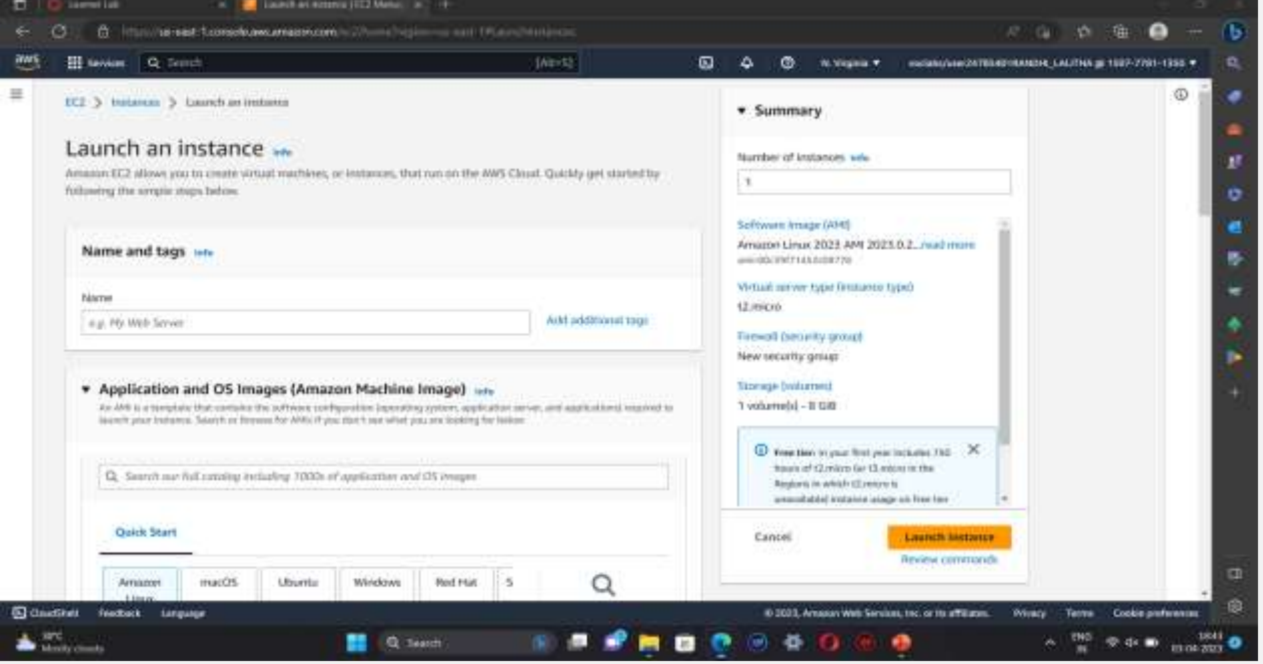
# AWS SERVICES

*20A31A05F0,*
*M .V.SOWJANYA.*

# AWS EC2 INSTANCE

## 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→ 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

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**

Name: e.g. My Web Server

**Application and OS Images (Amazon Machine Image)**

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.

**Summary**

Number of instances: 1

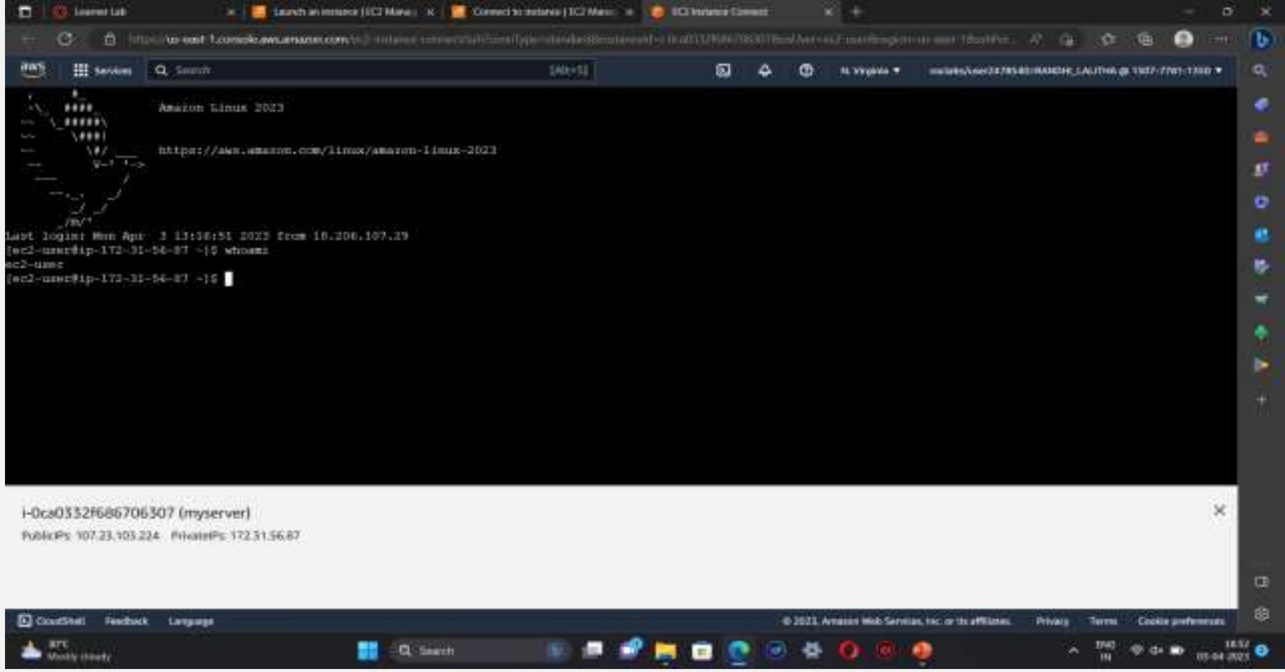
Software Image (AMI): Amazon Linux 2023 AMI 2023.0.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

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

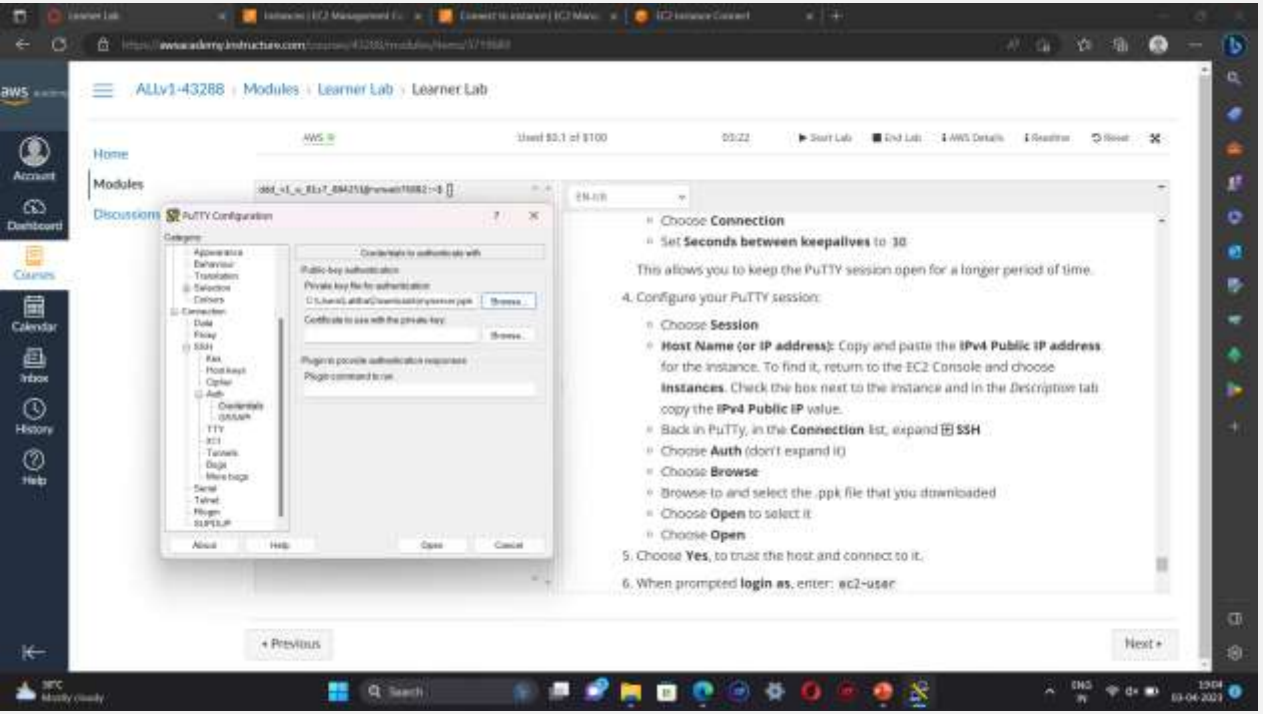
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Mon Apr  3 13:18:51 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

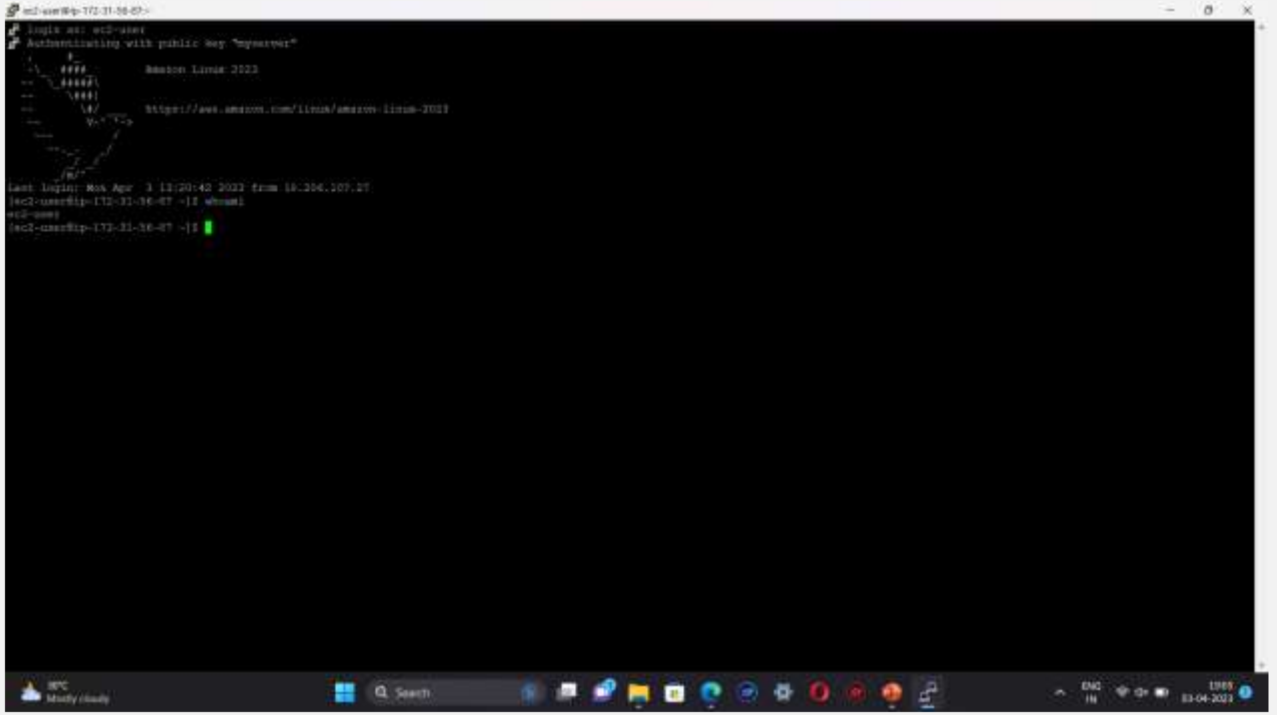


ALLv1-43288 › Modules › Learner Lab › Learner Lab
Home
Modules
Discussions
AWS
Used $0.1 of $100
03:22
Start Lab
End Lab
AWS Details
Readme
Reset
PuTTY Configuration
Credentials to authenticate with
Public-key authentication
Private key file for authentication:
Browse...
Certificate to use with the private key:
Plugin to provide authentication responses
Plugin command to run:
About
Help
Open
Cancel
Previous
Next

- Choose **Connection**
- Set **Seconds between keepalives** to 30

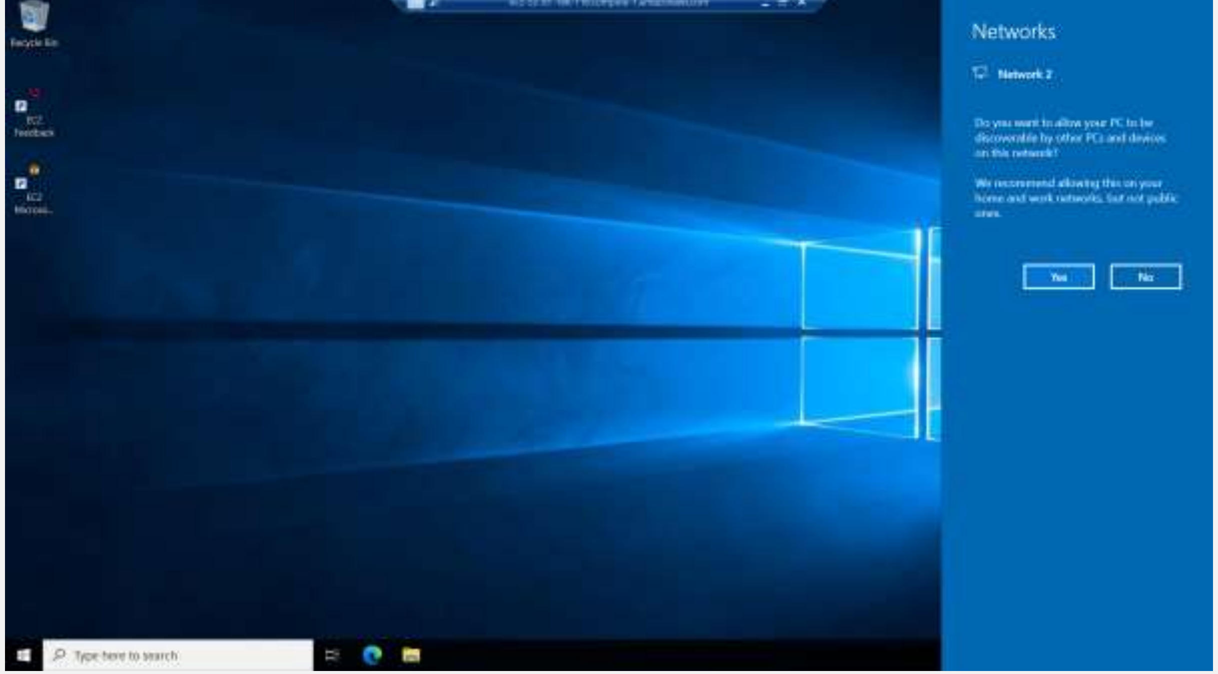
This allows you to keep the PuTTY session open for a longer period of time.

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`



ec2-user@ip-172-31-56-87:~

```
login as: ec2-user
Authenticating with public key "myserver"
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
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 ~]$
```



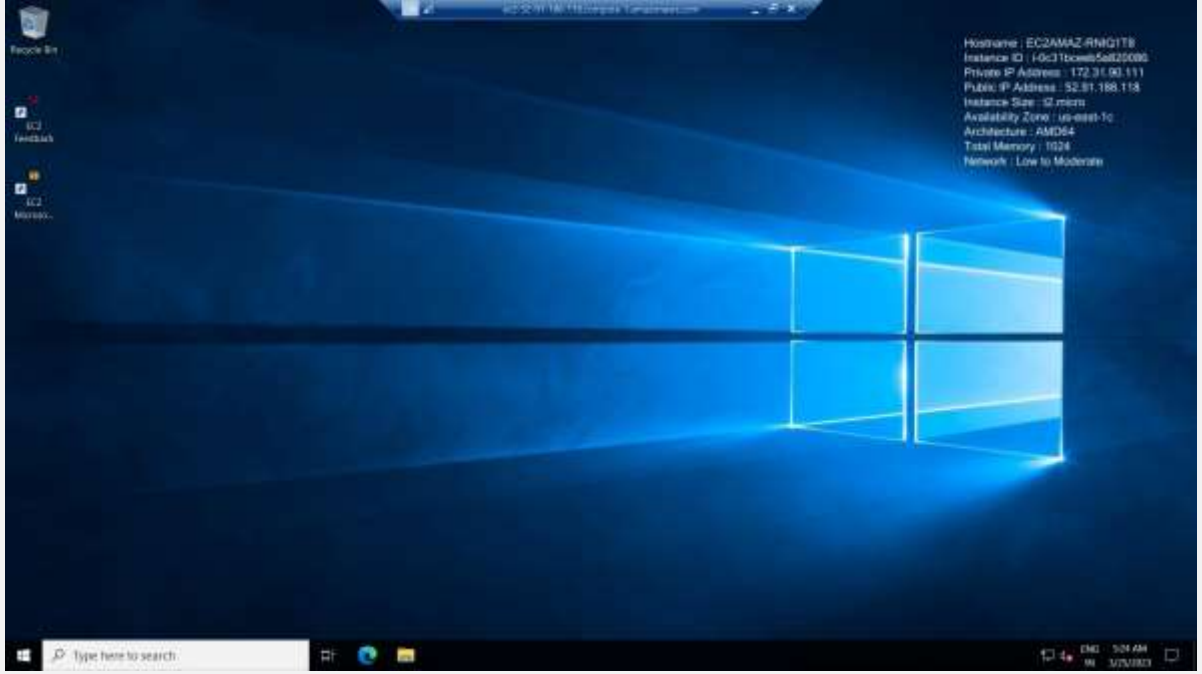
Networks

Network 2

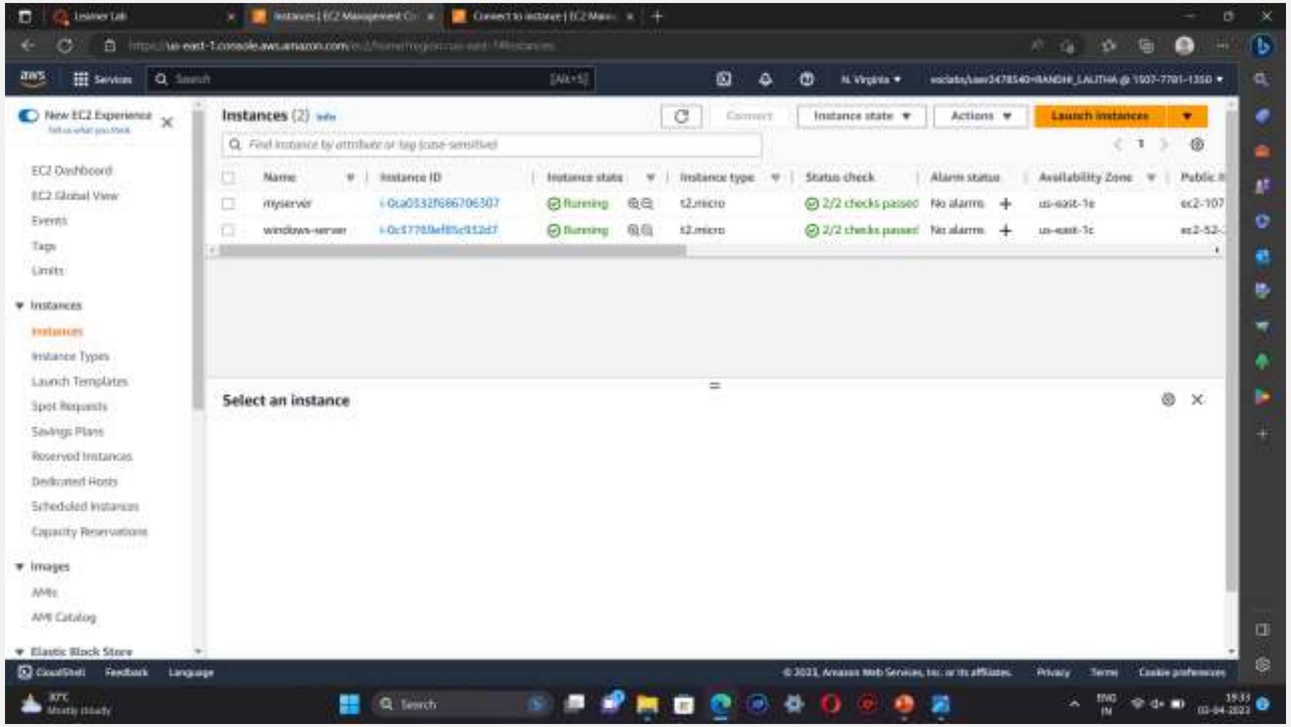
Do you want to allow your PC to be discoverable by other PCs and devices on this network?

We recommend allowing this on your home and work networks, but not public ones.

Yes No



Hostname : EC2AMAZ-RNIG1T8
Instance ID : i-0c31bceeb5e820080
Private IP Address : 172.31.90.111
Public IP Address : 52.91.188.118
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 | Public |
|---|---|---|---|---|---|---|---|---|
| □ | myserver | i-0ca0532f686706307 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-1e | ec2-107 |
| □ | windows-server | i-0c57769ef85c932d7 | Running | t2.micro | 2/2 checks passed | No alarms | us-east-1c | ec2-52- |

Select an instance

# AWS ELASTIC LOAD BALANCING (ELB)

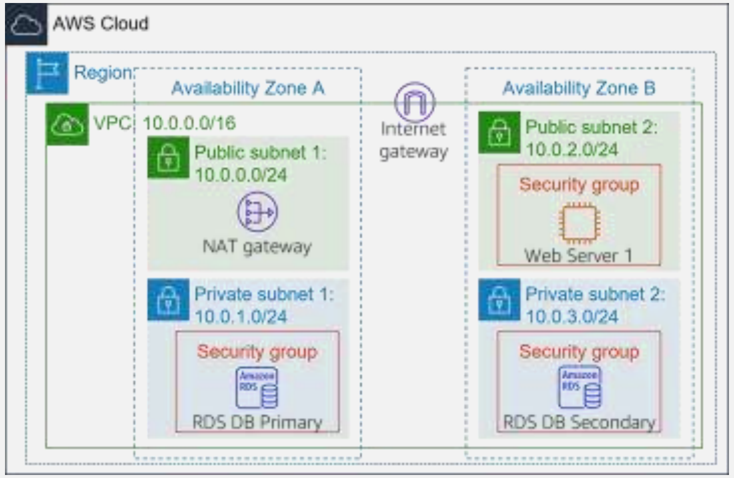
**Elastic Load Balancing** automatically distributes incoming application traffic across multiple Amazon EC2 instances.

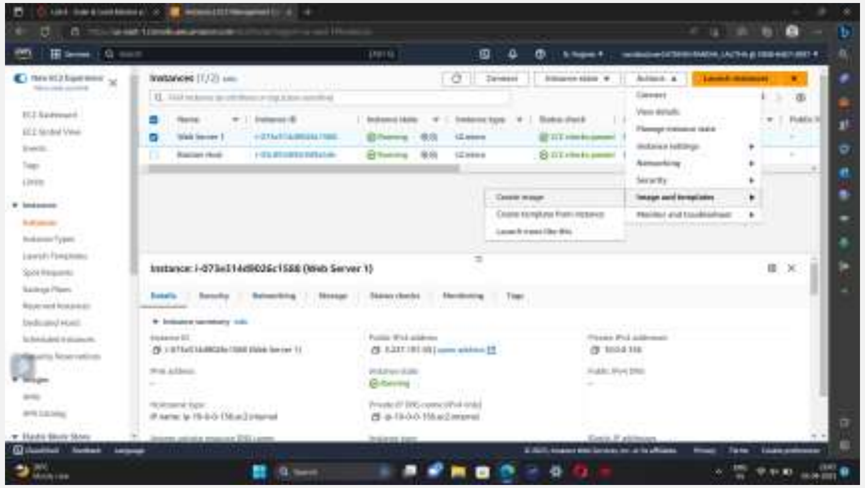
In this lab, We are provided with the given infrastructure.

Procedure:

Task1: Creating an AMI for Auto Scaling

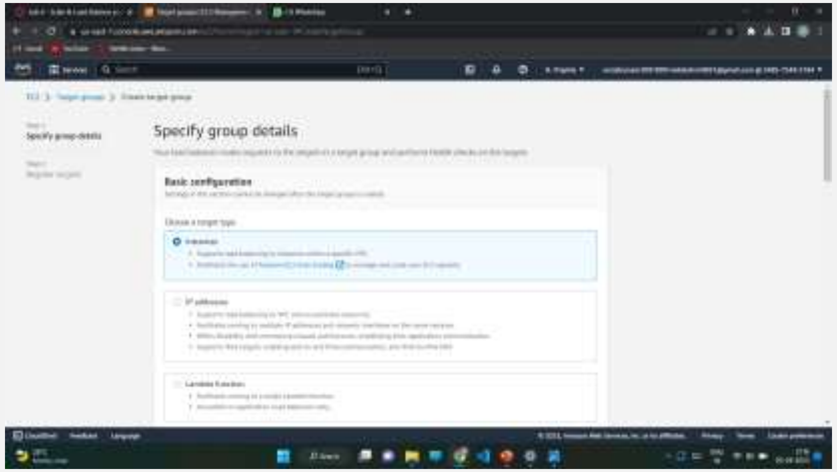
- ❖ Click start lab then click on AWS.
- ❖ You will navigate to AWS management console. Click on services and select EC2.
- ❖ Click instances. Make sure that **Status Checks** for **Web Server 1** displays 2/2 checks.
- ❖ Select Web Server 1 and in actions click images and templates > create image. Name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





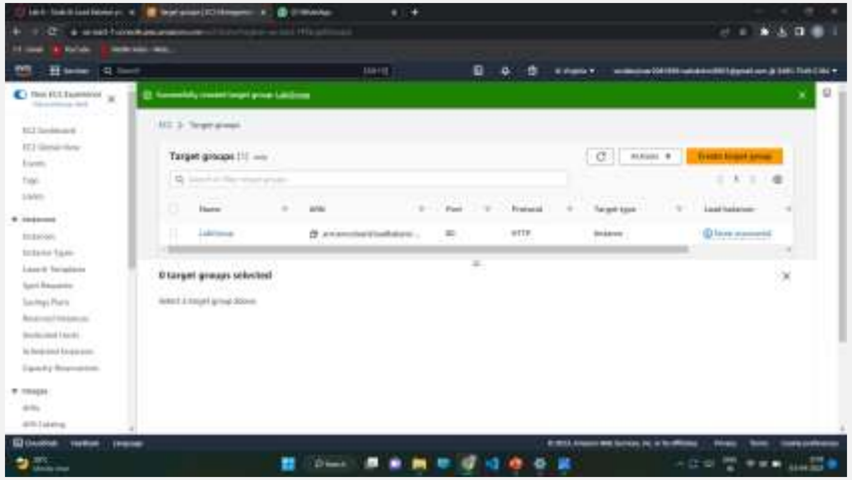
## Task 2: Creating a load balancer

- 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.



Specify group details

Basic configuration

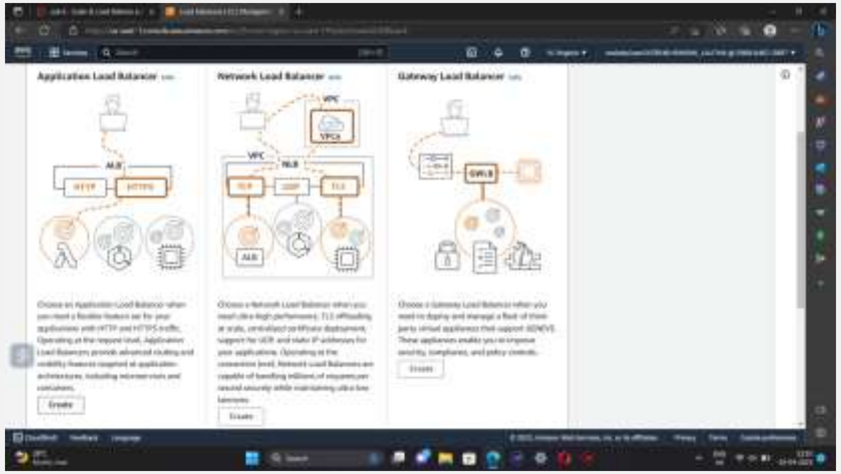


Target groups (1)

| Name | ARN | Port | Protocol | Target type | Load balancer |
|---|---|---|---|---|---|
| LabGroup | [illegible] | 80 | HTTP | Instance | [illegible] |

0 target groups selected

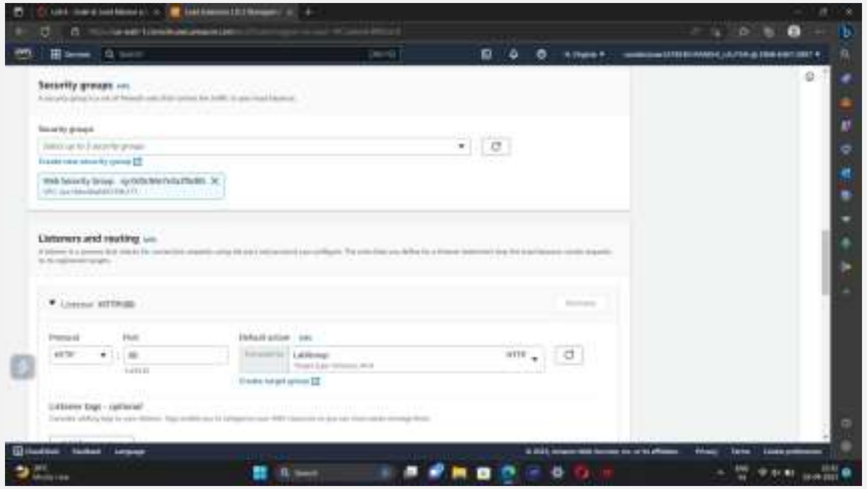
- From the left navigation pane , select Load Balancers. Click create load balancer.
- To create a application balancer, click create under Application Load Balancer and Name it.
- In Networking mapping, select Lab VPC and specify the subnets that the load balancer should use.
- In security groups, select 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

Network Load Balancer

Gateway Load Balancer



Security groups

Security groups

Web Security Group

Listeners and routing

Listener HTTP:80

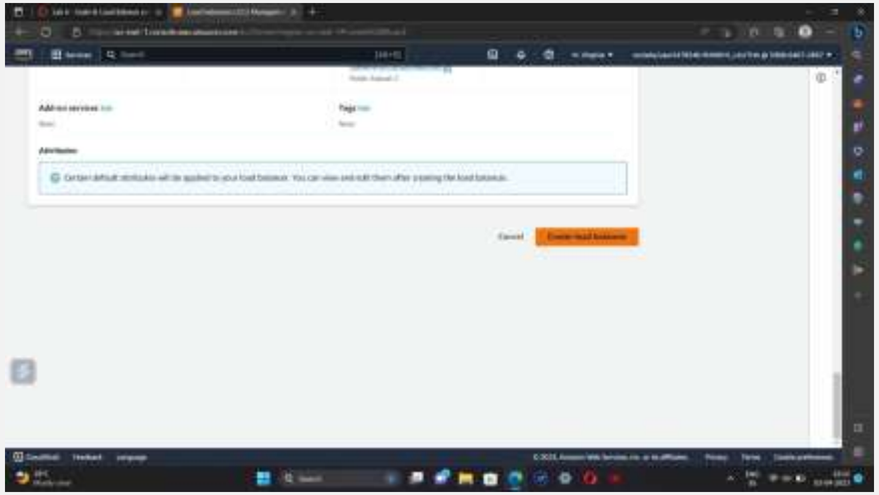
Protocol: HTTP

Port: 80

Default action

Forward to: LabGroup HTTP

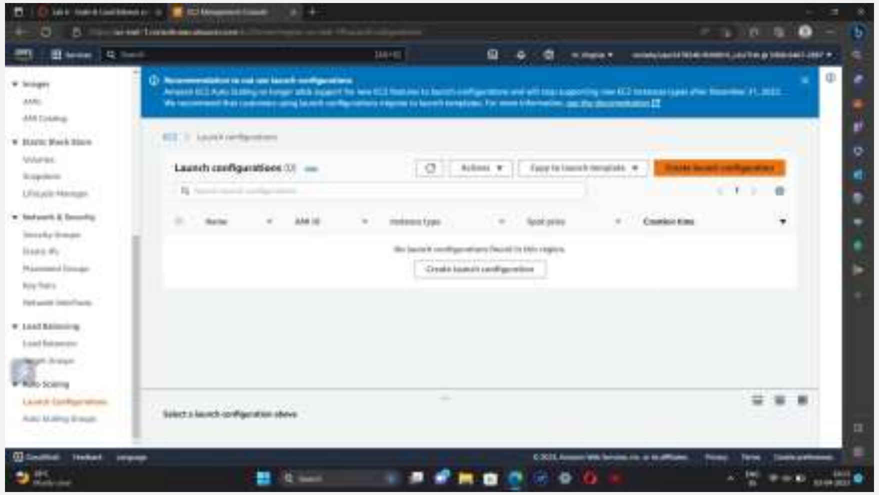
Listener tags - optional



Add-on services

Tags

Attributes



Launch configurations (0)

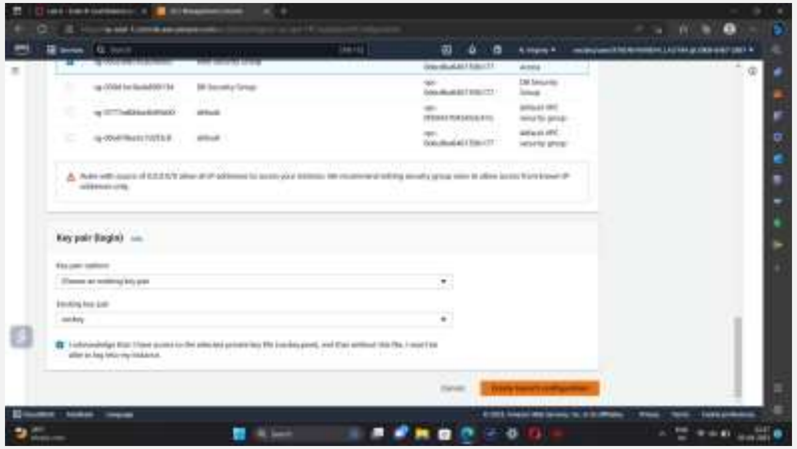
No launch configurations found in this region.

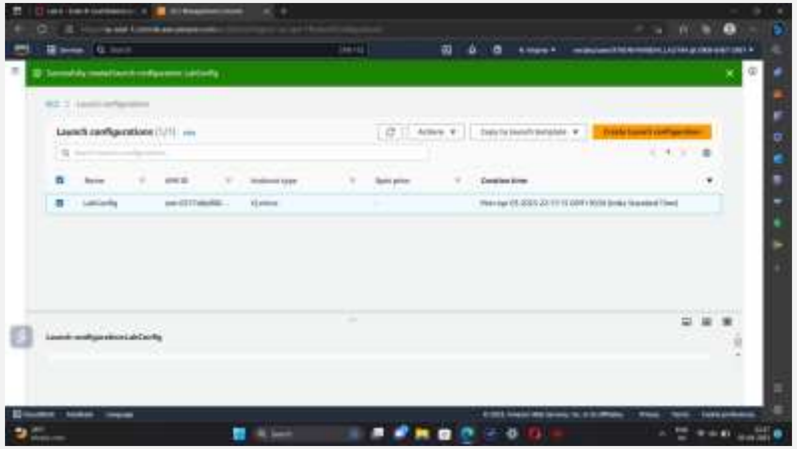
Select a launch configuration above

- Click create load balancer.

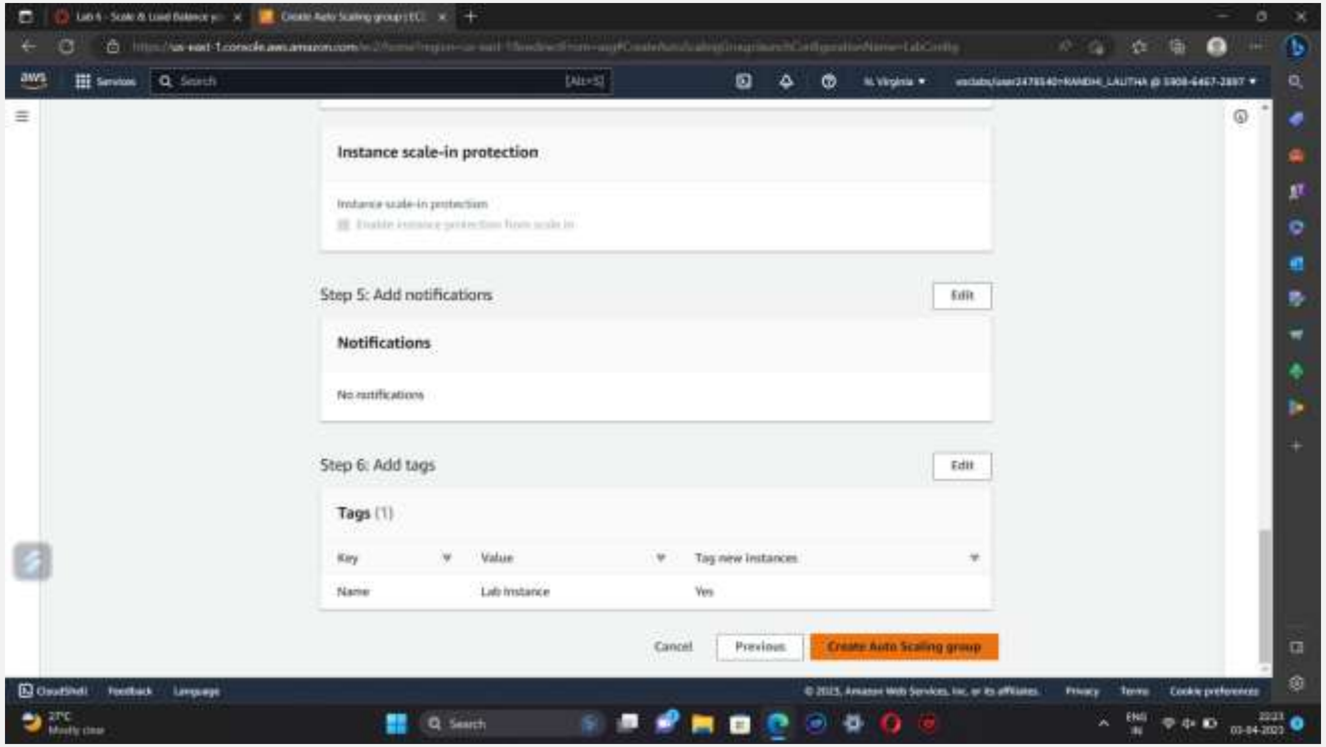
## Task 3: Create a Launch Configuration and an Auto Scaling Group

- 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.



Instance scale-in protection

Instance scale-in protection

□ Enable instance protection from scale-in

Step 5: Add notifications

Notifications

No notifications

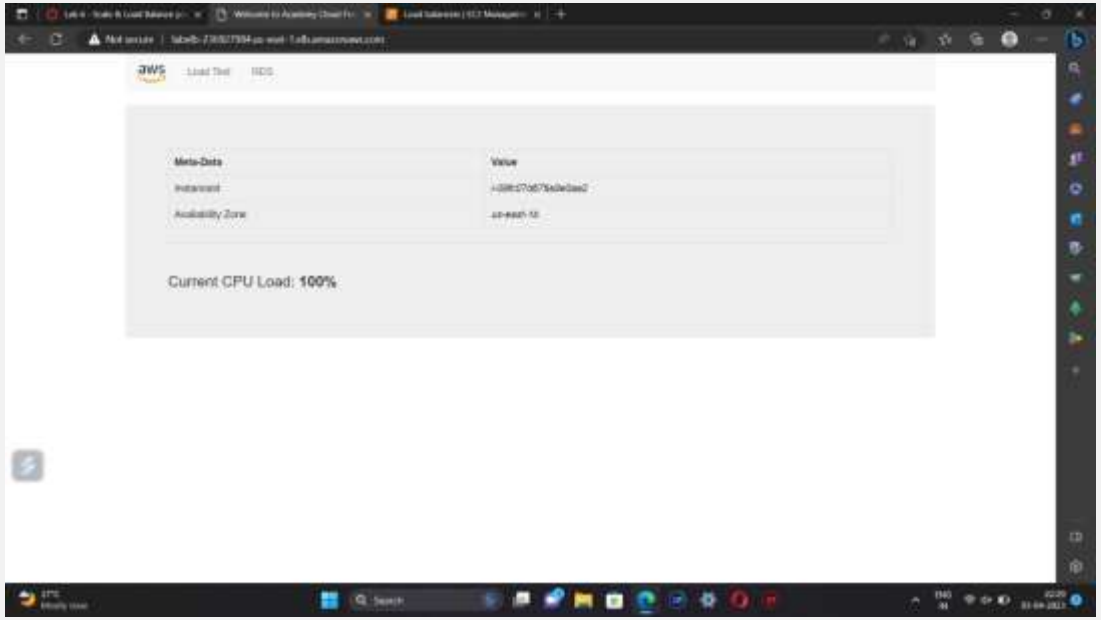
Step 6: Add tags

Tags (1)

| Key | Value | Tag new instances |
|---|---|---|
| Name | Lab Instance | Yes |

**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.



| Meta-Data | Value |
|---|---|
| InstanceId | [illegible] |
| Availability Zone | us-east-1b |

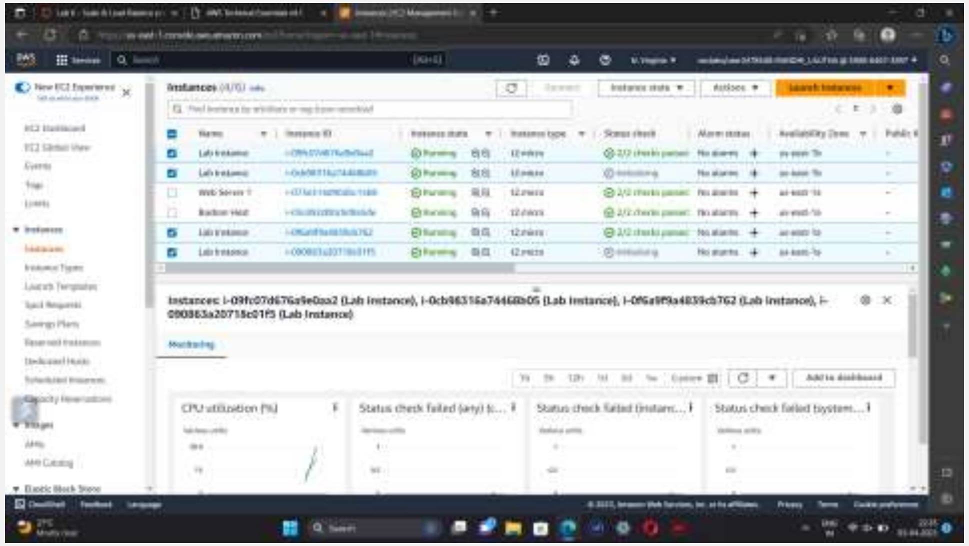
Current CPU Load: **100%**

## **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.



| | Name | Instance ID | Instance state | Instance type | Status check | Alarm status |
|---|---|---|---|---|---|---|
| ☑ | Lab Instance | [illegible] | Running | t2.micro | 2/2 checks passed | No alarms |
| ☑ | Lab Instance | [illegible] | Running | t2.micro | Initializing | No alarms |
| □ | Web Server 1 | [illegible] | Running | t2.micro | 2/2 checks passed | No alarms |
| □ | Bastion Host | [illegible] | Running | t2.micro | 2/2 checks passed | No alarms |
| ☑ | Lab Instance | [illegible] | Running | t2.micro | 2/2 checks passed | No alarms |
| ☑ | Lab Instance | [illegible] | Running | t2.micro | Initializing | No alarms |

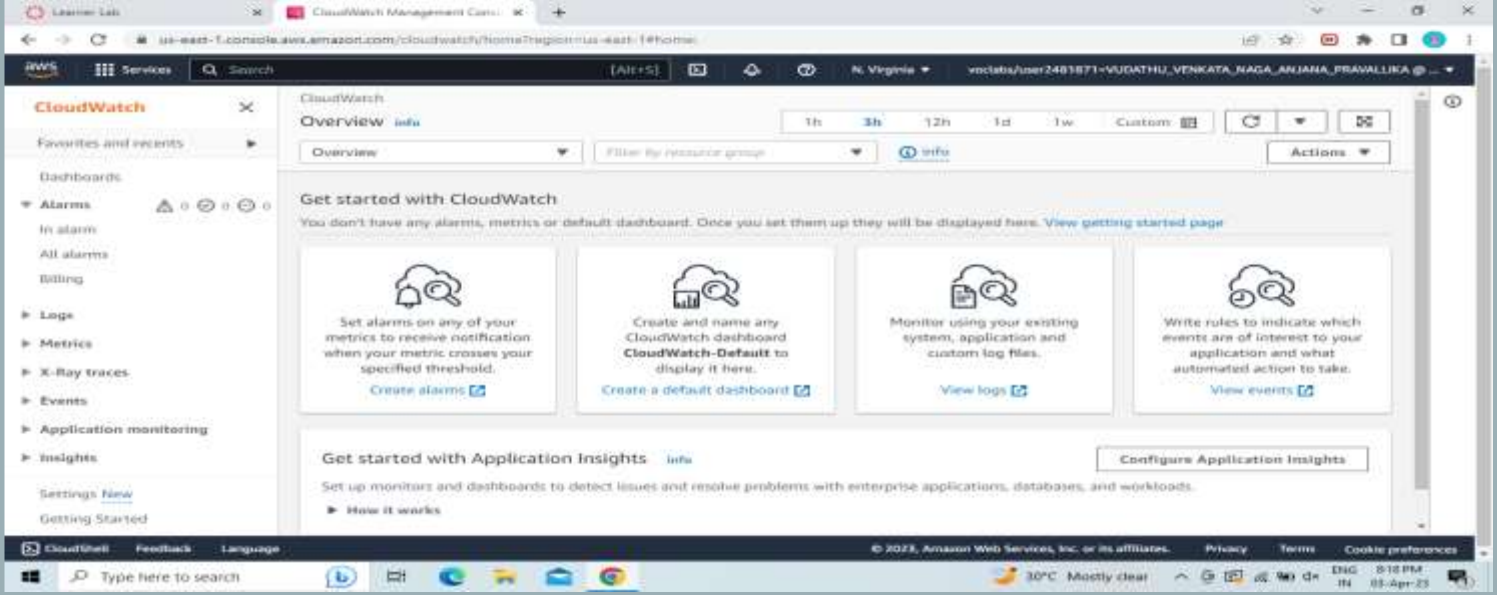
Instances: i-09fc07d676a9e0aa2 (Lab Instance), i-0cb96316a74468b05 (Lab Instance), i-0f6a9f9a4839cb762 (Lab Instance), i-090863a20718c01f5 (Lab Instance)

Instances
Launch instances
Actions
Instance state
Monitoring
Add to dashboard
CPU utilization (%)
Status check failed (any) (c...
Status check failed (instanc...
Status check failed (system...

# AWS CLOUDWATCH

## PROCEDURE

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



Get started with CloudWatch

You don't have any alarms, metrics or default dashboard. Once you set them up they will be displayed here. View getting started page

Set alarms on any of your metrics to receive notification when your metric crosses your specified threshold.

Create and name any CloudWatch dashboard **CloudWatch-Default** to display it here.

Monitor using your existing system, application and custom log files.

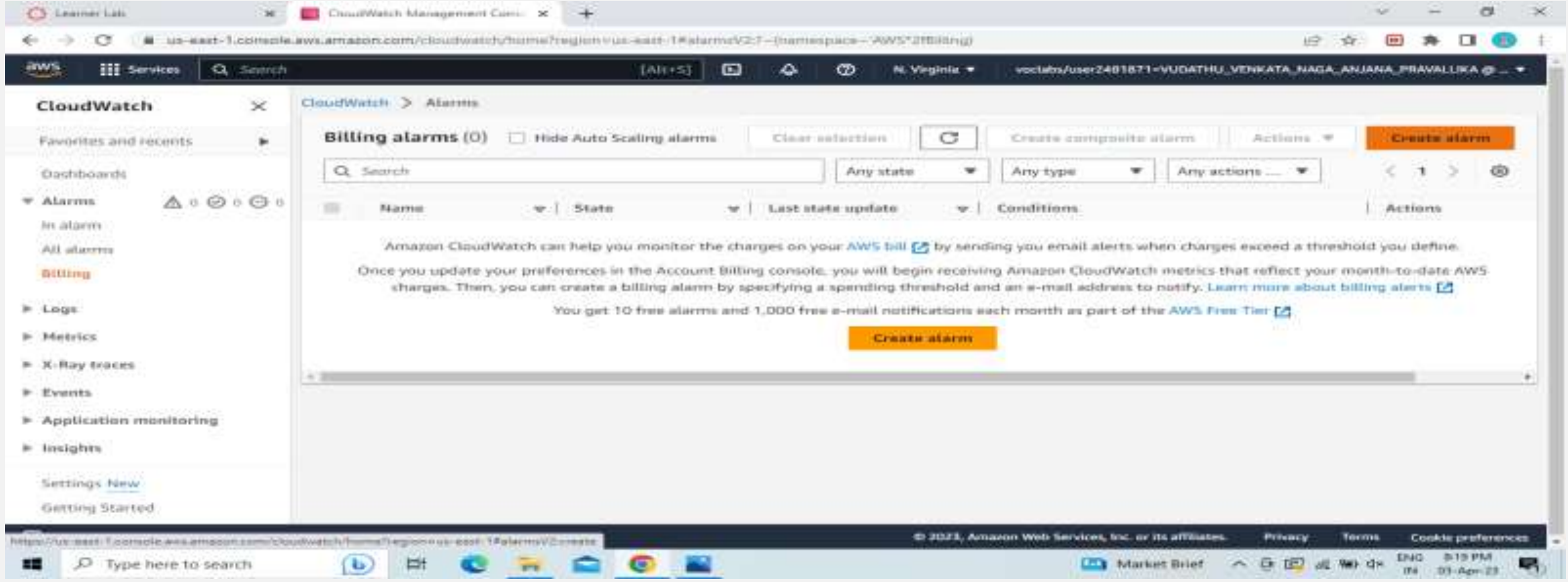
Write rules to indicate which events are of interest to your application and what automated action to take.

Get started with Application Insights

Set up monitors and dashboards to detect issues and resolve problems with enterprise applications, databases, and workloads.

© 2023, Amazon Web Services, Inc. or its affiliates.

2.Then Click on CREATE ALARM.



Billing alarms (0)

Amazon CloudWatch can help you monitor the charges on your AWS bill by sending you email alerts when charges exceed a threshold you define.

Once you update your preferences in the Account Billing console, you will begin receiving Amazon CloudWatch metrics that reflect your month-to-date AWS charges. Then, you can create a billing alarm by specifying a spending threshold and an e-mail address to notify. Learn more about billing alerts

You get 10 free alarms and 1,000 free e-mail notifications each month as part of the AWS Free Tier

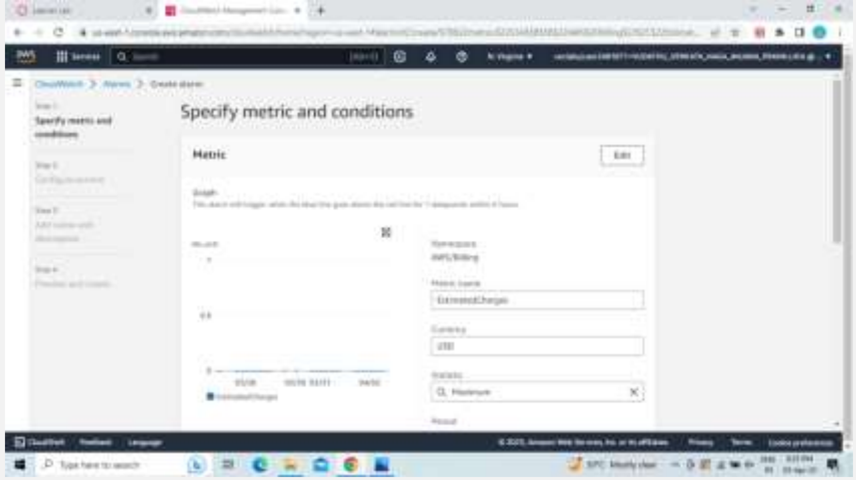
3.Then follow the steps.

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.



Specify metric and conditions

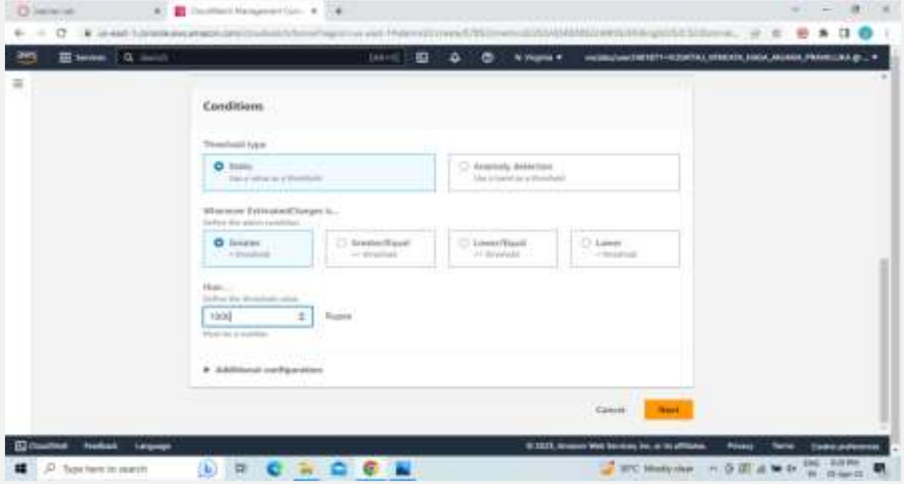
Metric

Namespace: AWS/Billing

Metric name: EstimatedCharges

Currency: USD

Statistic: Maximum



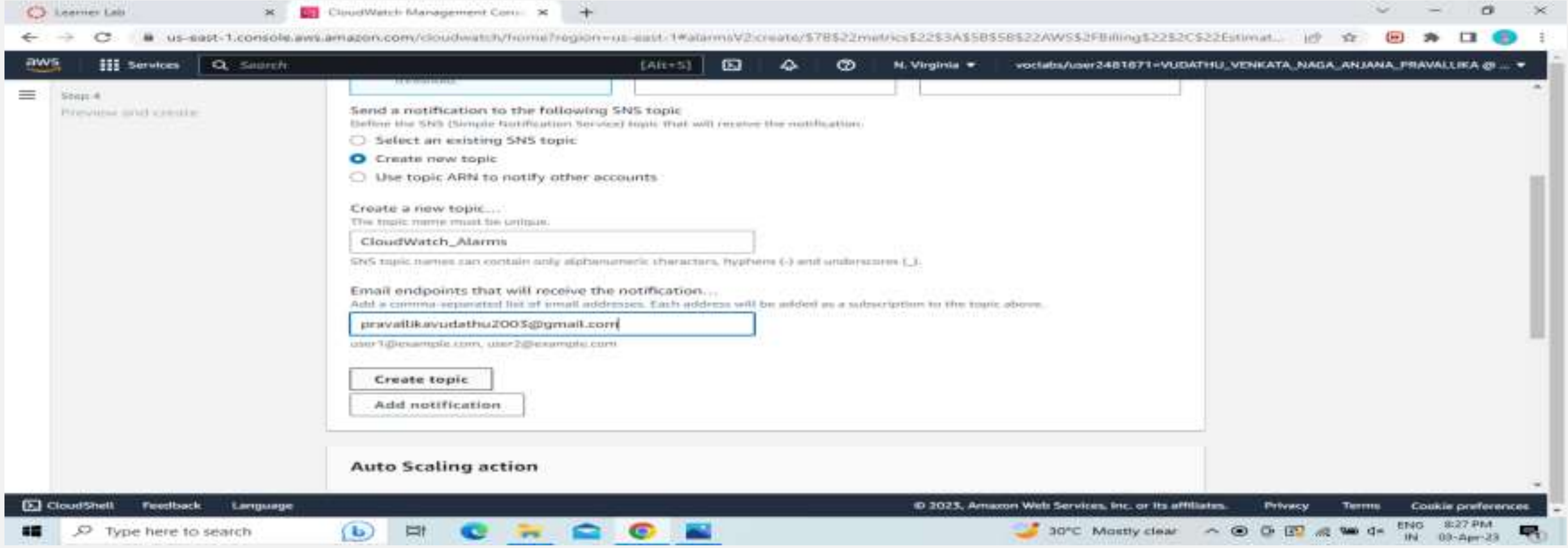
Conditions

Threshold type: ● Static (Use a value as a threshold) ○ Anomaly detection (Use a band as a threshold)

Whenever EstimatedCharges is...: ● Greater (> threshold) ○ Greater/Equal (>= threshold) ○ Lower/Equal (<= threshold) ○ Lower (< threshold)

than...: 1000 Rupee

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.



Send a notification to the following SNS topic

Define the SNS (Simple Notification Service) topic that will receive the notification.

○ Select an existing SNS topic

● Create new topic

○ Use topic ARN to notify other accounts

Create a new topic...

The topic name must be unique.

CloudWatch_Alarms

SNS topic names can contain only alphanumeric characters, hyphens (-) and underscores (_).

Email endpoints that will receive the notification...

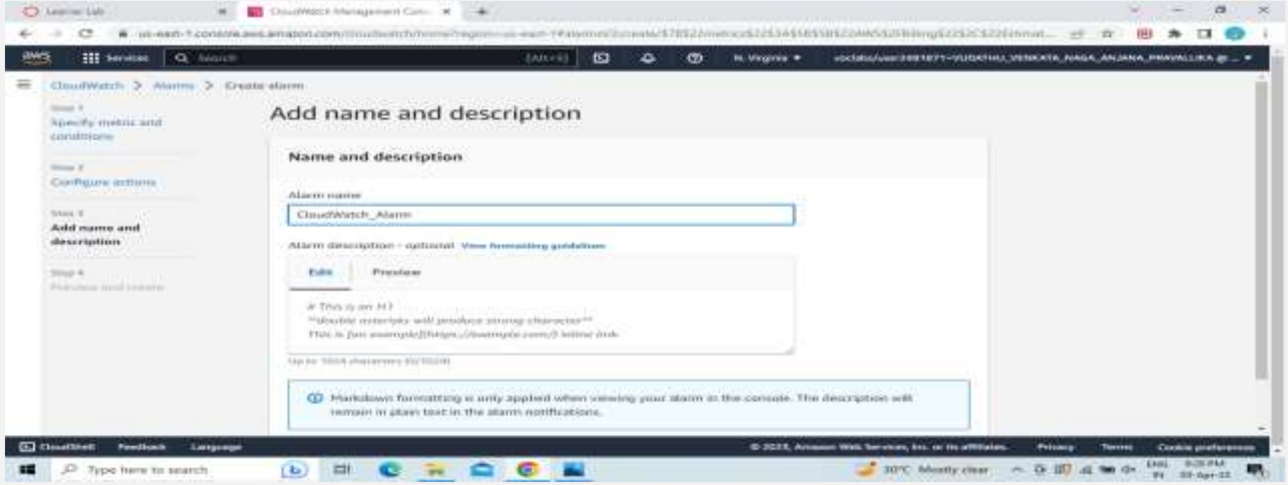
Add a comma-separated list of email addresses. Each address will be added as a subscription to the topic above.

pravallikavudathu2003@gmail.com

user1@example.com, user2@example.com

Learner Lab
CloudWatch Management Cons
us-east-1.console.aws.amazon.com/cloudwatch/home?region=us-east-1#alarmsV2:create/$7B$22metrics$22$3A$5B$5B$22AWS$2FBilling$22$2C$22Estimat...
aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user2481871=VUDATHU_VENKATA_NAGA_ANJANA_PRAVALLIKA @ ...
Step 4
Preview and create
Create topic
Add notification
Auto Scaling action
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
30°C Mostly clear
ENG
IN
8:27 PM
03-Apr-23

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



CloudWatch > Alarms > Create alarm

Add name and description

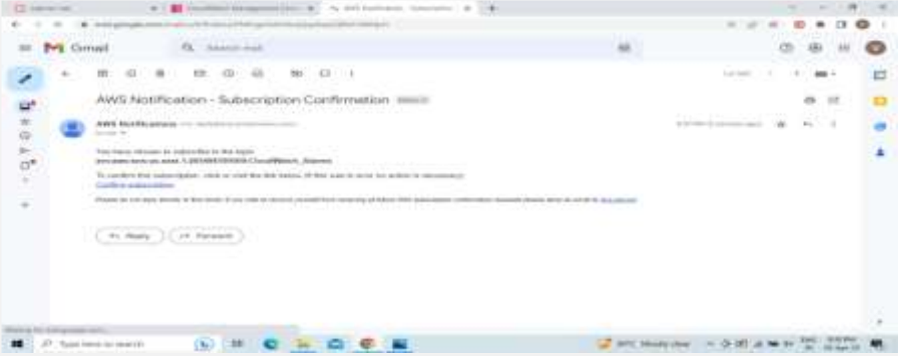
Name and description

Alarm name: CloudWatch_Alarm

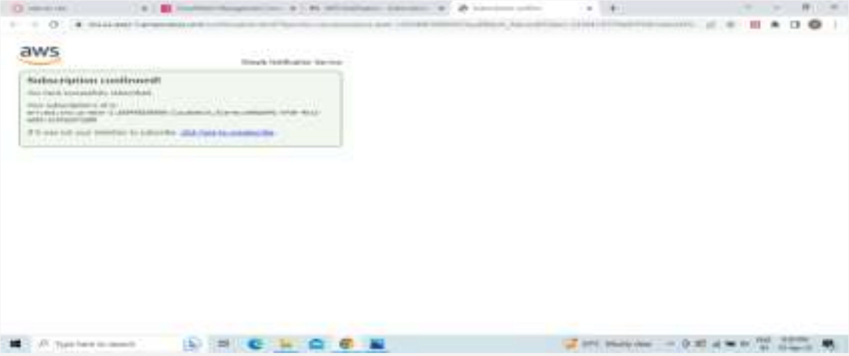
Alarm description - optional View formatting guidelines

Markdown formatting is only applied when viewing your alarm in the console. The description will remain in plain text in the alarm notifications.

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.



AWS Notification - Subscription Confirmation

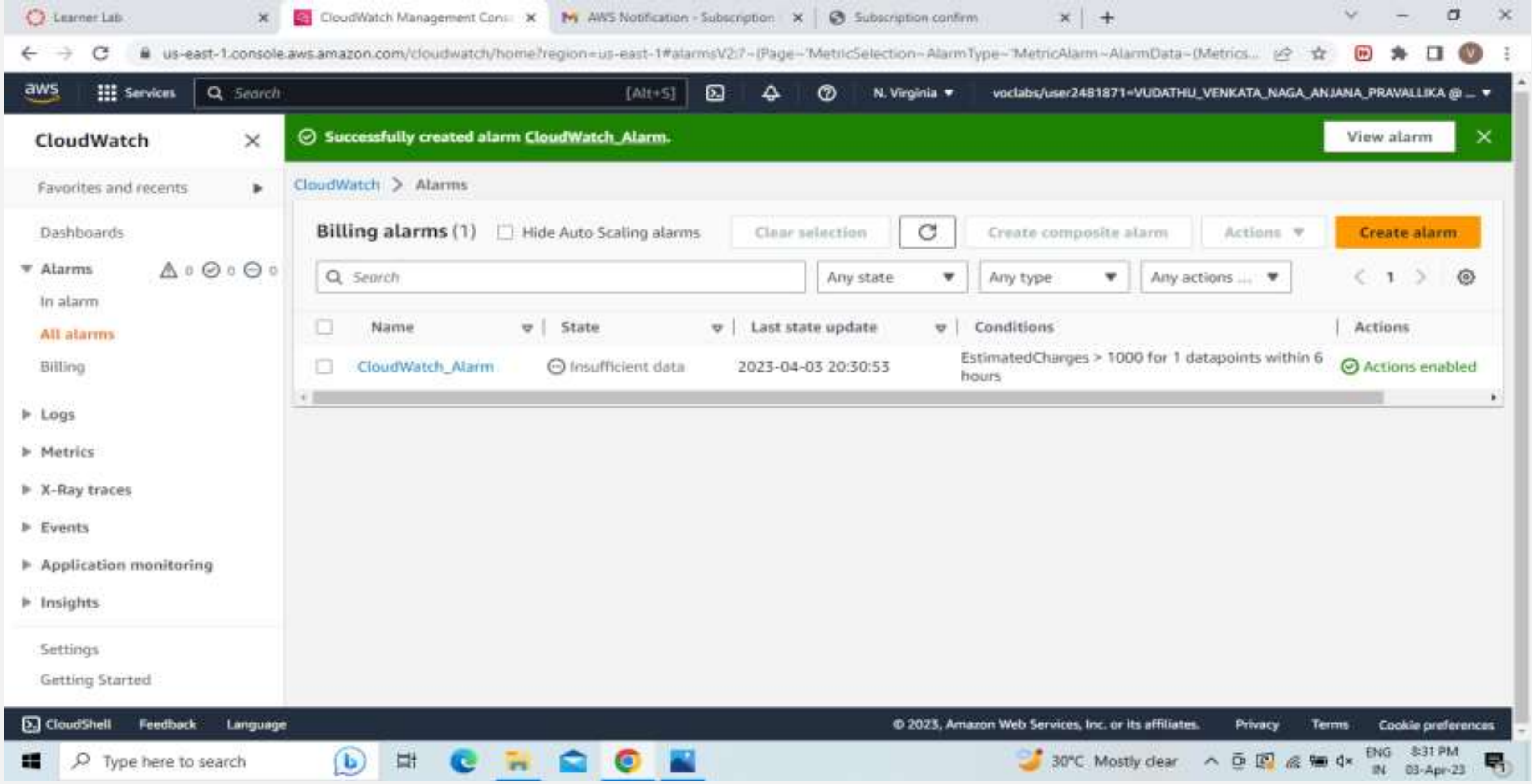


Simple Notification Service

Subscription confirmed!

8.Preview the details you have entered .

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



Successfully created alarm CloudWatch_Alarm.

Billing alarms (1)

| Name | State | Last state update | Conditions | Actions |
|---|---|---|---|---|
| CloudWatch_Alarm | Insufficient data | 2023-04-03 20:30:53 | EstimatedCharges > 1000 for 1 datapoints within 6 hours | Actions enabled |

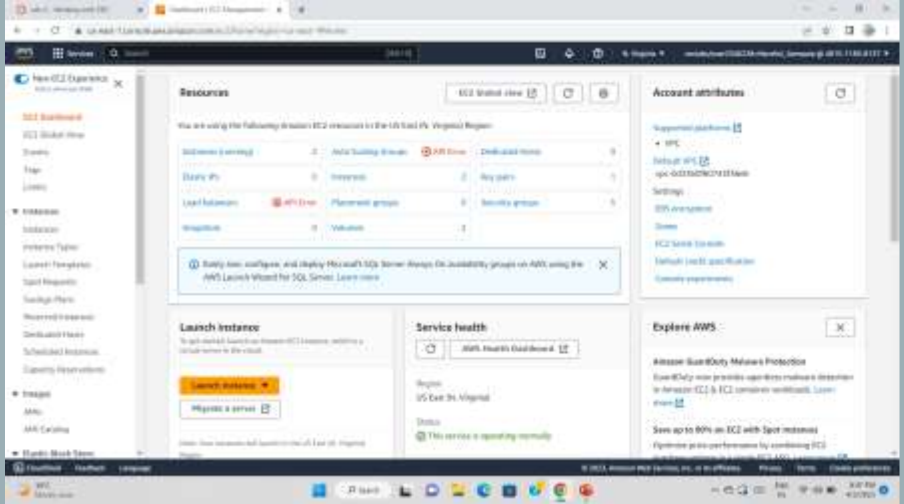
# AWS EBS(ELASTIC BLOCK STORE)

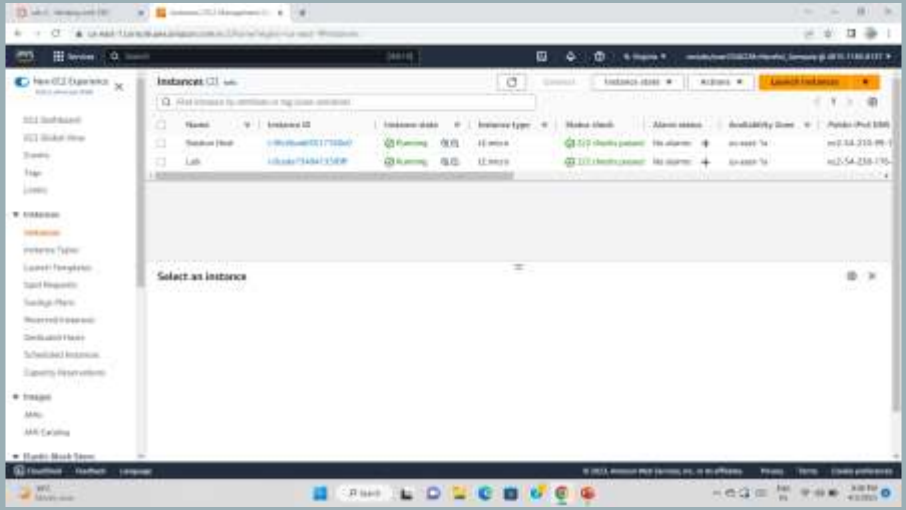
## CREATING A EBS VOLUME

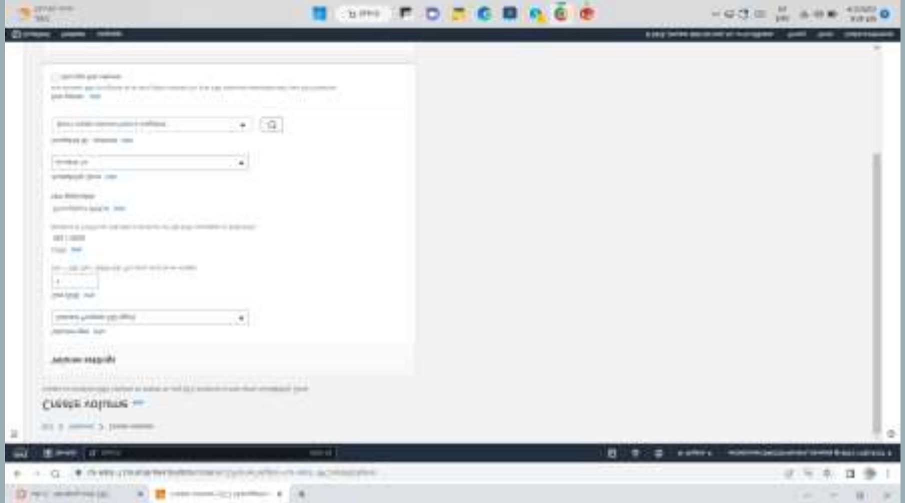
1. Open Management Console, on the services menu open Ec2
2. In the left navigation pane choose instances and create a instance with a name
3. Next, In the left navigation pane choose Volumes
4. Click on Create Volume
5. Select volume type, size(Gib),Availability Zone and in Add tag section add key and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

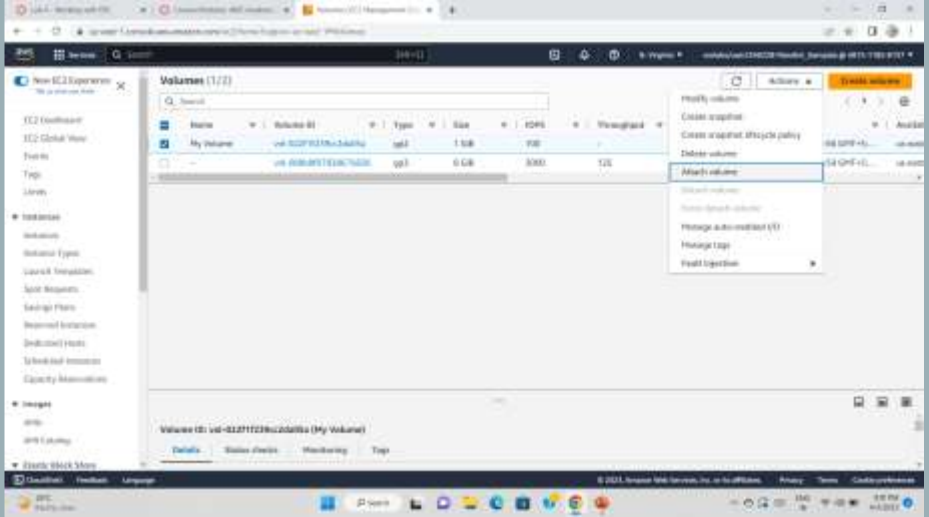
```
df –h
sudo mkfs –t ext3/dev/sdf                    etc.,
```

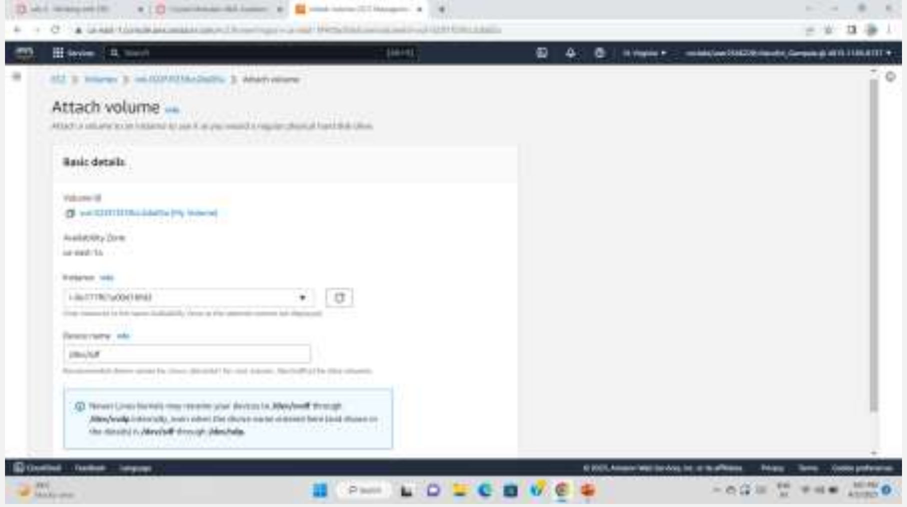
14. Create a EBS snapshot by giving the necessary fields.
15. Create a volume using snapshot.
16. Attach the volume to the created EC2 instance


Resources
EC2 Global view
Account attributes
Launch instance
Launch instance
Migrate a server
Service health
AWS Health Dashboard
Region
US East (N. Virginia)
Explore AWS
Amazon GuardDuty Malware Protection
Save up to 90% on EC2 with Spot Instances


Instances (2)
Launch instances
Select an instance


Create volume
Volume settings


Volumes (1/2)
Actions
Create volume
Modify volume
Create snapshot
Create snapshot lifecycle policy
Delete volume
Attach volume
Manage tags
My Volume
Details
Status checks
Monitoring
Tags



EC2 > Volumes > [illegible] > Attach volume

## Attach volume

Basic details

Volume ID

Availability Zone

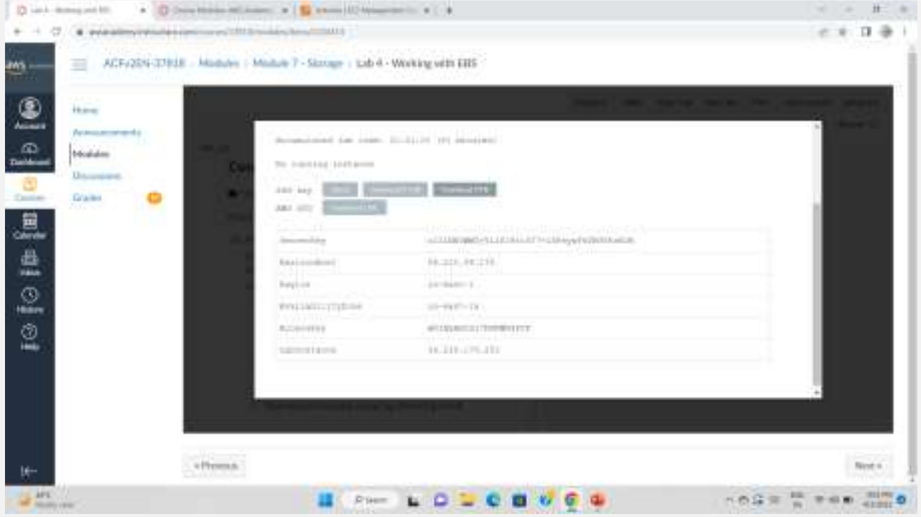
us-east-1a

Instance

Device name

/dev/sdf

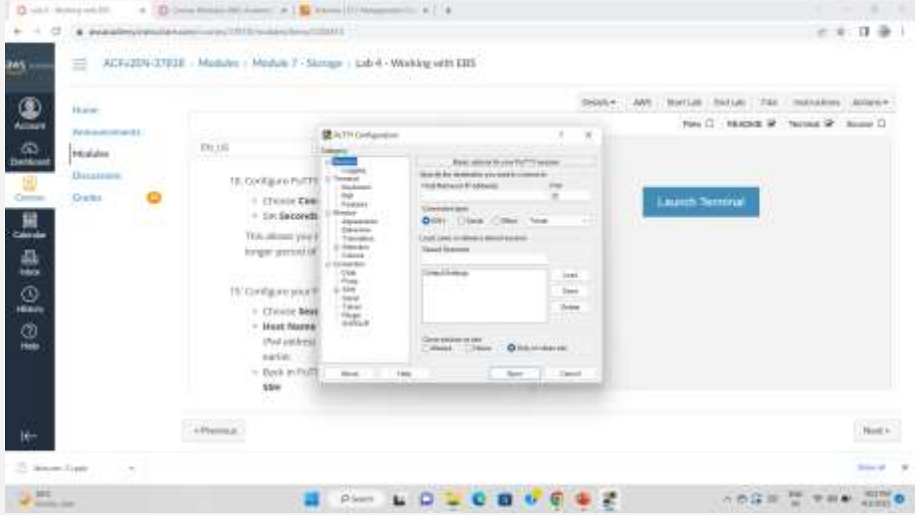
Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.



ACF-[illegible] › Modules › Module 7 - Storage › Lab 4 - Working with EBS

```
No running instance
```

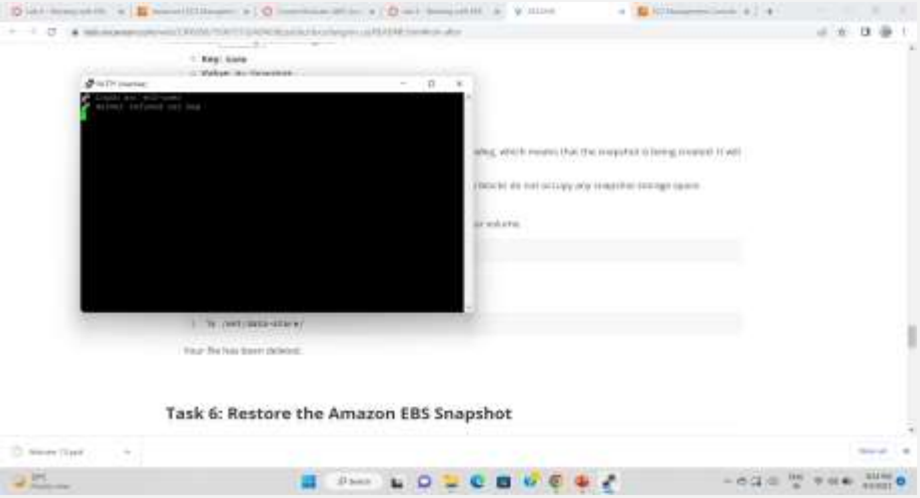
| | |
|---|---|
| SecretKey | [illegible] |
| Region | us-east-1 |
| AvailabilityZone | us-east-1a |
| [illegible] | [illegible] |
| [illegible] | [illegible] |
| [illegible] | [illegible] |



ACF-[illegible] › Modules › Module 7 - Storage › Lab 4 - Working with EBS

18. Configure PuTTY [illegible]

19. Configure your [illegible]

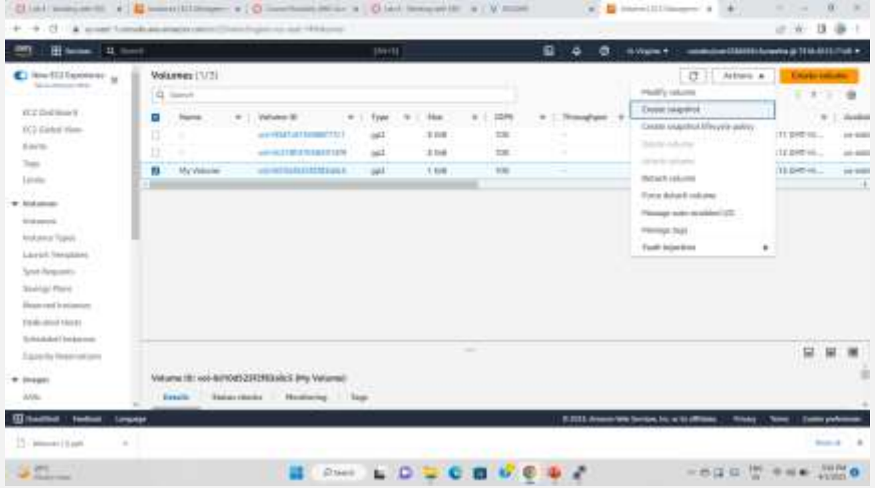


- Key: Name
- Value: [illegible]

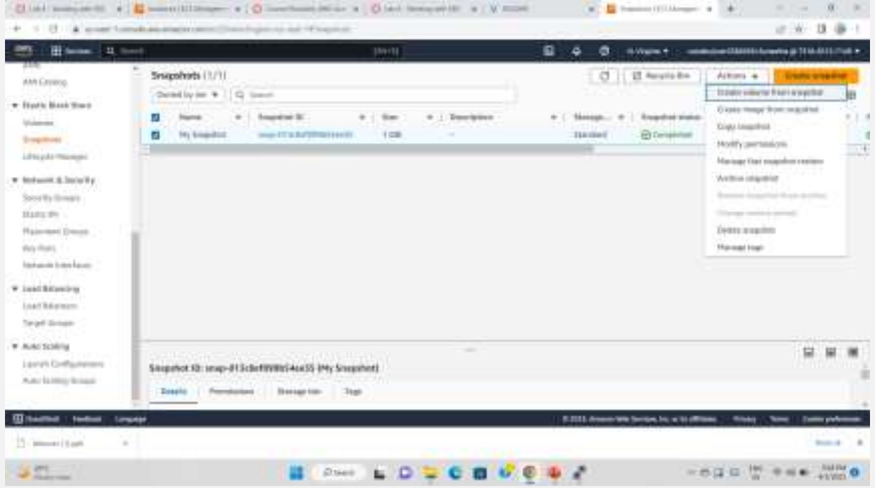
```
ls /mnt/data-store/
```

Your file has been deleted.

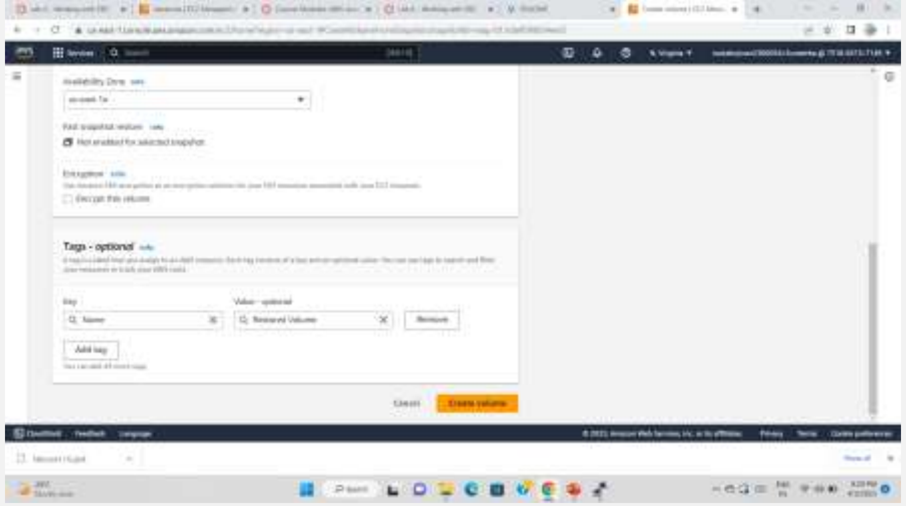
### Task 6: Restore the Amazon EBS Snapshot



Volumes (1/3)



Snapshots (1/1)



Availability Zone: us-east-1a

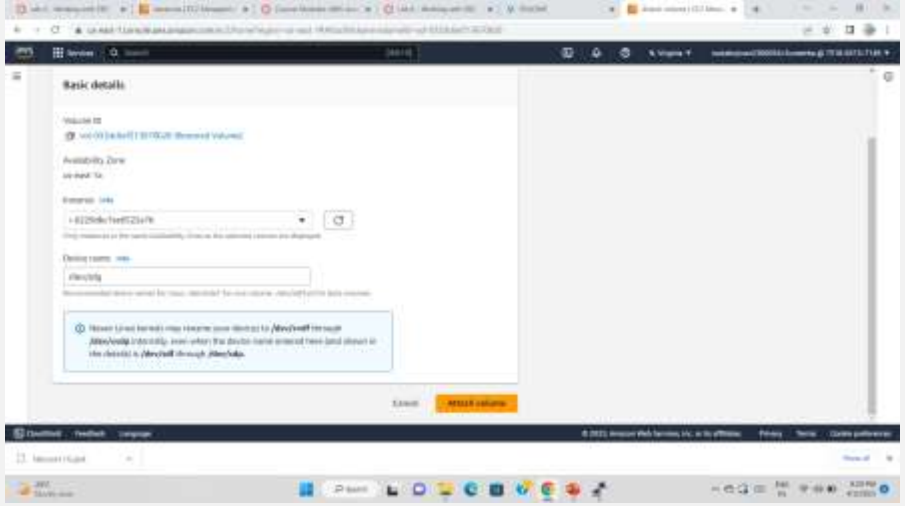
Fast snapshot restore: Not enabled for selected snapshot

Encryption: □ Encrypt this volume

Tags - optional

Key: Name

Value - optional: Restored Volume



Basic details

Volume ID: (Restored Volume)

Availability Zone: us-east-1a

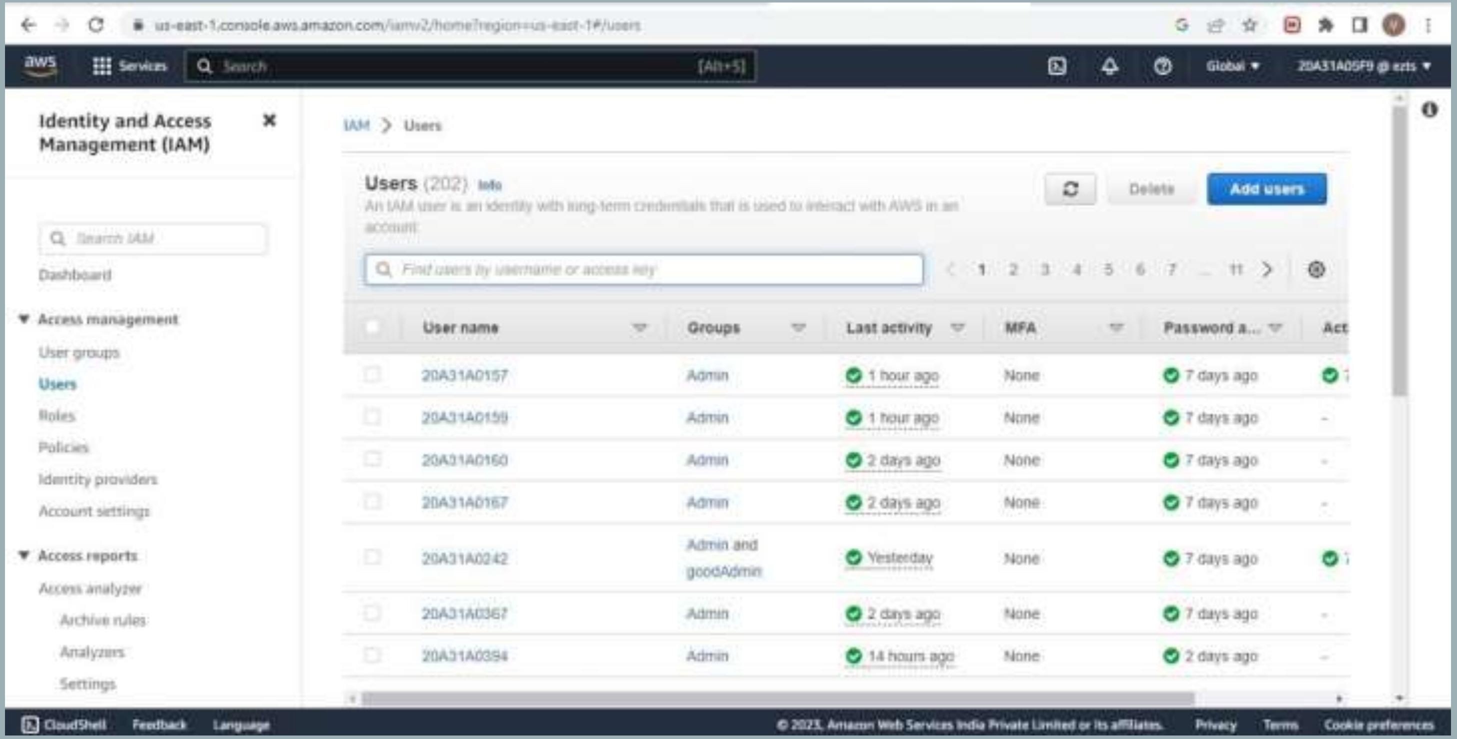
Instance

Device name: /dev/sdg

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.

# 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



Users (202) Info

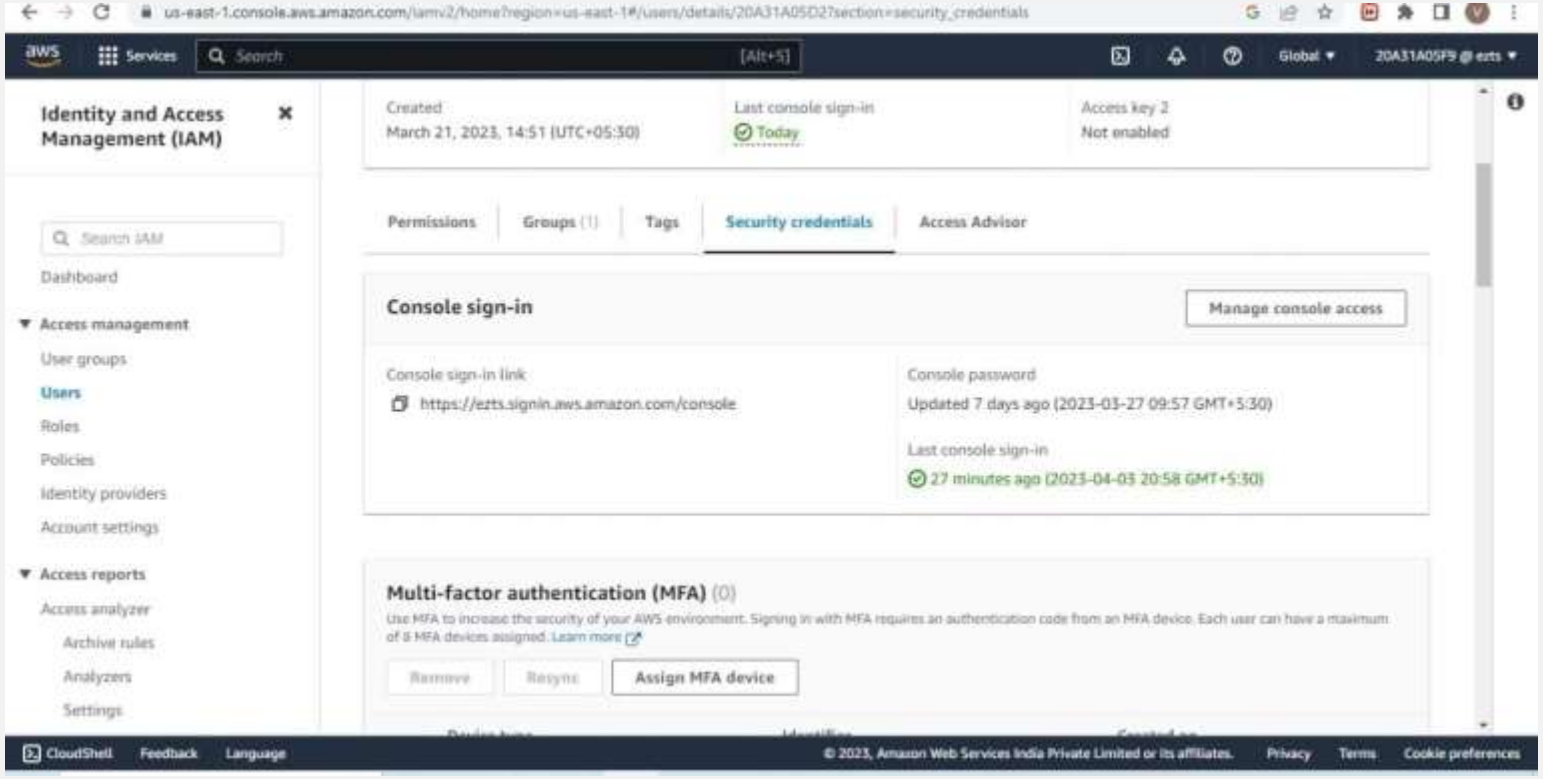
An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

| □ | User name | Groups | Last activity | MFA | Password a... | Act |
|---|---|---|---|---|---|---|
| □ | 20A31A0157 | Admin | ✓ 1 hour ago | None | ✓ 7 days ago | ✓ |
| □ | 20A31A0159 | Admin | ✓ 1 hour ago | None | ✓ 7 days ago | - |
| □ | 20A31A0160 | Admin | ✓ 2 days ago | None | ✓ 7 days ago | - |
| □ | 20A31A0167 | Admin | ✓ 2 days ago | None | ✓ 7 days ago | - |
| □ | 20A31A0242 | Admin and goodAdmin | ✓ Yesterday | None | ✓ 7 days ago | ✓ |
| □ | 20A31A0367 | Admin | ✓ 2 days ago | None | ✓ 7 days ago | - |
| □ | 20A31A0394 | Admin | ✓ 14 hours ago | None | ✓ 2 days ago | - |

© 2023, Amazon Web Services India Private Limited or its affiliates.

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.



Created
March 21, 2023, 14:51 (UTC+05:30)

Last console sign-in
Today

Access key 2
Not enabled

Console sign-in

Console sign-in link
https://ezts.signin.aws.amazon.com/console

Console password
Updated 7 days ago (2023-03-27 09:57 GMT+5:30)

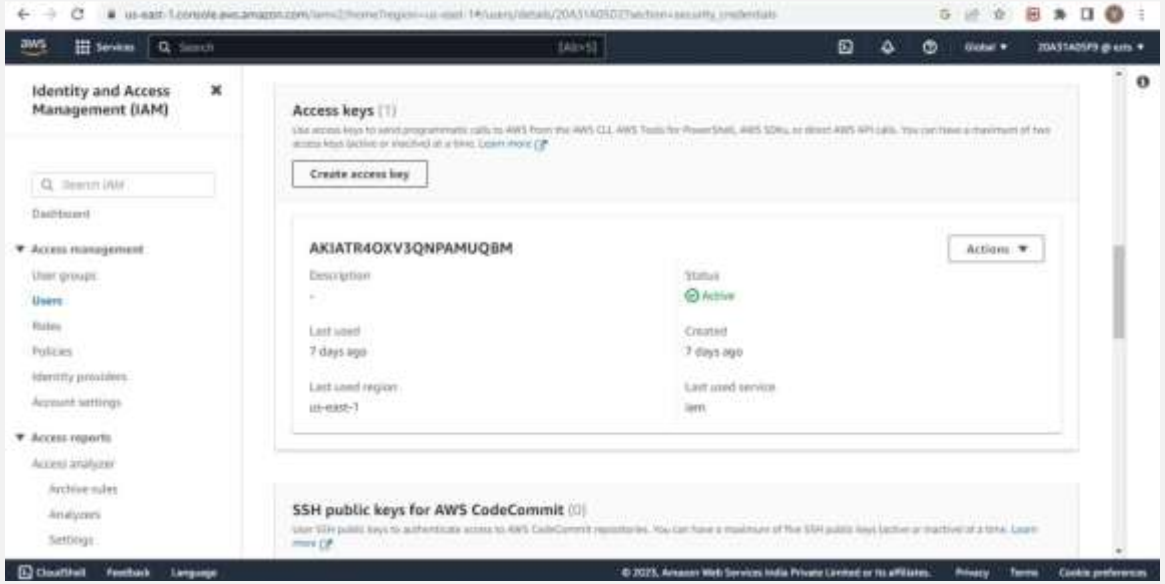
Last console sign-in
27 minutes ago (2023-04-03 20:58 GMT+5:30)

Multi-factor authentication (MFA) (0)

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more

© 2023, Amazon Web Services India Private Limited or its affiliates.

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



Access keys (1)

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more

Create access key

AKIATR4OXV3QNPAMUQBM

Description: -

Status: Active

Last used: 7 days ago

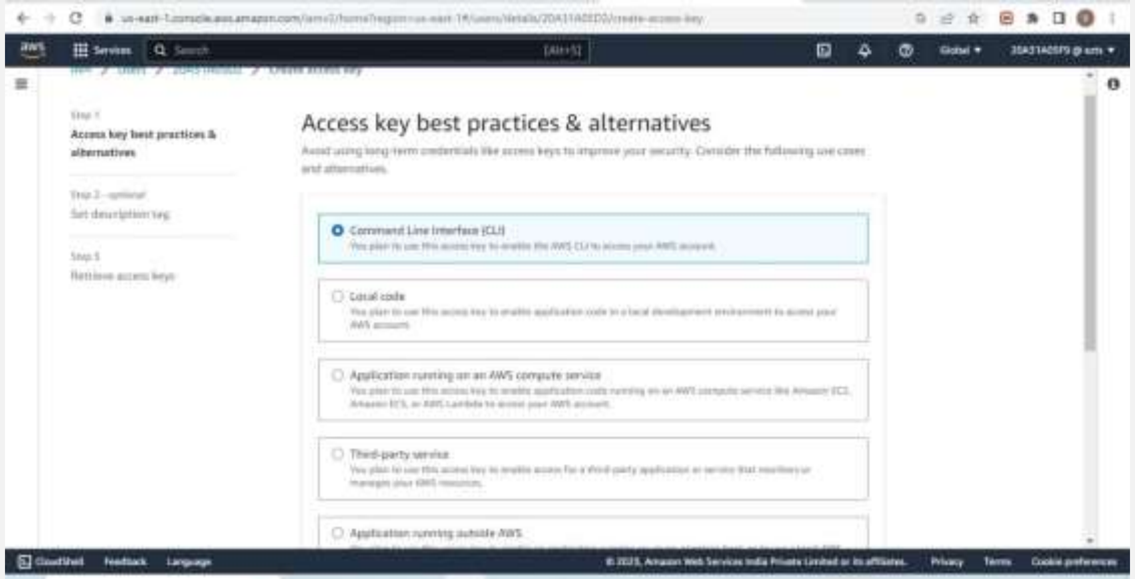
Created: 7 days ago

Last used region: us-east-1

Last used service: iam

SSH public keys for AWS CodeCommit (0)

Use SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. Learn more


Services
Search
[Alt+S]
Global ▾
20A31A05F9 @ ezts ▾
IAM > Users > 20A31A05D2 > Create access key
CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

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.

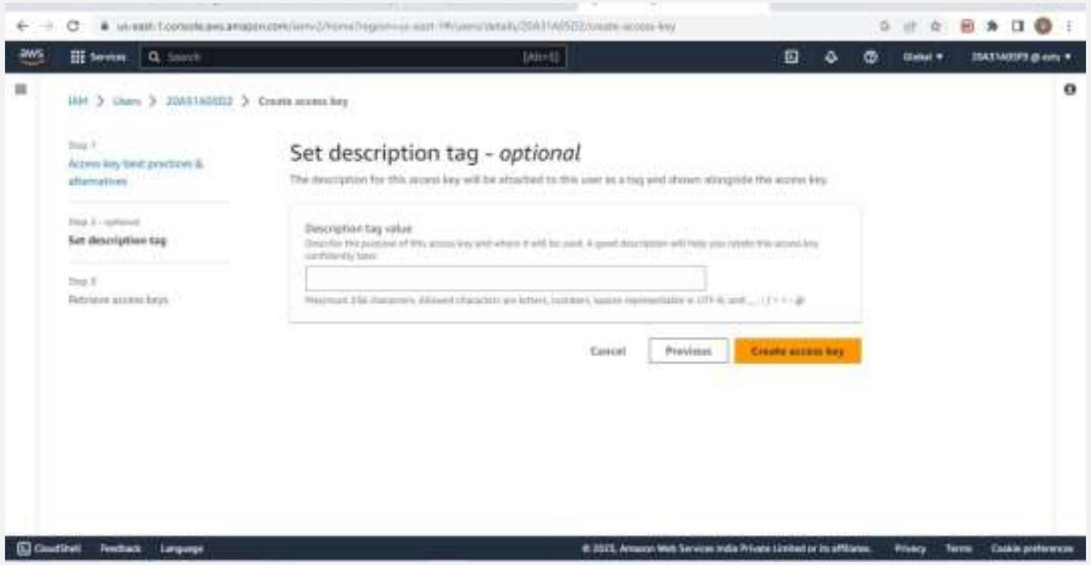
● Command Line Interface (CLI)
You plan to use this access key to enable the AWS CLI to access your AWS account.

○ 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



IAM > Users > 20A31A05D2 > Create access key

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

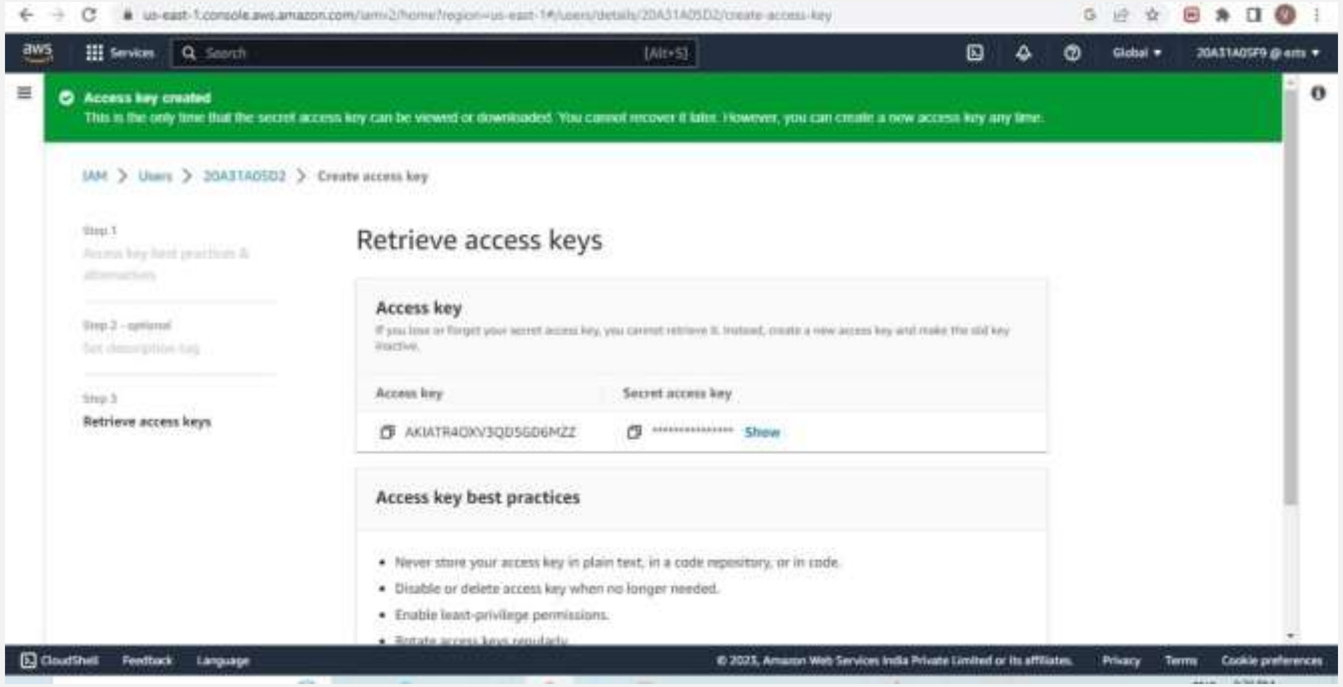
Step 3
Retrieve access keys

## 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: _ . : / = + - @



us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/users/details/20A31A05D2/create-access-key

**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.

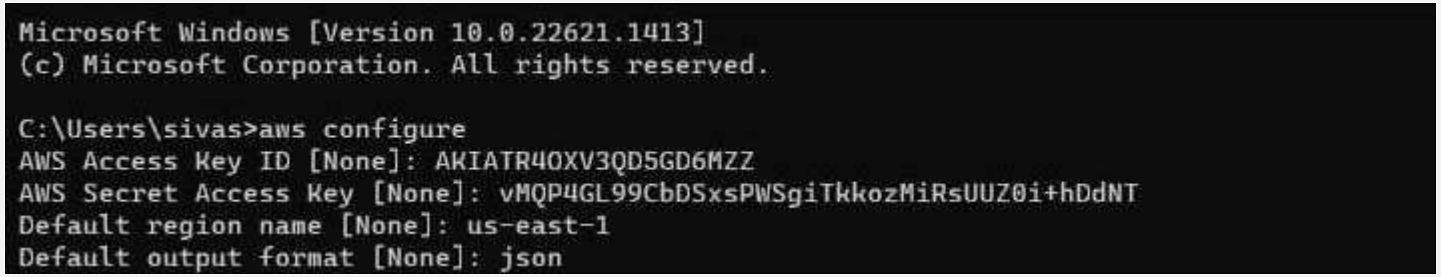
STEP 6 – Now you can use this access key to configure CLI

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)



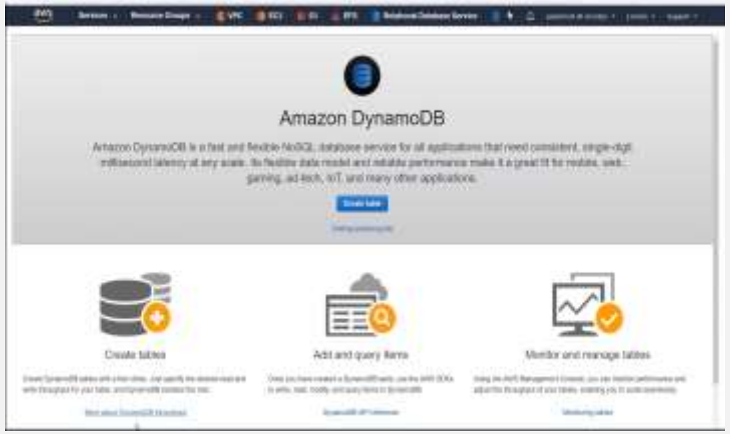
```
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sivas>aws configure
AWS Access Key ID [None]: AKIATR4OXV3QD5GD6MZZ
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
```

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

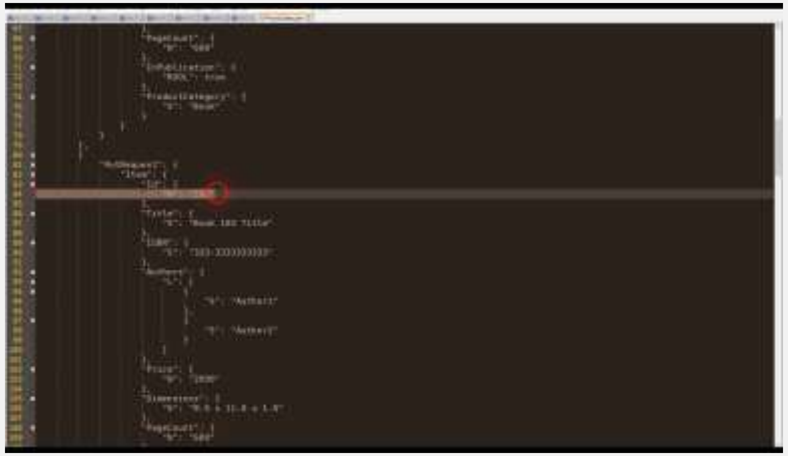
# AWS DYNAMO DB

- 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:



Amazon DynamoDB

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, IoT, and many other applications.



```
                "PageCount": {
                    "N": "500"
                },
                "InPublication": {
                    "BOOL": true
                },
                "ProductCategory": {
                    "S": "Book"
                }
            }
        }
    },
    {
        "PutRequest": {
            "Item": {
                "Id": {
                    "N": "102"
                },
                "Title": {
                    "S": "Book 102 Title"
                },
                "ISBN": {
                    "S": "333-3333333333"
                },
                "Authors": {
                    "L": [
                        {
                            "S": "Author1"
                        },
                        {
                            "S": "Author2"
                        }
                    ]
                },
                "Price": {
                    "N": "2000"
                },
                "Dimensions": {
                    "S": "8.5 x 11.0 x 1.8"
                },
                "PageCount": {
                    "N": "600"
```

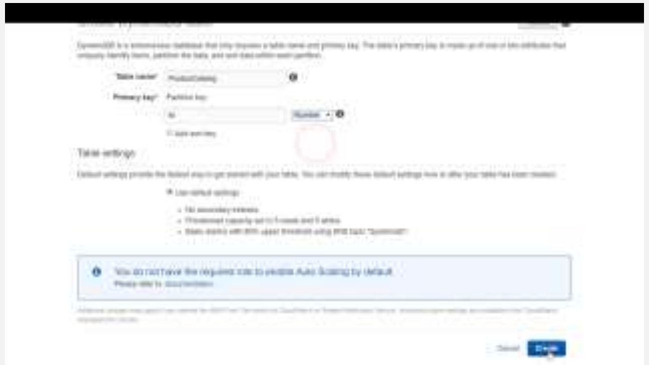


Table name* ProductCatalog

Primary key* Partition key

id Number

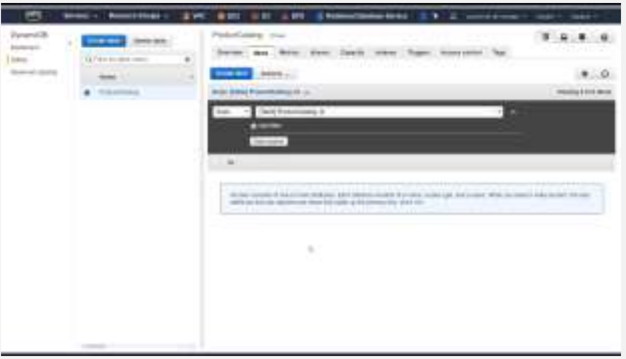
Table settings

● Use default settings

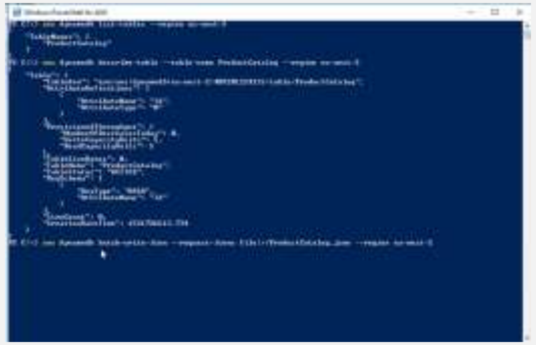
- No secondary indexes.
- Provisioned capacity set to 5 reads and 5 writes.

You do not have the required role to enable Auto Scaling by default.
Please refer to documentation

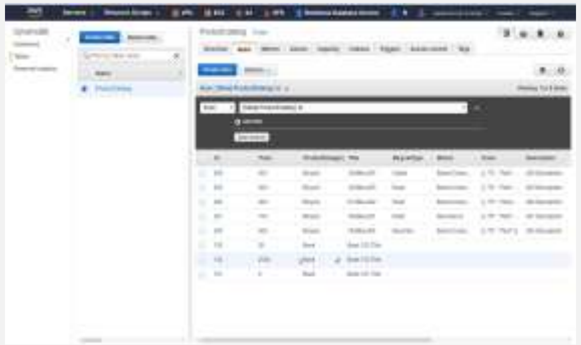
- After creating the table , we can see that

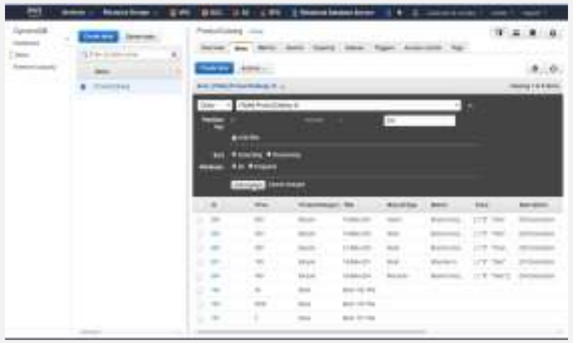


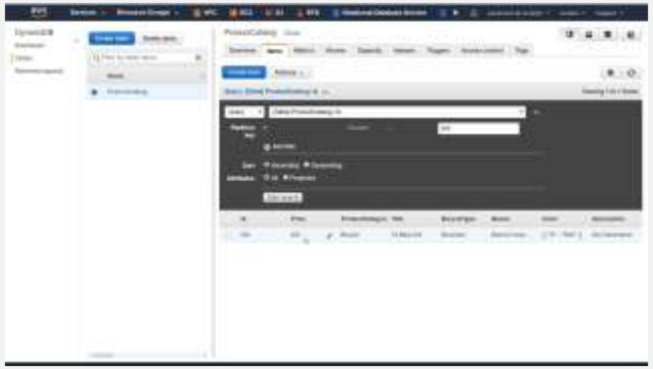
- So we will use the CLI to populate the table. Open powershell of AWS.



```
PS C:\> aws dynamodb list-tables --region us-east-1
{
    "TableNames": [
        "ProductCatalog"
    ]
}
PS C:\> aws dynamodb describe-table --table-name ProductCatalog --region us-east-1
{
    "Table": {
        [illegible]
    }
}
PS C:\> aws dynamodb batch-write-item --request-items file://ProductCatalog.json --region us-east-1
```







# AWS LIGHT SAIL

**PROCEDURE:**

On the home page, choose Create instance.

Select a location for your instance (an AWS Region and Availability Zone).Choose Change Region and zone to create your instance in another location.

Optionally, you can change the Availability Zone.Choose an Availability Zone from the dropdown list.

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:**

Must be unique within each AWS Region in your Lightsail account.

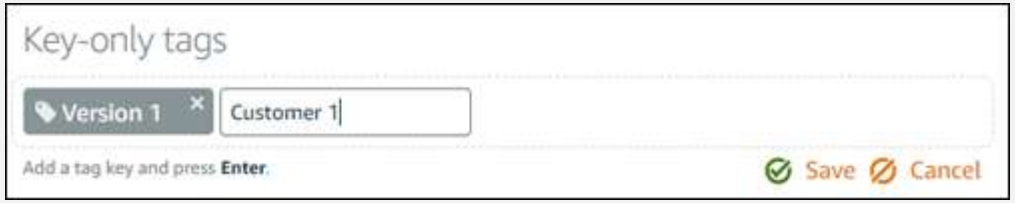
Must contain 2 to 255 characters.

Must start and end with an alphanumeric character or number.

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.



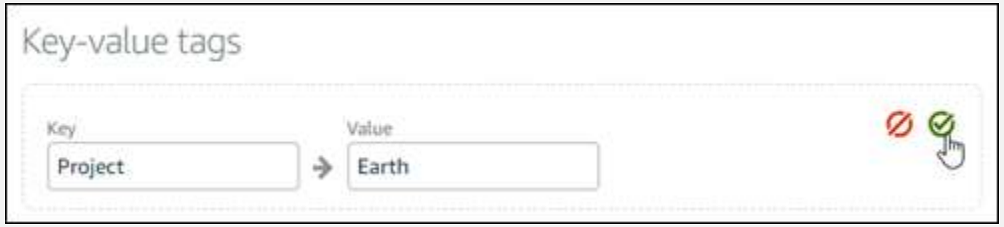
Key-only tags

Version 1 ×   Customer 1

Add a tag key and press **Enter**.

✓ Save ⊘ Cancel

- 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.



Key-value tags

Key: Project → Value: Earth

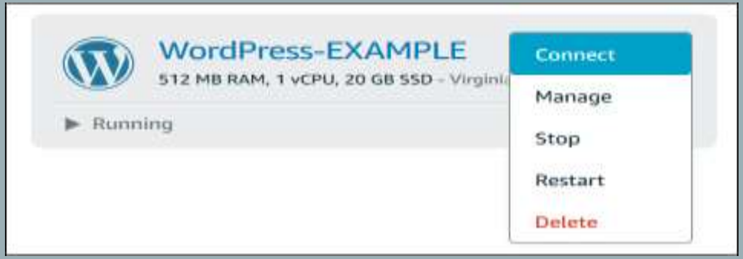
⊘ ✓

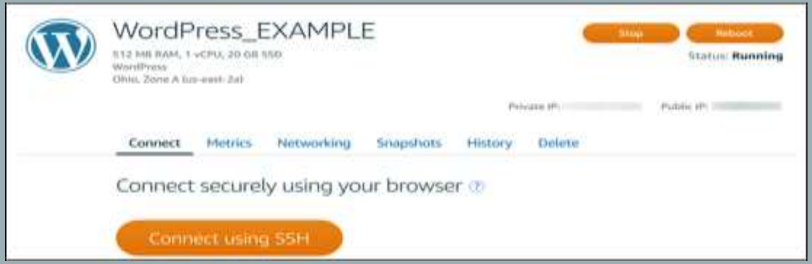
8. Choose Create instance.

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

# How to connect to your instance

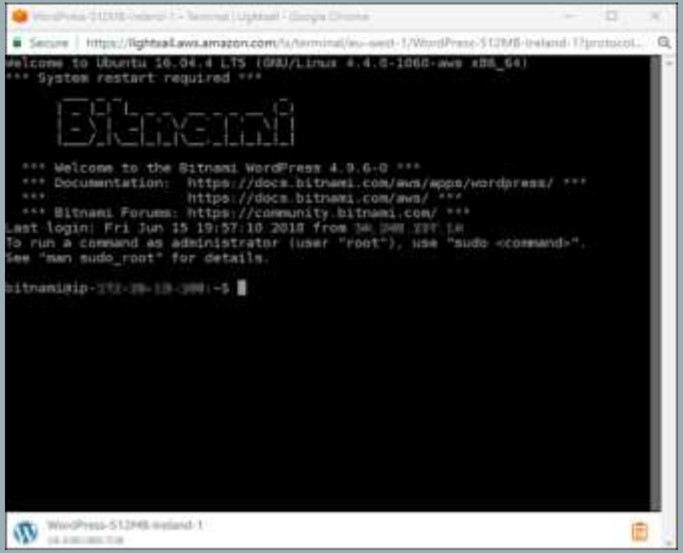
1. From the Lightsail home page, choose the menu on the right of your instance's name, and then choose connect.





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.


Secure | https://lightsail.aws.amazon.com/ls/terminal/eu-west-1/WordPress-512MB-Ireland-1?protocol...
WordPress-512MB-Ireland-1

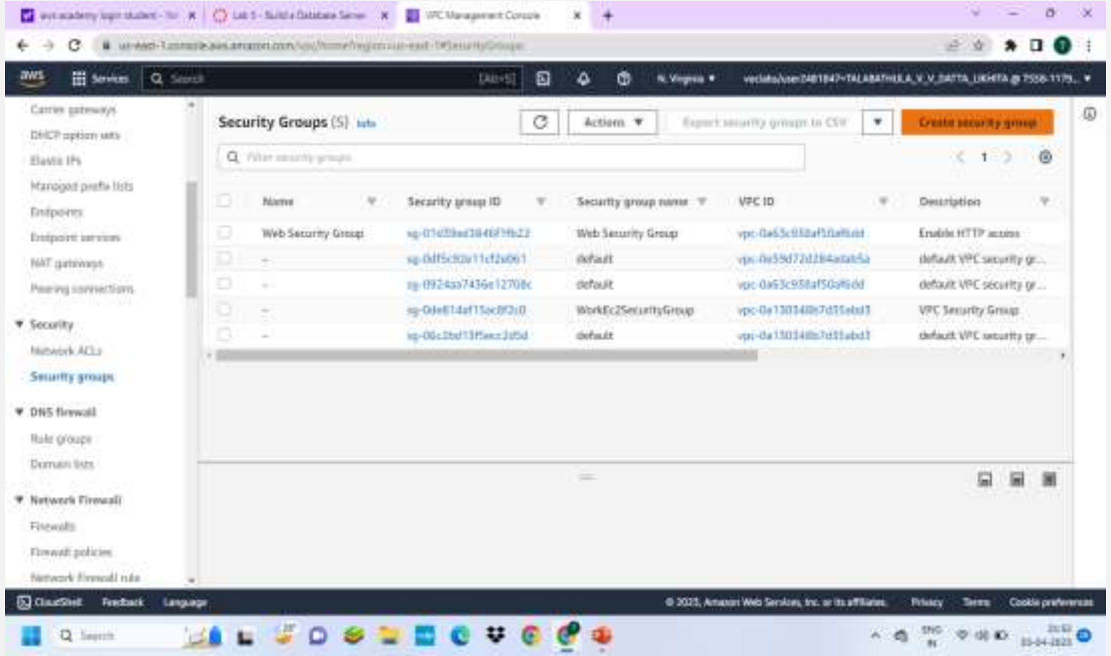
```
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1060-aws x86_64)
*** System restart required ***

  *** Welcome to the Bitnami WordPress 4.9.6-0 ***
  *** Documentation:  https://docs.bitnami.com/aws/apps/wordpress/ ***
  ***                 https://docs.bitnami.com/aws/ ***
  *** Bitnami Forums: https://community.bitnami.com/ ***
Last login: Fri Jun 15 19:57:10 2018 from [redacted]
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

bitnami@ip-[redacted]:~$
```

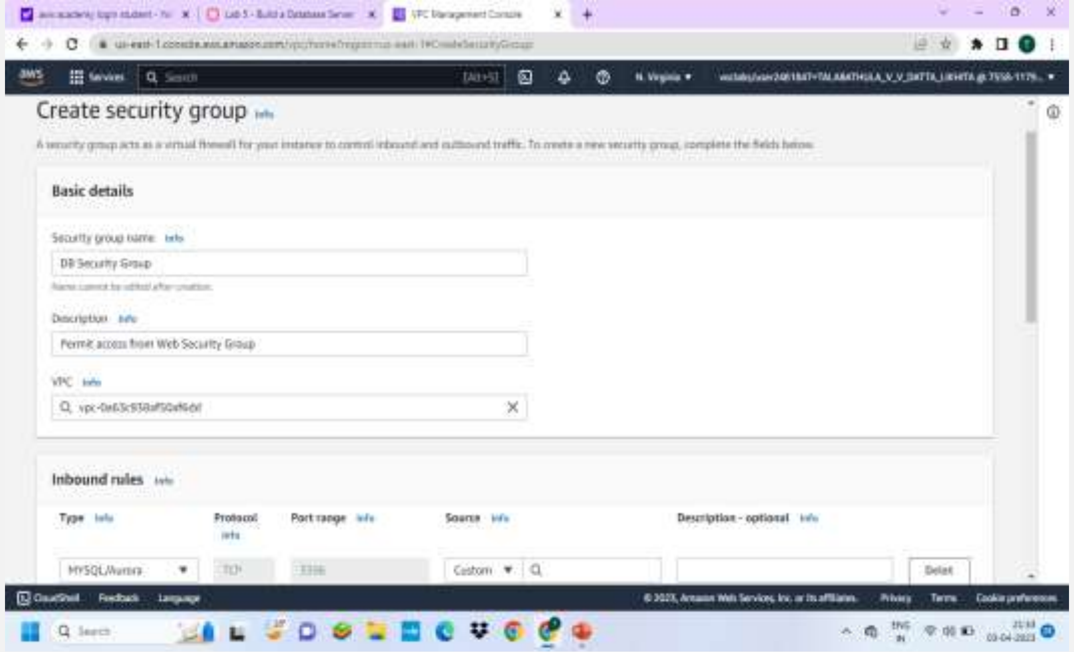
# AWS RDS

**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.



Security Groups (5)

|  | Name | Security group ID | Security group name | VPC ID | Description |
|---|---|---|---|---|---|
| □ | Web Security Group | sg-01d39ad384bf1fb22 | Web Security Group | vpc-0a63c938af50af6dd | Enable HTTP access |
| □ | – | sg-0df5c92e11cf2a061 | default | vpc-0e59d72d284adab5a | default VPC security gr... |
| □ | – | sg-0924aa7436e12708c | default | vpc-0a63c938af50af6dd | default VPC security gr... |
| □ | – | sg-0de614af15ac8f2c0 | WorkEc2SecurityGroup | vpc-0a130348b7d35abd3 | VPC Security Group |
| □ | – | sg-06c2bd13f5ecc2d5d | default | vpc-0a130348b7d35abd3 | default VPC security gr... |



Create security group

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

Basic details

Security group name: DB Security Group

Name cannot be edited after creation.

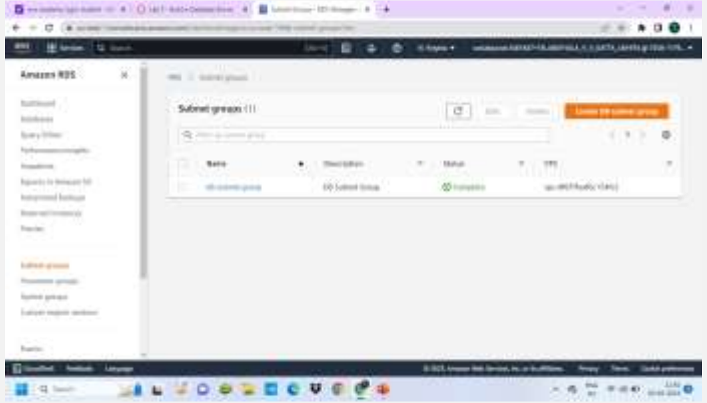
Description: Permit access from Web Security Group

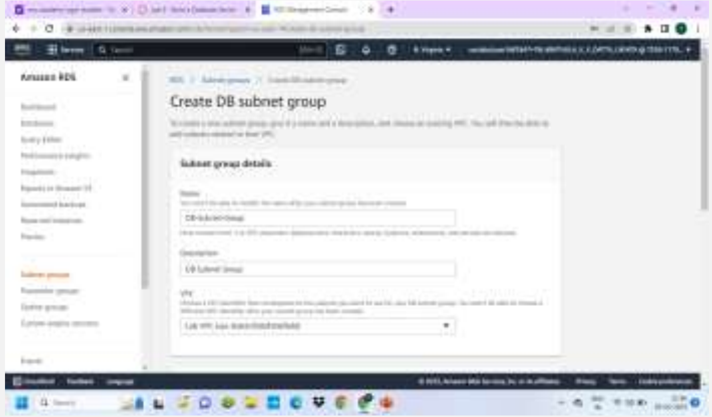
VPC: vpc-0a63c938af50af6dd

Inbound rules

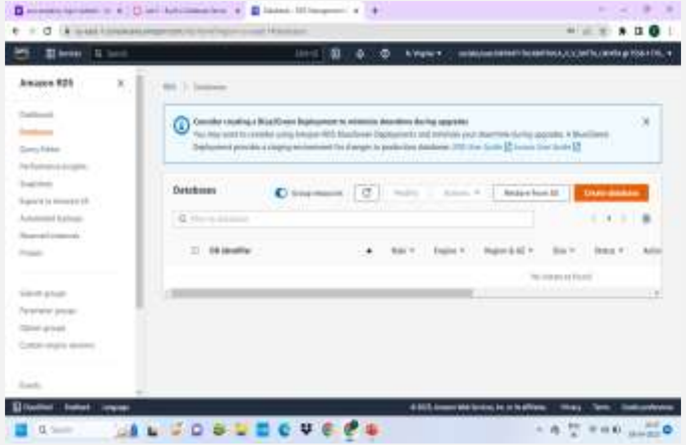
| Type | Protocol | Port range | Source | Description - optional |
|---|---|---|---|---|
| MYSQL/Aurora | TCP | 3306 | Custom |  |

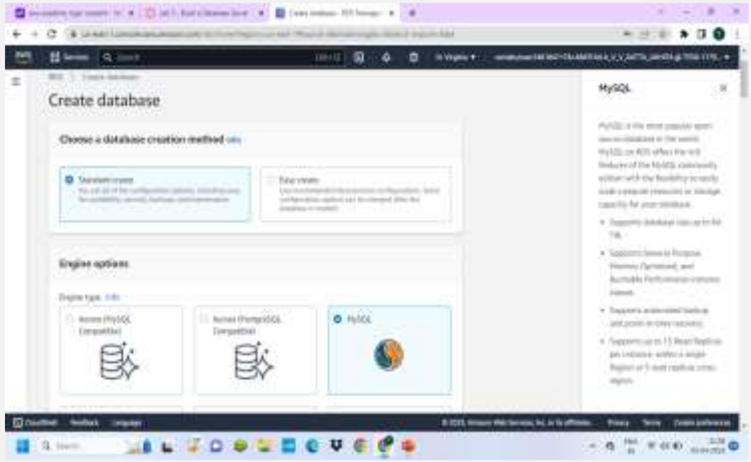
**Step 2 :** Create a DB Subnet Group.
Rds → subnet groups → choose create DB subnet group → add subnets → create DB subnet group.



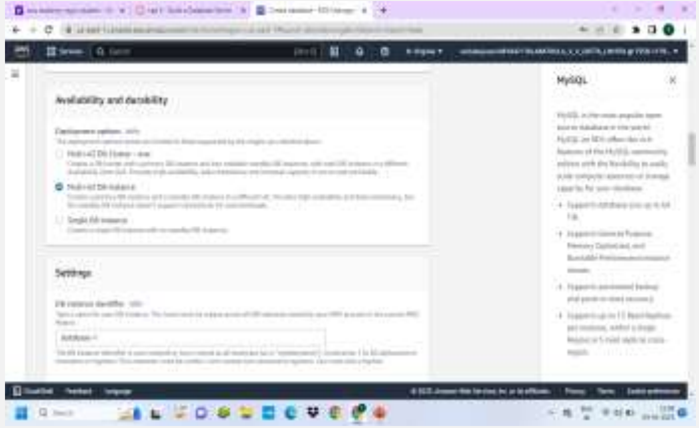


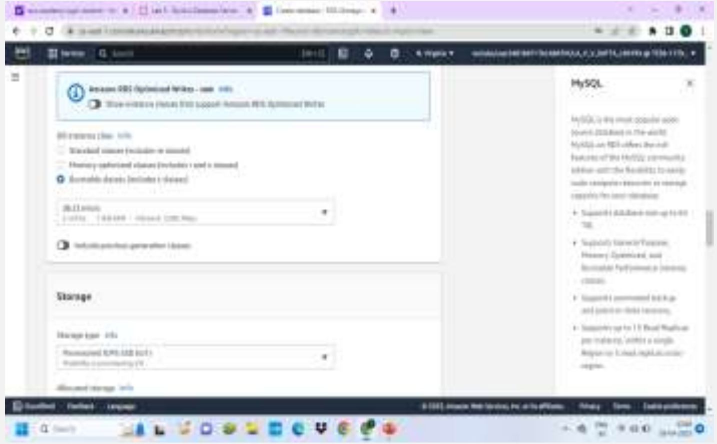
**Step 3**: In the left navigation pane, choose **Databases** → choose create database → MYSQL



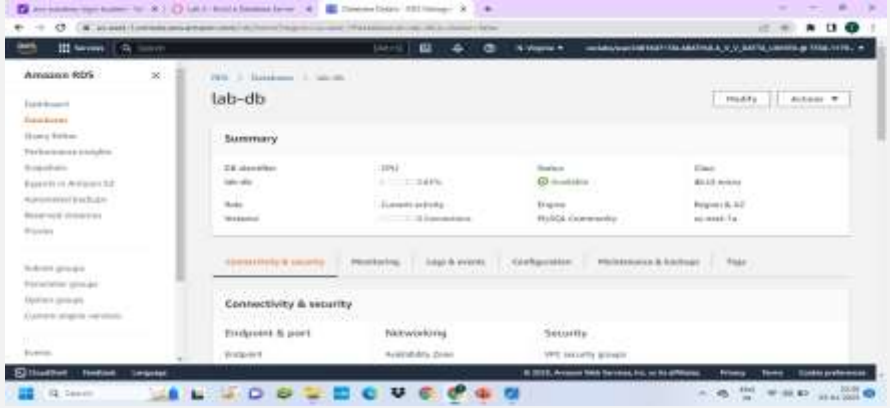


**Step 4**: In Availability and durability ,choose Multi –AZ DB instance then configure settings , DB instance class, Storage, connectivity, choose existing vpc security group and setup additional configuration.





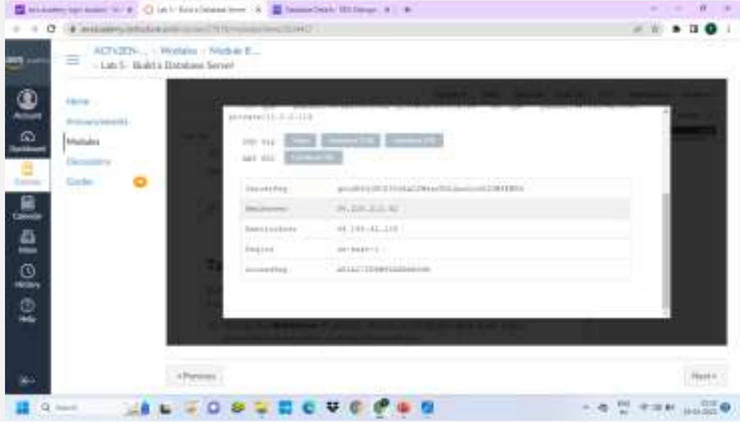
**Step 5:** Wait until Info changes to Modifying or Available.
Scroll down to the Connectivity & security section and copy the **Endpoint** field.

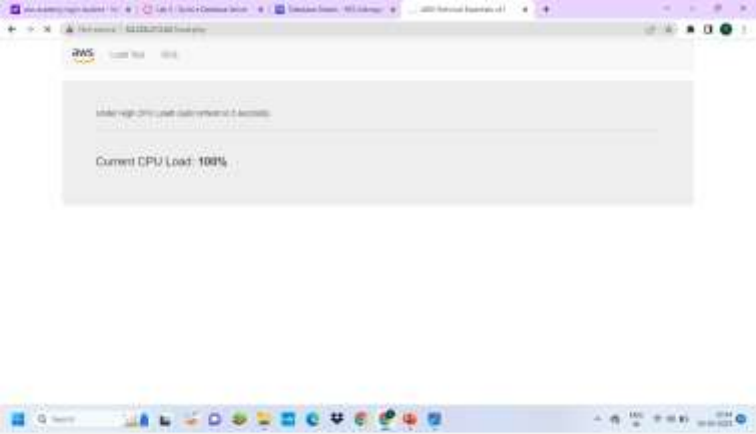


**Step 6** : Interact with Your Database.

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

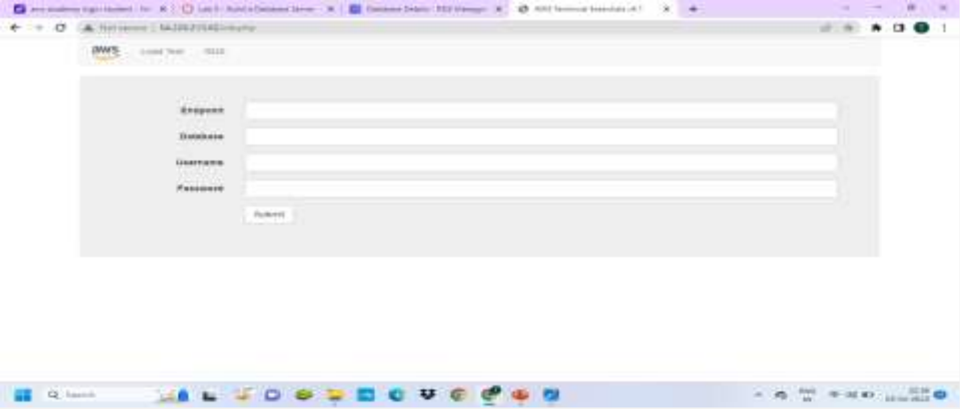
The web application will be displayed, showing information about the EC2 instance.





Current CPU Load: **100%**

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



Endpoint

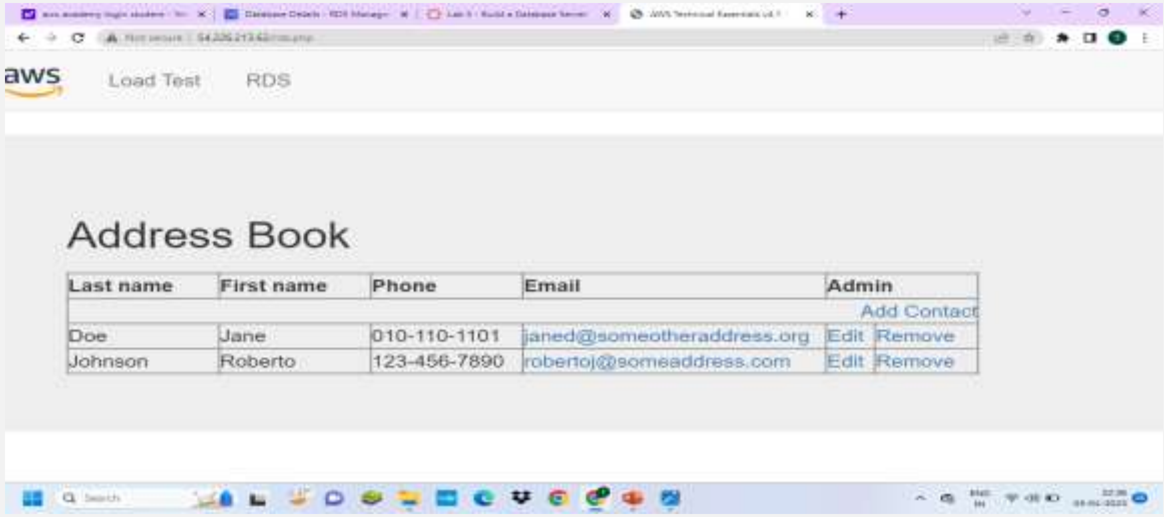
Database

Username

Password

Submit

**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.



Address Book

| Last name | First name | Phone | Email | Admin |
|---|---|---|---|---|
| | | | | Add Contact |
| Doe | Jane | 010-110-1101 | janed@someotheraddress.org | Edit Remove |
| Johnson | Roberto | 123-456-7890 | robertoj@someaddress.com | Edit Remove |

# 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.

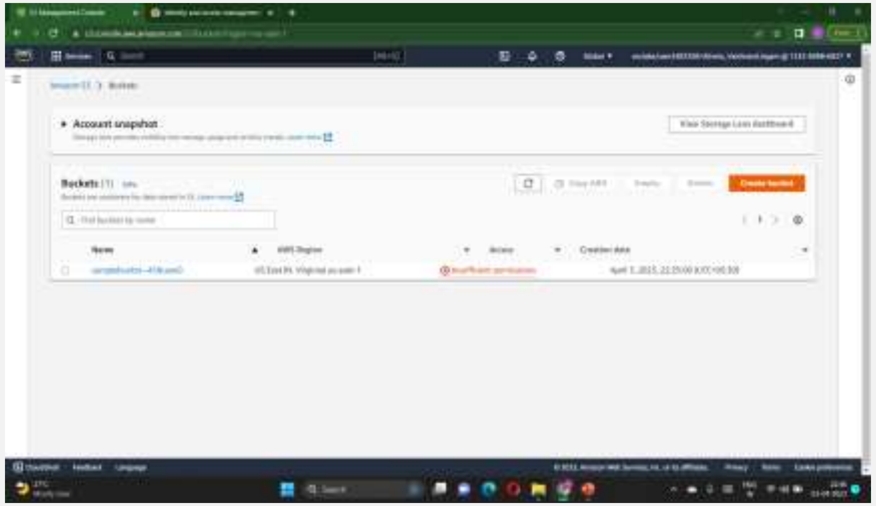
**STEPS :**

**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.

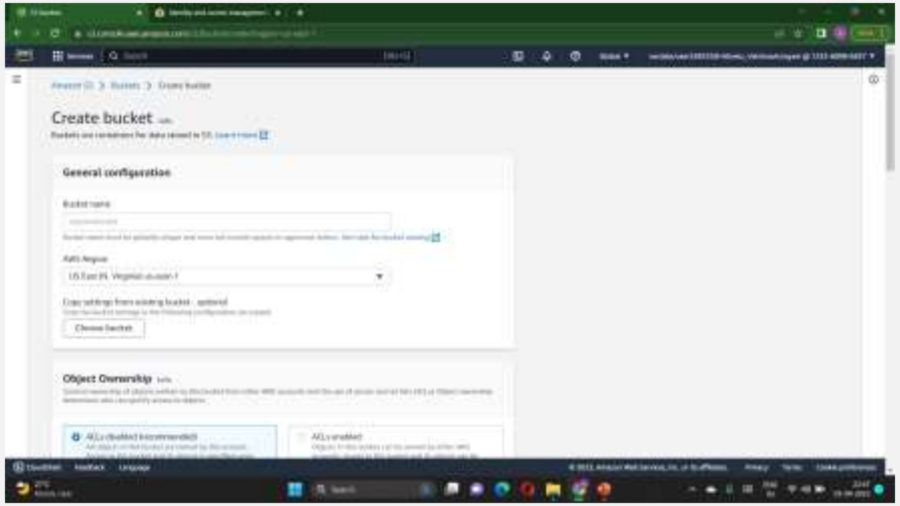
**Step 5:** Click on Add Files , and choose a file from your computer.
**Step 6:** After choosing your file, click on Next.
**Step 7:** Click on Upload.
**Step 8:**Now you have a private S3 bucket with a private object uploaded, which means you cannot visit it through Internet.
**Step 9:**Now you have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

CHANGE BUCKET PERMISSIONS:

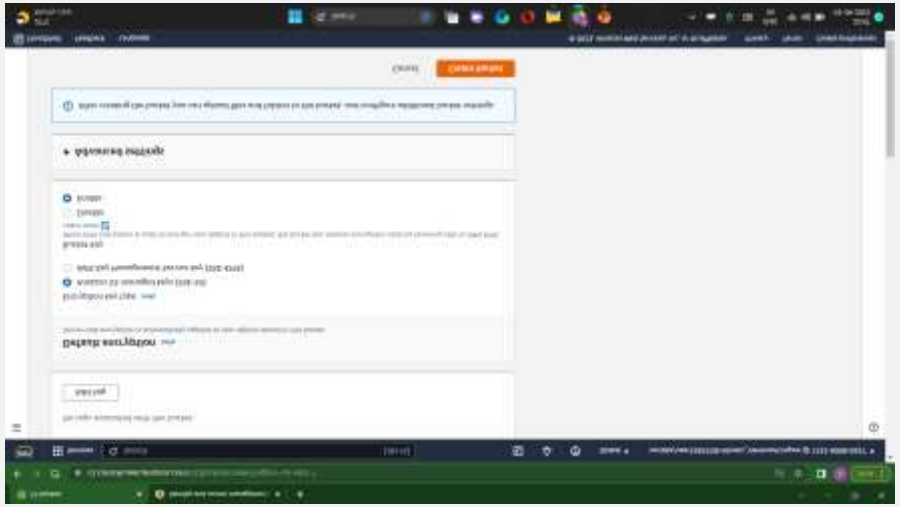
**Step 10:**Go back to your bcket and click on Permissions.
**Step 11:**Click on Everyone under the Public access, and click on Read object on the right of pop-up window. Then click on Save.
**Step 12 :**Now its state switches to Read Object - Yes
**Step 13:**Click on Overview, and click on your Object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



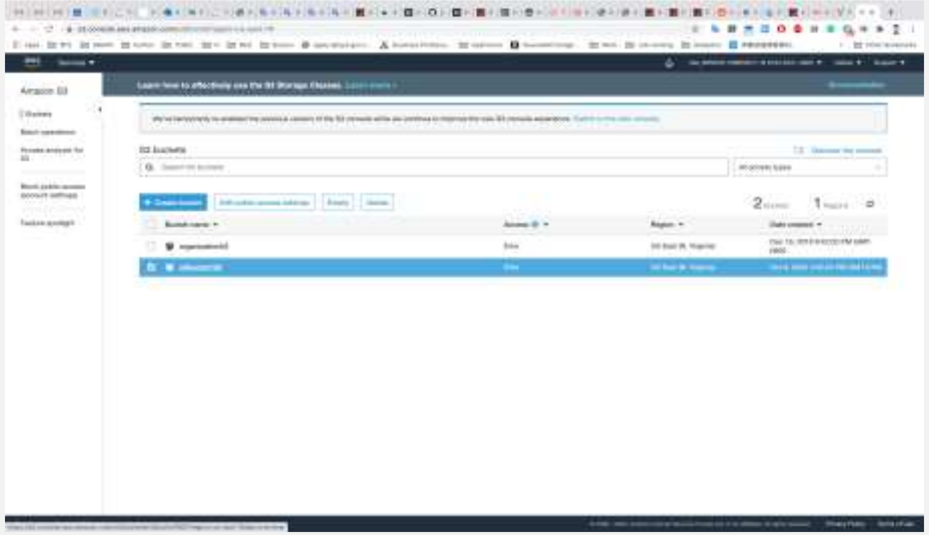
## Step 1



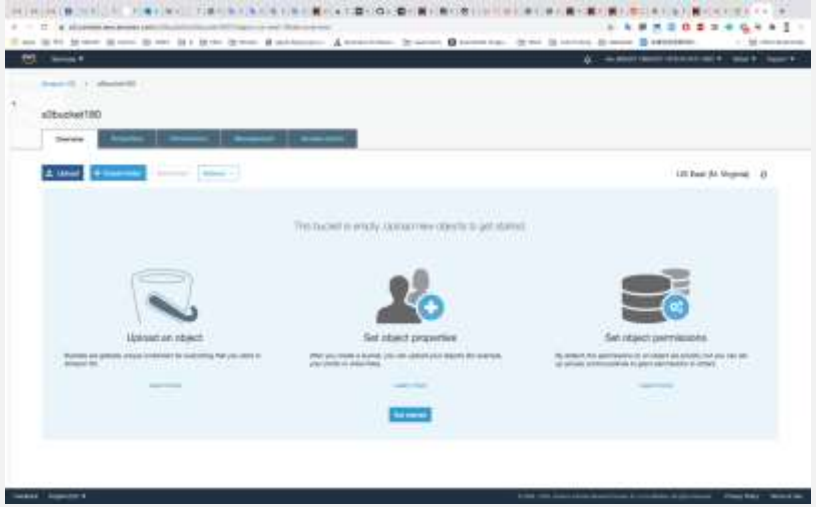
## Step 2



## Step 2

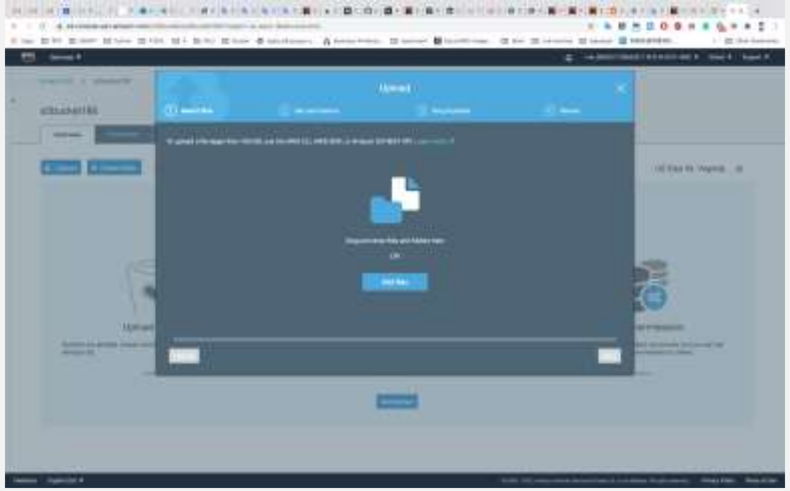


## Step 3

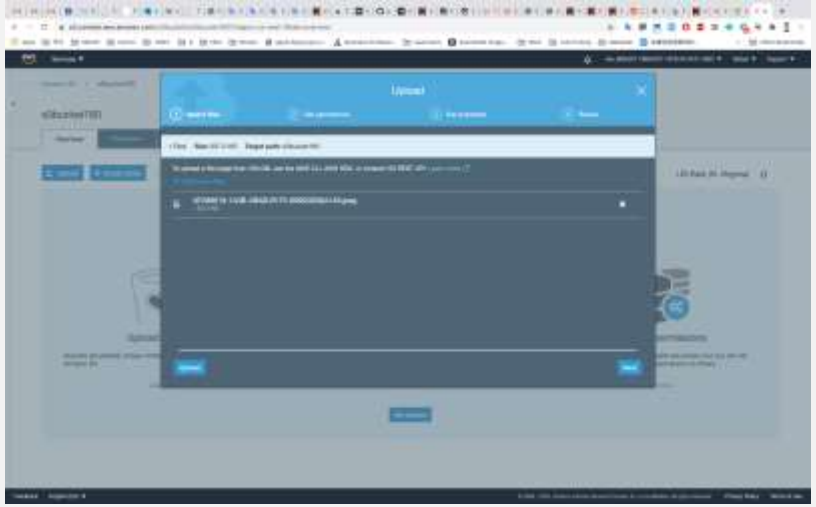


This bucket is empty. Upload new objects to get started.

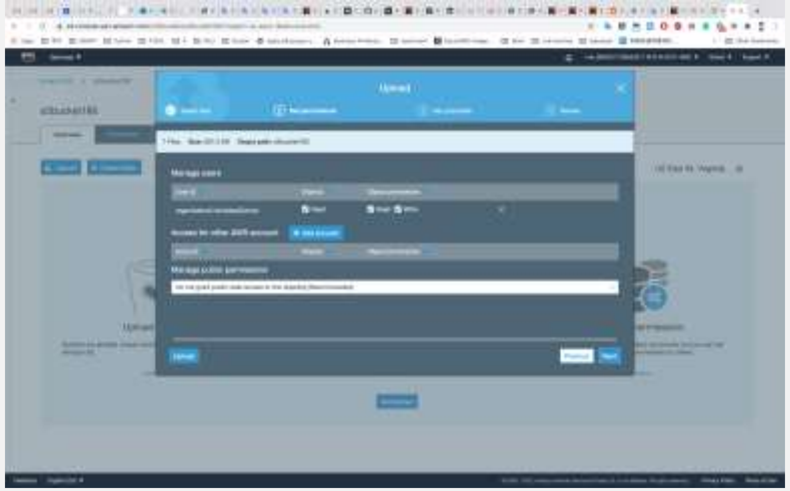
**Step 4**



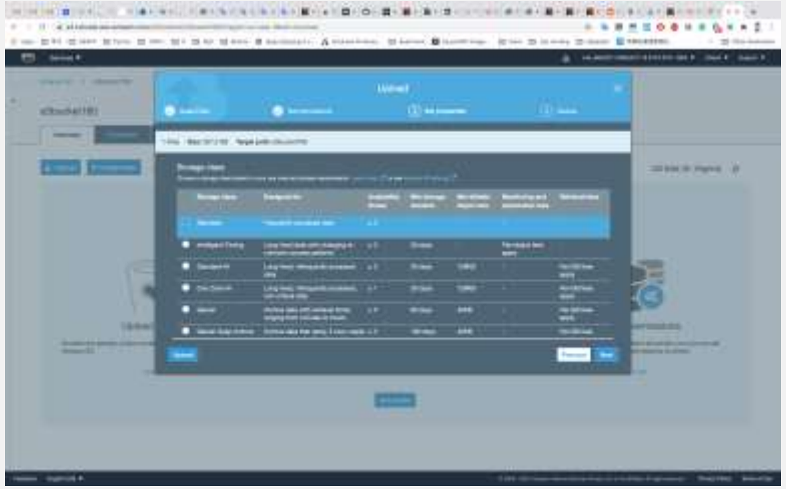
**Step 5**



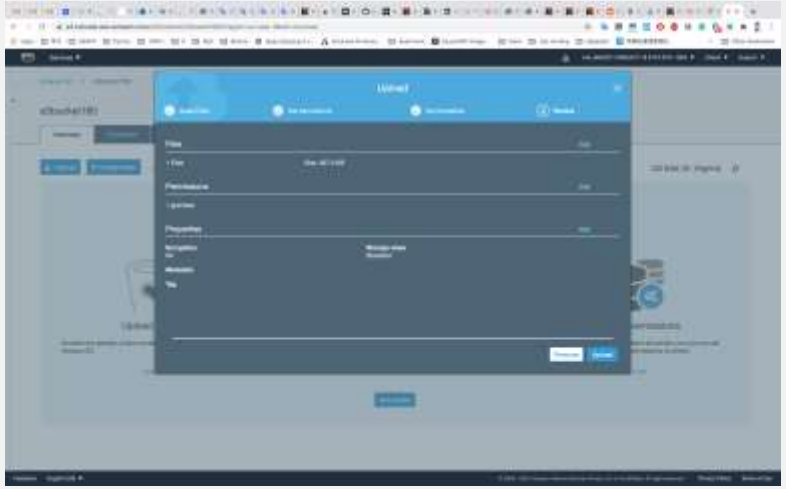
**Step 6**



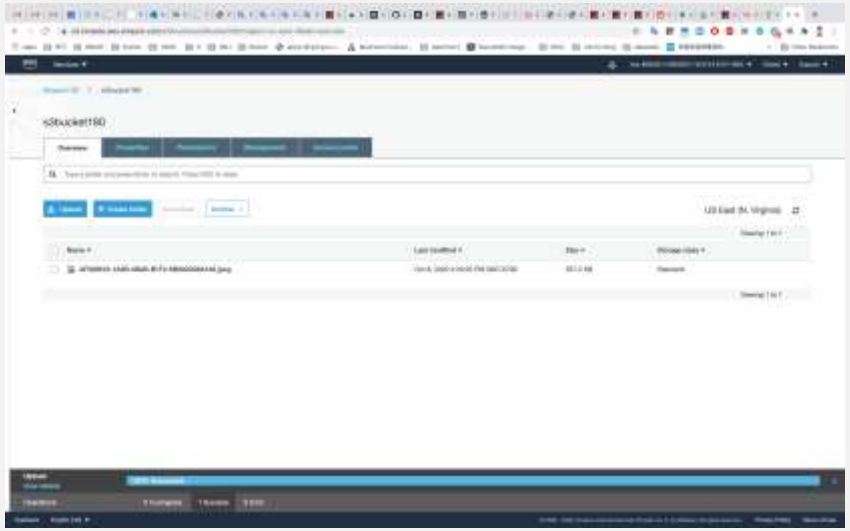
**Step 7**



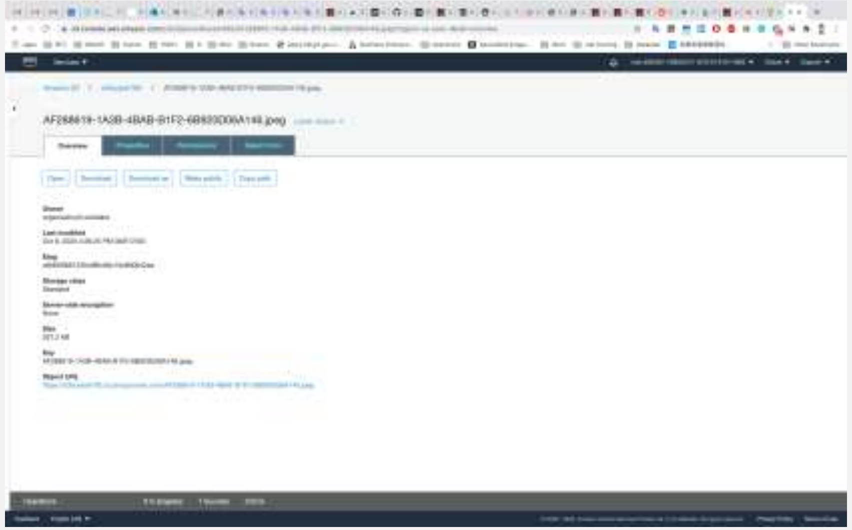
**Step 8**



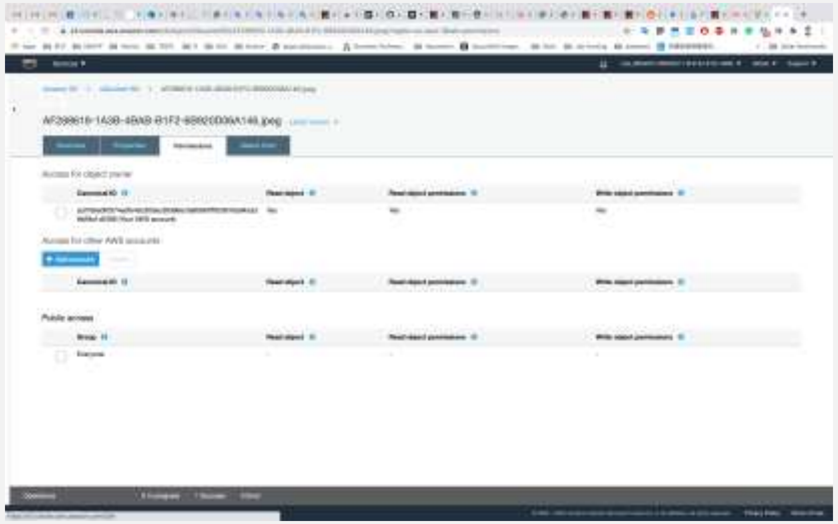
**Step 9**



**Step 10**


AF268619-1A3B-4BAB-B1F2-6B820D06A149.jpeg

**Step 11**



AF299619-1A3B-4BA9-B1F2-6B920D06A146.jpeg

Access for object owner

| | Canonical ID | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|
| ○ | [illegible] (Your AWS account) | Yes | Yes | Yes |

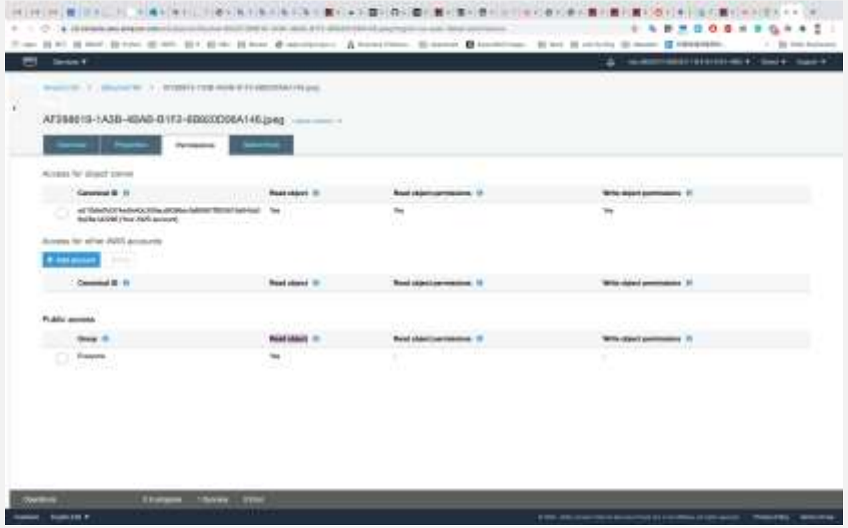
Access for other AWS accounts

+ Add account

| | Canonical ID | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|

Public access

| | Group | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|
| ○ | Everyone | - | - | - |



AF298619-1A3B-40A0-B1F3-6B920D06A146.jpeg

Access for object owner

| | Canonical ID | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|
| ○ | [illegible] (Your AWS account) | Yes | Yes | Yes |

Access for other AWS accounts

+ Add account

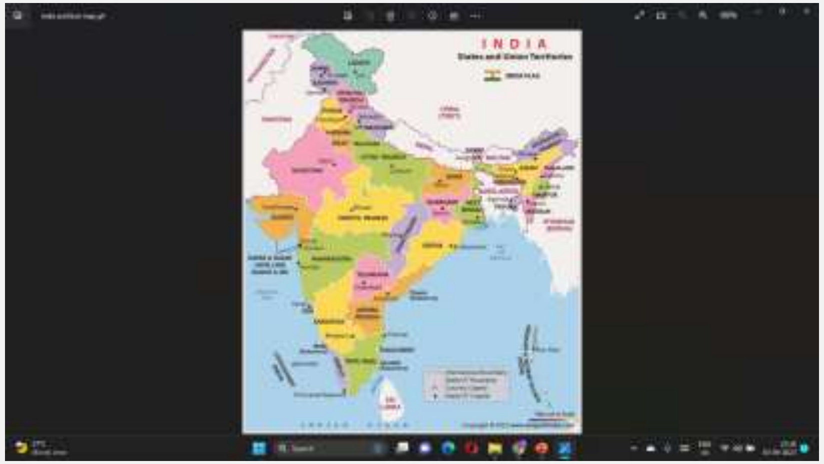
| | Canonical ID | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|

Public access

| | Group | Read object | Read object permissions | Write object permissions |
|---|---|---|---|---|
| ○ | Everyone | Yes | - | - |

**Step 12**

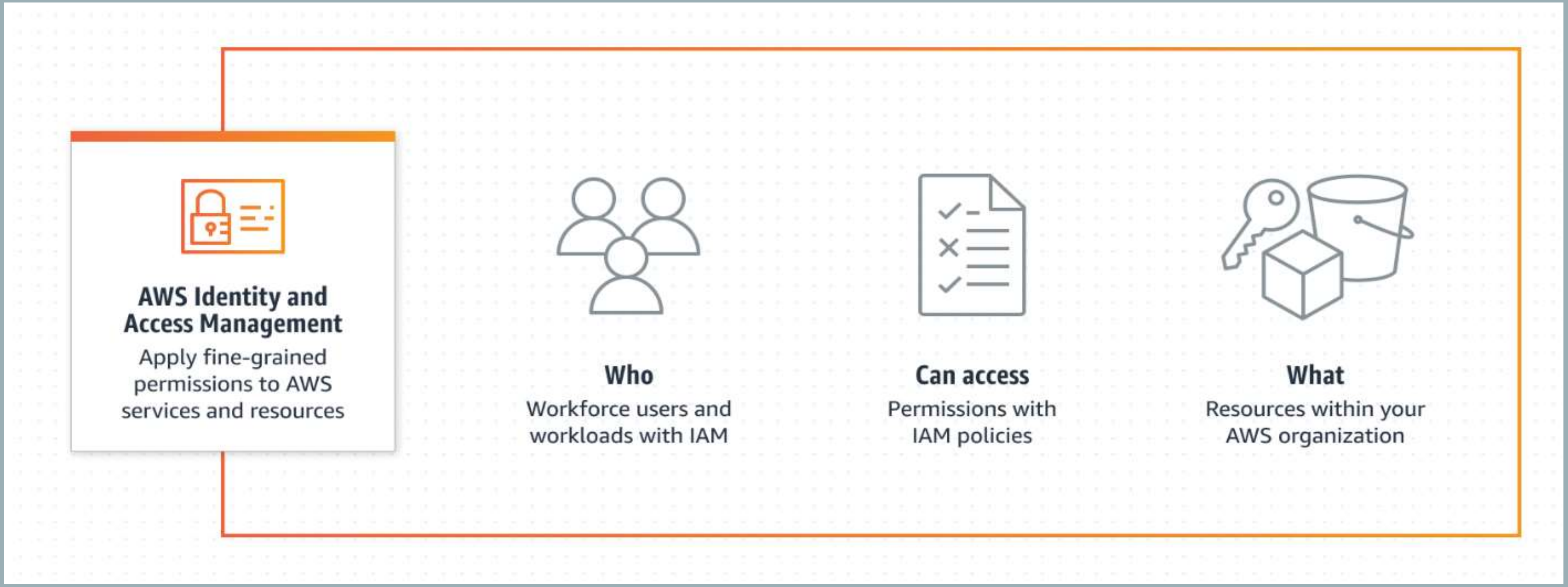
**Step 13**



**Step 14**

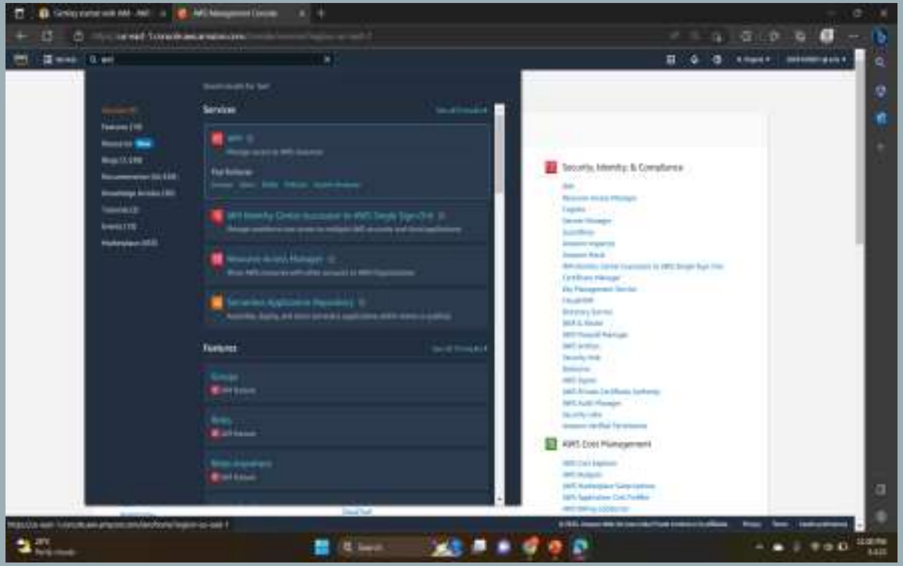
# AWS Identity and Access Management (IAM)

**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.

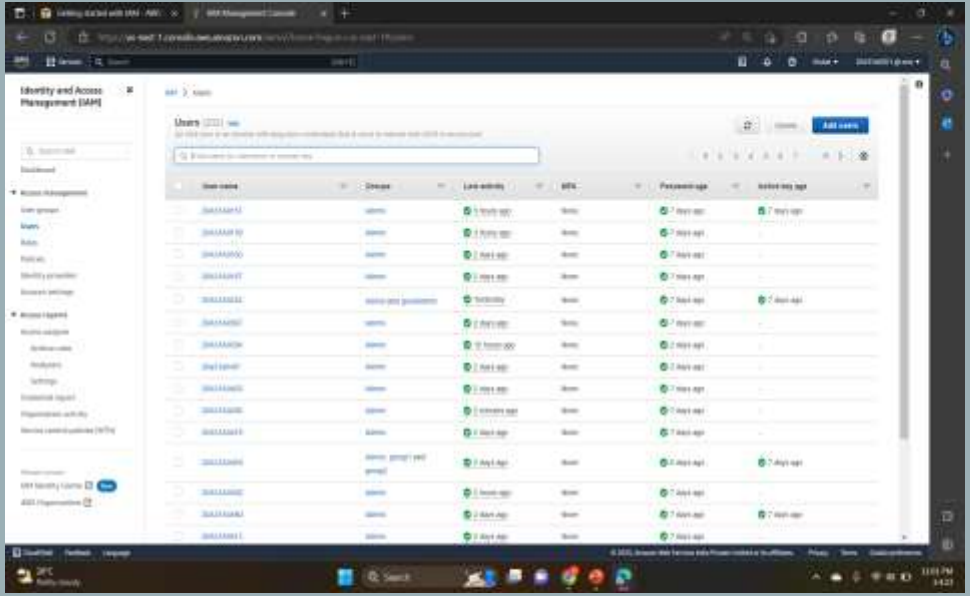

AWS Identity and Access Management
Apply fine-grained permissions to AWS services and resources
Who
Workforce users and workloads with IAM
Can access
Permissions with IAM policies
What
Resources within your AWS organization

# Steps to create IAM User and User Groups

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

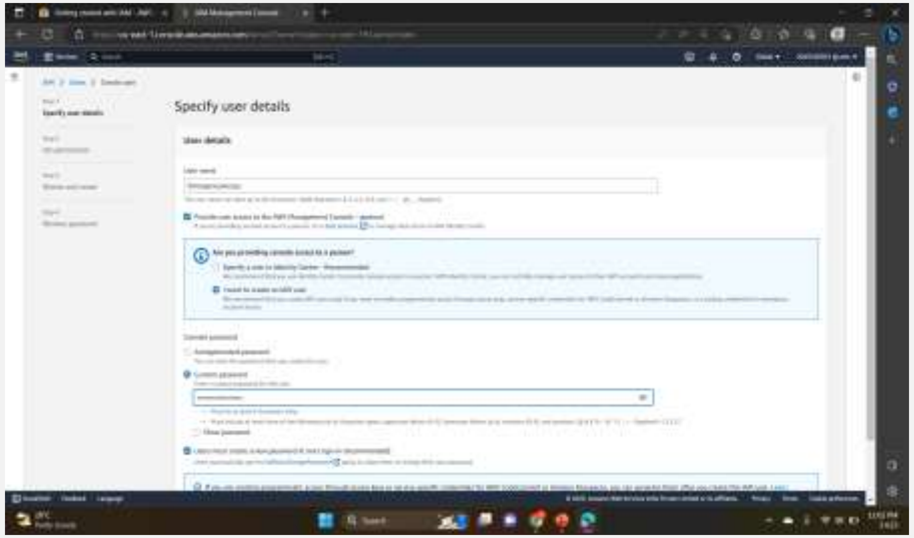


2.In the navigation pane, select **Users** and then select **Add users**.



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**.

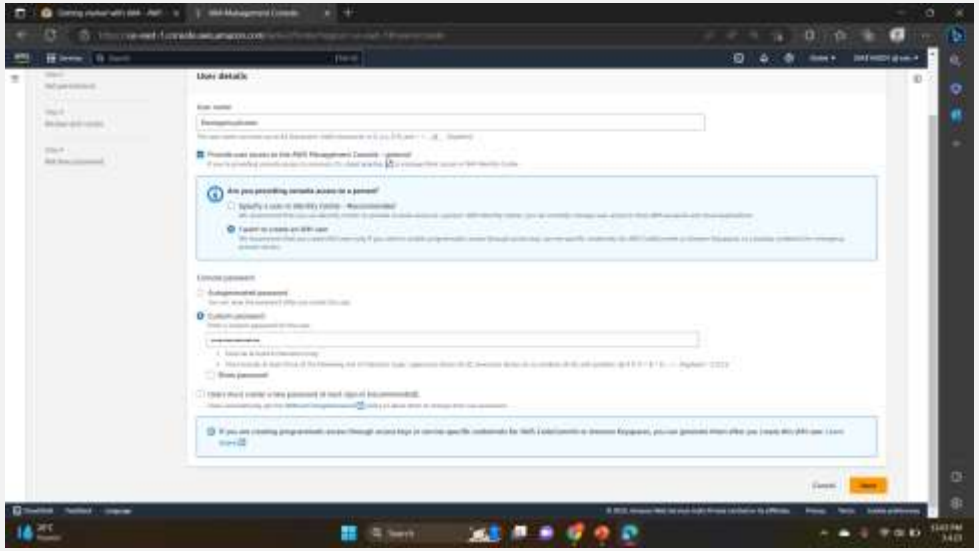
4.Under **Console password**, select **Custom Password** and create your own password.



Specify user details

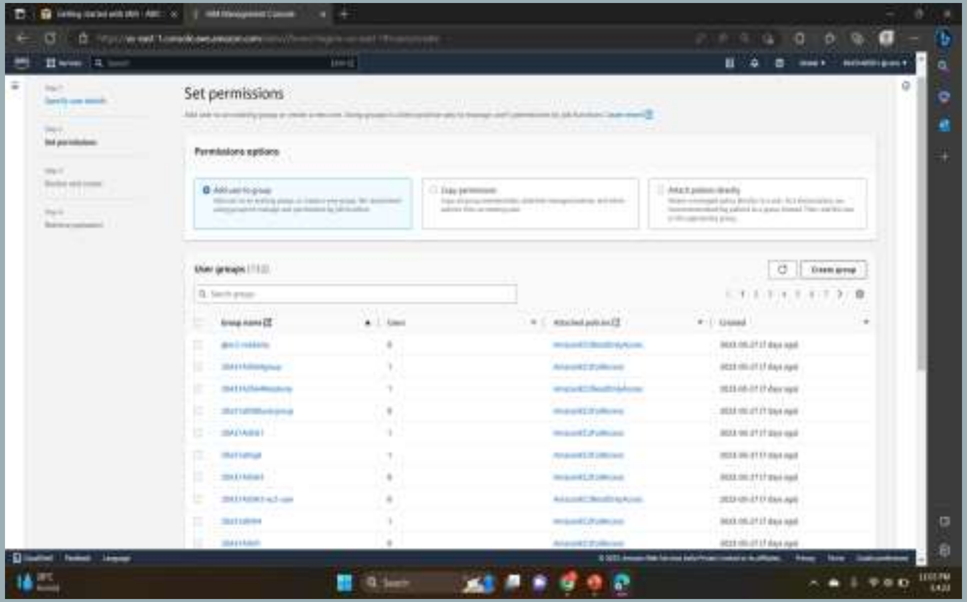
User details

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

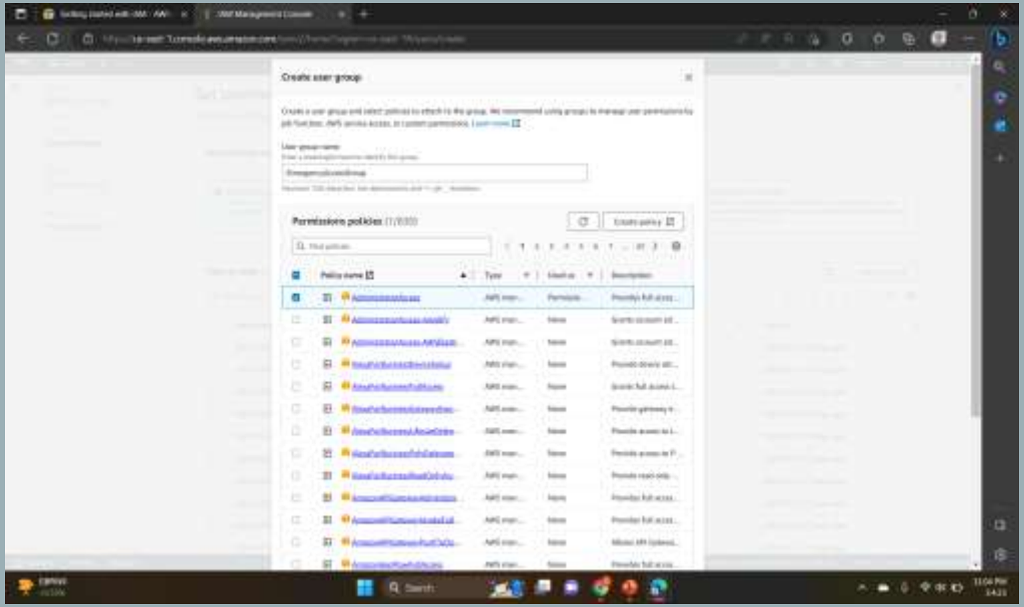


User details

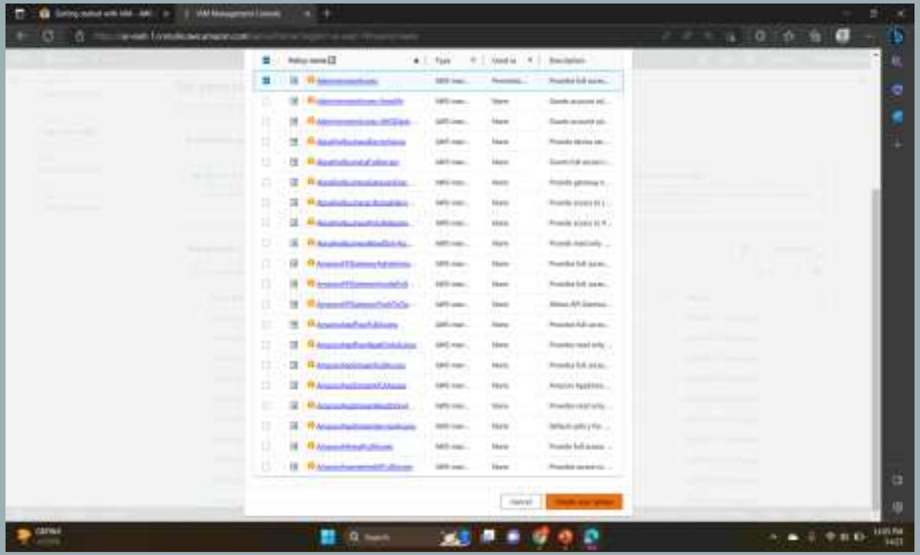
6. On the **Set permissions** page, under **Permissions options**, select **Add user to group**. Then, under **User groups**, select **Create group**.



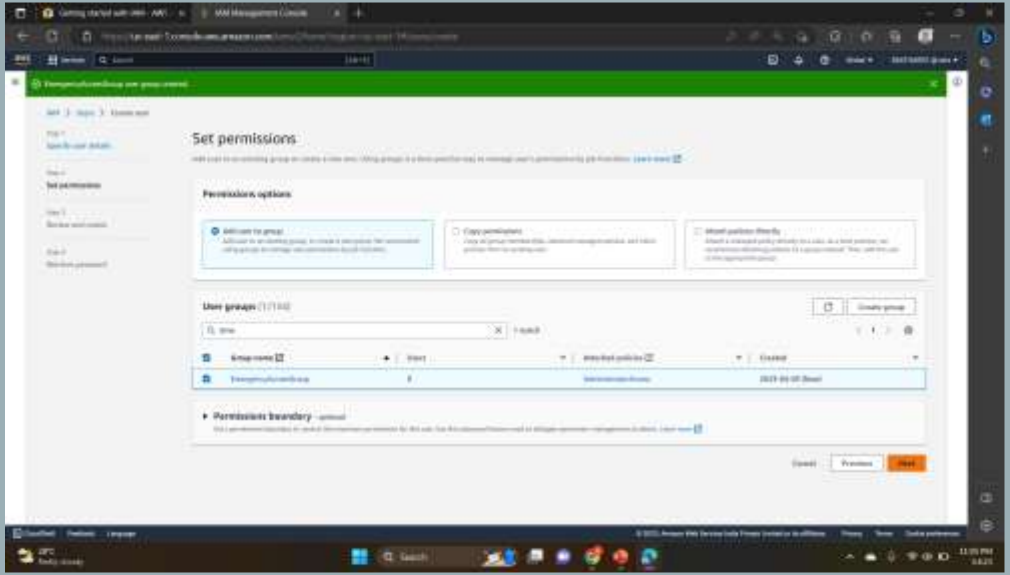
7. On the **Create user group** page, in **User group name**, enter **EmergencyAccessGroup**. Then, under **Permissions policies**, select **AdministratorAccess**.



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



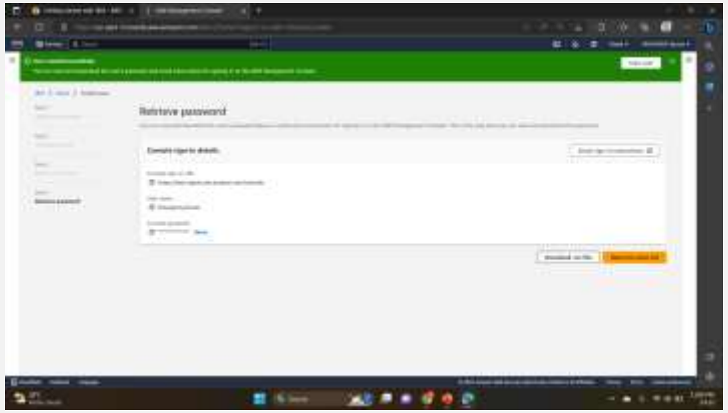
9. Select **Next** to proceed to the **Review and create** page.



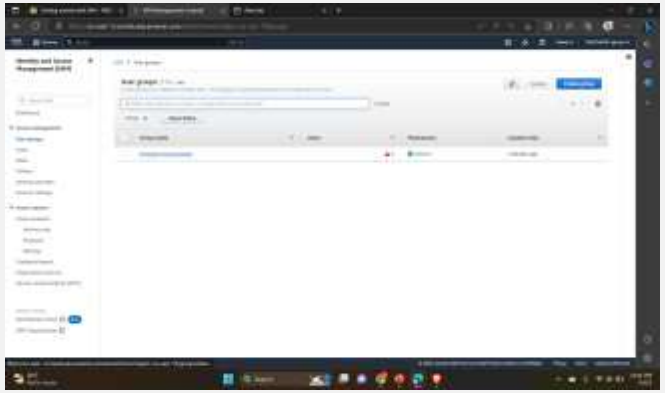
10.On the **Review and create** page, review the list of user group memberships to be added to the new user. When you are ready to proceed, select **Create user**.

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).

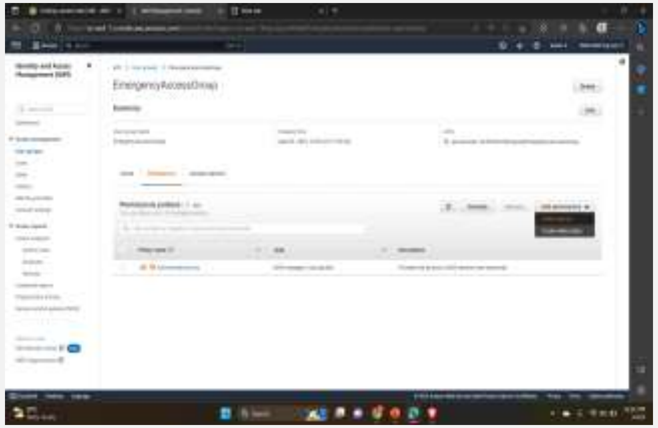
12. Save this file to use if you need to sign-in to IAM and do not have access to your federated identity provider.



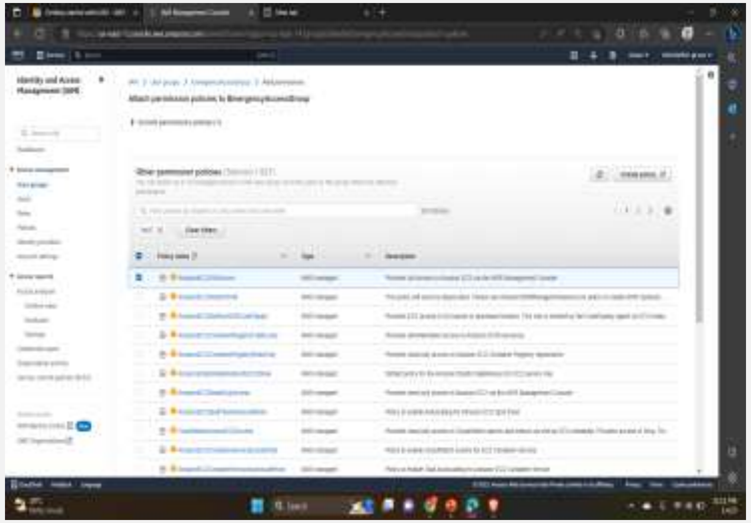
13. If you want to add permissions to the user group, then go to **User groups** and click on the respective **User group.**



**14. Go to Permissions → All permissions→ Attach policies**



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

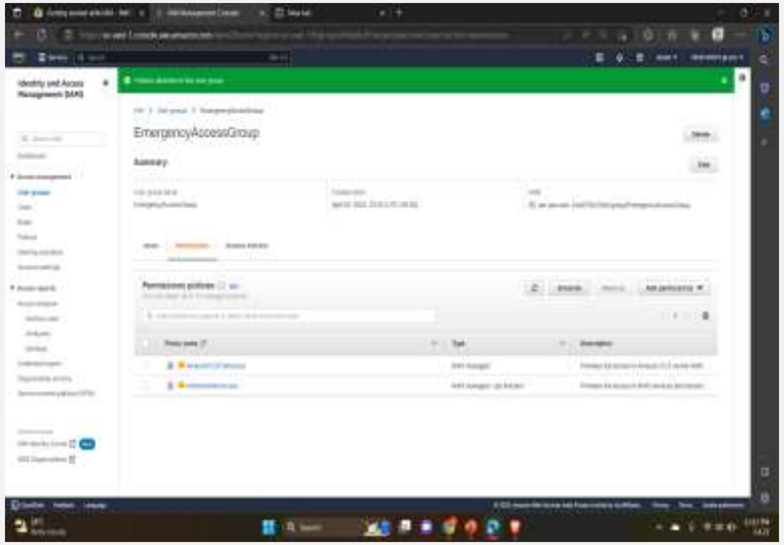


Identity and Access Management (IAM)

Attach permission policies to EmergencyAccessGroup

Other permission policies

| Policy name | Type | Description |
|---|---|---|



Identity and Access Management (IAM)

EmergencyAccessGroup

Summary

Permissions policies

| Policy name | Type | Description |
|---|---|---|

# 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)

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.

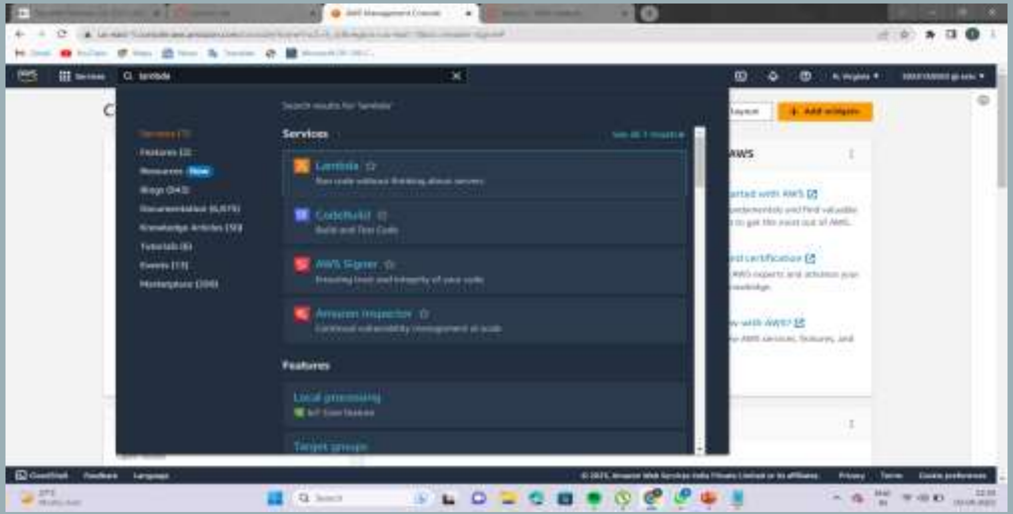
11)Verify that an EC2 instance named instance1 is running in your account, and copy the instance1 instance ID.

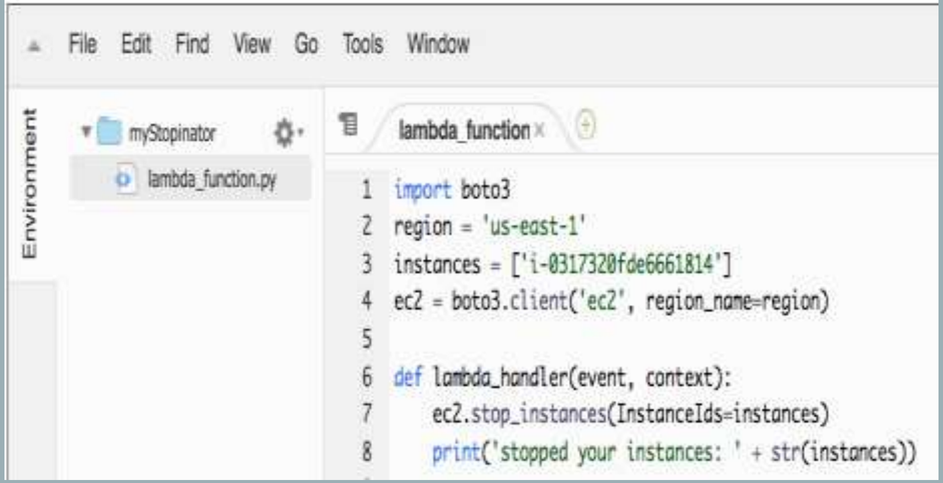
12)Return to the AWS Lambda console browser tab, and replace <REPLACE_WITH_INSTANCE_ID> with the actual instance ID that you just copied.

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

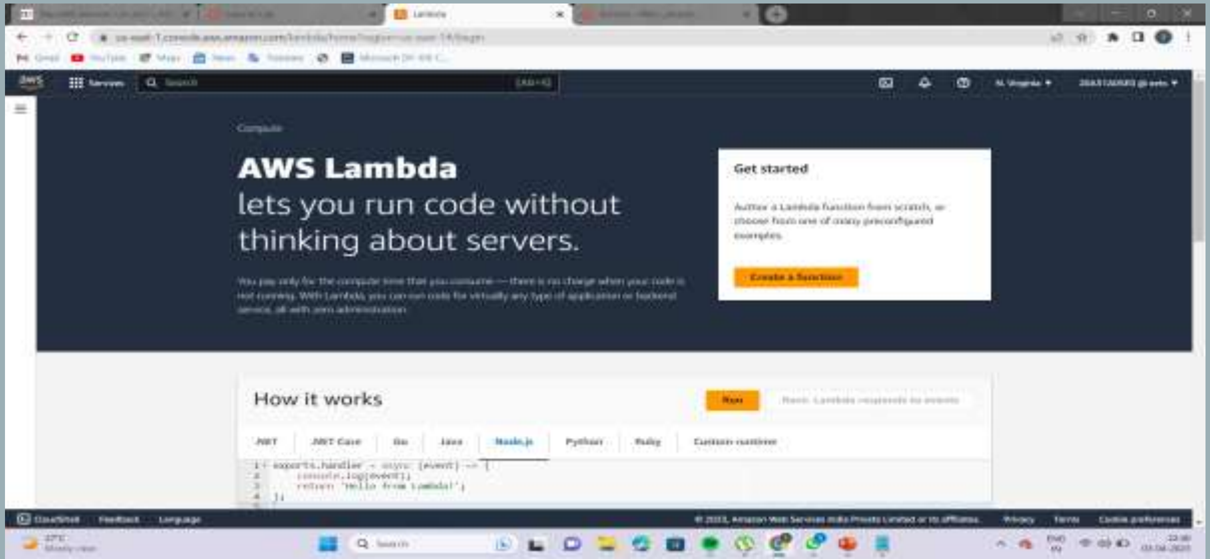
15)Return to the Amazon EC2 console browser tab and see if your instance was stopped.





```python
import boto3
region = 'us-east-1'
instances = ['i-0317320fde6661814']
ec2 = boto3.client('ec2', region_name=region)

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



**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

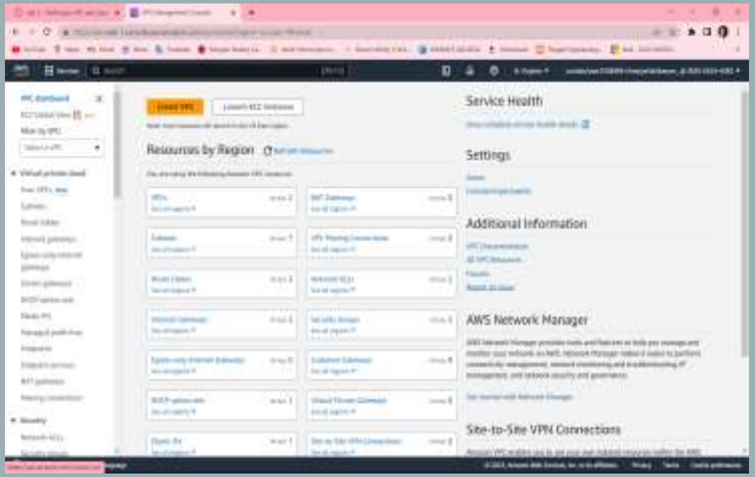
Author a Lambda function from scratch, or choose from one of many preconfigured examples.

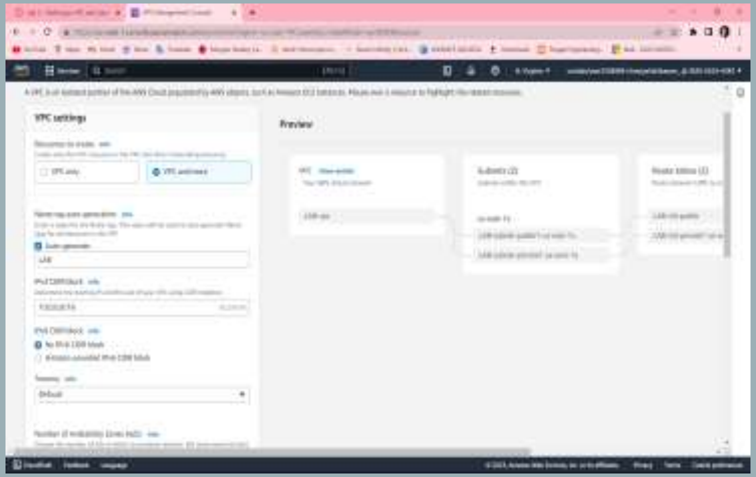
How it works

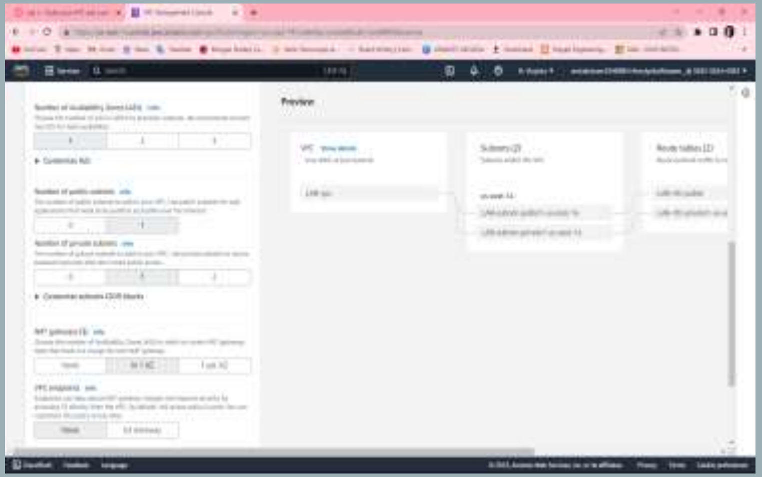
```javascript
exports.handler = async (event) => {
    console.log(event);
    return 'Hello from Lambda!';
};
```

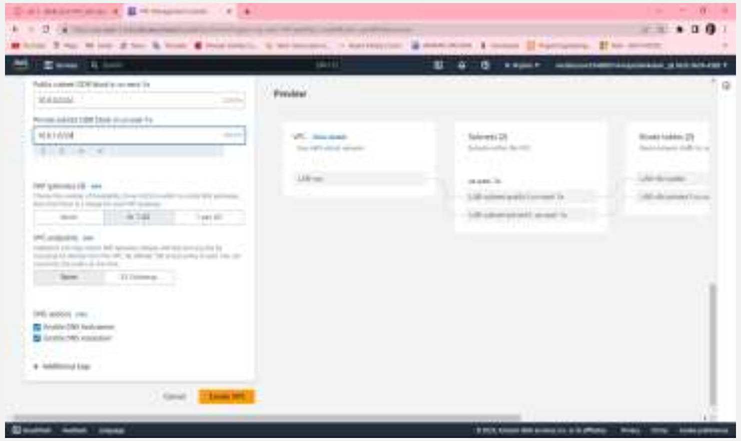
# BUILDING A VPC AND LAUNCHING A WEB SERVER

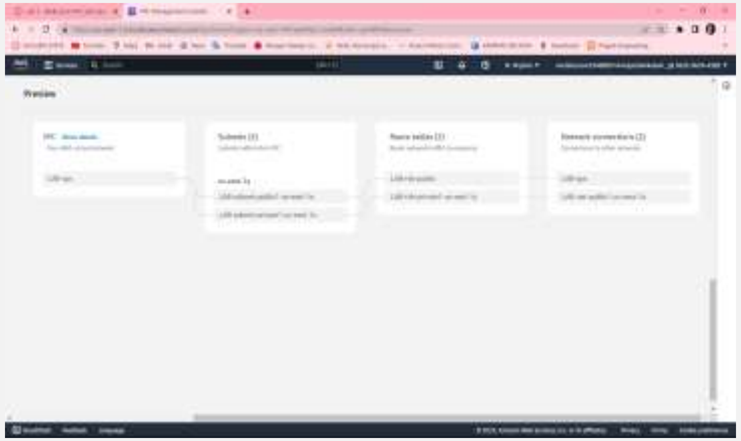
Step 1: First start the lab, when the lab status gets ready, it redirects to the AWS management console.
Step 2: Go to AWS services, search, and choose VPC to open the VPC console.
Step 3: Verify that your regions remain in the N-Virginia(us-east-1). Choose to create VPC in the VPC dashboard
Step 4: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

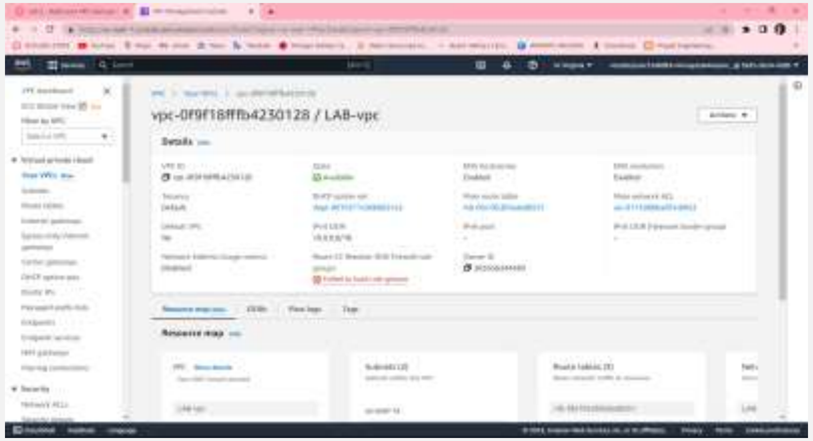












# CREATING ADDITIONAL SUBNETS :

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

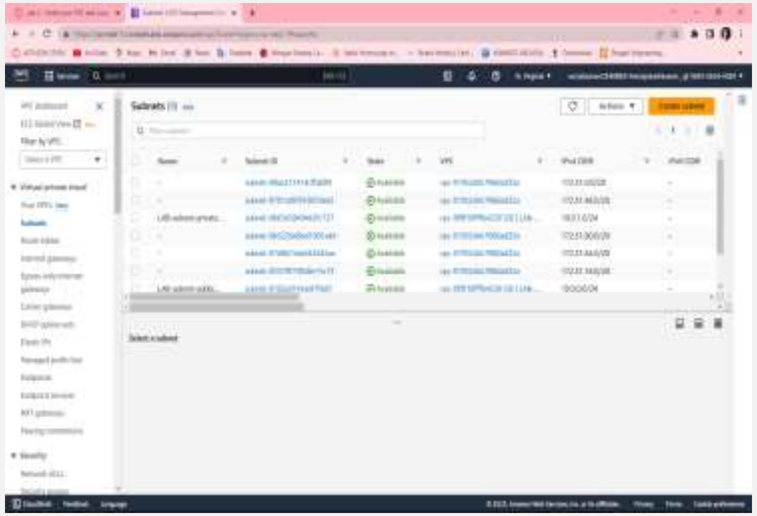
Step 5 : Leave lab-subnet-private1-us-east-1a and also select lab-subnet-private2

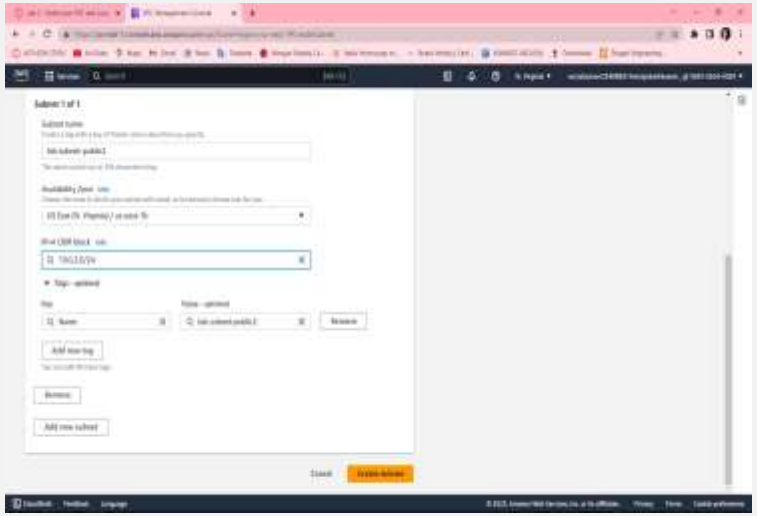
Step 6: Choose SAVE ASSOCIATIONS

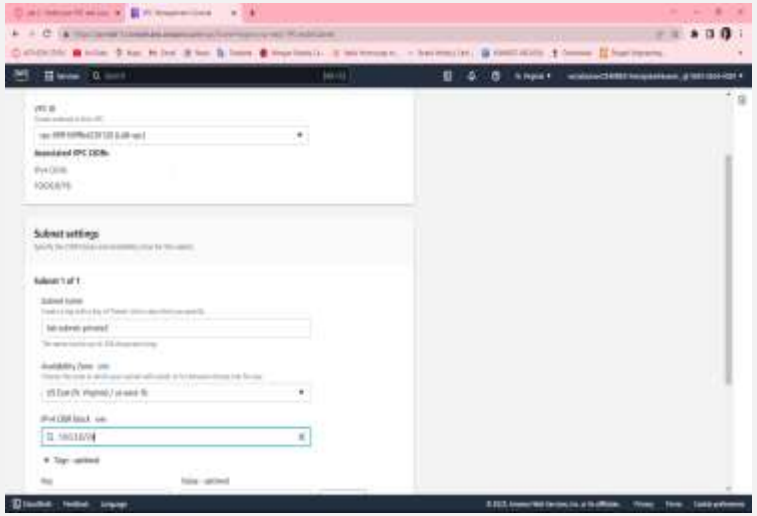
Step 7: Select lab-rtb-public route table and deselect any other subnet

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

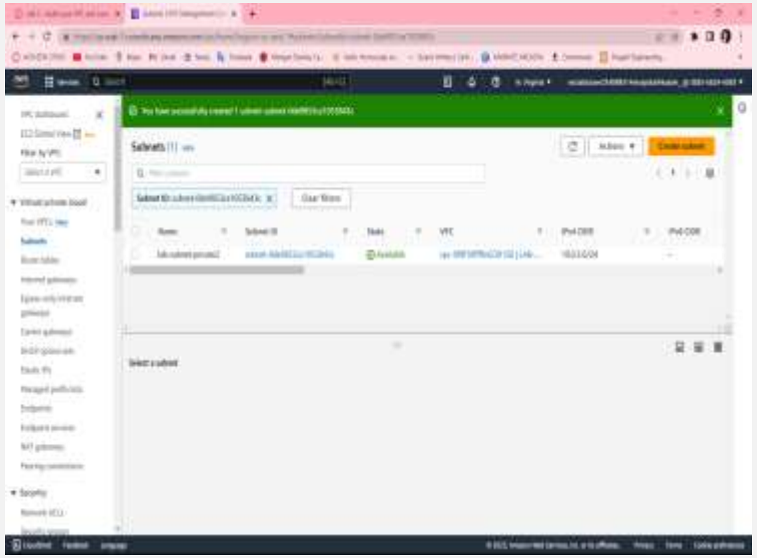
Step 9: Choose SAVE ASSOCIATIONS

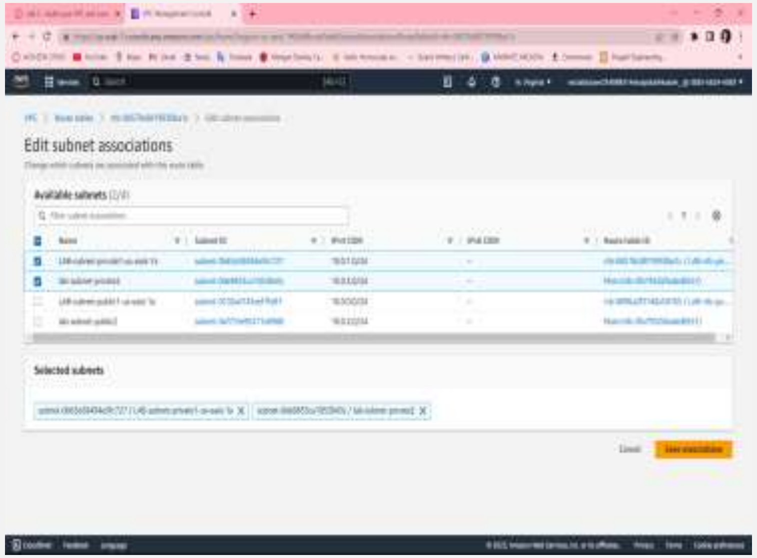

Subnets


Subnet 1 of 1


Subnet settings

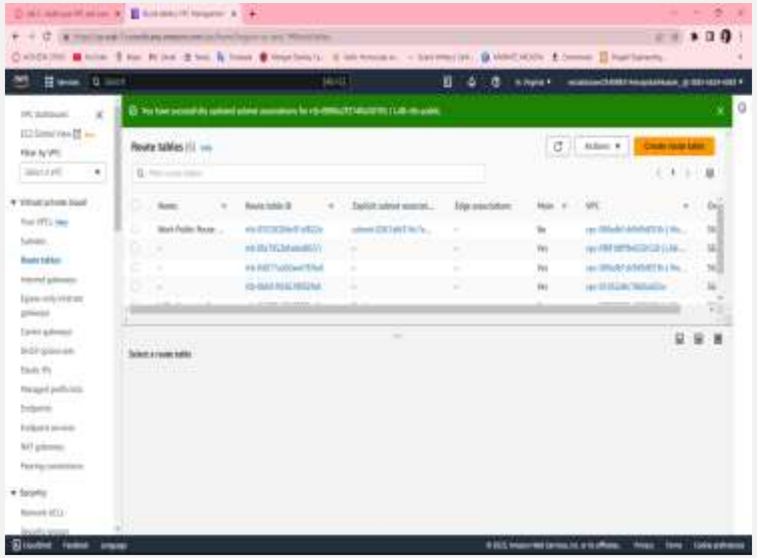
Subnet 1 of 1


Subnets


Edit subnet associations

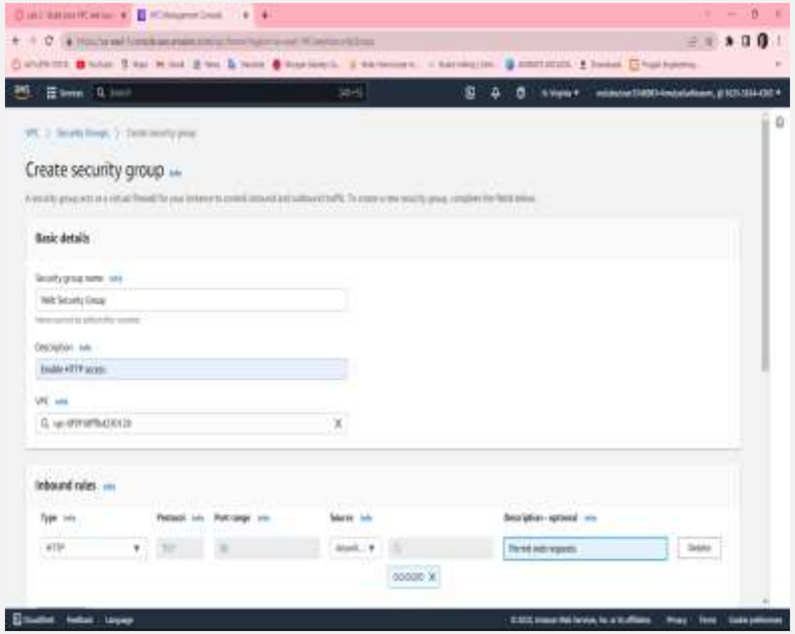
Available subnets

Selected subnets



# 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



Create security group

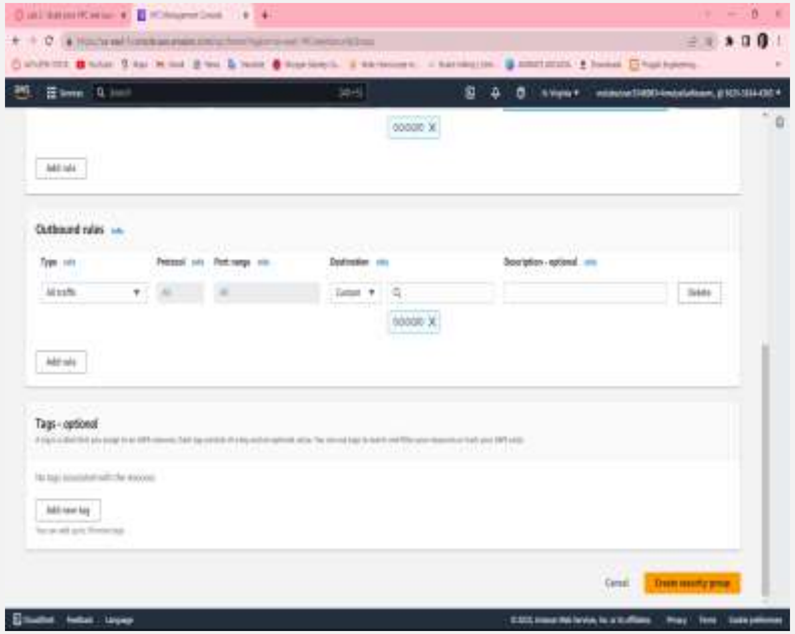
Basic details

Security group name: Web Security Group

Description: Enable HTTP access

Inbound rules

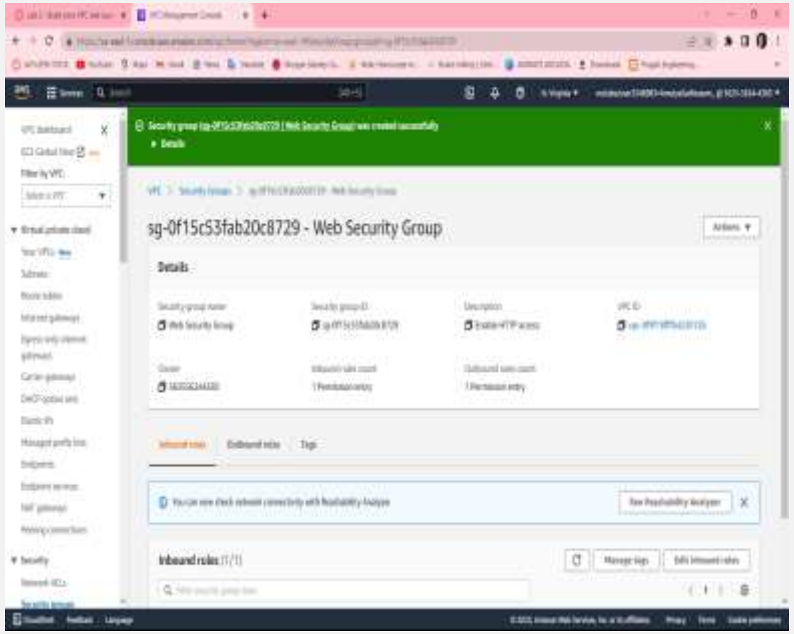
| Type | Protocol | Port range | Source | Description - optional |
|---|---|---|---|---|
| HTTP | TCP | 80 | Anywh... 0.0.0.0/0 | Permit web requests |



Outbound rules

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

Tags - optional



sg-0f15c53fab20c8729 - Web Security Group

Details

Security group name: Web Security Group

Description: Enable HTTP access

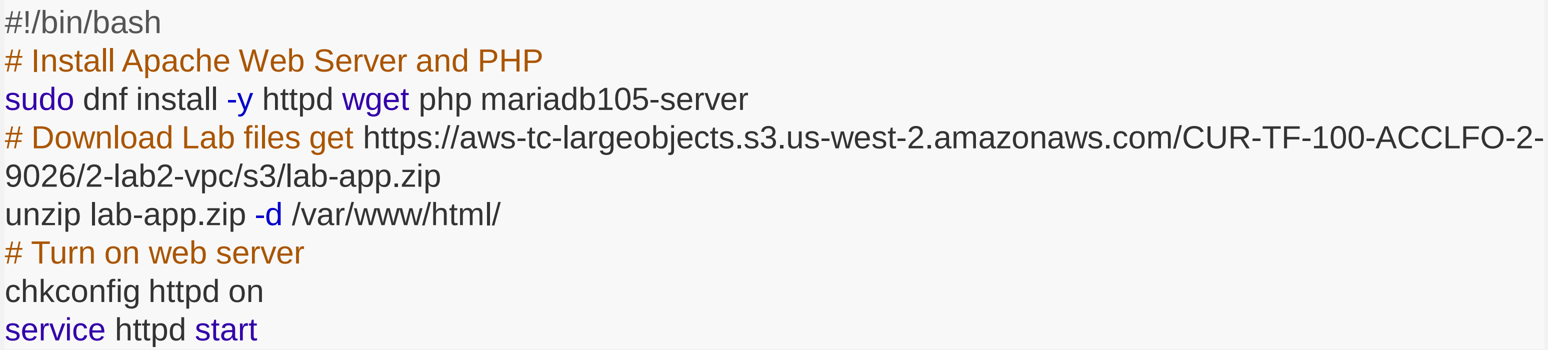
Inbound rules count: 1 Permission entry

Outbound rules count: 1 Permission entry

Inbound rules (1/1)

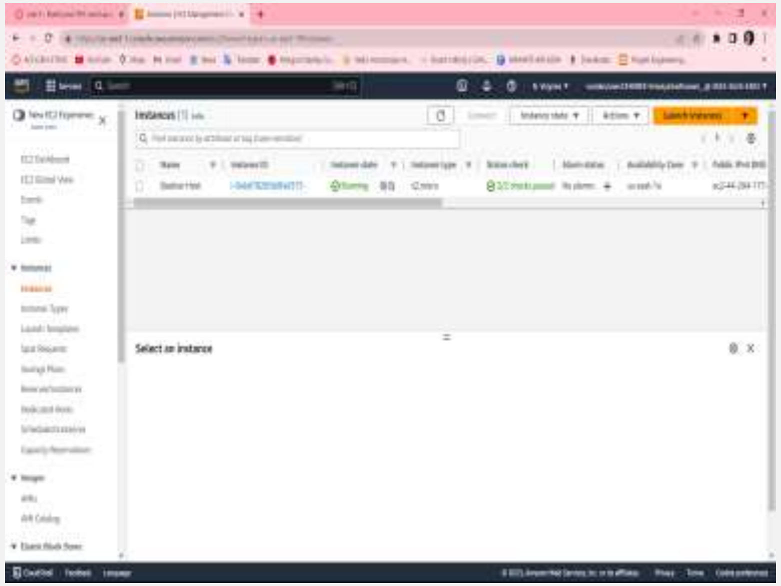
# LAUNCH A WEB SERVER INSTANCE

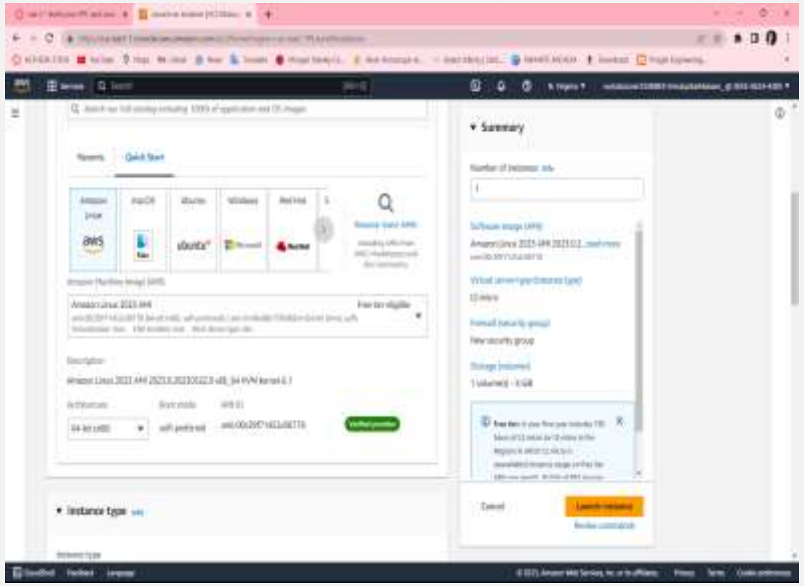
Step 1 : Choose EC2  to open EC2 console , from instances click launch instance  and give name as web server 1
Step 2 : Keep Amazon Linux selelct and select amazon linux 2023 AMI  , Choose t2.micro
Step 3 : Choose key pair name to vockey , in network settings choose edit select network to lab-vpc ,subnet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

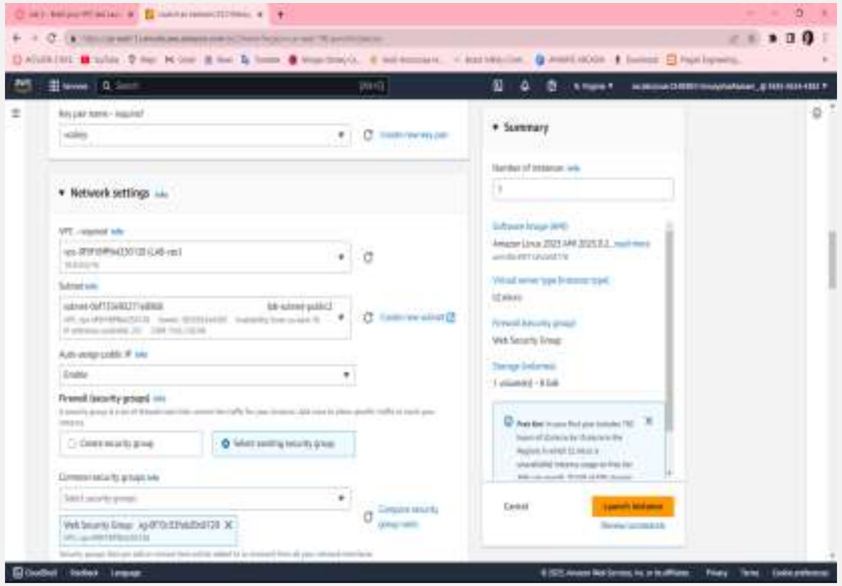


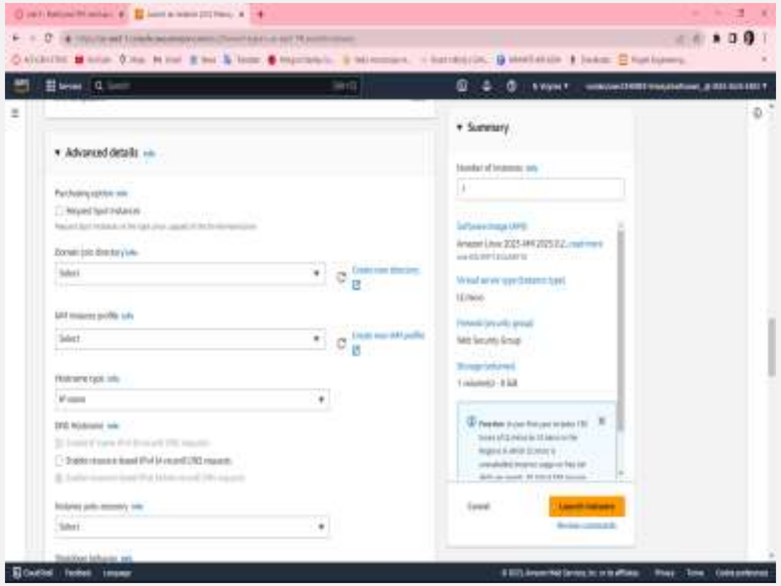
```
#!/bin/bash
# Install Apache Web Server and PHP
sudo dnf install -y httpd wget php mariadb105-server
# Download Lab files get https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-100-ACCLFO-2-9026/2-lab2-vpc/s3/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig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

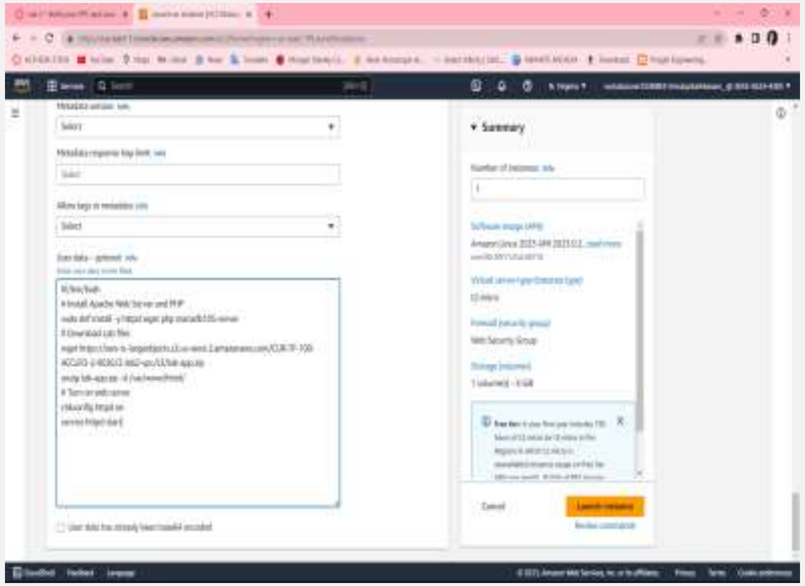
Step 6: At the bottom SUMMARY panel choose launch instance ,click on view instances
Step 7 : Wait until web server 1 shows 2/2 checks passed
Step 8 : Copy the public ipv4 dns value present in details tab
Step 9 : Open a new browser , copy the public dns
Step 10 : Finally we see a web page displaying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





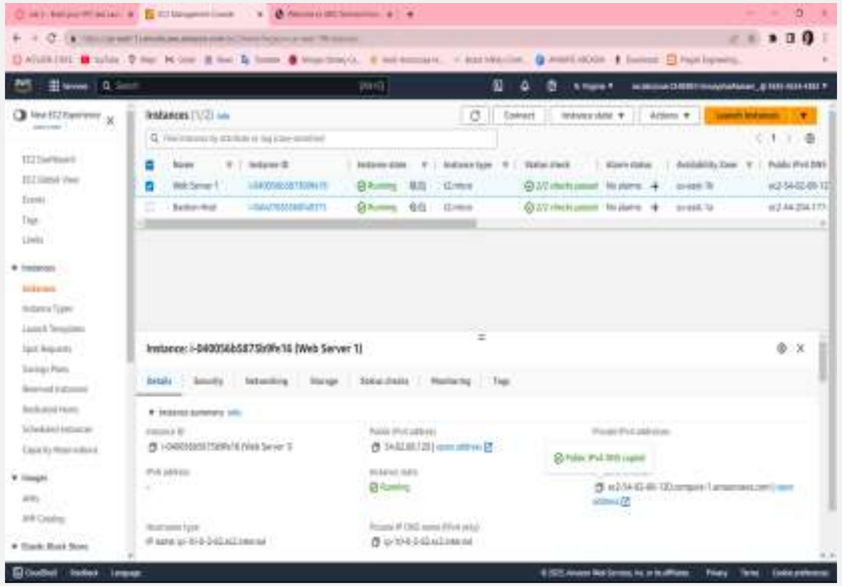


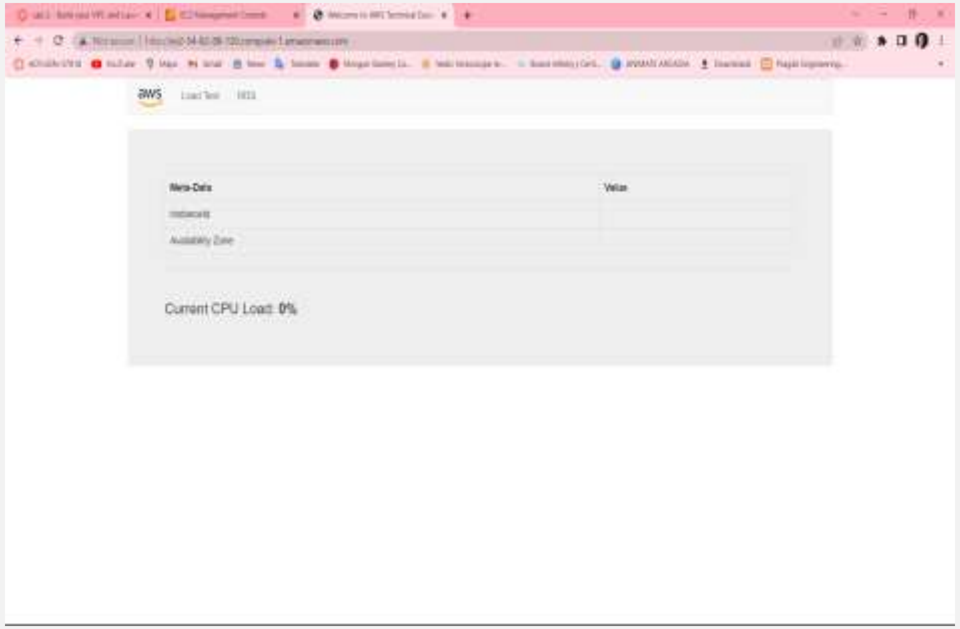




User data - optional

```
#!/bin/bash
# Install Apache Web Server and PHP
sudo dnf install -y httpd wget php mariadb105-server
# Download Lab files
wget 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
```





| Meta-Data | Value |
|---|---|
| InstanceId | |
| Availability Zone | |

Current CPU Load: **0%**

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

*THANK YOU*