# NAAN MUDHALVAN UPSKILLING PLATFORM

## SB8017 - CLOUD ESSENTIALS

Submitted by

**PAVITHRA K**

**(Reg. No. 421320104024)**

III Year / V Semester

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**KRISHNASAMY**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**ANNA UNIVERSITY :: CHENNAI 600 025**

**2022 – 2023**

# BONAFIDE CERTIFICATE

Certified that this report **“SB8017 - CLOUD ESSENTIALS**” is the bonafide work of **PAVITHRA K (Reg. No. 421320104024)** who carried out the project work under my supervision during **21.11.2022 to 02.12.2022**.

| | |
|---|---|
| **SIGNATURE** | **SIGNATURE** |
| **COURSE COORDINATOR(S)** | **HEAD OF THE DEPARTMENT** |
| Er. P.M. KAMATCHI, AP / CSE<br>Er. R. SIVARANJANI, AP / CSE | Er. C. REIKHA, AsP / CSE |

# INDEX

| S.No. | Title | Page No. | Marks | Signature |
|---|---|---|---|---|
| 1 | LINUX AND INTRODUCTION TO CLOUD COMPUTING – HOMEWORK 1 | 4 | | |
| 2 | AWS COMPUTE SERVICES – HOMEWORK 3 | 7 | | |
| 3 | NETWORKING IN AWS – HOMEWORK 4 | 12 | | |
| 4 | LOAD BALANCING AND SCALING IN AWS - HOMEWORK 5 | 23 | | |
| 5 | AWS DATABASE SERVICES – HOMEWORK 6 | 36 | | |
| 6 | SECURITY AND ARCHITECTURAL DESIGN IN AWS – HOMEWORK 7 | 52 | | |
| 7 | MIGRATING TO CLOUD AND AZURE OVERVIEW – HOMEWORK 8 | 56 | | |
| 8 | MICROSOFT AZURE AND GCP OVERVIEW – HOMEWORK 9 | 58 | | |
| 9 | CLOUD ESSENTIALS – PROJECT | 60 | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

# LINUX AND INTRODUCTION TO CLOUD COMPUTING

## HOMEWORK-1

**1.Open the file /van/log/message in the vi editor and delete line number 150.**

To open a file/van/log/messages in the vi editor and delete line number 150 the following command are used

TO OPEN:

```
$vi van.txt
```

TO DELETE:

```
$sed  '150d' van
```

**2.Write a shell script to add two numbers?**

- using  expr command with quotes sum =’expr $num 1+$num2’
- use expr command inclosed with brackets and start with dollar symbol.
- sum=$(expr $ num 1+num2)
- this is my preffered way to directly with the shell.
- sum=$(($ num 1+&num2)).

**3.User root wants to copy /etc,including all subdirectries and files to /tmp .how will you achieve this task?**

To copy files and directives in linux by executing following command

```
$cp file.txt dir file1.txt file2.txt dir1
```

**4.Create a file that contains only the username and the user id of all the users present on the server**

To List users in linux using the simple command line methods present here

```
cat/etc/password
```

**5.How will you provide a count of all users on the system except for adm user?**

To provide a list of all linux users and accounts we use "who" in the command line

**6.How will you list all files in/tmp in increasing order of their size ?**

- To list all files and sort them by size,use the -S option.
- By defult,it displays output in descending order (biggest to smallest in size).

**7.What command is used to clean history on the linux server ?**

*If you want to delete a particular command,enter history -d<line number>,to clean the entire contents os the history life,execute history-c.

**8.Technical use study on migration of charity trust operations to cloud infrastructure.**

**<u>TECHNICAL CASE STUDY</u>:**

**TOPIC: MIGRATION OF CHARITY TRUST OPERATIONS TO CLOUD INFRASTRUCTURES**

**<u>INTRODUCTION:</u>**

Adopting a resilient cloud-first strategy ensures the charity can rapidly deploy and manage services for users, drive enhanced agility, and deliver better outc and undertake the complex transition of core infrastructure,applications and services to the cloud

**<u>PROBLEM</u>: On deploying a website as a solution, the firm traces the various challenges/objectives**

**<u>SOLUTION FOR THE PROBLEM:</u>**

**Making the Difficult Feel Easy**

Highly experienced at helping organisations fast track the migration of infrastructure and workloads to public and private cloud platforms, Bell Integration first needed to analyse the integrations and dependencies underpinning each service and assess each service's readiness to move to the cloud. To ensure business continuity was maintained at all times, Bell's team also mapped all workloads before defining a detailed road map for instigating the charity's new environment.

"Our initial discussions about the best transition approach were incredibly detailed," explains the Global Service Delivery Manager. "For a move of this complexity, we knew that a simple 'lift and shift' method was not an option. Instead, Bell Integration developed an agile and iterative switchover plan that was an exact fit for our requirements,"

With the clock ticking down, the Bell Integration team began the painstaking process of decommissioning the on-premises network appliances supporting the organisation.

During the migration itself, Bell Integration transitioned a number of core network architectures including active directories, switches, servers and storage – and implemented the new Cisco Meraki full stack infrastructure complete with end-point protection that enables the Global Service Delivery Manager to securely administer the new cloud-based network from any location.

“The migration phase itself proved to be a highly collaborative process – Bell’s team of experts kept our teams fully informed of every execution point and procedure, consulting with us on each critical decision,”

**<u>RESULT</u>:** “As a result, our move to the cloud was completed in a hyper-controlled and staged manner with no disruptions to user services.”

# AWS COMPUTE SERVICES

## HOMEWORK-3

### 1. How to create Amazon EBS Lifecycle policy.

The following process shows how to create amazon EBS lifecycle policy,

**To create a snapshot policy:**

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose **Elastic Block Store**, **Lifecycle Manager**, and then choose **Create lifecycle policy**.
3. On the **Select policy type** screen, choose **EBS snapshot policy** and then choose **Next**.
4. In the **Target resources** section, do the following:
   - ➢ For **Target resource types**, choose the type of resource to back up. Choose Volume to create snapshots of individual volumes, or choose Instance to create multi-volume snapshots from the volumes attached to an instance.
   - ➢ For **Target resource location**, specify where the target resources are located.
   - ➢ If the target resources are located in an AWS Region, choose **AWS Region**. Amazon Data Lifecycle Manager backs up all resources of the specified type that have matching target tags in the current Region only. If the resource is located in a Region, snapshots created by the policy will be stored in the same Region.
   - ➢ If the target resources are located on an Outpost in your account, choose **AWS Outpost**. Amazon Data Lifecycle Manager backs up all resources of the specified type that have matching target tags across all of the Outposts in your account. If the resource is located on an Outpost, snapshots created by the policy can be stored in the same Region or on the same Outpost as the resource.
   - ➢ If you do not have any Outposts in your account, this option is hidden and AWS Region is selected for you.
   - ➢ For **Target resource tags**, choose the resource tags that identify the volumes or instances to back up. Only resources that have the specified tag key and value pairs are backed up by the policy.
5. For **Description**, enter a brief description for the policy.
6. For **IAM role**, choose the IAM role that has permissions to manage snapshots and to describe volumes and instances. To use the default role provided by Amazon Data Lifecycle Manager. choose **Default role**. Alternatively, to use a custom IAM role that you previously created, choose **Choose another role** and then select the role to use.
7. For **Policy tags**, add the tags to apply to the lifecycle policy. You can use these tags to identify and categorize your policies.

8. For **Policy status**, choose **Enable** to start the policy runs at the next scheduled time, or **Disable policy** to prevent the policy from running. If you do not enable the policy now, it will not start creating snapshots until you manually enable it after creation.
9. By default, Amazon Data Lifecycle Manager will create snapshots of all the volumes attached to targeted instances. However, you can choose to create snapshots of a subset of the attached volumes. In the **Parameters** section, do the following:
    - If you do not want to create snapshots of the root volumes attached to the targeted instances, select **Exclude root volume**. If you select this option, only the data (non-root) volumes that are attached to targeted instances will be included in the multi-volume snapshot sets.
    - If you want to create snapshots of a subset of the data (non-root) volumes attached to the targeted instances, select **Exclude specific data volumes**, and then specify the tags that are to be used to identify the data volumes that should not be snapshotted. Amazon Data Lifecycle Manager will not create snapshots of data volumes that have any of the specified tags. Amazon Data Lifecycle Manager will create snapshots only of data volumes that do not have any of the specified tags.
    - Choose **Next**.
    - On the **Configure schedule** screen, configure the policy schedules.
    - A policy can have up to 4 schedules.
    - Schedule 1 is mandatory.
    - Schedules 2, 3, and 4 are optional.
    - For each policy schedule that you add, do the following:

In the **Schedule details** section do the following:

i. For **Schedule name**, specify a descriptive name for the schedule.

ii. For **Frequency** and the related fields, configure the interval between policy runs. You can configure policy runs on a daily, weekly, monthly, or yearly schedule. Alternatively, choose **Custom cron expression** to specify an interval of up to one year.

iii. For **Starting at**, specify the time at which the policy runs are scheduled to start. The first policy run starts within an hour after the scheduled time. The time must be entered in the hh:mm UTC format.

iv. For **Retention type**, specify the retention policy for snapshots created by the schedule. You can retain snapshots based on either their total count or their age.

- (Count-based retention) If you do not enable snapshot archiving, the range is 1 to 1000. If you enable snapshot archiving, the range is 0 to 1000. If you specify a count of 0, snapshots are archived immediately after creation.
- (Age-based retention) If you do not enable snapshot archiving, the range is 1 day to 100 years. If you enable snapshot archiving, the range is 0 days to 100 years. If you specify 0 days, snapshots are archived immediately after creation.

i. For **Snapshot destination**, specify the destination for snapshots created by the policy.

- If the policy targets resources in a Region, snapshots must be created in the same Region. AWS Region is selected for you.
- If the policy targets resources on an Outpost, you can choose to create snapshots on the same Outpost as the source resource, or in the Region that is associated with the Outpost.
- If you do not have any Outposts in your account, this option is hidden and AWS Region is selected for you.

a) In the **Tagging** section, do the following:

- To copy all of the user-defined tags from the source volume to the snapshots created by the schedule, select **Copy tags from source**.
- To specify additional tags to assign to snapshots created by this schedule, choose **Add tags**.

In the **Snapshot archiving** section, do the following:

ii. Specify the retention rule for snapshots in the archive tier.

- For **count-based schedules**, specify the number of snapshots to retain in the archive tier. When the retention threshold is reached, the oldest snapshot is permanently deleted from the archive tier. For example, if you specify 3, the schedule will retain a maximum of 3 snapshots in the archive tier. When the fourth snapshot is archived, the oldest of the three existing snapshots in the archive tier is deleted.
- For **age-based schedules**, specify the time period for which to retain snapshots in the archive tier. When the retention threshold is reached, the oldest snapshot is permanently deleted from the archive tier. For example, if you specify 120 days, the schedule will automatically delete snapshots from the archive tier when they reach that age.

**Important:**

The minimum retention period for archived snapshots is 90 days. You must specify a retention rule that retains the snapshot for at least 90 days.

a. To enable fast snapshot restore for snapshots created by the schedule, in the **Fast snapshot restore** section, select **Enable fast snapshot restore**. If you enable fast snapshot restore, you must choose the Availability Zones in which to enable it. If the schedule uses an age-based retention schedule, you must specify the period for which to enable fast snapshot restore for each snapshot. If the schedule uses count-based retention, you must specify the maximum number of snapshots to enable for fast snapshot restore.

b. If the schedule creates snapshots on an Outpost, you can't enable fast snapshot restore. Fast snapshot restore is not supported with local snapshots that are stored on an Outpost.

c. To copy snapshots created by the schedule to an Outpost or to a different Region, in the **Cross-Region copy** section, select **Enable cross-Region copy**.

d. If the schedule creates snapshots in a Region, you can copy the snapshots to up to three additional Regions or Outposts in your account. You must specify a separate cross-Region copy rule for each destination Region or Outpost.

e. For each Region or Outpost, you can choose different retention policies and you can choose whether to copy all tags or no tags. If the source snapshot is encrypted, or if encryption by default is enabled, the copied snapshots are encrypted. If the source snapshot is unencrypted, you can enable encryption. If you do not specify a KMS key, the snapshots are encrypted using the default KMS key for EBS encryption in each destination Region. If you specify a KMS key for the destination Region, then the selected IAM role must have access to the KMS key.

f. In the **Cross-account sharing**, configure the policy to automatically share the snapshots created by the schedule with other AWS accounts. Do the following:

i. To enable sharing with other AWS accounts, select **Enable cross-account sharing**.

ii. To add the accounts with which to share the snapshots, choose **Add account**, enter the 12-digit AWS account ID, and choose **Add**.

iii. To automatically unshare shared snapshots after a specific period, select**Unshare automatically**. If you choose to automatically unshare shared snapshots, the period after which to automatically unshare the snapshots cannot be longer than the period for which the policy retains its snapshots. For example, if the policy's retention configuration retains snapshots for a period of 5 days, you can configure the policy to automatically unshare shared snapshots after periods up to 4 days. This applies to policies with age-based and count-based snapshot retention configurations.

iv. If you do not enable automatic unsharing, the snapshot is shared until it is deleted.

g. To add additional schedules, choose **Add another schedule**, which is located at the top of the screen. For each additional schedule, complete the fields as described previously in this topic.

h. After you have added the required schedules, choose **Review policy**.
Review the policy summary, and then choose **Create policy**.

**2. How to create Amazon EFS Replication?**

The following process shows how to create amazon EFS replication,

**Creating a replication configuration:**

1. Sign in to the AWS Management Console and open the Amazon EFS console at https://console.aws.amazon.com/efs/.
2. In the left navigation pane, choose **File systems**.
3. In the **File systems** list, choose the Amazon EFS file system that you want to replicate. The file system that you choose cannot be a source or destination file system in an existing replication configuration.

4. Choose the **Replication** tab to display the file system's replication section. The section should be blank. If it is not, choose a different file system to be the source.
5. If the file system is not already replicated, choose **Create replication** to display the **Create replication** page.
6. For **Replication settings**, choose the following options:
    - **Destination Region** – The AWS Region in which you want to create the destination file system.
    - **Availability and durability** – Choose **Regional** or **One Zone**.
    - To create a file system that has the highest levels of availability and durability, choose **Regional**. The destination file system will use EFS Standard storage. With EFS Standard storage, your file system data and metadata are stored redundantly across multiple geographically separated Availability Zones within an AWS Region. For more information, see EFS storage classes.
    - To create a file system that uses EFS One Zone storage, choose **One Zone**. With EFS One Zone storage, your file system data and metadata are stored redundantly within a single Availability Zone within an AWS Region. For more information, see EFS storage classes.
7. Choose **Create replication**. The Replication section is displayed, showing the replication details. The **Replication state** value is initially **Enabling**, and **Last synced** is blank. After the state reads **Enabled**, **Last synced** shows **Initial sync in progress**.
8. To see the destination file system's configuration information, choose the file system ID above **Destination file system**. The **File system details** page for the destination file system displays in a new browser tab (depending on your browser settings).

**3. How is stopping and terminating an instance different from each other?**

When an instance is stopped, the instance performs a normal shutdown and then transitions to a stopped state. When an instance is terminated, the instance performs a normal shutdown, then the attached Amazon EBS volumes are deleted unless the volume's deleteOnTermination attribute is set to false.

**4. How is a Spot instance different from an On-Demand instance or Reserved Instance?**

While both Spot Instances and Reserved Instances are great for cost optimization, they operate under different rules and pricing methodologies. As Spot Instances focus on spare EC2 capacity, Reserved Instances give you the benefit of booking ahead of time.

# NETWORKING IN AWS

# HOMEWORK-4

## 1. How to create VPC and subnets using the AWS CLI.

The following process shows how to create VPC and subnets using the AWS CLI(Command line interface),This process uses AWS CLI commands to create a nondefault VPC with an IPv4 CIDR block, and a public and private subnet in the VPC. After you've created the VPC and subnets, you can launch an instance in the public subnet and connect to it.

create the following AWS resources:

- A VPC
- Two subnets
- An internet gateway
- A route table
- An EC2 instance

## TASKS:

- **Prerequisites**
- **STEP-1: Create a VPC and subnets**
- **STEP-2: Make your subnet public**
- **STEP-3: Launch an instance into your subnet**

---

## Prerequisites

Before you begin, install and configure the AWS CLI. When you configure the AWS CLI, you specify your AWS credentials. The examples in this tutorial assume you configured a default Region. Otherwise, add the --region option to each command.

---

## Step 1: Create a VPC and subnets

The first step is to create a VPC and two subnets. This example uses the CIDR block 10.0.0.0/16 for the VPC, but you can choose a different CIDR block.

## To create a VPC and subnets using the AWS CLI

1. Create a VPC with a 10.0.0.0/16 CIDR block using the following create-vpc command.

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --query Vpc.VpcId --output text
```

The command returns the ID of the new VPC. The following is an example.

```
vpc-2f09a348
```

2. Using the VPC ID from the previous step, create a subnet with a 10.0.1.0/24 CIDR block using the following create-subnet command.

```
aws ec2 create-subnet --vpc-id vpc-2f09a348 --cidr-block 10.0.1.0/24
```

3. Create a second subnet in your VPC with a 10.0.0.0/24 CIDR block.

```
aws ec2 create-subnet --vpc-id vpc-2f09a348 --cidr-block 10.0.0.0/24
```

---

## Step 2: Make your subnet public

After you've created the VPC and subnets, you can make one of the subnets a public subnet by attaching an internet gateway to your VPC, creating a custom route table, and configuring routing for the subnet to the internet gateway.

## To make your subnet a public subnet

1. Create an internet gateway using the following create-internet-gateway command.

```
aws ec2 create-internet-gateway --query InternetGateway.InternetGatewayId --output text
```

The command returns the ID of the new internet gateway. The following is an example.

```
igw-1ff7a07b
```

2. Using the ID from the previous step, attach the internet gateway to your VPC using the following create-internet-gateway command.

```
aws ec2 attach-internet-gateway --vpc-id vpc-2f09a348 --internet-gateway-id igw-1ff7a07b
```

3. Create a custom route table for your VPC using the following create-route-table command.

```
aws ec2 create-route-table --vpc-id vpc-2f09a348 --query RouteTable.RouteTableId --output text
```

The command returns the ID of the new route table. The following is an example.

```
rtb-c1c8faa6
```

4. Create a route in the route table that points all traffic (0.0.0.0/0) to the internet gateway using the following create-route command.

```
aws ec2 create-route --route-table-id rtb-c1c8faa6 --destination-cidr-block 0.0.0.0/0 --gateway-id igw-1ff7a07b
```

5. (Optional) To confirm that your route has been created and is active, you can describe the route table using the following describe-route-table command.

```
aws ec2 describe-route-tables --route-table-id rtb-c1c8faa6

{

    "RouteTables": [

        {

            "Associations": [],

            "RouteTableId": "rtb-c1c8faa6",

            "VpcId": "vpc-2f09a348",

            "PropagatingVgws": [],

            "Tags": [],

            "Routes": [

                {

                    "GatewayId": "local",

                    "DestinationCidrBlock": "10.0.0.0/16",

                    "State": "active",

                    "Origin": "CreateRouteTable"

                },
                {

                    "GatewayId": "igw-1ff7a07b",

                    "DestinationCidrBlock": "0.0.0.0/0",

                    "State": "active",

                    "Origin": "CreateRoute"

                }   ]  }    ] }
```

6. The route table is currently not associated with any subnet. You need to associate it with a subnet in your VPC so that traffic from that subnet is routed to the internet gateway. Use the following describe-subnets command to get the subnet IDs. The --filter option restricts the subnets to your new VPC only, and the --query option returns only the subnet IDs and their CIDR blocks.

```
aws ec2 describe

-subnets

--filters

"Name=vpc-id,Values=vpc-2f09a348"

--query "Subnets[*].{ID:SubnetId,CIDR:CidrBlock}"

[

    {

        "CIDR": "10.0.1.0/24",

        "ID": "subnet-b46032ec"

    },

    {

        "CIDR": "10.0.0.0/24",

        "ID": "subnet-a46032fc"

    }]
```

7. You can choose which subnet to associate with the custom route table, for example, subnet-b46032ec, and associate it using the associate-route-table command. This subnet is your public subnet.

```
aws ec2 associate-route-table  --subnet-id subnet-b46032ec --route-table-id rtb-c1c8faa6
```

8. (Optional) You can modify the public IP addressing behavior of your subnet so that an instance launched into the subnet automatically receives a public IP address using the following modify-subnet-attribute command. Otherwise, associate an Elastic IP address with your instance after launch so that the instance is reachable from the internet.

```
aws ec2 modify-subnet-attribute --subnet-id subnet-b46032ec --map-public-ip-on-launch
```

---

## Step 3: Launch an instance into your subnet

To test that your subnet is public and that instances in the subnet are accessible over the internet, launch an instance into your public subnet and connect to it. First, you must create a security group to associate with your instance, and a key pair with which you'll connect to your instance

### To launch and connect to an instance in your public subnet

1. Create a key pair and use the --query option and the --output text option to pipe your private key directly into a file with the .pem extension.

```
aws ec2 create-key-pair --key-name MyKeyPair --query "KeyMaterial" --output text > MyKeyPair.pem
```

In this example, you launch an Amazon Linux instance. If you use an SSH client on a Linux or Mac OS X operating system to connect to your instance, use the following command to set the permissions of your private key file so that only you can read it.

```
chmod 400 MyKeyPair.pem
```

2. Create a security group in your VPC using the create-security-group command.

```
aws ec2 create-security-group --group-name SSHAccess --description "Security group for SSH access" --vpc-id vpc-2f09a348

{

   "GroupId": "sg-e1fb8c9a"

}
```

Add a rule that allows SSH access from anywhere using the authorize-security-group-ingress command.

```
aws ec2 authorize-security-group-ingress --group-id sg-e1fb8c9a --protocol tcp --port
22 --cidr 0.0.0.0/0
```

3. Launch an instance into your public subnet, using the security group and key pair you've created. In the output, take note of the instance ID for your instance.

```
aws ec2 run-instances --image-id ami-a4827dc9 --count 1 --instance-type t2.micro --
key-name MyKeyPair --security-group-ids sg-e1fb8c9a --subnet-id subnet-b46032ec
```

**Note**

In this example, the AMI is an Amazon Linux AMI in the US East (N. Virginia) Region. If you're in a different Region, you'll need the AMI ID for a suitable AMI in your Region. Your instance must be in the running state in order to connect to it. Use the following command to describe the state and IP address of your instance.

```
aws ec2 describe-instances --instance-id i-0146854b7443af453 --query
"Reservations[*].Instances[*].{State:State.Name,Address:PublicIpAddress}"
```

The following is example output.

```
[   [       {

            "State": "running",

            "Address": "52.87.168.235"

        }   ] ]
```

4. When your instance is in the running state, you can connect to it using an SSH client on a Linux or Mac OS X computer by using the following command:

```
ssh -i "MyKeyPair.pem" ec2-user@52.87.168.235
```

## 2. How to migrate existing VPCs from IPv4 to IPv6?

The following process shows how to migrate existing VPCs from IPV4 to IPv6,

**CONTENTS:**

- **STEP-1: Associate an IPv6 CIDR block with your VPC and subnet**
- **STEP-2: Update your route tables**
- **STEP-3: update your security group rules**
- **STEP-4: Change your instance type**

- **STEP-5: Assign IPv6 addresses to your instances**

## Step 1: Associate an IPv6 CIDR block with your VPC and subnets

You can associate an IPv6 CIDR block with your VPC, and then associate a /64 CIDR block from that range with each subnet.

### To associate an IPv6 CIDR block with a VPC

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Your VPCs**.
3. Select your VPC, choose **Actions**, **Edit CIDRs**.
4. Choose **Add IPv6 CIDR**, choose one of the following options, and then choose **Select CIDR**:
   - **Amazon-provided IPv6 CIDR block**: Requests an IPv6 CIDR block from Amazon's pool of IPv6 addresses. For **Network Border Group**, select the group from which AWS advertises IP addresses.
   - **IPv6 CIDR owned by me**: (BYOIP) Allocates an IPv6 CIDR block from your IPv6 address pool. For **Pool,** choose the IPv6 address pool from which to allocate the IPv6 CIDR block.

### To associate an IPv6 CIDR block with a subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Subnets**.
3. Select your subnet, choose **Subnet Actions**, **Edit IPv6 CIDRs**.
4. Choose **Add IPv6 CIDR**. Specify the hexadecimal pair for the subnet (for example, 00) and confirm the entry by choosing the tick icon.
5. Choose **Close**. Repeat the steps for the other subnets in your VPC.

## Step 2: Update your route tables

For a public subnet, you must update the route table to enable instances (such as web servers) to use the internet gateway for IPv6 traffic.

For a private subnet, you must update the route table to enable instances (such as database instances) to use an egress-only internet gateway for IPv6 traffic.

### To update your route table for a public subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.

2. In the navigation pane, choose **Route Tables** and select the route table that's associated with the public subnet.
3. On the **Routes** tab, choose **Edit routes**.
4. Choose **Add route**. Specify ::/0 for **Destination**, select the ID of the internet gateway for **Target**, and then choose **Save changes**.

### To update your route table for a private subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. If you're using a NAT device in your private subnet, it does not support IPv6 traffic. Instead, create an egress-only internet gateway for your private subnet to enable outbound communication to the internet over IPv6 and prevent inbound communication. An egress-only internet gateway supports IPv6 traffic only.In the navigation pane, choose **Route Tables** and select the route table that's associated with the private subnet.
3. On the **Routes** tab, choose **Edit routes**.
4. Choose **Add route**. For **Destination**, specify ::/0. For **Target**, select the ID of the egress-only internet gateway, and then choose **Save changes**.

---

## Step 3: Update your security group rules

To enable your instances to send and receive traffic over IPv6, you must update your security group rules to include rules for IPv6 addresses.

For example, in the example above, you can update the web server security group (sg-11aa22bb11aa22bb1) to add rules that allow inbound HTTP, HTTPS, and SSH access from IPv6 addresses. You do not need to make any changes to the inbound rules for your database security group; the rule that allows all communication from sg-11aa22bb11aa22bb1 includes IPv6 communication by default.

### To update your security group rules

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Security Groups** and select your web server security group.
3. In the **Inbound Rules** tab, choose **Edit**.
4. For each rule, choose **Add another rule**, and choose **Save** when you're done. For example, to add a rule that allows all HTTP traffic over IPv6, for **Type**, select **HTTP** and for **Source**, enter ::/0.

By default, an outbound rule that allows all IPv6 traffic is automatically added to your security groups when you associate an IPv6 CIDR block with your VPC. However, if you modified the original outbound rules for your security group, this rule is not automatically added, and you must add equivalent outbound rules for IPv6 traffic.

## Update your network ACL rules

If you associate an IPv6 CIDR block with your VPC, we automatically add rules to the default network ACL to allow IPv6 traffic, provided you haven't modified its default rules. If you've modified your default network ACL or if you've created a custom network ACL with rules to control the flow of traffic to and from your subnet, you must manually add rules for IPv6 traffic.

## Step 4: Change your instance type

All current generation instance types support IPv6.If your instance type does not support IPv6, you must resize the instance to a supported instance type. In the example above, the database instance is an m3.large instance type, which does not support IPv6. You must resize the instance to a supported instance type, for example, m4.large.

To resize your instance, be aware of the compatibility limitations.In this scenario, if your database instance was launched from an AMI that uses HVM virtualization, you can resize it to an m4.large instance type by using the following procedure.

### Important

To resize your instance, you must stop it. Stopping and starting an instance changes the public IPv4 address for the instance, if it has one. If you have any data stored on instance store volumes, the data is erased.

### To resize your instance

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose **Instances**, and select the database instance.
3. Choose **Actions**, **Instance State**, **Stop**.
4. In the confirmation dialog box, choose **Yes, Stop**.
5. With the instance still selected, choose **Actions**, **Instance Settings**, **Change Instance Type**.
6. For **Instance Type**, choose the new instance type, and then choose **Apply**.
7. To restart the stopped instance, select the instance and choose **Actions**, **Instance State**, **Start**. In the confirmation dialog box, choose **Yes, Start**.

If your instance is an instance store-backed AMI, you can't resize your instance using the earlier procedure. Instead, you can create an instance store-backed AMI from your instance, and launch a new instance from your AMI using a new instance type.You may not be able to migrate to a new instance type if there are compatibility limitations. For example, if your instance was launched from an AMI that uses PV virtualization, the only instance type that supports both PV virtualization and IPv6 is C3. This instance type may not be suitable for your needs. In this case, you may have to reinstall your software on a base HVM AMI, and launch a new instance.

If you launch an instance from a new AMI, you can assign an IPv6 address to your instance during launch.

---

## Step 5: Assign IPv6 addresses to your instances

After you've verified that your instance type supports IPv6, you can assign an IPv6 address to your instance using the Amazon EC2 console. The IPv6 address is assigned to the primary network interface (eth0) for the instance.

### To assign an IPv6 address to your instance

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, choose **Instances**.
3. Select your instance, and choose **Actions**, **Networking**, **Manage IP Addresses**.
4. Under **IPv6 Addresses**, choose **Assign new IP**. You can enter a specific IPv6 address from the range of your subnet, or you can leave the default Auto-Assign value to let Amazon choose one for you.
5. Choose **Yes, Update**.

Alternatively, if you launch a new instance (for example, if you were unable to change the instance type and you created a new AMI instead), you can assign an IPv6 address during launch.

### To assign an IPv6 address to an instance during launch

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. Select your AMI and an IPv6-compatible instance type, and choose **Next: Configure Instance Details**.
3. On the **Configure Instance Details** page, select a VPC for **Network** and a subnet for **Subnet**. For **Auto-assign IPv6 IP**, select **Enable**.
4. Follow the remaining steps in the wizard to launch your instance.

You can connect to an instance using its IPv6 address. If you're connecting from a local computer, ensure that your local computer has an IPv6 address and is configured to use IPv6.

**3. Can I connect my corporate data center to the Amazon Cloud?**

Yes we can connect a data center to amazon cloud using AWS Direct Connect which enables you to securely connect your AWS environment to your on-premises data center or office location over a standard 1 gigabit or 10 gigabit Ethernet fiber-optic connection.

**4. Is it possible to change the private IP addresses of an EC2 while it is running/stopped in a VPC?**

No, It is not possible the private IP address of an Amazon EC2 instance will never change. It will not change while an instance is running. It will not change while an instance is stopped. You cannot change a private IP address.

**5. If Im using Amazon CloudFront, can I use Direct Connect to transfer objects from my own data center?**

Yes. Amazon CloudFront supports custom origins including origins you run outside of AWS.

**6. If my AWS Direct Connect fails, will I lose my connectivity?**

Traffic to/from public resources, such as Amazon S3, will be routed over the internet. **If you do not have a backup AWS Direct Connect link or an IPsec VPN link, then Amazon VPC traffic will be dropped in the event of a failure**. Traffic to/from public resources will be routed over the internet.

# LOAD BALANCING AND AUTO SCALING IN AWS

## HOMEWORK-5

**1. Setup Sticky sessions for your Application Load Balancer.**

### Sticky sessions for your Application Load Balancer

By default, an Application Load Balancer routes each request independently to a registered target based on the chosen load-balancing algorithm. However, you can use the sticky session feature (also known as session affinity) to enable the load balancer to bind a user's session to a specific target. This ensures that all requests from the user during the session are sent to the same target. This feature is useful for servers that maintain state information in order to provide a continuous experience to clients. To use sticky sessions, the client must support cookies.

Application Load Balancers support both duration-based cookies and application-based cookies. Sticky sessions are enabled at the target group level. You can use a combination of duration-based stickiness, application-based stickiness, and no stickiness across your target groups.

The key to managing sticky sessions is determining how long your load balancer should consistently route the user's request to the same target. If your application has its own session cookie, then you can use application-based stickiness and the load balancer session cookie follows the duration specified by the application's session cookie. If your application does not have its own session cookie, then you can use duration-based stickiness to generate a load balancer session cookie with a duration that you specify.

The content of load balancer generated cookies are encrypted using a rotating key. You cannot decrypt or modify load balancer generated cookies.

For both stickiness types, the Application Load Balancer resets the expiry of the cookies it generates after every request. If a cookie expires, the session is no longer sticky and the client should remove the cookie from its cookie store.

### Requirements

- An HTTP/HTTPS load balancer.
- At least one healthy instance in each Availability Zone.

## Considerations

- Sticky sessions are not supported if cross-zone load balancing is disabled Attempting to enable sticky sessions while cross-zone load balancing is disabled will fail.
- For application-based cookies, cookie names have to be specified individually for each target group. However, for duration-based cookies, AWSALB is the only name used across all target groups.
- If you are using multiple layers of Application Load Balancers, you can enable sticky sessions across all layers with application-based cookies. However, with duration-based cookies, you can enable sticky sessions only on one layer, because AWSALB is the only name available.
- Application-based stickiness does not work with weighted target groups.
- If you have a forward action with multiple target groups, and sticky sessions are enabled for one or more of the target groups, you must enable stickiness at the target group level.
- WebSocket connections are inherently sticky. If the client requests a connection upgrade to WebSockets, the target that returns an HTTP 101 status code to accept the connection upgrade is the target used in the WebSockets connection. After the WebSockets upgrade is complete, cookie-based stickiness is not used.
- Application Load Balancers use the Expires attribute in the cookie header instead of the Max-Age attribute.
- Application Load Balancers do not support cookie values that are URL encoded.

---

## Duration-based stickiness

Duration-based stickiness routes requests to the same target in a target group using a load balancer generated cookie (AWSALB). The cookie is used to map the session to the target. If your application does not have its own session cookie, you can specify your own stickiness duration and manage how long your load balancer should consistently route the user's request to the same target.

When a load balancer first receives a request from a client, it routes the request to a target (based on the chosen algorithm), and generates a cookie named AWSALB. It encodes information about the selected target, encrypts the cookie, and includes the cookie in the response to the client. The load balancer generated cookie has its own expiry of 7 days which is non-configurable.

In subsequent requests, the client should include the AWSALB cookie. When the load balancer receives a request from a client that contains the cookie, it detects it and routes the request to the same target. If the cookie is present but cannot be decoded, or if it refers to a target that was deregistered or is unhealthy, the load balancer selects a new target and updates the cookie with information about the new target.

With cross-origin resource sharing (CORS) requests, some browsers require SameSite=None; Secure to enable stickiness. In this case, the load balancer generates a second stickiness cookie, AWSALBCORS, which includes the same information as the original stickiness cookie plus the SameSite attribute. Clients receive both cookies.

### To enable duration-based stickiness using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. On the navigation pane, under **Load Balancing**, choose **Target Groups**.
3. Choose the name of the target group to open its details page.
4. On the **Group details** tab, in the **Attributes** section, choose **Edit**.
5. On the **Edit attributes** page, do the following:
   a) Select **Stickiness**.
   b) For **Stickiness type**, select **Load balancer generated cookie**.
   c) For **Stickiness duration**, specify a value between 1 second and 7 days.
   d) Choose **Save changes**.

### To enable duration-based stickiness using the AWS CLI

Use the modify-target-group-attributes command with the stickiness.enabled and stickiness.lb_cookie.duration_seconds attributes

Use the following command to enable duration-based stickiness.

```
aws elbv2 modify-target-group-attributes --target-group-arn ARN --attributes
Key=stickiness.enabled,Value=true
Key=stickiness.lb_cookie.duration_seconds,Value=time-in-seconds
```

Your output should be similar to the following example.

```
{
  "Attributes": [
    ...
     {
         "Key": "stickiness.enabled",
         "Value": "true"
```

```
        },
        {
            "Key": "stickiness.lb_cookie.duration_seconds",
            "Value": "86500"
                },          ... ]     }
```

## Application-based stickiness

Application-based stickiness gives you the flexibility to set your own criteria for client-target stickiness. When you enable application-based stickiness, the load balancer routes the first request to a target within the target group based on the chosen algorithm. The target is expected to set a custom application cookie that matches the cookie configured on the load balancer to enable stickiness. This custom cookie can include any of the cookie attributes required by the application.

When the Application Load Balancer receives the custom application cookie from the target, it automatically generates a new encrypted application cookie to capture stickiness information. This load balancer generated application cookie captures stickiness information for each target group that has application-based stickiness enabled.

The load balancer generated application cookie does not copy the attributes of the custom cookie set by the target. It has its own expiry of 7 days which is non-configurable. In the response to the client, the Application Load Balancer only validates the name with which the custom cookie was configured at the target group level and not the value or the expiry attribute of the custom cookie. As long as the name matches, the load balancer sends both cookies, the custom cookie set by the target, and the application cookie generated by the load balancer, in the response to the client.

In subsequent requests, clients have to send back both cookies to maintain stickiness. The load balancer decrypts the application cookie, and checks whether the configured duration of stickiness is still valid. It then uses the information in the cookie to send the request to the same target within the target group to maintain stickiness. The load balancer also proxies the custom application cookie to the target without inspecting or modifying it. In subsequent responses, the expiry of the load balancer generated application cookie and the duration of stickiness configured on the load balancer are reset. To maintain stickiness between client and target, the expiry of the cookie, and the duration of stickiness should not elapse.

If a target fails or becomes unhealthy, the load balancer stops routing requests to that target, and chooses a new healthy target based on the chosen load balancing algorithm. The load balancer treats the session as now being "stuck" to the new healthy target, and continues routing requests to the new healthy target even if the failed target comes back.

With cross-origin resource sharing (CORS) requests, to enable stickiness, the load balancer adds the SameSite=None; Secure attributes to the load balancer generated application cookie only if the user-agent version is Chromium80 or above.

Since most browsers limit cookies to 4K in size, the load balancer shards application cookies greater than 4K into multiple cookies. Application Load Balancers support cookies up to 16K in size and can therefore create up to 4 shards that it sends to the client. The application cookie name that the client sees begins with “AWSALBAPP-" and includes a fragment number. For example, if the cookie size is 0-4K, the client sees AWSALBAPP-0. If the cookie size is 4-8k, the client sees AWSALBAPP-0 and AWSALBAPP-1, and so on.

## To enable application-based stickiness using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. On the navigation pane, under **Load Balancing**, choose **Target Groups**.
3. Choose the name of the target group to open its details page.
4. On the **Group details** tab, in the **Attributes** section, choose **Edit**.
5. On the **Edit attributes** page, do the following:

Select **Stickiness**.

a) For **Stickiness type**, select **Application-based cookie**.
b) For **Stickiness duration**, specify a value between 1 second and 7 days.
c) For **App cookie name**, enter a name for your application-based cookie.
d) Do not use AWSALB, AWSALBAPP, or AWSALBTG for the cookie name; they're reserved for use by the load balancer.
e) Choose **Save changes**.

## To enable application-based stickiness using the AWS CLI

Use the modify-target-group-attributes command with the following attributes:

- stickiness.enabled
- stickiness.type
- stickiness.app_cookie.cookie_name
- stickiness.app_cookie.duration_seconds

Use the following command to enable application-based stickiness.

```
aws elbv2 modify-target-group-attributes --target-group-arn ARN --attributes
```

```
Key=stickiness.enabled,Value=true          Key=stickiness.type,Value=app_cookie
Key=stickiness.app_cookie.cookie_name,Value=my-cookie-name
Key=stickiness.app_cookie.duration_seconds,Value=time-in-seconds
```

Your output should be similar to the following example.

```
{
   "Attributes": [
      ...
      {
         "Key": "stickiness.enabled",
         "Value": "true"
      },
      {
         "Key": "stickiness.app_cookie.cookie_name",
         "Value": "MyCookie"
      },
      {
         "Key": "stickiness.type",
         "Value": "app_cookie"
      },
      {
         "Key": "stickiness.app_cookie.duration_seconds",
         "Value": "86500"
      },
      ... ]       }
```

### Manual rebalancing

When scaling up, if the number of targets increase considerably, there is potential for unequal distribution of load due to stickiness. In this scenario, you can rebalance the load on your targets using the following two options:

- Set an expiry on the cookie generated by the application that is prior to the current date and time. This will prevent clients from sending the cookie to the Application Load Balancer, which will restart the process of establishing stickiness.
- Set a very short duration on the load balancer's application-based stickiness configuration, for example, 1 second. This forces the Application Load Balancer to reestablish stickiness even if the cookie set by the target has not expired.

## 2. Setup a scaled and load-balanced application

### Set up a scaled and load-balanced application

### Important

Registering your Auto Scaling group with an Elastic Load Balancing load balancer helps you set up a load-balanced application. Elastic Load Balancing works with Amazon EC2 Auto Scaling to distribute incoming traffic across your healthy Amazon EC2 instances. This increases the scalability and availability of your application. You can enable Elastic Load Balancing within multiple Availability Zones to increase the fault tolerance of your applications.

**TASKS:**

- **Prerequisites**
- **STEP-1: Set up a launch template or launch configuration**
- **STEP-2: Create an auto scaling group**
- **STEP-3: Verify that your load balancing is attached**

### Prerequisites

- A load balancer and target group. Make sure to choose the same Availability Zones for the load balancer that you plan to use for your Auto Scaling group.
- A security group for your launch template or launch configuration. The security group must allow access from the load balancer on both the listener port (usually port 80 for HTTP traffic) and the port that you want Elastic Load Balancing to use for health checks. Optionally, if your instances will have public IP addresses, you can allow SSH traffic for connecting to the instances.
- A virtual private cloud (VPC). This tutorial refers to the default VPC, but you can use your own. If using your own VPC, make sure that it has a subnet mapped to each Availability Zone of the Region you are working in. At minimum, you must have two public subnets available to create the load balancer. You must also have either two

private subnets or two public subnets to create your Auto Scaling group and register it with the load balancer.

---

## Step 1: Set up a launch template or launch configuration

Use either a launch template or a launch configuration

If you already have a launch template that you'd like to use, select it by using the following procedure.

### To select an existing launch template

1. Open the launch templates page of the Amazon EC2 console.
2. On the navigation bar at the top of the screen, choose the Region where the load balancer was created.
3. Select a launch template.
4. Choose **Actions**, **Create Auto Scaling group**.
5. Alternatively, to create a new launch template, use the following procedure.

### To create a launch template

1. Open the launch templates page of the Amazon EC2 console.
2. On the navigation bar at the top of the screen, choose the Region where the load balancer was created.
3. Choose **Create launch template**.
4. Enter a name and provide a description for the initial version of the launch template.
5. For **Application and OS Images (Amazon Machine Image)**, choose the ID of the AMI for your instances. You can search through all available AMIs, or select an AMI from the **Recents** or **Quick Start** list. If you don't see the AMI that you need, choose **Browse more AMIs** to browse the full AMI catalog.
6. For **Instance type**, select a hardware configuration for your instances that is compatible with the AMI that you specified.
7. For **Network settings**, expand **Advanced network configuration** and do the following:
    a) Choose **Add network interface** to configure the primary network interface.
    b) For **Security group ID**, specify a security group for your instances from the same VPC as the load balancer.
    c) For **Delete on termination**, choose **Yes**. This deletes the network interface when the Auto Scaling group scales in, and terminates the instance to which the network interface is attached.
8. Choose **Create launch template**.
9. On the confirmation page, choose **Create Auto Scaling group**.

## Select or create a launch configuration

If you already have a launch configuration that you'd like to use, select it by using the following procedure.

### To select an existing launch configuration

1. Open the launch configuration page of the Amazon EC2 console.
2. On the navigation bar at the top of the screen, choose the Region where the load balancer was created.
3. Select a launch configuration.
4. Choose **Actions**, **Create Auto Scaling group**.

Alternatively, to create a new launch configuration, use the following procedure.

### To create a launch configuration

1. Open the launch configuration page of the Amazon EC2 console.
2. On the navigation bar at the top of the screen, choose the Region where the load balancer was created.
3. Choose **Create launch configuration**, and enter a name for your launch configuration.
4. For **Amazon machine image (AMI)**, enter the ID of the AMI for your instances as search criteria.
5. For **Instance type**, select a hardware configuration for your instance.
6. Under **Additional configuration**, pay attention to the following fields:
7. For **Security groups**, choose an existing security group from the same VPC as the load balancer. If you keep the **Create a new security group** option selected, a default SSH rule is configured for Amazon EC2 instances running Linux. A default RDP rule is configured for Amazon EC2 instances running Windows.
8. For **Key pair (login)**, choose an option under **Key pair options**.

   If you've already configured an Amazon EC2 instance key pair, you can choose it here.

   If you don't already have an Amazon EC2 instance key pair, choose **Create a new key pair** and give it a recognizable name. Choose **Download key pair** to download the key pair to your computer.
9. Select the acknowledgment check box, and then choose **Create launch configuration**.
10. Select the check box next to the name of your new launch configuration and choose **Actions**, **Create Auto Scaling group**.

---

## Step 2: Create an Auto Scaling group

Use the following procedure to continue where you left off after creating or selecting your launch template or launch configuration.

## To create an Auto Scaling group

1. On the **Choose launch template or configuration** page, for **Auto Scaling group name**, enter a name for your Auto Scaling group.
2. [Launch template only] For **Launch template**, choose whether the Auto Scaling group uses the default, the latest, or a specific version of the launch template when scaling out.
3. Choose **Next**.
4. The **Choose instance launch options** page appears, allowing you to choose the VPC network settings you want the Auto Scaling group to use and giving you options for launching On-Demand and Spot Instances (if you chose a launch template).
5. In the **Network** section, for **VPC**, choose the VPC that you used for your load balancer. If you chose the default VPC, it is automatically configured to provide internet connectivity to your instances. This VPC includes a public subnet in each Availability Zone in the Region.
6. For **Availability Zones and subnets**, choose one or more subnets from each Availability Zone that you want to include. [Launch template only] In the **Instance type requirements** section, use the default setting to simplify this step. (Do not override the launch template.) you will launch only On-Demand Instances using the instance type specified in your launch template.
7. Choose **Next** to go to the **Configure advanced options** page.
8. To attach the group to an existing load balancer, in the **Load balancing** section, choose **Attach to an existing load balancer**. You can choose **Choose from your load balancer target groups** or **Choose from Classic Load Balancers**. You can then choose the name of a target group for the Application Load Balancer or Network Load Balancer you created, or choose the name of a Classic Load Balancer.
9. (Optional) To use Elastic Load Balancing health checks, for **Health checks**, choose **ELB** under **Health check type**.
10. When you have finished configuring the Auto Scaling group, choose **Skip to review**.
11. On the **Review** page, review the details of your Auto Scaling group. You can choose **Edit** to make changes. When you are finished, choose **Create Auto Scaling group**.

After you have created the Auto Scaling group with the load balancer attached, the load balancer automatically registers new instances as they come online. You have only one instance at this point, so there isn't much to register. However, you can add additional instances by updating the desired capacity of the group.

## Step 3: Verify that your load balancer is attached

### To verify that your load balancer is attached

1. From the auto scaling groups page of the Amazon EC2 console, select the check box next to your Auto Scaling group.
2. On the **Details** tab, **Load balancing** shows any attached load balancer target groups or Classic Load Balancers.
3. On the **Activity** tab, in **Activity history**, you can verify that your instances launched successfully. The **Status** column shows whether your Auto Scaling group has successfully launched instances. If your instances fail to launch, you can find troubleshooting ideas for common instance launch issues in troubleshoot amazon EC2 auto scaling.
4. On the **Instance management** tab, under **Instances**, you can verify that your instances are ready to receive traffic. Initially, your instances are in the Pending state. After an instance is ready to receive traffic, its state is inservice. The **Health status** column shows the result of the Amazon EC2 Auto Scaling health checks on your instances. Although an instance may be marked as healthy, the load balancer will only send traffic to instances that pass the load balancer health checks.
5. Verify that your instances are registered with the load balancer. Open the target groups of the Amazon EC2 console. Select your target group, and then choose the **Targets** tab. If the state of your instances is initial, it's probably because they are still in the process of being registered, or they are still undergoing health checks. When the state of your instances is healthy, they are ready for use.

## 3. How to Transition to latency-based routing in Amazon Route 53.

### Transitioning to latency-based routing in Amazon Route 53

With latency-based routing, Amazon Route 53 can direct your users to the lowest-latency AWS endpoint available. For example, you might associate a DNS name like www.example.com with an ELB Classic, Application, or Network Load Balancer, or with Amazon EC2 instances or Elastic IP addresses that are hosted in the US East (Ohio) and Europe (Ireland) regions. The Route 53 DNS servers decide, based on network conditions of the past couple of weeks, which instances in which regions should serve particular users. A user in London will likely be directed to the Europe (Ireland) instance, a user in Chicago will likely be directed to the US East (Ohio) instance, and so on. Route 53 supports latency-based routing for A, AAAA, TXT, and CNAME records, as well as aliases to A and AAAA records.

**Note**

Data about the latency between users and your resources is based entirely on traffic between users and AWS data centers. If you aren't using resources in an AWS Region, the actual latency between your users and your resources can vary significantly from AWS latency data. This is true even if your resources are located in the same city as an AWS Region.

For a smooth, low-risk transition, you can combine weighted and latency records to gradually migrate from standard routing to latency-based routing with full control and rollback capability at each stage. Let's consider an example in which www.example.com is currently hosted on an Amazon EC2 instance in the US East (Ohio) region. The instance has the Elastic IP address W.W.W.W. Suppose you want to continue routing traffic to the US East (Ohio) region when applicable while also beginning to direct users to additional Amazon EC2 instances in the US West (N. California) region (Elastic IP X.X.X.X) and in the Europe (Ireland) region (Elastic IP Y.Y.Y.Y). The Route 53 hosted zone for example.com already has a record for www.example.com that has a Type of A and a Value (an IP address) of W.W.W.W.

- When you're finished with the following example, you'll have two weighted alias records:

- You'll convert your existing record for www.example.com into a weighted alias record that continues to direct the majority of your traffic to your existing Amazon EC2 instance in the US East (Ohio) region.
- You'll create another weighted alias record that initially directs only a small portion of your traffic to your latency records, which route traffic to all three regions.

- By updating the weights in these weighted alias records, you can gradually shift from routing traffic only to the US East (Ohio) region to routing traffic to all three regions in which you have Amazon EC2 instances.

**To transition to latency-based routing**

1. Make a copy of the record for www.example.com, but use a new domain name, for example, copy-www.example.com. Give the new record the same Type (A) and Value (W.W.W.W) as the record for www.example.com.
2. Update the existing A record for www.example.com to make it a weighted alias record:
   - For Value/Route traffic to, choose Alias to another record in this hosted zone, and specify copy-www.example.com.

- For Weight, specify 100.
- When you're finished with the update, Route 53 will continue to use this record to route all traffic to the resource that has an IP address of W.W.W.W.

3. Create a latency record for each of your Amazon EC2 instances, for example:
   - US East (Ohio), Elastic IP address W.W.W.W
   - US West (N. California), Elastic IP address X.X.X.X
   - Europe (Ireland), Elastic IP address Y.Y.Y.Y

   Give all of the latency records the same domain name, for example, www-lbr.example.com and the same type, A.

   When you're finished creating the latency records, Route 53 will continue to route traffic using the record that you updated in Step 2.

   You can use www-lbr.example.com for validation testing, for example, to ensure that each endpoint can accept requests.
4. Let's now add the www-lbr.example.com latency record into the www.example.com weighted record and begin routing limited traffic to the corresponding Amazon EC2 instances. This means that the Amazon EC2 instance in the US East (Ohio) region will be getting traffic from both weighted records.

   Create another weighted alias record for www.example.com:
   - For Value/Route traffic to, choose Alias to another record in this hosted zone, and specify www-lbr.example.com.
   - For Weight, specify 1.

   When you finish and your changes are synchronized to Route 53 servers, Route 53 will begin to route a tiny fraction of your traffic (1/101) to the Amazon EC2 instances for which you created latency records in Step 3.
5. As you develop confidence that your endpoints are adequately scaled for the incoming traffic, adjust the weights accordingly. For example, if you want 10% of your requests to be based on latency-based routing, change the weights to 90 and 10, respectively.

### 4. What is the difference between Scalability and Elasticity?

Scalability is used to fulfill the static needs while elasticity is used to fulfill the dynamic need of the organization. Scalability is a similar kind of service provided by the cloud where the customers have to pay-per-use.

### 5. How do I decide which load balancer to select for my application?

Elastic Load Balancing (ELB) supports four types of load balancers. You can select the appropriate load balancer based on your application needs. If you need to load balance HTTP requests, It is recommend to use the Application Load Balancer (ALB).

# AWS DATABASE SERVICES

# HOMEWORK-6

**1. Problem Statement: Create a web server and an Amazon RDS DB instance**

## Create a web server and an Amazon RDS DB instance

The web server runs on an Amazon EC2 instance using Amazon Linux, and the MySQL database is a MySQL DB instance. Both the Amazon EC2 instance and the DB instance run in a virtual private cloud (VPC) based on the Amazon VPC service.

we can create an EC2 instance that uses the default VPC, subnets, and security group for your AWS account. The following process shows you how to create the DB instance and automatically set up connectivity with the EC2 instance that you created. Then shows you how to install the web server on the EC2 instance. You connect your web server to your DB instance in the VPC using the DB instance endpoint.

- Launch an EC2 instance
- Create a database instance
- Install a web server on your EC2 instance

Launch an EC2 instance:

Create an Amazon EC2 instance in the public subnet of your VPC.

### To launch an EC2 instance

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the upper-right corner of the AWS Management Console, choose the AWS Region where you want to create the EC2 instance.
3. Choose **EC2 Dashboard**, and then choose **Launch instance**, as shown following.
4. Make sure you have opted into the new launch experience.
5. Under **Name and tags**, for **Name**, enter tutorial-ec2-instance-web-server.
6. Under **Application and OS Images (Amazon Machine Image)**, choose **Amazon Linux**, and then choose the **Amazon Linux 2 AMI**. Keep the defaults for the other choices.
7. Under **Instance type**, choose **t2.micro**.

8. Under **Key pair (login)**, choose a **Key pair name** to use an existing key pair. To create a new key pair for the Amazon EC2 instance, choose **Create new key pair** and then use the **Create key pair** window to create it.
9. Under **Network settings**, set these values and keep the other values as their defaults:
   - For **Allow SSH traffic from**, choose the source of SSH connections to the EC2 instance.
   - You can choose **My IP** if the displayed IP address is correct for SSH connections.
   - Otherwise, you can determine the IP address to use to connect to EC2 instances in your VPC using Secure Shell (SSH). To determine your public IP address, in a different browser window or tab, you can use the service at https://checkip.amazonaws.com. An example of an IP address is 203.0.113.25/32.

   In many cases, you might connect through an internet service provider (ISP) or from behind your firewall without a static IP address. If so, make sure to determine the range of IP addresses used by client computers.

   **Warning**

   If you use 0.0.0.0/0 for SSH access, you make it possible for all IP addresses to access your public instances using SSH. This approach is acceptable for a short time in a test environment, but it's unsafe for production environments. In production, authorize only a specific IP address or range of addresses to access your instances using SSH.
   - Turn on **Allow HTTPs traffic from the internet**.
   - Turn on **Allow HTTP traffic from the internet**.
10. Leave the default values for the remaining sections.
11. Review a summary of your instance configuration in the **Summary** panel, and when you're ready, choose **Launch instance**.
12. On the **Launch Status** page, shown following, note the identifier for your new EC2 instance, for example: i-03a6ad47e97ba9dc5.
13. Choose **View all instances** to find your instance.
14. Wait until **Instance state** for your instance is **Running** before continuing.
15. Complete create a DB instance.

Create a DB instance:

Create an Amazon RDS for MySQL DB instance that maintains the data used by a web application.

### To create a MySQL DB instance

1. Sign in to the AWS Management Console and open the Amazon RDS console at https://console.aws.amazon.com/rds/.
2. In the upper-right corner of the AWS Management Console, check the AWS Region. It should be the same as the one where you created your EC2 instance.

3. In the navigation pane, choose **Databases**.
4. Choose **Create database**.
5. On the **Create database** page, shown following, make sure that the **Standard create** option is chosen, and then choose **MySQL**.
6. In the **Templates** section, choose **Free tier**.
7. In the **Availability and durability** section, keep the defaults.
8. In the **Settings** section, set these values:
   - **DB instance identifier** – Type tutorial-db-instance.
   - **Master username** – Type tutorial username.
   - **Auto generate a password** – Leave the option turned off.
   - **Master password** – Type a password.
   - **Confirm password** – Retype the password.
9. In the **Instance configuration** section, set these values:
   - **Burstable classes (includes t classes)**
   - **db.t3.micro**
10. In the **Storage** section, keep the defaults.
11. In the **Connectivity** section, set these values and keep the other values as their defaults:
    - For **Compute resource**, choose **Connect to an EC2 compute resource**.
    - For **EC2 instance**, choose the EC2 instance you created previously, such as **tutorial-ec2-instance-web-server**.
12. In the **Database authentication** section, make sure **Password authentication** is selected.
13. Open the **Additional configuration** section, and enter sample for **Initial database name**. Keep the default settings for the other options.
14. To create your MySQL DB instance, choose **Create database**.

    Your new DB instance appears in the **Databases** list with the status **Creating**.
15. Wait for the **Status** of your new DB instance to show as **Available**. Then choose the DB instance name to show its details.
16. In the **Connectivity & security** section, view the **Endpoint** and **Port** of the DB instance. Note the endpoint and port for your DB instance. You use this information to connect your web server to your DB instance.
17. Complete install a web server on EC2 instance.

Install a web server on your EC2 instance:

Install a web server on the EC2 instance you created in Launch an EC2 instance. The web server connects to the Amazon RDS DB instance that you created in Create a DB instance.

## Install an Apache web server with PHP and MariaDB

Connect to your EC2 instance and install the web server.

## To connect to your EC2 instance and install the Apache web server with PHP

1. Connect to the EC2 instance that you created earlier by following the steps in Connect to your linux instance. Get the latest bug fixes and security updates by updating the software on your EC2 instance. To do this, use the following command.
2. After the updates complete, install the PHP software using the amazon-linux-extras install command. This command installs multiple software packages and related dependencies at the same time.

```
sudo amazon-linux-extras install php8.0 mariadb10.5
```

If you receive an error stating sudo: amazon-linux-extras: command not found, your instance wasn't launched with an Amazon Linux 2 AMI. You might be using the Amazon Linux AMI instead. You can view your version of Amazon Linux using the following command.

```
cat /etc/system-release
```

3. Install the Apache web server.

```
sudo yum install -y httpd
```

4. Start the web server with the command shown following.

```
sudo systemctl start httpd
```

You can test that your web server is properly installed and started. To do this, enter the public Domain Name System (DNS) name of your EC2 instance in the address bar of a web browser, for example: http://ec2-42-8-168-21.us-west-1.compute.amazonaws.com. If your web server is running, then you see the Apache test page.

If you don't see the Apache test page, check your inbound rules for the VPC security group that you created in Create a VPC for use with a DB instance(IPv4 only). Make sure that your inbound rules include one allowing HTTP (port 80) access for the IP address to connect to the web server.

5. Configure the web server to start with each system boot using the systemctl command.

```
sudo systemctl enable httpd
```

To allow ec2-user to manage files in the default root directory for your Apache web server, modify the ownership and permissions of the /var/www directory. There are many ways to accomplish this task. In this tutorial, you add ec2-user to the apache group, to give the apache group ownership of the /var/www directory and assign write permissions to the group.

## To set file permissions for the Apache web server

1. Add the ec2-user user to the apache group.

```
sudo usermod -a -G apache ec2-user
```

2. Log out to refresh your permissions and include the new apache group.

```
exit
```

3. Log back in again and verify that the apache group exists with the groups command.

```
groups
```

Your output looks similar to the following:

```
ec2-user adm wheel apache systemd-journal
```

4. Change the group ownership of the /var/www directory and its contents to the apache group.

```
sudo chown -R ec2-user:apache /var/www
```

5. Change the directory permissions of /var/www and its subdirectories to add group write permissions and set the group ID on subdirectories created in the future.

6. sudo chmod 2775 /var/www

```
find /var/www -type d -exec sudo chmod 2775 {} \;
```

7. Recursively change the permissions for files in the /var/www directory and its subdirectories to add group write permissions.

```
find /var/www -type f -exec sudo chmod 0664 {} \;
```

Now, ec2-user (and any future members of the apache group) can add, delete, and edit files in the Apache document root. This makes it possible for you to add content, such as a static website or a PHP application.

**Note**

A web server running the HTTP protocol provides no transport security for the data that it sends or receives. When you connect to an HTTP server using a web browser, much information is visible to eavesdroppers anywhere along the network pathway. This information includes the URLs that you visit, the content of webpages that you receive, and the contents (including passwords) of any HTML forms.

The best practice for securing your web server is to install support for HTTPS (HTTP Secure). This protocol protects your data with SSL/TLS encryption.

## Connect your Apache web server to your DB instance

Next, you add content to your Apache web server that connects to your Amazon RDS DB instance.

### To add content to the Apache web server that connects to your DB instance

1. While still connected to your EC2 instance, change the directory to /var/www and create a new subdirectory named inc.

```
2.    cd /var/www

3.    mkdir inc
```

```
cd inc
```

4. Create a new file in the inc directory named dbinfo.inc, and then edit the file by calling nano (or the editor of your choice).

```
5.    >dbinfo.inc
```

```
nano dbinfo.inc
```

6. Add the following contents to the dbinfo.inc file. Here, *db_instance_endpoint* is your DB instance endpoint, without the port, and *master password* is the master password for your DB instance.

```
<?php

define('DB_SERVER', 'db_instance_endpoint');

define('DB_USERNAME', 'tutorial_user');

define('DB_PASSWORD', 'master password');

define('DB_DATABASE', 'sample');

?>
```

7. Save and close the dbinfo.inc file.
8. Change the directory to /var/www/html.

```
cd /var/www/html
```

9. Create a new file in the html directory named SamplePage.php, and then edit the file by calling nano (or the editor of your choice).

```
>SamplePage.php
```

10. nano SamplePage.php

11. Add the following contents to the SamplePage.php file:

```
<?php include "../inc/dbinfo.inc"; ?>

<html>

<body>

<h1>Sample page</h1>

<?php

  /* Connect to MySQL and select the database. */

  $connection = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD);

if (mysqli_connect_errno()) echo "Failed to connect to MySQL: " . mysqli_connect_error();

$database = mysqli_select_db($connection, DB_DATABASE);

  /* Ensure that the EMPLOYEES table exists. */

VerifyEmployeesTable($connection, DB_DATABASE);

/* If input fields are populated, add a row to the EMPLOYEES table. */

$employee_name = htmlentities($_POST['NAME']);

 $employee_address = htmlentities($_POST['ADDRESS']);

  if (strlen($employee_name) || strlen($employee_address)) {

  AddEmployee($connection, $employee_name, $employee_address);

  }

?>

<!-- Input form -->
```

```
<form action="<?PHP echo $_SERVER['SCRIPT_NAME'] ?>" method="POST">
  <table border="0">
    <tr>
     <td>NAME</td>
      <td>ADDRESS</td>
   </tr>
  <tr>
    <td>
     <input type="text" name="NAME" maxlength="45" size="30" />
     </td>
    <td>
        <input type="text" name="ADDRESS" maxlength="90" size="60" />
      </td>
    <td>
     <input type="submit" value="Add Data" />
     </td>
    </tr>
  </table>
</form>
<!-- Display table data. -->
<table border="1" cellpadding="2" cellspacing="2">
  <tr>
    <td>ID</td>
    <td>NAME</td>
    <td>ADDRESS</td>
```

```
 </tr>
<?php
$result = mysqli_query($connection, "SELECT * FROM EMPLOYEES");
while($query_data = mysqli_fetch_row($result)) {
  echo "<tr>";
  echo "<td>",$query_data[0], "</td>",
       "<td>",$query_data[1], "</td>",
       "<td>",$query_data[2], "</td>";
  echo "</tr>";
}
?>
</table>
<!-- Clean up. -->
<?php
mysqli_free_result($result);
  mysqli_close($connection);
?>
</body>
</html>
<?php
/* Add an employee to the table. */
function AddEmployee($connection, $name, $address) {
 $n = mysqli_real_escape_string($connection, $name);
   $a = mysqli_real_escape_string($connection, $address);
$query = "INSERT INTO EMPLOYEES (NAME, ADDRESS) VALUES ('$n', '$a');";
```

```
   if(!mysqli_query($connection, $query)) echo("<p>Error adding employee data.</p>");
}
   /* Check whether the table exists and, if not, create it. */
   function VerifyEmployeesTable($connection, $dbName) {
     if(!TableExists("EMPLOYEES", $connection, $dbName))
     {
   $query = "CREATE TABLE EMPLOYEES (
    ID int(11) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
           NAME VARCHAR(45),
            ADDRESS VARCHAR(90)
          )";
        if(!mysqli_query($connection, $query)) echo("<p>Error creating table.</p>");
     }
   }
   /* Check for the existence of a table. */
   function TableExists($tableName, $connection, $dbName) {
     $t = mysqli_real_escape_string($connection, $tableName);
     $d = mysqli_real_escape_string($connection, $dbName);
   $checktable = mysqli_query($connection, SELECT TABLE_NAME FROM information_schema.TABLES WHERE TABLE_NAME = '$t' AND TABLE_SCHEMA = '$d'");
   if(mysqli_num_rows($checktable) > 0)
   return true;
   return false;
}?>
```

12. Save and close the SamplePage.php file.
13. Verify that your web server successfully connects to your DB instance by opening a web browser and browsing to http://*EC2 instance endpoint*/SamplePage.php, for example: http://ec2-55-122-41-31.us-west-2.compute.amazonaws.com/SamplePage.php.

You can use SamplePage.php to add data to your DB instance. The data that you add is then displayed on the page. To verify that the data was inserted into the table, install MySQL client on the Amazon EC2 instance. Then connect to the DB instance and query the table.

To make sure that your DB instance is as secure as possible, verify that sources outside of the VPC can't connect to your DB instance.

After you have finished testing your web server and your database, you should delete your DB instance and your Amazon EC2 instance.

## 2. How to monitor your estimated charges using CloudWatch?

Monitor your estimated charges using CloudWatch:

In this scenario, you create an Amazon CloudWatch alarm to monitor your estimated charges. When you enable the monitoring of estimated charges for your AWS account, the estimated charges are calculated and sent several times daily to CloudWatch as metric data.

Billing metric data is stored in the US East (N. Virginia) Region and reflects worldwide charges. This data includes the estimated charges for every service in AWS that you use, as well as the estimated overall total of your AWS charges.

You can choose to receive alerts by email when charges have exceeded a certain threshold. These alerts are triggered by CloudWatch and messages are sent using Amazon Simple Notification Service (Amazon SNS).

**Tasks:**

- STEP-1: Enable billing alerts
- STEP-2:Create a billing alarm
- STEP-3: Check the alarm status
- STEP-4: Edit a billing alarm

- STEP-5: Delete a billing alarm

---

## Step 1: Enable billing alerts

Before you can create an alarm for your estimated charges, you must enable billing alerts, so that you can monitor your estimated AWS charges and create an alarm using billing metric data. After you enable billing alerts, you cannot disable data collection, but you can delete any billing alarms that you created.

After you enable billing alerts for the first time, it takes about 15 minutes before you can view billing data and set billing alarms.

### Requirements

- You must be signed in using account root user credentials or as an IAM user that has been given permission to view billing information.
- For consolidated billing accounts, billing data for each linked account can be found by logging in as the paying account. You can view billing data for total estimated charges and estimated charges by service for each linked account, in addition to the consolidated account.
- In a consolidated billing account, member linked account metrics are captured only if the payer account enables the **Receive Billing Alerts** preference. If you change which account is your management/payer account, you must enable the billing alerts in the new management/payer account.
- The account must not be part of the Amazon Partner Network (APN) because billing metrics are not published to CloudWatch for APN accounts.

### To enable monitoring of your estimated charges

1. Open the AWS Billing console at https://console.aws.amazon.com/billing/.
2. In the navigation pane, choose **Preferences**.
3. Select **Receive Billing Alerts**.
4. Choose **Save preferences**.

---

## Step 2: Create a billing alarm

### Important

Before you create a billing alarm, you must set your Region to US East (N. Virginia). Billing metric data is stored in this Region and represents worldwide charges. You also must enable billing alerts for your account or in the management/payer account (if you are using consolidated billing).

In this procedure, you create an alarm that sends a notification when your estimated charges for AWS exceed a defined threshold.

## To create a billing alarm using the CloudWatch console

1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Alarms**, and then choose **All alarms**.
3. Choose **Create alarm**.
4. Choose **Select metric**. In **Browse**, choose **Billing**, and then choose **Total Estimated Charge**.

### Note

If you dont't see the **Billing/Total Estimated Charge** metric, enable billing alerts, and change your Region to US East (N. Virginia).

5. Select the box for the **EstimatedCharges** metric, and then choose **Select metric**.
6. For **Threshold type**, choose **Static**.
7. For **Whenever EstimatedCharges is . . .**, choose **Greater**.
8. For **than . . .**, define a threshold value that triggers your alarm (for example, 200 USD).

### Note

After you define a threshold value, the preview graph displays your estimated charges for the current month.

9. Choose **Next**.
10. Under **Notification**, specify an Amazon SNS topic to be notified when your alarm is in the ALARM state. You can select an existing Amazon SNS topic, create a new Amazon SNS topic, or use a topic ARN to notify other account. If you want your alarm to send multipl notifications for the same alarm state or for different alarm states, choose **Add notification**.
11. Choose **Next**.
12. Under **Name and description**, enter a name for your alarm.
    i) (Optional) Enter a description of your alarm.
13. Under **Preview and create**, make sure that your configuration is correct, and then choose **Create alarm**.

---

## Step 3: Check the alarm status

Now, check the status of the billing alarm that you just created.

### To check the alarm status

1. Open the CloudWatch console at [https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/).
2. If necessary, change the Region to US East (N. Virginia). Billing metric data is stored in this Region and reflects worldwide charges.
3. In the navigation pane, choose **Alarms**.
4. Select the check box next to the alarm. Until the subscription is confirmed, it is shown as "Pending confirmation". After the subscription is confirmed, refresh the console to show the updated status.

---

## Step 4: Edit a billing alarm

For example, you may want to increase the amount of money you spend with AWS each month from $200 to $400. You can edit your existing billing alarm and increase the monetary amount that must be exceeded before the alarm is triggered.

### To edit a billing alarm

1. Open the CloudWatch console at [https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/).
2. If necessary, change the Region to US East (N. Virginia). Billing metric data is stored in this Region and reflects worldwide charges.
3. In the navigation pane, choose **Alarms**.
4. Select the check box next to the alarm and choose **Actions**, **Modify**.
5. For **Whenever my total AWS charges for the month exceed**, specify the new amount that must be exceeded to trigger the alarm and send an email notification.
6. Choose **Save Changes**.

---

## Step 5: Delete a billing alarm

If you no longer need your billing alarm, you can delete it.

### To delete a billing alarm

1. Open the CloudWatch console at [https://console.aws.amazon.com/cloudwatch/](https://console.aws.amazon.com/cloudwatch/).
2. If necessary, change the Region to US East (N. Virginia). Billing metric data is stored in this Region and reflects worldwide charges.
3. In the navigation pane, choose **Alarms**.
4. Select the check box next to the alarm and choose **Actions**, **Delete**.
5. When prompted for confirmation, choose **Yes, Delete**.

**3. How are Amazon RDS, DynamoDB, and Redshift different?**

**RDS -** RDS's storage limit depends on which engine you're running, but it tops out at 64 TB using Amazon Aurora. SQL accommodates 16 TB, and all the other engines allow for 32TB**.**

**Redshift -** Redshift's max capacity is much higher at 2PB.

**DynamoDB -** DynamoDB has limitless storage capacity.

**4. Can I run more than one DB instance for Amazon RDS for free?**

**Yes**. You can run more than one Single-AZ Micro DB instance simultaneously and be eligible for usage counted under the AWS Free Tier for Amazon RDS**.**

**5. Can I retrieve only a specific element of the data if I have nested JSON data in DynamoDB?**

**Yes**, when using GetItem, BatchGetItem, Query, or Scan APIs you can define a ProjectionExpression to determine which attributes should be retrieved from the table.

**6. What happens to my backups and DB Snapshots if I delete my DB Instance?**

Automated backups are deleted when the DB instance is deleted. **Only manually created DB Snapshots are retained after the DB Instance is deleted**.

**7. How can I load my data to Amazon Redshift from different data sources like Amazon RDS, Amazon DynamoDB, and Amazon EC2?**

On the contrary, using the COPY command is the suggested way to load large volumes of data on an Amazon Redshift cluster.

**COPY command parameters:**

- Data source, the place where we pull the data.
- Table Name, the destination of where we store the data.
- Authorization, credentials to access the data on the data source.

## 8. What platforms does the CloudWatch Logs Agent support?

**Supported operating systems:**

The CloudWatch agent is supported on x86-64 architecture on the following operating systems:

- Amazon Linux version 2014.03.02 or later
- Amazon Linux 2
- Ubuntu Server versions 20.04, 18.04, 16.04, and 14.04
- CentOS version 8 stream and version 7
- Red Hat Enterprise Linux (RHEL) versions 8.4, 8.3, 8.2, 8.1, 8.0, 7.7, 7.6, 7.5, 7.4, 7.2, and 7.0
- Debian version 10
- SUSE Linux Enterprise Server (SLES) version 15 and version 12
- Oracle Linux versions 8.6, 8.4, 8.3, 8.2, 8.1, 7.8, 7.6, and 7.5
- macOS, including EC2 Mac1 instances
- 64-bit versions of Windows Server 2022, Windows Server 2019, Windows Server 2016, and Windows Server 2012 R2
- 64-bit Windows 10 running on x86-64 computers

**The agent is supported on ARM64 architecture on the following operating systems:**

- Amazon Linux 2
- Ubuntu Server versions 20.04 and 18.04
- Red Hat Enterprise Linux (RHEL) version 7.6
- SUSE Linux Enterprise Server 15

# SECURITY AND ARCHITECTURAL DESIGN IN AWS

## HOMEWORK-7

### 1. Problem Statement: Configure WAF to Protect Website from Attacks

Resolution

Migrate from AWS WAF Classic to AWS WAF (if applicable)

If you're using AWS WAF Classic, it's recommended that you migrate to AWS WAF. To make this migration, you can leverage the automated migration tool.

Run penetration testing against your application to identify vulnerabilities

Every application receives its own type of requests. As a result, the firewall rules that protect your application must be customized.

Run penetration testing against your application to understand its specific vulnerabilities.

Review incoming requests to optimize your custom rules

Before creating custom rules to protect your application, review the incoming requests in your environment.

First, generate logs using:

- AWS WAF
- Amazon CloudFront
- Application Load Balancer
- Amazon API Gateway

Then, store these logs on Amazon Simple Storage Service (Amazon S3). Finally, use Amazon athena to query the logs and identify patterns. For example, you might see patterns like:

- Requests made to your environment for URIs that don't exist
  - To recognize this pattern, you must know every supported URI
  - Example Athena query performed on AWS WAF logs to count requests for each URI:

```
SELECT count("httprequest"."uri") as URIcount, "httprequest"."uri"
FROM waf_logs
GROUP BY "httprequest"."uri"
ORDER BY URIcount DESC
```

- Requests that contain an HTTP Host header that's unsupported by your webserver -OR- requests that contain an IP address instead of your website's domain name
  - Example Athena query performed on AWS WAF logs to count requests with different Host header values:

```
SELECT header.value as HostHeader, count(header) as count
FROM waf_logs, UNNEST(httprequest.headers) AS x(header)
WHERE "header"."name" = 'Host'
GROUP BY  header
ORDER BY count DESC
```

After identifying a pattern, you can create AWS WAF rules in COUNT mode to verify that the rule is configured to match those requests. Then, move the rule to BLOCK mode.

For example, if your application only supports Host header "www.example.com":

- Create a NOT match on the HOST header with value "www.example.com"
- Set the action to BLOCK

Any requests to your environment that don't have a Host header of "www.example.com" are now blocked.
Note: This rule also blocks requests to the AWS-provided fully qualified domain name (FQDN).

Use AWS Managed Rules to protect against common attacks

Use AWS managed rules to prevent common attacks that apply to most applications, including requests that:

- Don't contain a User-Agent
- Represent bot requests
- Use “localhost” as the HTTP Host header
- Use the PROPFIND HTTP method

Include these baseline rule groups in your web access control list (web ACL) in COUNT mode. Be sure to choose “Enable Count mode” in the rule group. Then, review the AWS WAF logs and CLOUDWATCH metrics to determine whether the managed rule matches any legitimate traffic. If it doesn't, move the rule group to BLOCK by disabling “Enable Count mode”. To disable a specific rule in the AWS Managed Rule Group, choose “Override rules action” for that rule.

**Important**: AWS Managed Rules are designed to protect you from common web threats. When used in accordance with the documentation, AWS Managed Rules rule groups add another layer of security for your applications. However, AWS Managed Rules rule groups aren't intended as a replacement for your security responsibilities, which are determined by the AWS resources that you select.

### Baseline your AWS WAF using the rate of legitimate requests

Perform an analysis of your traffic to identify the number of requests made by legitimate client IP addresses using Amazon athena or Amazon Quicksight on the AWS WAF logs. Using the information you get from this analysis, baseline your AWS WAF to the rate of requests made by a legitimate client. Then, set up a threshold while configuring the AWS WAF rate-based rule.

Sample Athena query performed on AWS WAF logs to count the number of requests from a single IP address (x.x.x.x) between a given timeframe (Nov 16th 2020 9AM-10AM):

```
SELECT  "httprequest"."clientip", "count"(*) "count", "httprequest"."country"
FROM waf_logs
WHERE httprequest.clientip LIKE 'x.x.x.x' and
date_format(from_unixtime("timestamp"/1000), '%Y-%m-%d %h:%i:%s') between
'2020-11-16 09:00:00' and '2020-11-16 10:00:00'
GROUP BY "httprequest"."clientip", "httprequest"."country"
```

Sample Athena query performed on AWS WAF logs to count the number of requests from all IP addresses between the same time frame:

```
SELECT "httprequest"."clientip", "count"(*) "count", "httprequest"."country"
FROM waf_logs
WHERE date_format(from_unixtime("timestamp"/1000), '%Y-%m-%d %h:%i:%s')
between '2020-11-16 09:00:00' and '2020-11-16 10:00:00'
GROUP BY "httprequest"."clientip", "httprequest"."country"
ORDER BY "count" DESC
```

### Use the AWS WAF Security Automations template to prevent common attacks

Use the AWS WAF Security Automation template to provide additional protection from common attacks. For example, you can enable protection against:

- Scanners and probes
- Bad bots
- Bad IP addresses

Note: This solution uses other AWS services that incur costs.

Protect against SQL injection and cross-site scripting

To protect your applications against SQL injection and cross-site scripting (XSS) attacks, use the built-in SQL injection and cross-site scripting engines. Remember that attacks can be performed on different parts of the HTTP request, such as the HTTP header, query string, or URI. Configure the AWS WAF rules to inspect different parts of the HTTP request against the built-in mitigation engines.

Restrict access from CloudFront (if you're using AWS WAF on CloudFront)

- Restrict access based on cloudfront IP addresses
- Add a customer header in CloudFront for origin requests. On the origin, allow access only if the custom header and value are present. If the origin is an Application Load Balancer or API Gateway, use AWS WAF on the origin to allow requests that contain the custom header and value.

**2. Why do you want to migrate to the cloud?**

**Improved Efficiency**: After migrating to the cloud, you no longer need to worry about power requirements, space considerations, expensive computer hardware, or software updates. You get to keep your entire company focused on generating revenue and relationships, not on IT.

**3. Explain the three phases of migration AWS.**

**Assess phase**

- You build the business case for the migration. Mobilize phase.
- You prepare the organization and mobilize the resources needed for the migration.
- Migrate and modernize phase of a large migration project.
- You use your strategy, plan, and the best practices to migrate and modernize.

**4. Where is my data encrypted if I use AWS KMS?**

You can request that AWS KMS generate data keys and return them for use in your own application. The data keys are encrypted **under a root key you define in AWS KMS** so that you can safely store the encrypted data key along with your encrypted data.

**5. How can I prevent GuardDuty from looking at my logs and data sources?**

You can **disable the feature in the console or by using the API**. In the GuardDuty console, you can disable GuardDuty EKS Protection for your accounts on the GuardDuty EKS Protection console page. If you have a GuardDuty administrator account, you can also disable this feature for your member accounts.

# MIGRATING TO CLOUD AND AZURE OVERVIEW

## HOMEWORK-8

**1. Do you think it's possible to automate all cloud migrations? If yes, what would be your approach?**

While **it may be possible to automate all cloud migrations, I think it would be difficult to do so**. There are so many different potential scenarios that could come up during a cloud migration that it would be hard to account for all of them in an automated process.

**2. When moving from on-premises infrastructure to the cloud, how do you select which applications should be migrated first?**

The biggest factor to consider when choosing an on-premises-to-cloud migration method are how much data you need to move and how quickly you have to move it. Although an online migration might be the easiest and most popular choice, it can be problematic for organisation with large amounts of data and a strict migration time frame.

**3. Does AWS Application Migration Service support agentless replication? Explain Briefly.**

AWS application migration service (AWS MGN) supports agentless replication from VMware vCenter versions 6.7 and 7.0 to the AWS Cloud. AWS Application Migration Service is the primary service for lift-and-shift migrations to AWS.

The agentless replication feature is intended for users who want to rehost their applications to AWS but cannot install the AWS Replication Agent on individual servers due to company policies or technical restrictions. You can perform agentless snapshot replication from your vCenter source environment to AWS by installing the AWS MGN vCenter Client in your vCenter environment.

AWS Application Migration Service minimizes time-intensive, error-prone manual processes by automatically converting your source servers from physical, virtual, and cloud infrastructure to run natively on AWS. You can use the same automated process to migrate a wide range of applications to AWS without making changes to applications, their architecture, or the migrated servers. By using AWS Application Migration Service, you can more quickly realize the benefits of the AWS Cloud — and leverage additional AWS services to further modernize your applications.
When possible, use AWS Application Migration Service's agent-based replication option as it enables continuous replication and shortens cutover windows.

**4. How does AWS Migration Hub help us to track the progress of our application migrations.**

AWS Migration Hub **provides a central location to collect server and application inventory data for the assessment, planning, and tracking of migrations to AWS**. Migration Hub can also help accelerate application modernization following migration.

**5. How do you connect on-premises Applications to Cloud Services?**

A **hybrid cloud** is a type of cloud computing that combines on-premises infrastructure or a private cloud with a public cloud. Hybrid clouds allow data and apps to move between the two environments.

**6. What are Roles in Azure, and why do we use them?**

**ROLE DEFINITION:**

It's typically just called a role. A role definition **lists the actions that can be performed, such as read, write, and delete**. Roles can be high-level, like owner, or specific, like virtual machine reader. Azure includes several built-in roles that you can use.

**USE OF ROLE:**

Azure role-based access control (Azure RBAC) **helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to**. Azure RBAC is an authorization system built on Azure Resource Manager that provides fine-grained access management to Azure resources.

# MICROSOFT AZURE AND GCP OVERVIEW

# HOMEWORK—09

**1. What is Microsoft Azure, and why is it used?**

The Azure cloud platform is more than 200 products and cloud services designed to help you bring new solutions to life—to solve today's challenges and create the future. Build, run, and manage applications across multiple clouds, on-premises, and at the edge, with the tools and frameworks of your choice.

**2. Which service in Azure is used to manage resources in Azure?**

Azure Resource Manager is the deployment and management service for Azure. It provides a management layer that enables you to create, update, and delete resources in your Azure account.

**3. What are Roles, and why do we use them?**

A role is a comprehensive pattern of behaviour that is socially recognized, providing a means of identifying and placing an individual in a society. It also serves as a strategy for coping with recurrent situations and dealing with the roles of others (e.g., parent–child roles).

**4. Is it possible to create a Virtual Machine using Azure Resource Manager in a Virtual Network that was created using classic deployment?**

You can't use Resource Manager to deploy a virtual machine into a virtual network that was created using classic deployment.

**5. What are virtual machine scale sets in Azure?**

Azure Virtual Machine Scale Sets let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide the following key benefits: Easy to create and manage multiple VMs.

**6. What is an Availability Set?**

An availability set is a logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability. We recommended that two or more VMs are created within an availability set to provide for a highly available application and to meet the 99.95% Azure SLA.

### 7. What is a VNet?

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks.

### 8. What are the main advantages of using the Google Cloud Platform?

**Advantages of GCP:**

- Better Pricing Than Competitors.
- Private Global Fiber Network.
- Live Migration of Virtual Machines.
- Improved Performance.

# CLOUD ESSENTIALS

## PROJECT

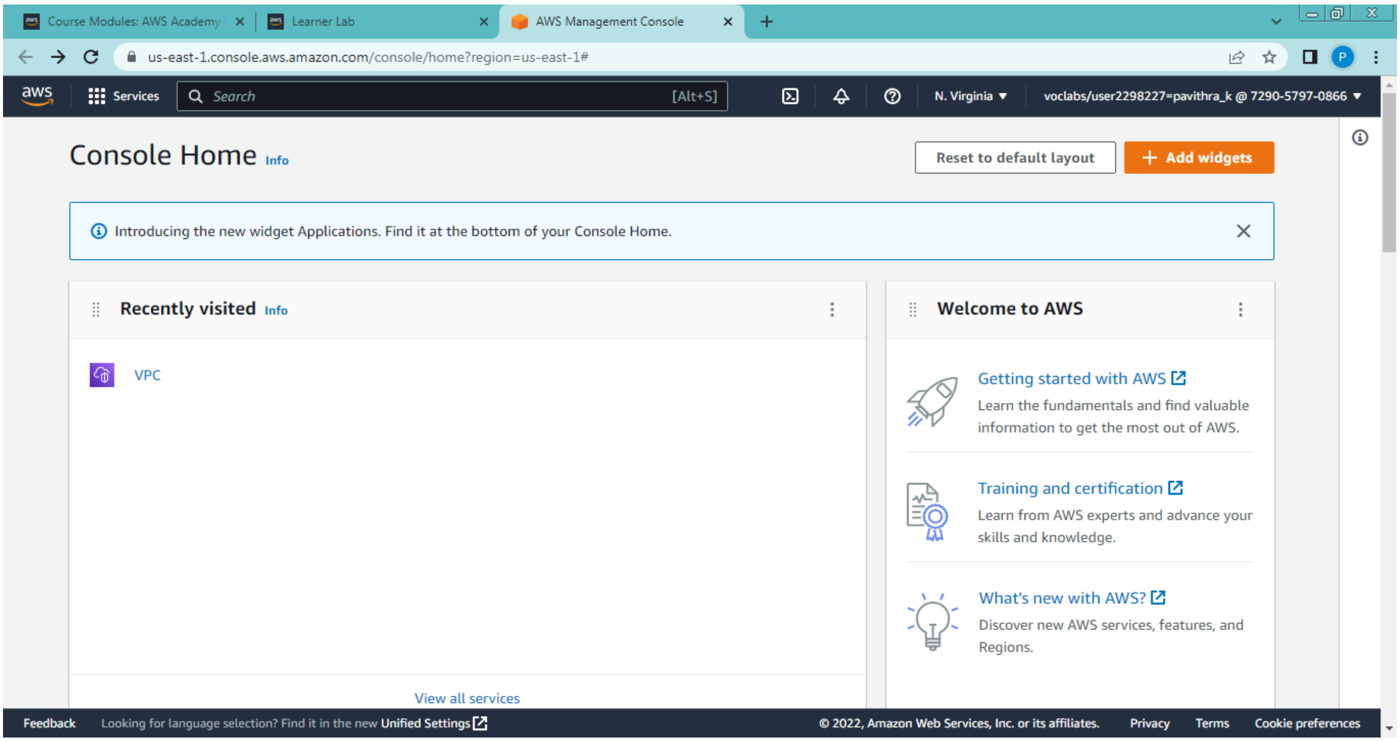
## PROBLEM STATEMENT:

John is a newbie to the cloud computing domain; he is exploring AWS and is comfortable with creating most of the AWS services. However, he struggles in creating a Virtual Private Cloud (VPC) using the console in the AWS platform. He would need you to assist him in creating a Virtual Private Cloud. While creating a VPC make sure that you:

- **Create an Amazon VPC using the VPC wizard, and it should be displayed on the dashboard**
- **Associate an Elastic IP address with it**
- **Explore various resources of VPC such as Internet Gateway, NAT Gateway, Subnets, Security Groups**
- **Launch a NAT Gateway so that internet access is provided to private resources**
- **Introduce a Public subnet for resources facing the internet such as a web server and a Private subnet for resources at the back end such as database server**
- **Define security groups with appropriate inbound rules**
- **Ensure proper routes and corresponding Route tables entries specifying the traffic moving out of the subnet**
- **Make use of Network ACLs for controlling inbound and outbound traffic in the VPC**

- **<u>Create an Amazon VPC using the VPC wizard, and it should be displayed on the dashboard</u>**

## STEP-1:



Introducing the new widget Applications. Find it at the bottom of your Console Home.

Learn the fundamentals and find valuable information to get the most out of AWS.

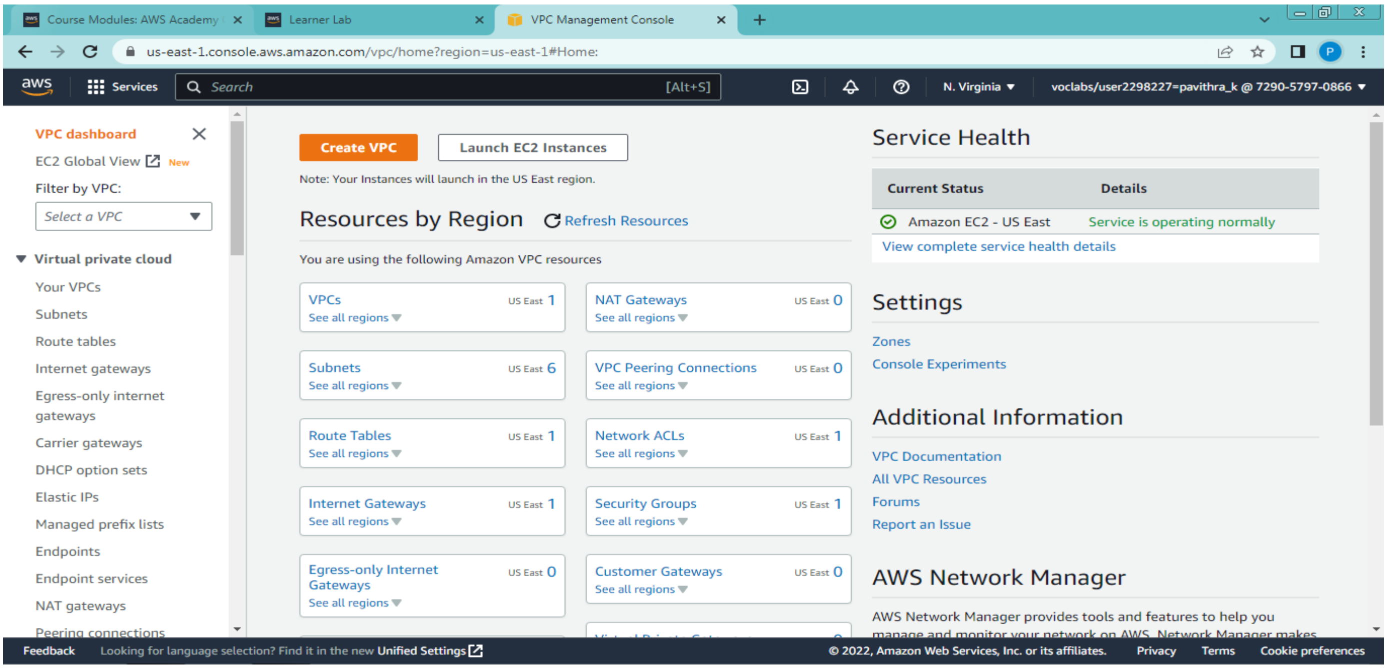
Learn from AWS experts and advance your skills and knowledge.

Discover new AWS services, features, and Regions.

Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates.

## STEP - 2:



Note: Your Instances will launch in the US East region.

You are using the following Amazon VPC resources

| Resource | Region | Count | Resource | Region | Count |
|---|---|---|---|---|---|
| VPCs<br>See all regions | US East | 1 | NAT Gateways<br>See all regions | US East | 0 |
| Subnets<br>See all regions | US East | 6 | VPC Peering Connections<br>See all regions | US East | 0 |
| Route Tables<br>See all regions | US East | 1 | Network ACLs<br>See all regions | US East | 1 |
| Internet Gateways<br>See all regions | US East | 1 | Security Groups<br>See all regions | US East | 1 |
| Egress-only Internet Gateways<br>See all regions | US East | 0 | Customer Gateways<br>See all regions | US East | 0 |

| Current Status | Details |
|---|---|
| Amazon EC2 - US East | Service is operating normally |

View complete service health details

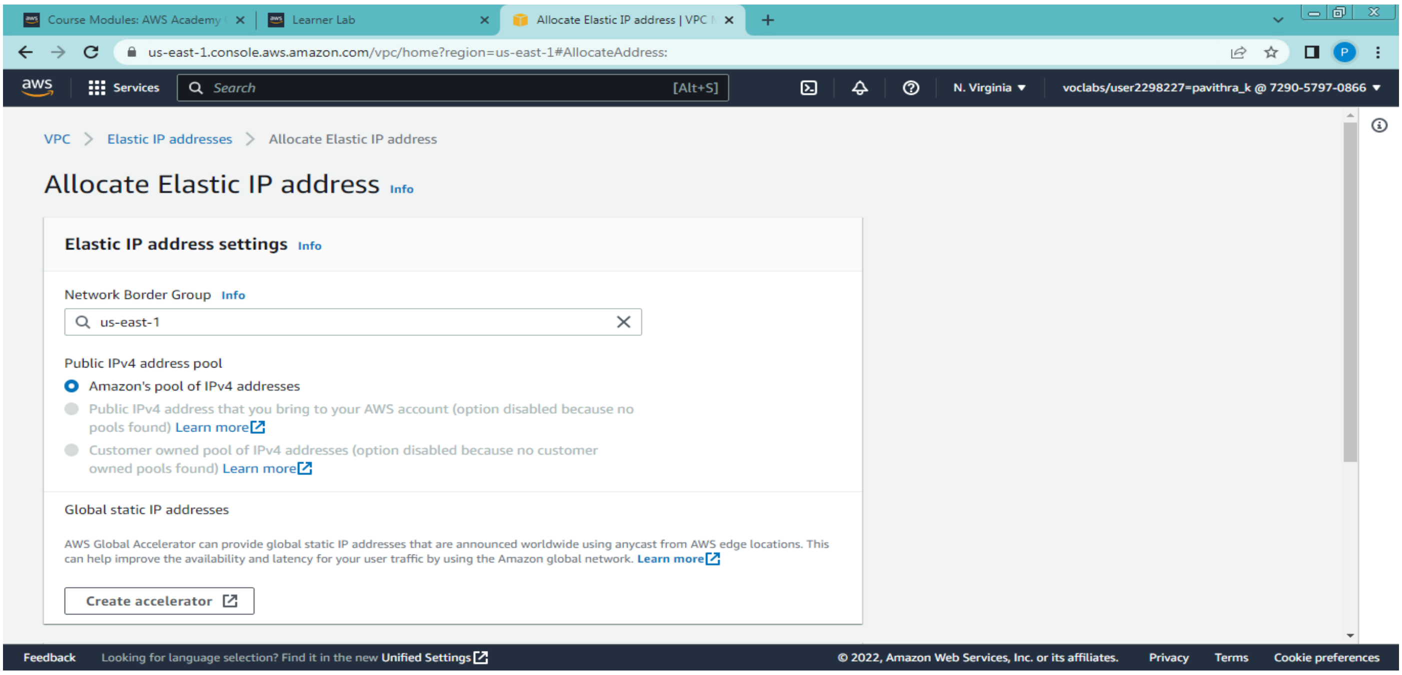
AWS Network Manager provides tools and features to help you manage and monitor your network on AWS. Network Manager makes

Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates.

- **Associate an Elastic IP address with it**

# STEP - 3:



VPC > Elastic IP addresses > Allocate Elastic IP address

Allocate Elastic IP address Info

Elastic IP address settings Info

Network Border Group Info: us-east-1

Public IPv4 address pool

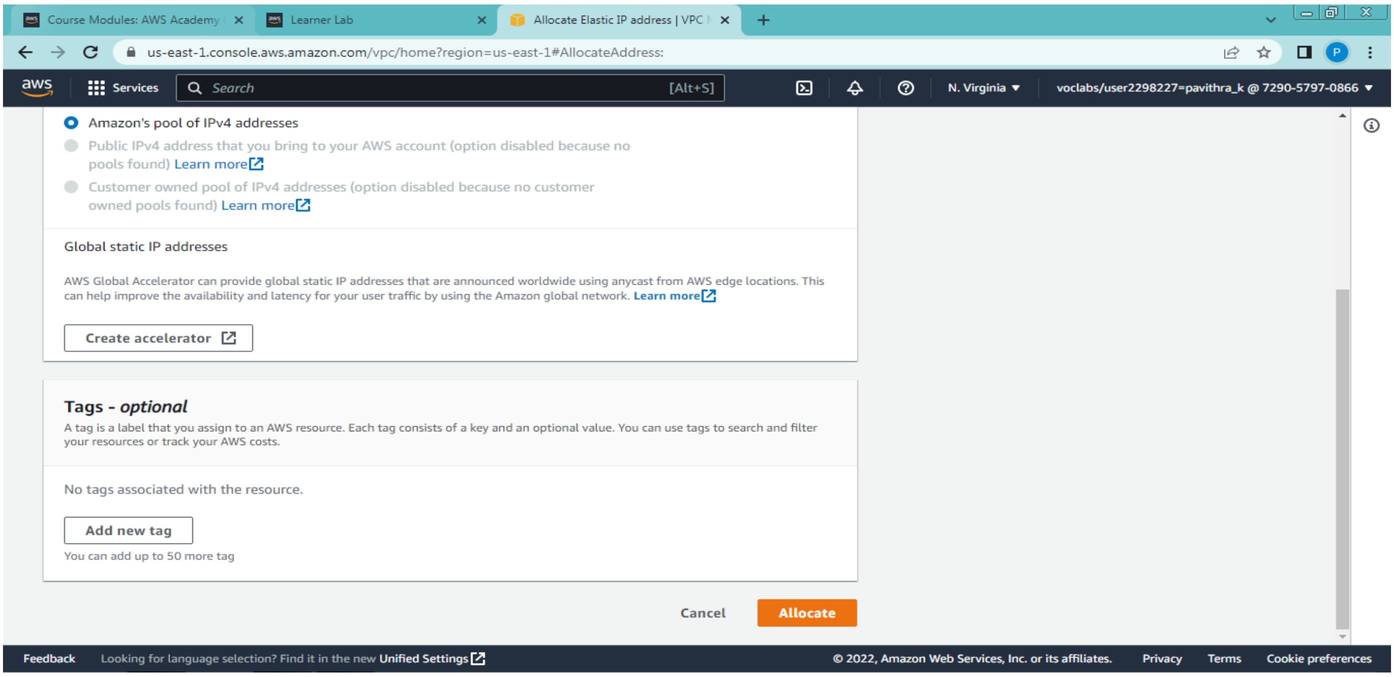
● Amazon's pool of IPv4 addresses

○ Public IPv4 address that you bring to your AWS account (option disabled because no pools found) Learn more

○ Customer owned pool of IPv4 addresses (option disabled because no customer owned pools found) Learn more

Global static IP addresses

AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. Learn more



● Amazon's pool of IPv4 addresses

○ Public IPv4 address that you bring to your AWS account (option disabled because no pools found) Learn more

○ Customer owned pool of IPv4 addresses (option disabled because no customer owned pools found) Learn more

Global static IP addresses

AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. Learn more

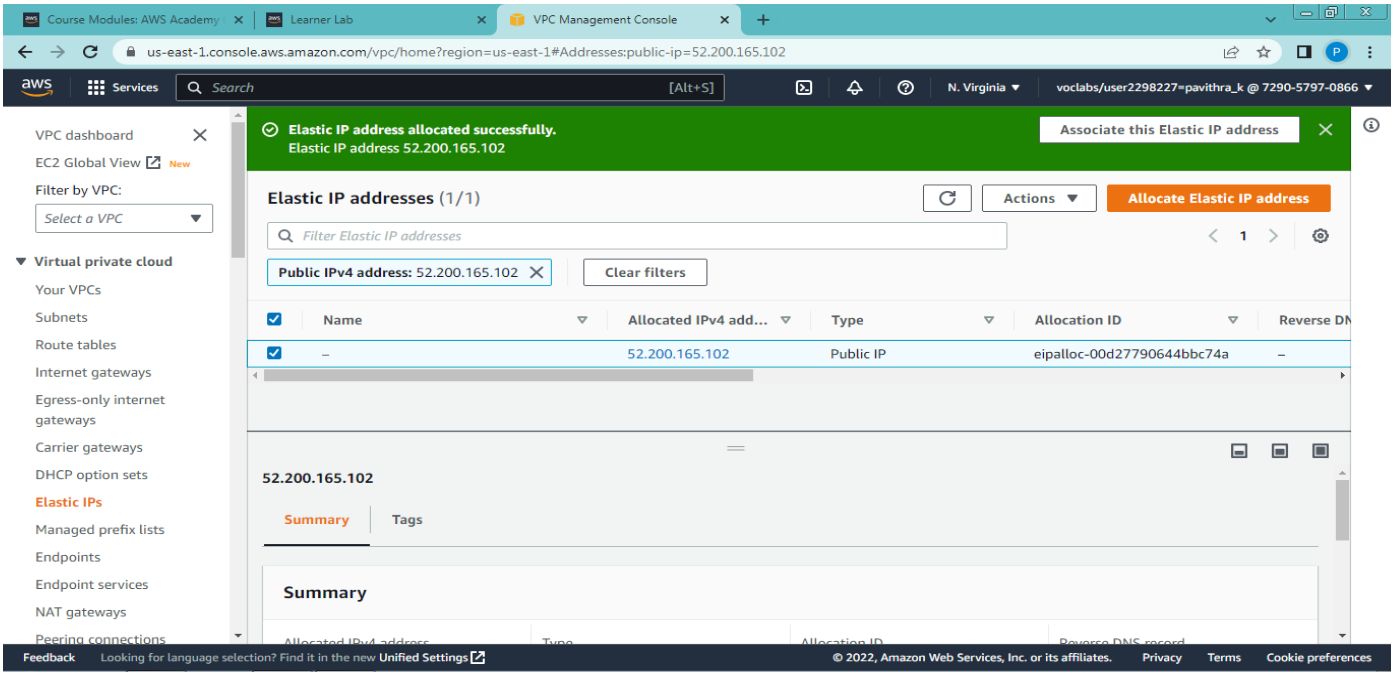
Tags - *optional*

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

No tags associated with the resource.

You can add up to 50 more tag

# STEP 4:


Course Modules: AWS Academy
Learner Lab
VPC Management Console
us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#Addresses:public-ip=52.200.165.102
Services
Search
[Alt+S]
N. Virginia
vocIabs/user2298227=pavithra_k @ 7290-5797-0866
VPC dashboard
EC2 Global View New
Filter by VPC:
Select a VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
Associate this Elastic IP address
Actions
Allocate Elastic IP address
Filter Elastic IP addresses
Clear filters
Summary
Tags
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Elastic IP address allocated successfully.
Elastic IP address 52.200.165.102

Elastic IP addresses (1/1)

Public IPv4 address: 52.200.165.102

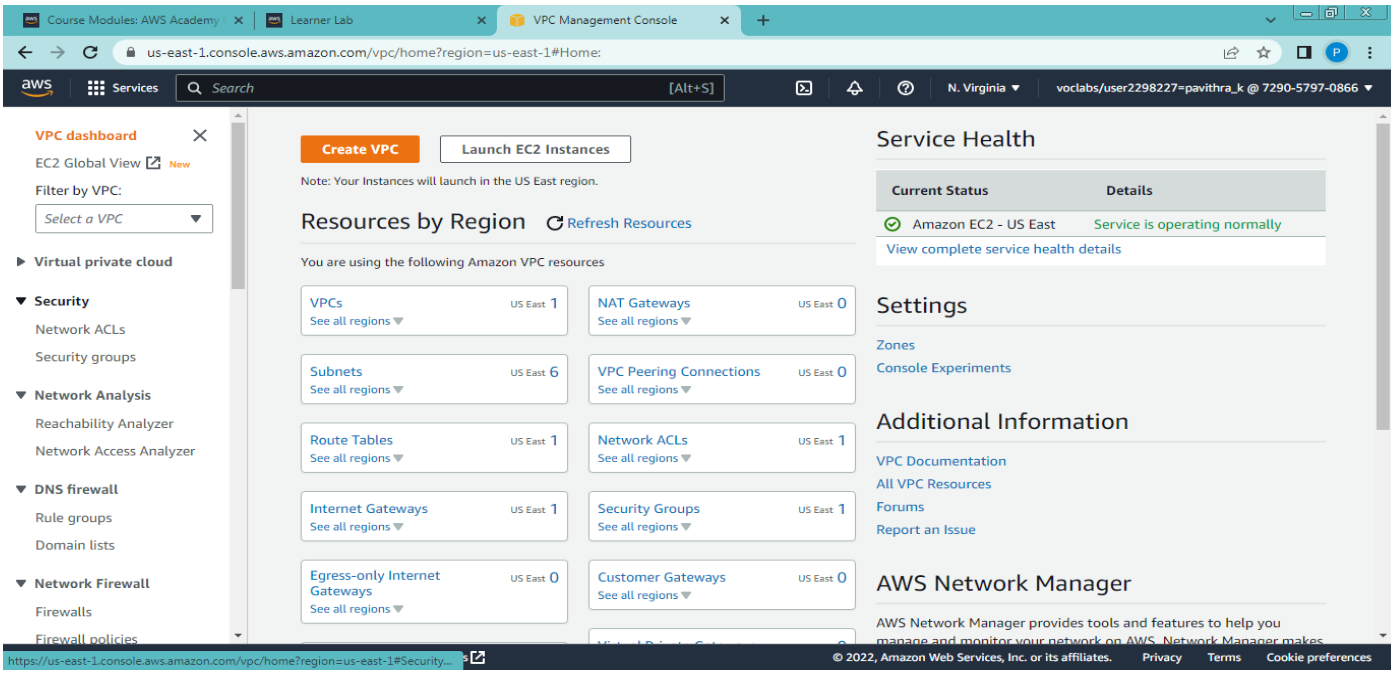
| ☑ | Name | Allocated IPv4 add... | Type | Allocation ID | Reverse DN |
|---|---|---|---|---|---|
| ☑ | – | 52.200.165.102 | Public IP | eipalloc-00d27790644bbc74a | – |

52.200.165.102

Summary

Allocated IPv4 address | Type | Allocation ID | Reverse DNS record

# STEP – 5:



Note: Your Instances will launch in the US East region.

Resources by Region

You are using the following Amazon VPC resources

| Resource | US East |
|---|---|
| VPCs | 1 |
| NAT Gateways | 0 |
| Subnets | 6 |
| VPC Peering Connections | 0 |
| Route Tables | 1 |
| Network ACLs | 1 |
| Internet Gateways | 1 |
| Security Groups | 1 |
| Egress-only Internet Gateways | 0 |
| Customer Gateways | 0 |

Service Health

| Current Status | Details |
|---|---|
| Amazon EC2 - US East | Service is operating normally |

View complete service health details

Settings

Zones
Console Experiments

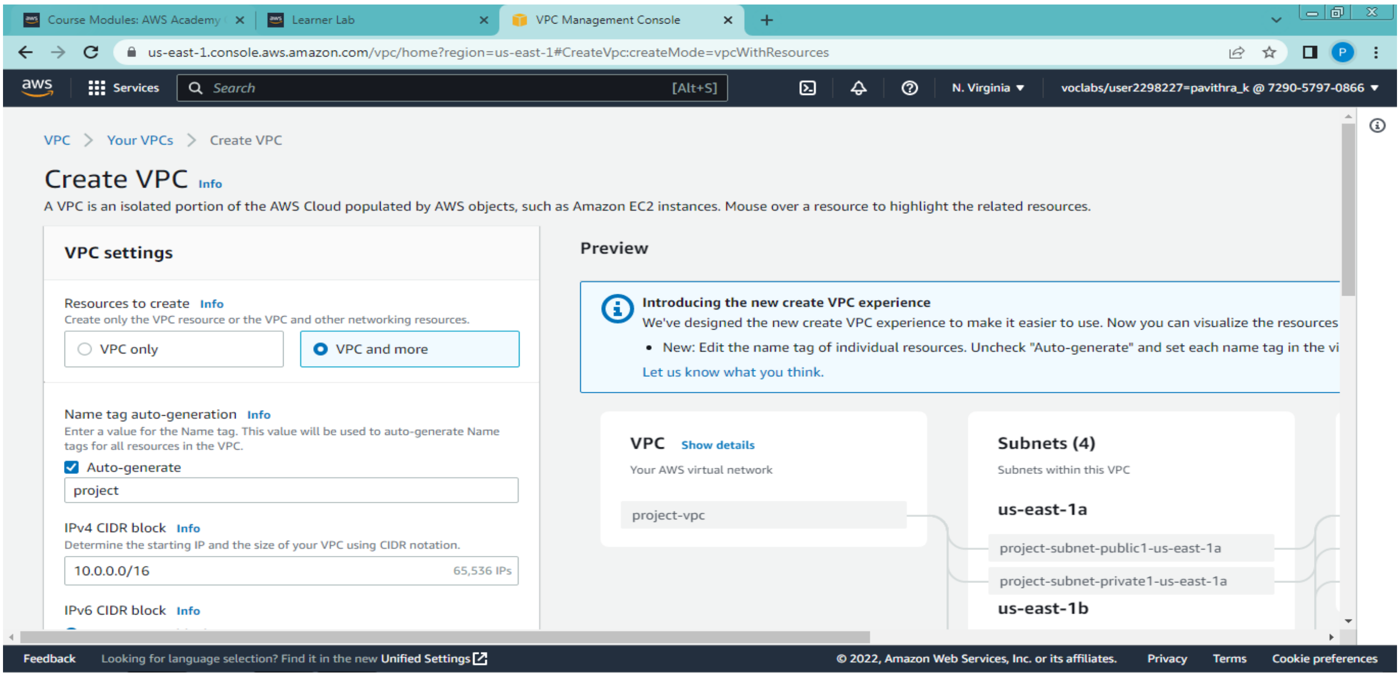
Additional Information

VPC Documentation
All VPC Resources
Forums
Report an Issue

AWS Network Manager

AWS Network Manager provides tools and features to help you manage and monitor your network on AWS. Network Manager makes

# STEP – 6:



us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#CreateVpc:createMode=vpcWithResources

VPC > Your VPCs > Create VPC

Create VPC Info

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

VPC settings

Resources to create Info

Create only the VPC resource or the VPC and other networking resources.

○ VPC only ● VPC and more

Name tag auto-generation Info

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☑ Auto-generate

project

IPv4 CIDR block Info

Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs

IPv6 CIDR block Info

Preview

Introducing the new create VPC experience

We've designed the new create VPC experience to make it easier to use. Now you can visualize the resources

- New: Edit the name tag of individual resources. Uncheck "Auto-generate" and set each name tag in the vi

Let us know what you think.

VPC Show details

Your AWS virtual network

project-vpc

Subnets (4)

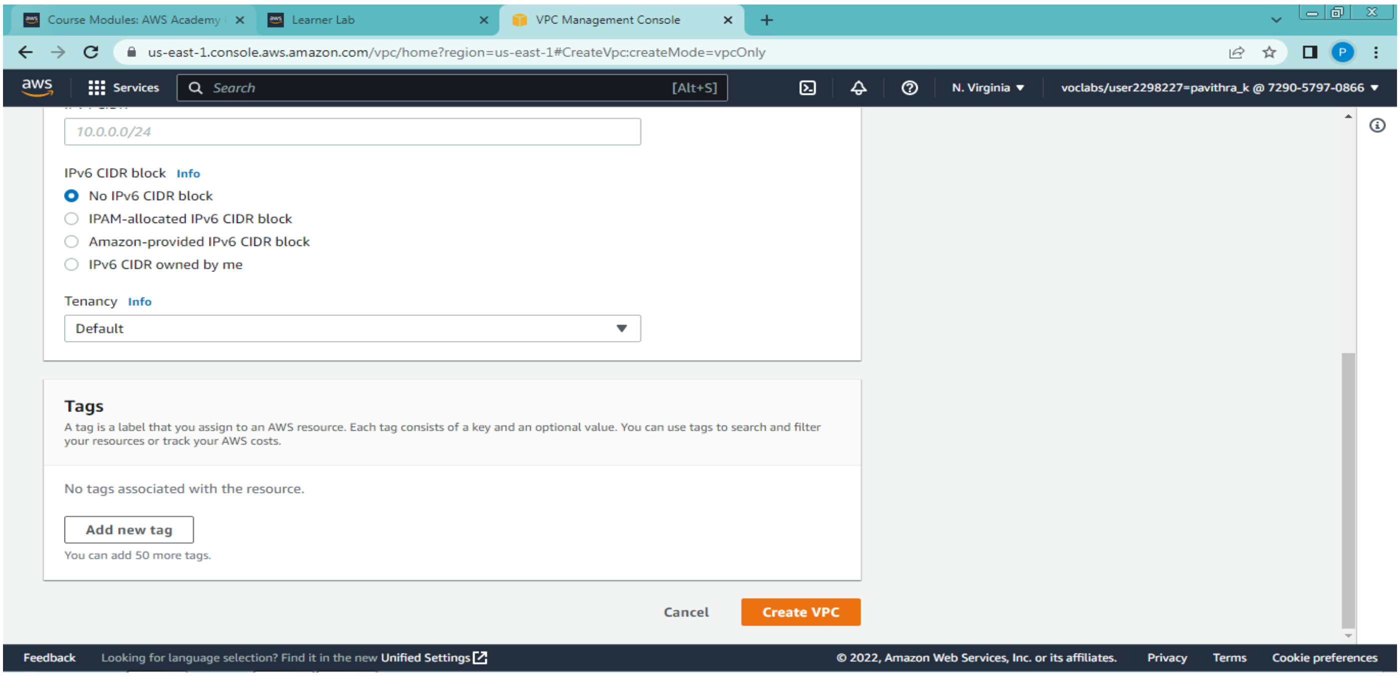
Subnets within this VPC

us-east-1a

project-subnet-public1-us-east-1a

project-subnet-private1-us-east-1a

us-east-1b



us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#CreateVpc:createMode=vpcOnly

10.0.0.0/24

IPv6 CIDR block Info

● No IPv6 CIDR block

○ IPAM-allocated IPv6 CIDR block

○ Amazon-provided IPv6 CIDR block

○ IPv6 CIDR owned by me

Tenancy Info

Default

Tags

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

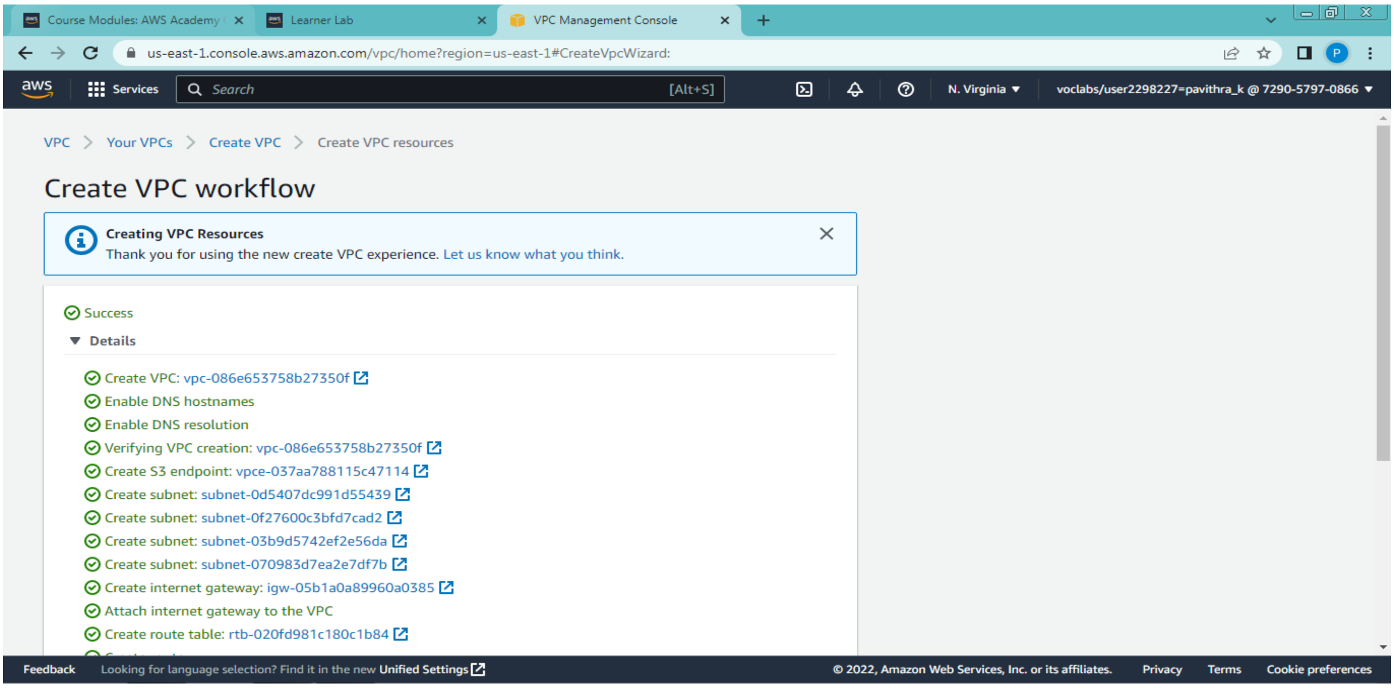
No tags associated with the resource.

Add new tag

You can add 50 more tags.

Cancel Create VPC

## STEP – 7:


Course Modules: AWS Academy
Learner Lab
VPC Management Console
us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#CreateVpcWizard:
Services
Search
[Alt+S]
N. Virginia
voclabs/user2298227=pavithra_k @ 7290-5797-0866
VPC > Your VPCs > Create VPC > Create VPC resources
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Create VPC workflow

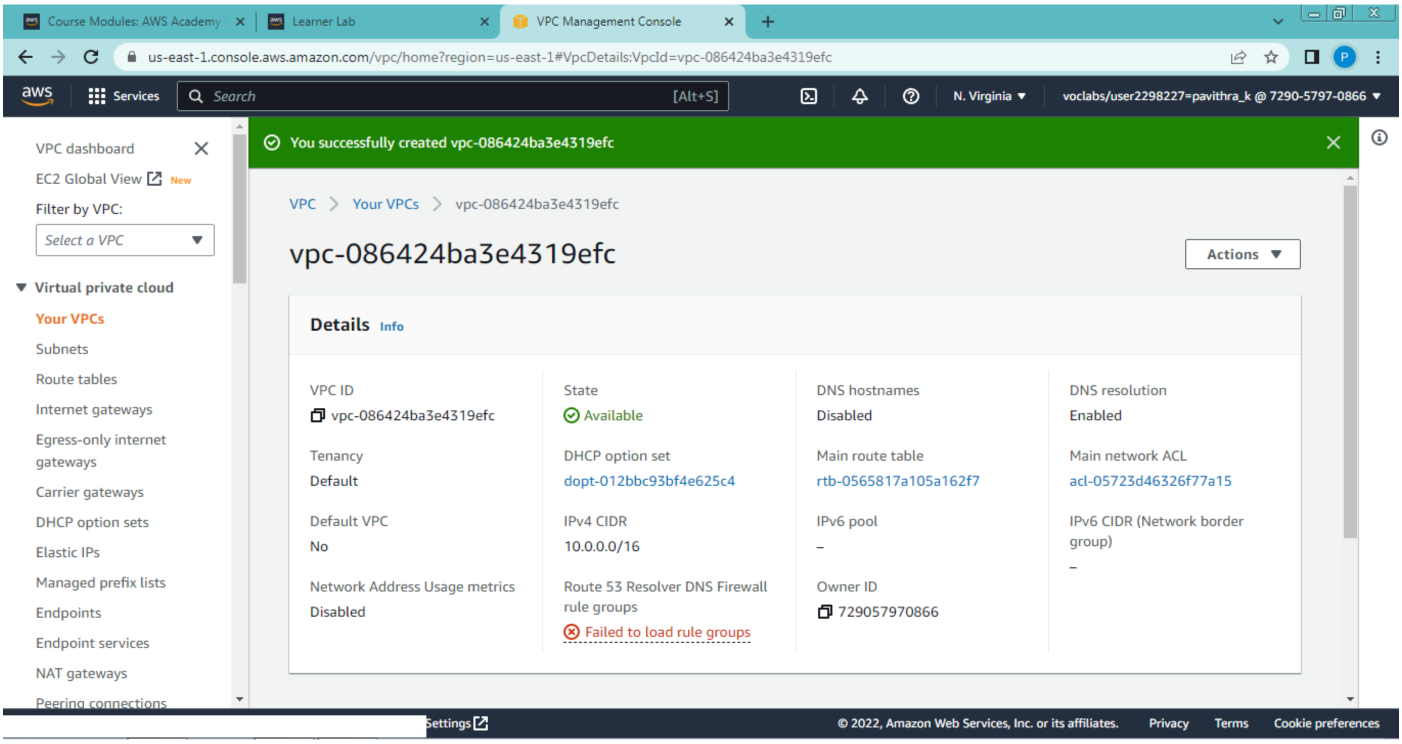
**Creating VPC Resources**
Thank you for using the new create VPC experience. Let us know what you think.

✓ Success

▼ Details

- ✓ Create VPC: vpc-086e653758b27350f
- ✓ Enable DNS hostnames
- ✓ Enable DNS resolution
- ✓ Verifying VPC creation: vpc-086e653758b27350f
- ✓ Create S3 endpoint: vpce-037aa788115c47114
- ✓ Create subnet: subnet-0d5407dc991d55439
- ✓ Create subnet: subnet-0f27600c3bfd7cad2
- ✓ Create subnet: subnet-03b9d5742ef2e56da
- ✓ Create subnet: subnet-070983d7ea2e7df7b
- ✓ Create internet gateway: igw-05b1a0a89960a0385
- ✓ Attach internet gateway to the VPC
- ✓ Create route table: rtb-020fd981c180c1b84

## STEP – 8:



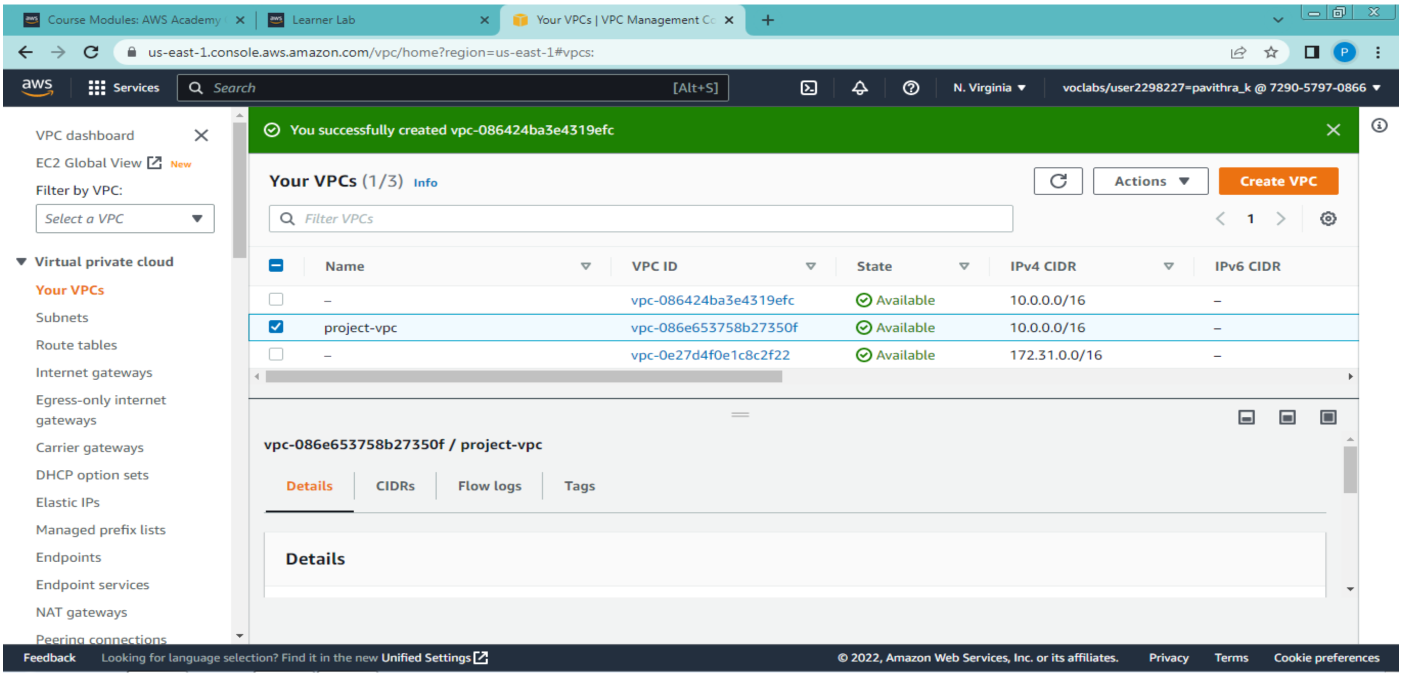
✓ You successfully created vpc-086424ba3e4319efc

vpc-086424ba3e4319efc

**Details** Info

| VPC ID | State | DNS hostnames | DNS resolution |
|---|---|---|---|
| vpc-086424ba3e4319efc | ✓ Available | Disabled | Enabled |
| **Tenancy** | **DHCP option set** | **Main route table** | **Main network ACL** |
| Default | dopt-012bbc93bf4e625c4 | rtb-0565817a105a162f7 | acl-05723d46326f77a15 |
| **Default VPC** | **IPv4 CIDR** | **IPv6 pool** | **IPv6 CIDR (Network border group)** |
| No | 10.0.0.0/16 | – | – |
| **Network Address Usage metrics** | **Route 53 Resolver DNS Firewall rule groups** | **Owner ID** | |
| Disabled | ⊗ Failed to load rule groups | 729057970866 | |

# STEP – 9:



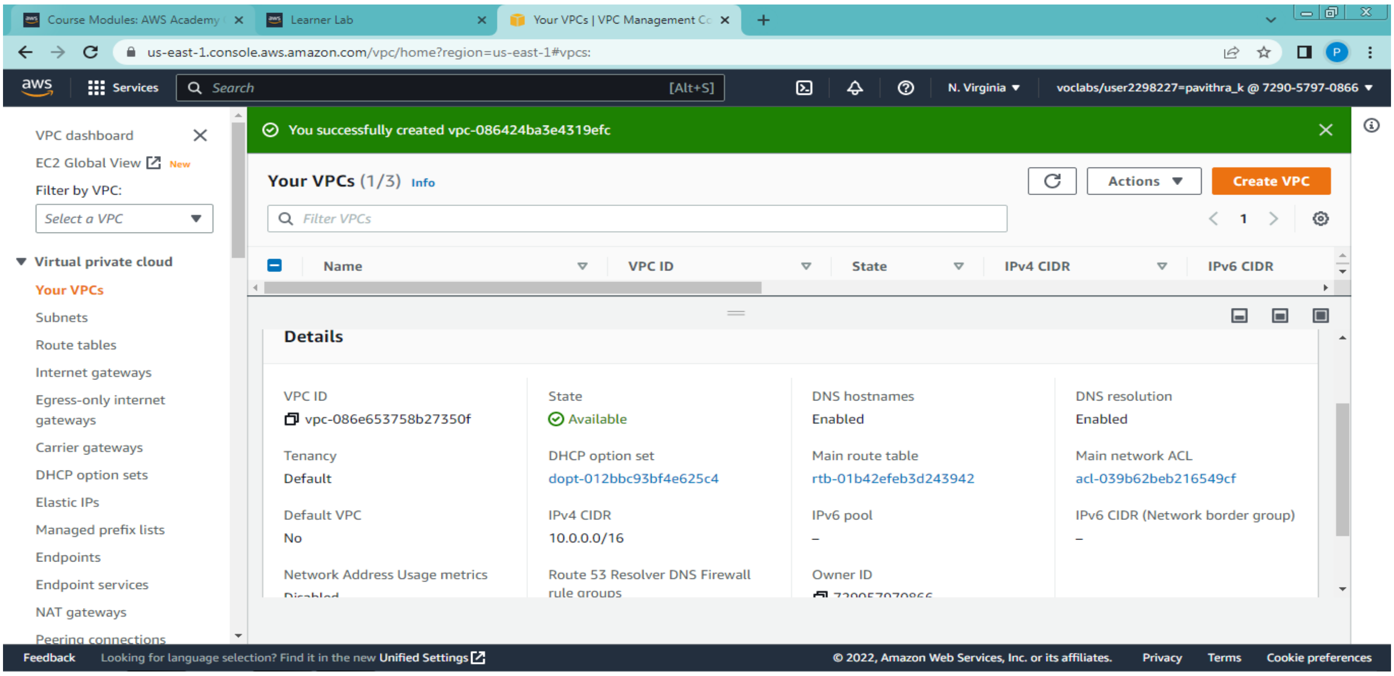
You successfully created vpc-086424ba3e4319efc

Your VPCs (1/3) Info

| | Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR |
|---|---|---|---|---|---|
| □ | – | vpc-086424ba3e4319efc | Available | 10.0.0.0/16 | – |
| ☑ | project-vpc | vpc-086e653758b27350f | Available | 10.0.0.0/16 | – |
| □ | – | vpc-0e27d4f0e1c8c2f22 | Available | 172.31.0.0/16 | – |

vpc-086e653758b27350f / project-vpc

Details



You successfully created vpc-086424ba3e4319efc

Your VPCs (1/3) Info

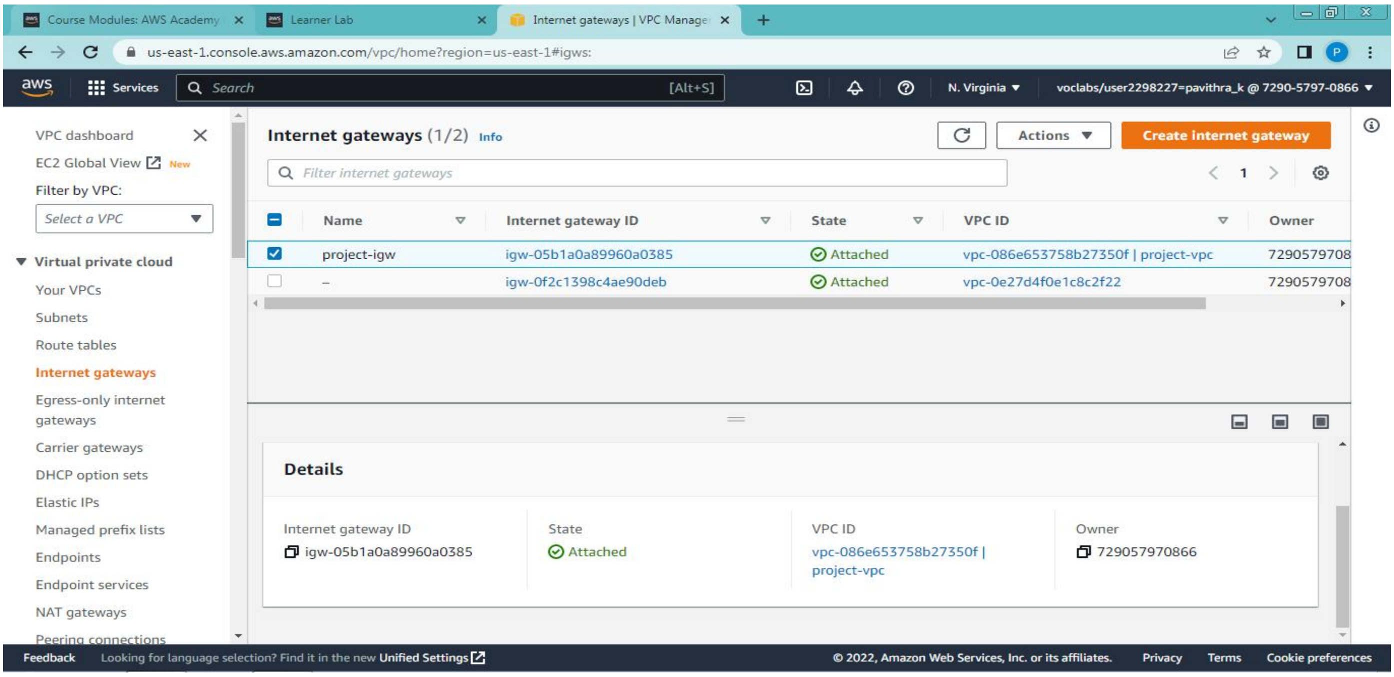
| | Name | VPC ID | State | IPv4 CIDR | IPv6 CIDR |
|---|---|---|---|---|---|

Details

| | | | |
|---|---|---|---|
| VPC ID<br>vpc-086e653758b27350f | State<br>Available | DNS hostnames<br>Enabled | DNS resolution<br>Enabled |
| Tenancy<br>Default | DHCP option set<br>dopt-012bbc93bf4e625c4 | Main route table<br>rtb-01b42efeb3d243942 | Main network ACL<br>acl-039b62beb216549cf |
| Default VPC<br>No | IPv4 CIDR<br>10.0.0.0/16 | IPv6 pool<br>– | IPv6 CIDR (Network border group)<br>– |
| Network Address Usage metrics<br>Disabled | Route 53 Resolver DNS Firewall rule groups | Owner ID<br>729057970866 | |

- **Explore various resources of VPC such as Internet Gateway, NAT Gateway, Subnets, Security Groups**

## STEP – 10:



us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#igws:

Internet gateways (1/2)

| | Name | Internet gateway ID | State | VPC ID | Owner |
|---|---|---|---|---|---|
| ☑ | project-igw | igw-05b1a0a89960a0385 | ✓ Attached | vpc-086e653758b27350f \| project-vpc | 7290579708 |
| □ | – | igw-0f2c1398c4ae90deb | ✓ Attached | vpc-0e27d4f0e1c8c2f22 | 7290579708 |

Details

Internet gateway ID: igw-05b1a0a89960a0385

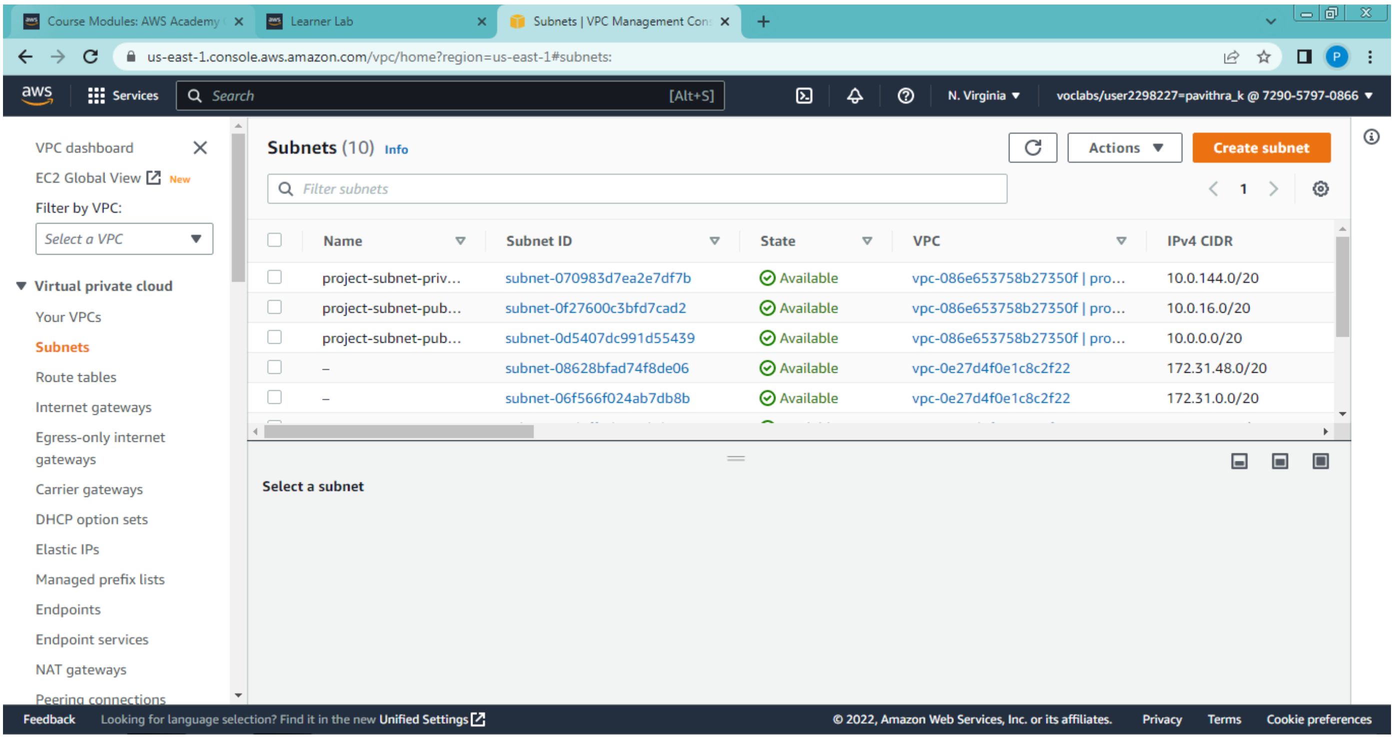
State: ✓ Attached

VPC ID: vpc-086e653758b27350f | project-vpc

Owner: 729057970866

- **Introduce a Public subnet for resources facing the internet such as a web server and a Private subnet for resources at the back end such as database server**

## STEP – 11:



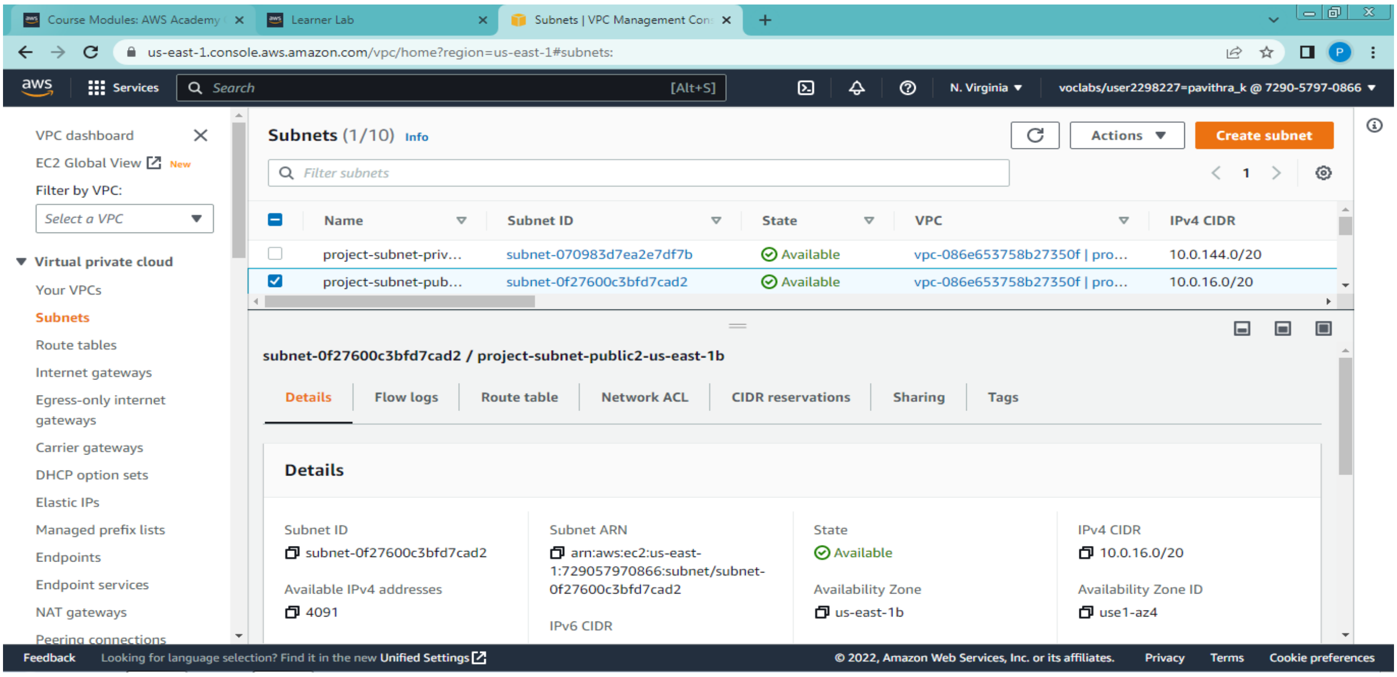
us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

Subnets (10)

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv... | subnet-070983d7ea2e7df7b | ✓ Available | vpc-086e653758b27350f \| pro... | 10.0.144.0/20 |
| □ | project-subnet-pub... | subnet-0f27600c3bfd7cad2 | ✓ Available | vpc-086e653758b27350f \| pro... | 10.0.16.0/20 |
| □ | project-subnet-pub... | subnet-0d5407dc991d55439 | ✓ Available | vpc-086e653758b27350f \| pro... | 10.0.0.0/20 |
| □ | – | subnet-08628bfad74f8de06 | ✓ Available | vpc-0e27d4f0e1c8c2f22 | 172.31.48.0/20 |
| □ | – | subnet-06f566f024ab7db8b | ✓ Available | vpc-0e27d4f0e1c8c2f22 | 172.31.0.0/20 |

Select a subnet

# STEP – 12:



us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

N. Virginia · voclabs/user2298227=pavithra_k @ 7290-5797-0866

**Subnets** (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv… | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |
| ☑ | project-subnet-pub… | subnet-0f27600c3bfd7cad2 | Available | vpc-086e653758b27350f \| pro… | 10.0.16.0/20 |

subnet-0f27600c3bfd7cad2 / project-subnet-public2-us-east-1b

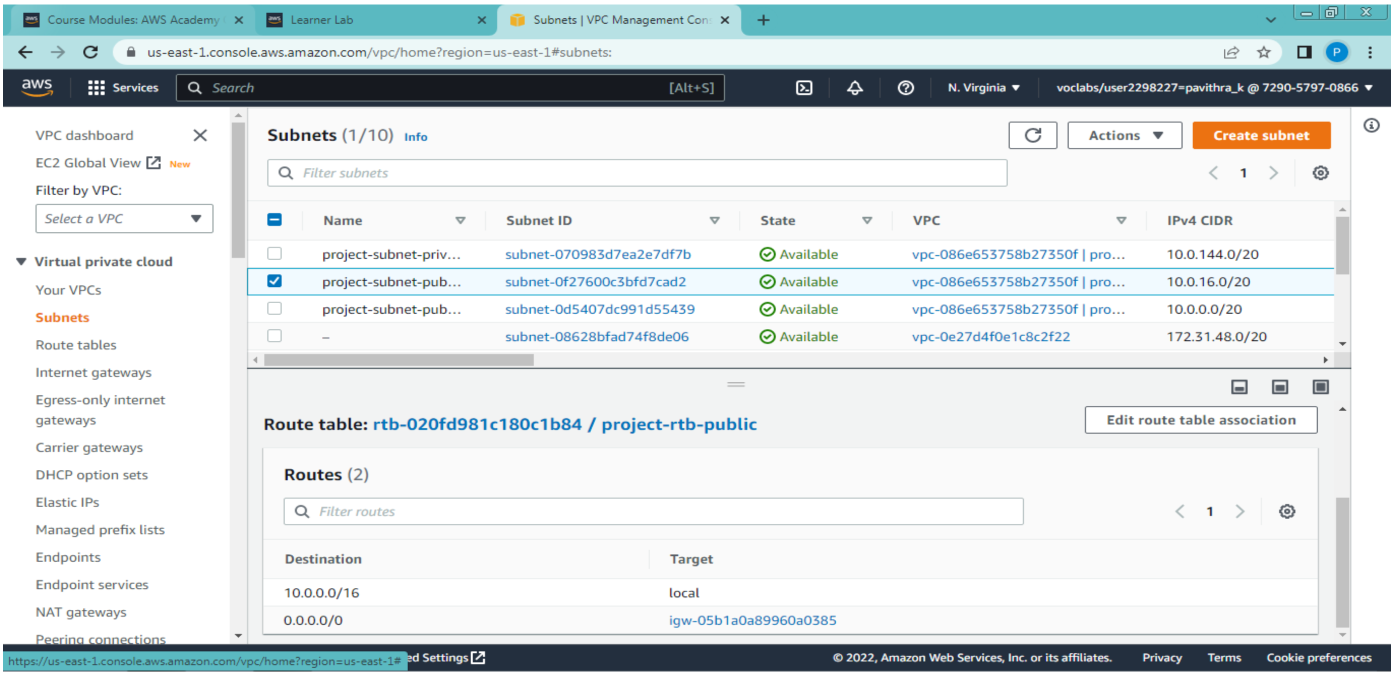
Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

**Details**

| Subnet ID | Subnet ARN | State | IPv4 CIDR |
|---|---|---|---|
| subnet-0f27600c3bfd7cad2 | arn:aws:ec2:us-east-1:729057970866:subnet/subnet-0f27600c3bfd7cad2 | Available | 10.0.16.0/20 |
| **Available IPv4 addresses** | **IPv6 CIDR** | **Availability Zone** | **Availability Zone ID** |
| 4091 | | us-east-1b | use1-az4 |

- **Ensure proper routes and corresponding Route tables entries specifying the traffic moving out of the subnet [Public subnet]**

# STEP – 13:



us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

N. Virginia · voclabs/user2298227=pavithra_k @ 7290-5797-0866

**Subnets** (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv… | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |
| ☑ | project-subnet-pub… | subnet-0f27600c3bfd7cad2 | Available | vpc-086e653758b27350f \| pro… | 10.0.16.0/20 |
| □ | project-subnet-pub… | subnet-0d5407dc991d55439 | Available | vpc-086e653758b27350f \| pro… | 10.0.0.0/20 |
| □ | – | subnet-08628bfad74f8de06 | Available | vpc-0e27d4f0e1c8c2f22 | 172.31.48.0/20 |

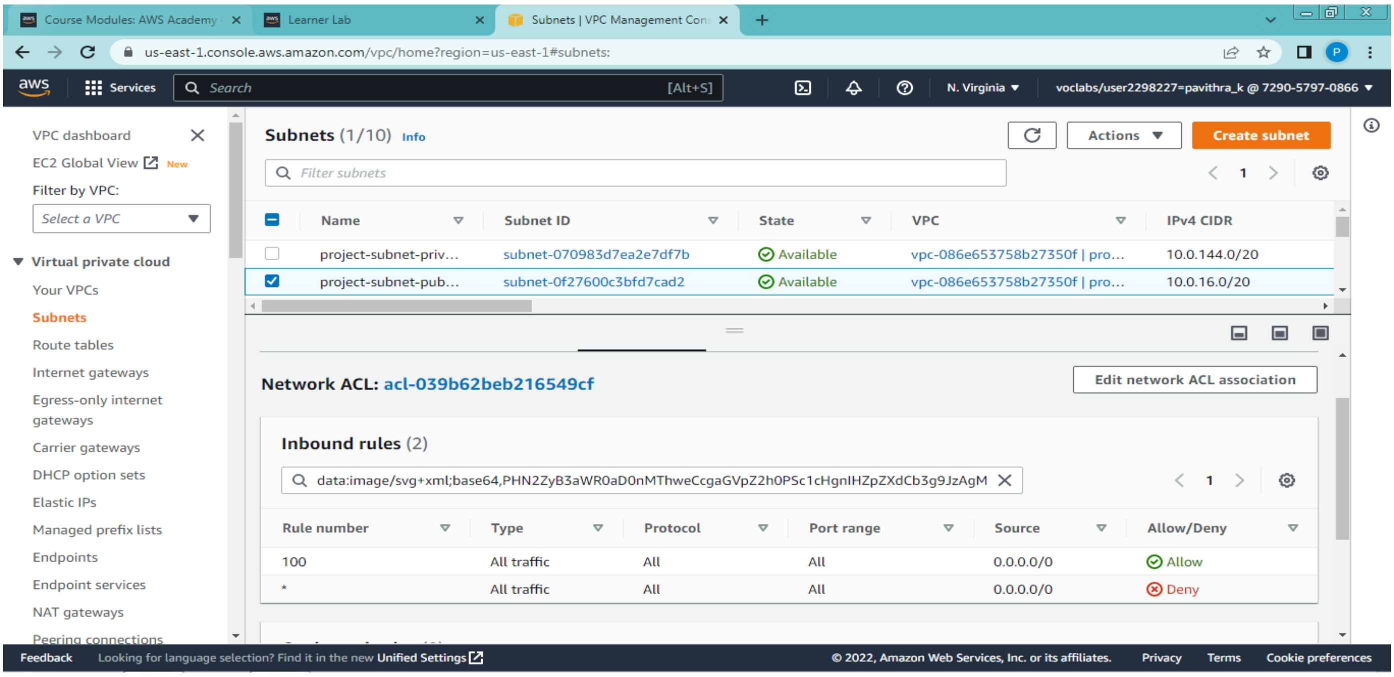
Route table: rtb-020fd981c180c1b84 / project-rtb-public

**Routes** (2)

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | igw-05b1a0a89960a0385 |

- **Make use of Network ACLs for controlling inbound and outbound traffic in the VPC [Public subnet]**

## STEP – 14:



Course Modules: AWS Academy | Learner Lab | Subnets | VPC Management Con

us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

Services | Search [Alt+S] | N. Virginia | voclabs/user2298227=pavithra_k @ 7290-5797-0866

**Subnets** (1/10) Info

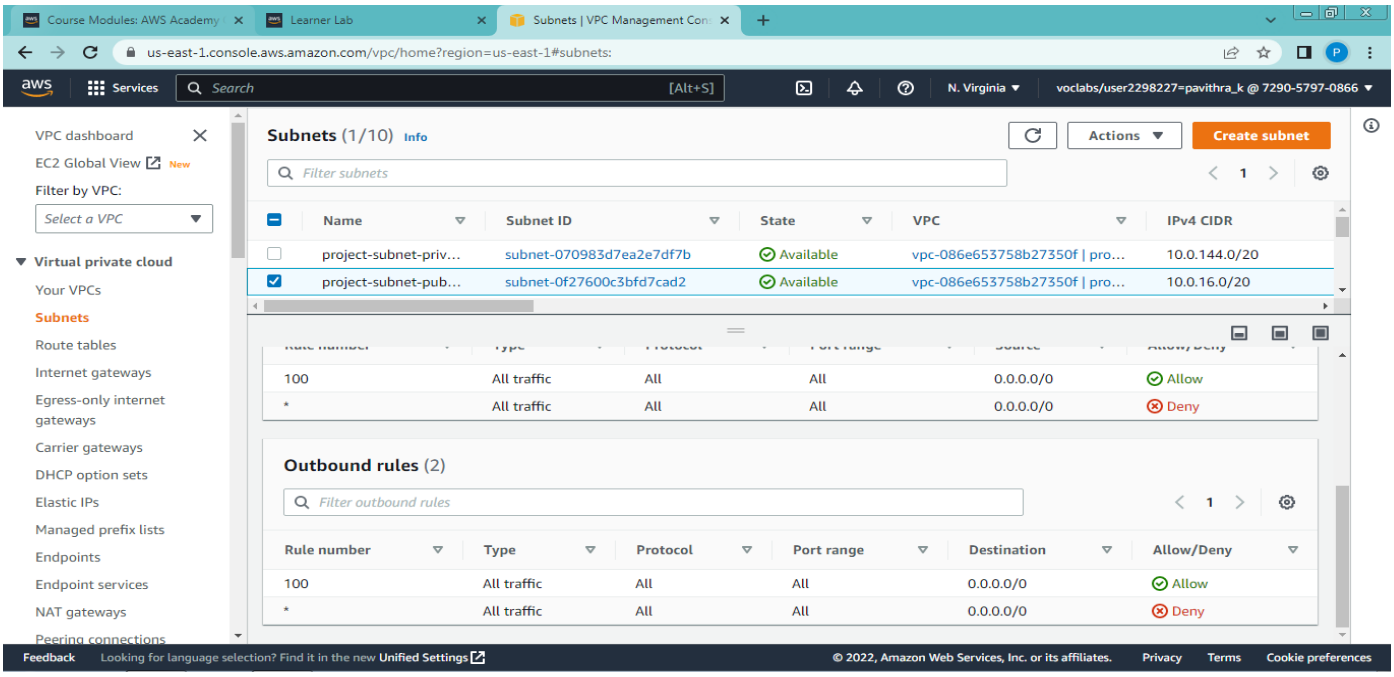
| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv... | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro... | 10.0.144.0/20 |
| ☑ | project-subnet-pub... | subnet-0f27600c3bfd7cad2 | Available | vpc-086e653758b27350f \| pro... | 10.0.16.0/20 |

**Network ACL:** acl-039b62beb216549cf

**Inbound rules** (2)

data:image/svg+xml;base64,PHN2ZyB3aWR0aD0nMThweCcgaGVpZ2h0PSc1cHgnIHZpZXdCb3g9JzAgM

| Rule number | Type | Protocol | Port range | Source | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | Allow |
| * | All traffic | All | All | 0.0.0.0/0 | Deny |



Course Modules: AWS Academy | Learner Lab | Subnets | VPC Management Con

us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

Services | Search [Alt+S] | N. Virginia | voclabs/user2298227=pavithra_k @ 7290-5797-0866

**Subnets** (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv... | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro... | 10.0.144.0/20 |
| ☑ | project-subnet-pub... | subnet-0f27600c3bfd7cad2 | Available | vpc-086e653758b27350f \| pro... | 10.0.16.0/20 |

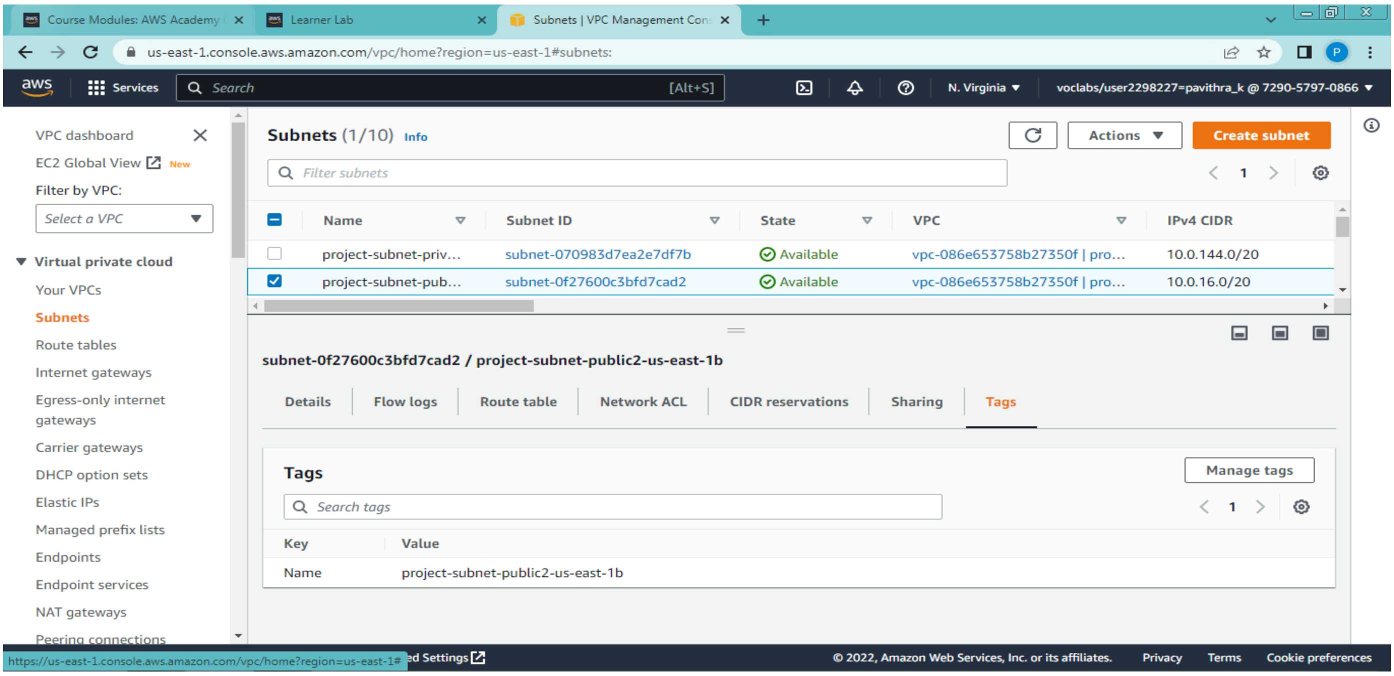
| Rule number | Type | Protocol | Port range | Source | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | Allow |
| * | All traffic | All | All | 0.0.0.0/0 | Deny |

**Outbound rules** (2)

Filter outbound rules

| Rule number | Type | Protocol | Port range | Destination | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | Allow |
| * | All traffic | All | All | 0.0.0.0/0 | Deny |

## STEP – 15:



Subnets (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| □ | project-subnet-priv… | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |
| ☑ | project-subnet-pub… | subnet-0f27600c3bfd7cad2 | Available | vpc-086e653758b27350f \| pro… | 10.0.16.0/20 |

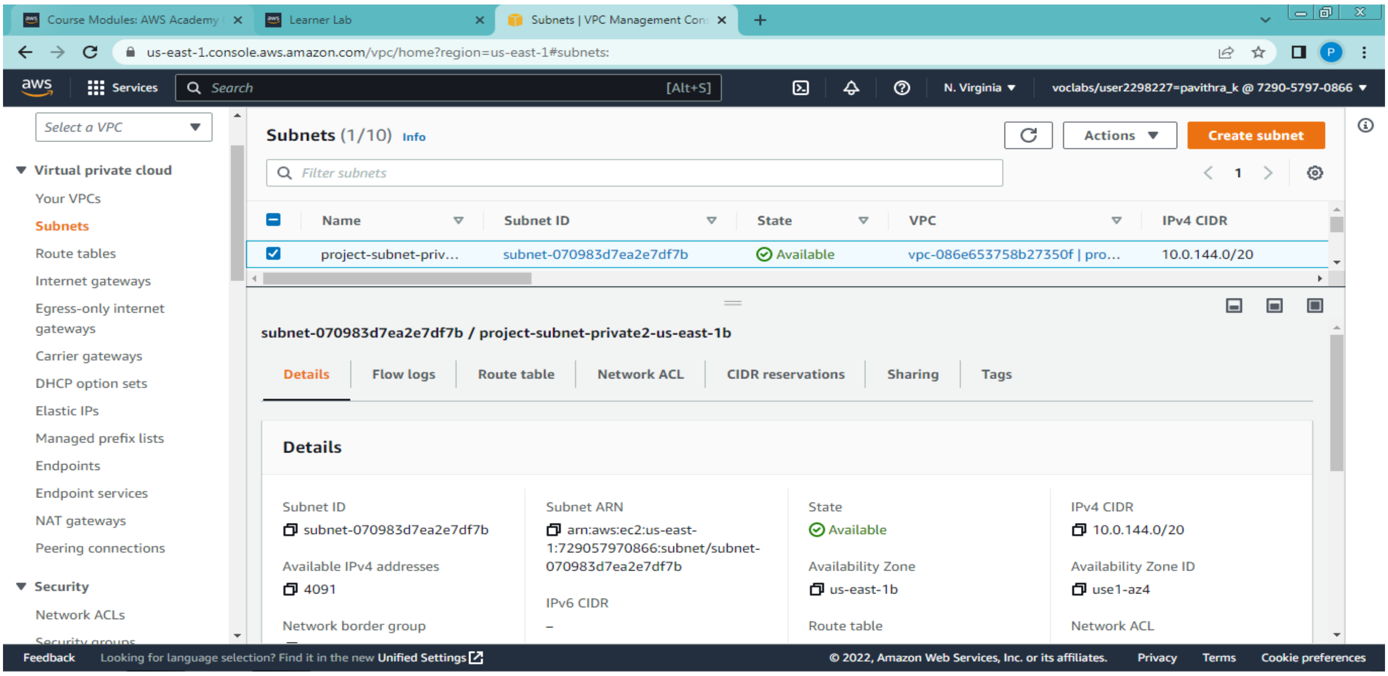
subnet-0f27600c3bfd7cad2 / project-subnet-public2-us-east-1b

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

Tags

| Key | Value |
|---|---|
| Name | project-subnet-public2-us-east-1b |

## STEP – 16:



Subnets (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| ☑ | project-subnet-priv… | subnet-070983d7ea2e7df7b | Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |

subnet-070983d7ea2e7df7b / project-subnet-private2-us-east-1b

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

Details

Subnet ID: subnet-070983d7ea2e7df7b

Subnet ARN: arn:aws:ec2:us-east-1:729057970866:subnet/subnet-070983d7ea2e7df7b

State: Available

IPv4 CIDR: 10.0.144.0/20

Available IPv4 addresses: 4091

IPv6 CIDR: –

Availability Zone: us-east-1b

Availability Zone ID: use1-az4

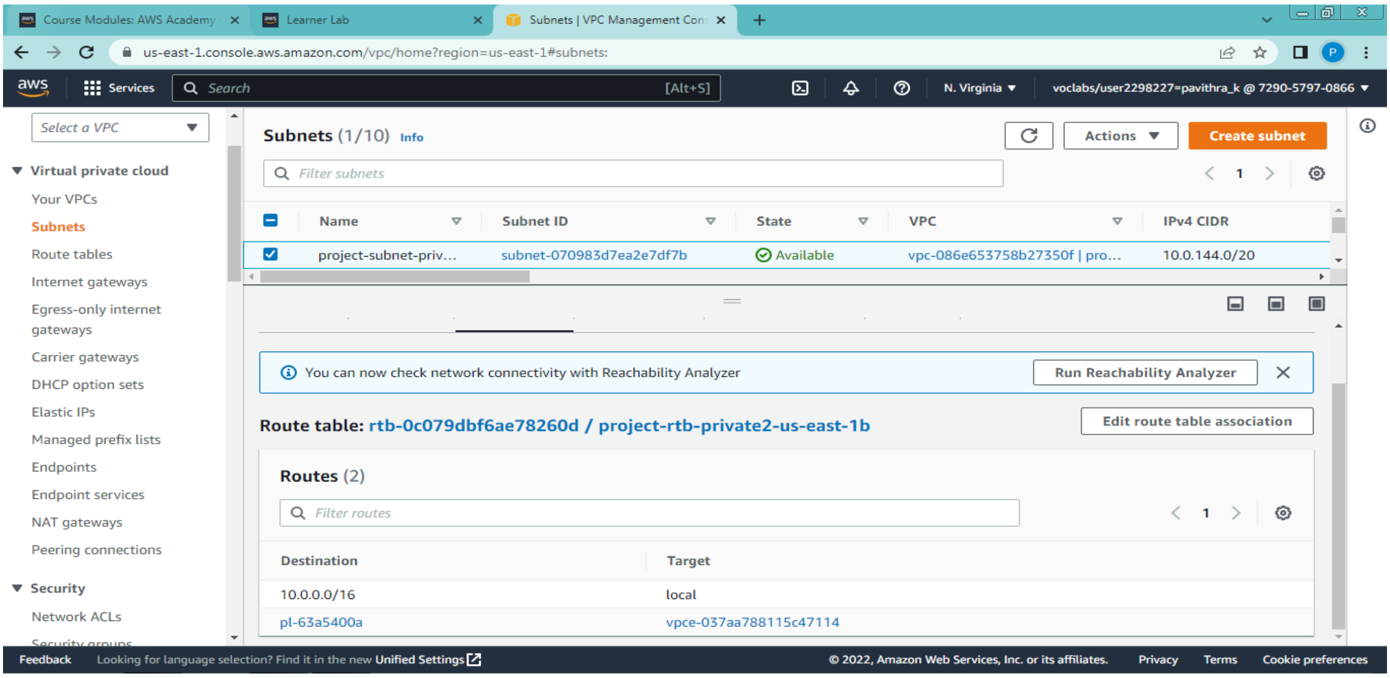
Network border group: –

Route table

Network ACL

- **Ensure proper routes and corresponding Route tables entries specifying the traffic moving out of the subnet [Private subnet]**

## STEP – 17:



Course Modules: AWS Academy | Learner Lab | Subnets | VPC Management Con

us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

aws | Services | Search [Alt+S] | N. Virginia ▼ | voclabs/user2298227=pavithra_k @ 7290-5797-0866 ▼

Select a VPC ▼

▼ Virtual private cloud
- Your VPCs
- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- Endpoints
- Endpoint services
- NAT gateways
- Peering connections

▼ Security
- Network ACLs
- Security groups

Subnets (1/10) Info | Actions ▼ | Create subnet

Filter subnets

| ☐ | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| ☑ | project-subnet-priv… | subnet-070983d7ea2e7df7b | ✓ Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |

ⓘ You can now check network connectivity with Reachability Analyzer | Run Reachability Analyzer | ✕

Route table: rtb-0c079dbf6ae78260d / project-rtb-private2-us-east-1b | Edit route table association

Routes (2)

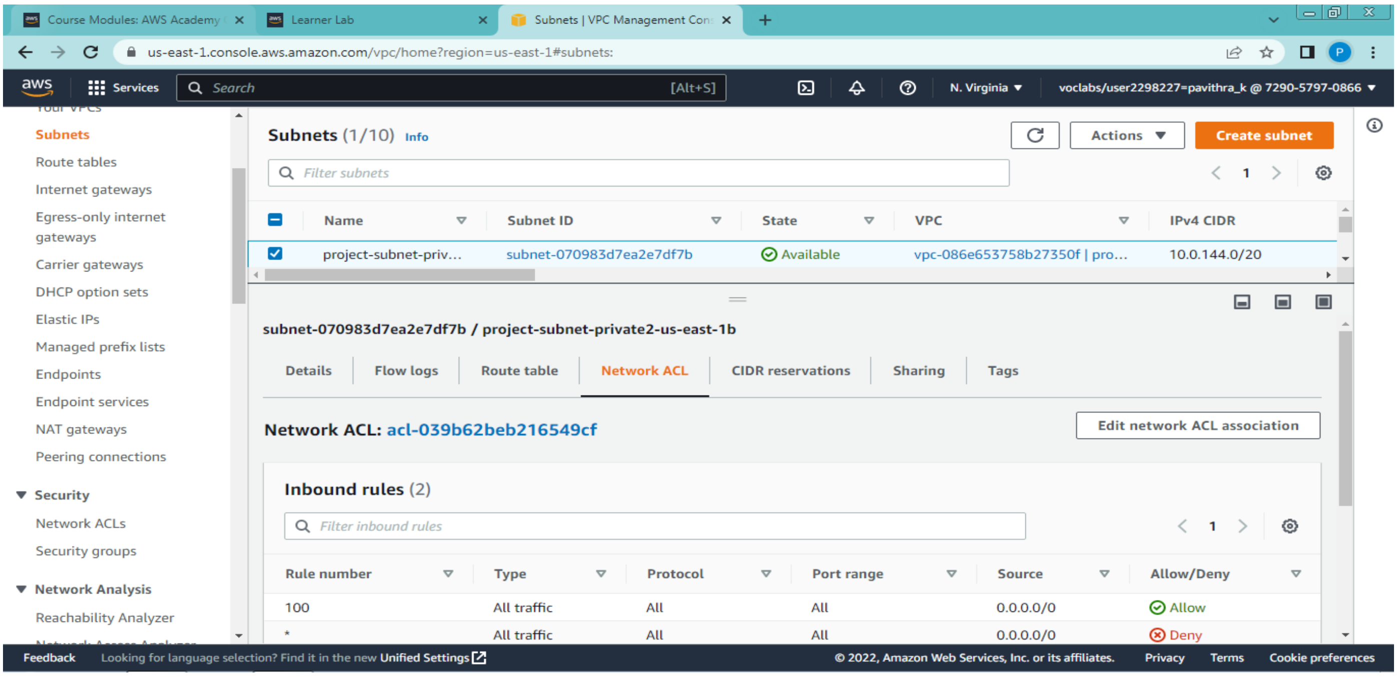
Filter routes

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| pl-63a5400a | vpce-037aa788115c47114 |

Feedback | Looking for language selection? Find it in the new Unified Settings | © 2022, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

- **Make use of Network ACLs for controlling inbound and outbound traffic in the VPC [Private subnet]**

## STEP – 18:



Course Modules: AWS Academy | Learner Lab | Subnets | VPC Management Con

us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets:

aws | Services | Search [Alt+S] | N. Virginia ▼ | voclabs/user2298227=pavithra_k @ 7290-5797-0866 ▼

- Your VPCs
- Subnets
- Route tables
- Internet gateways
- Egress-only internet gateways
- Carrier gateways
- DHCP option sets
- Elastic IPs
- Managed prefix lists
- Endpoints
- Endpoint services
- NAT gateways
- Peering connections

▼ Security
- Network ACLs
- Security groups

▼ Network Analysis
- Reachability Analyzer
- Network Access Analyzer

Subnets (1/10) Info | Actions ▼ | Create subnet

Filter subnets

| ☐ | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| ☑ | project-subnet-priv… | subnet-070983d7ea2e7df7b | ✓ Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |

subnet-070983d7ea2e7df7b / project-subnet-private2-us-east-1b

Details | Flow logs | Route table | Network ACL | CIDR reservations | Sharing | Tags

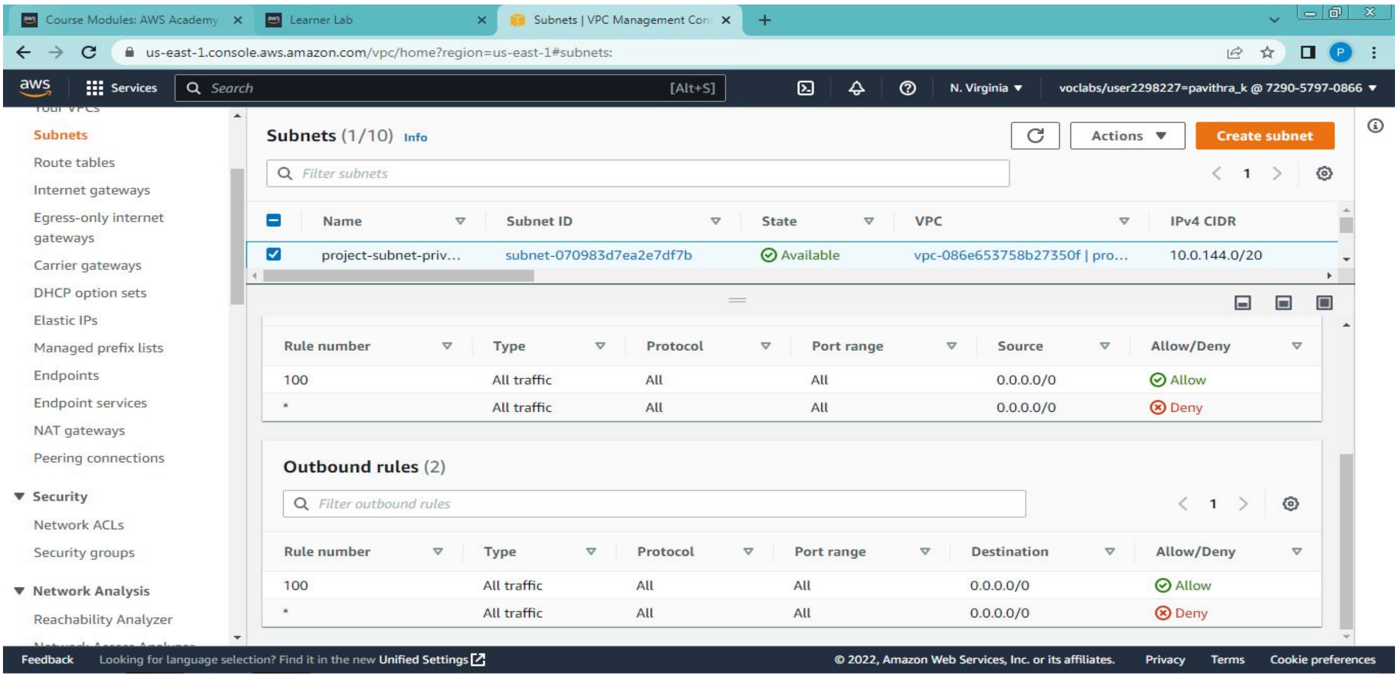
Network ACL: acl-039b62beb216549cf | Edit network ACL association

Inbound rules (2)

Filter inbound rules

| Rule number | Type | Protocol | Port range | Source | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | ✓ Allow |
| * | All traffic | All | All | 0.0.0.0/0 | ⊗ Deny |

Feedback | Looking for language selection? Find it in the new Unified Settings | © 2022, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences



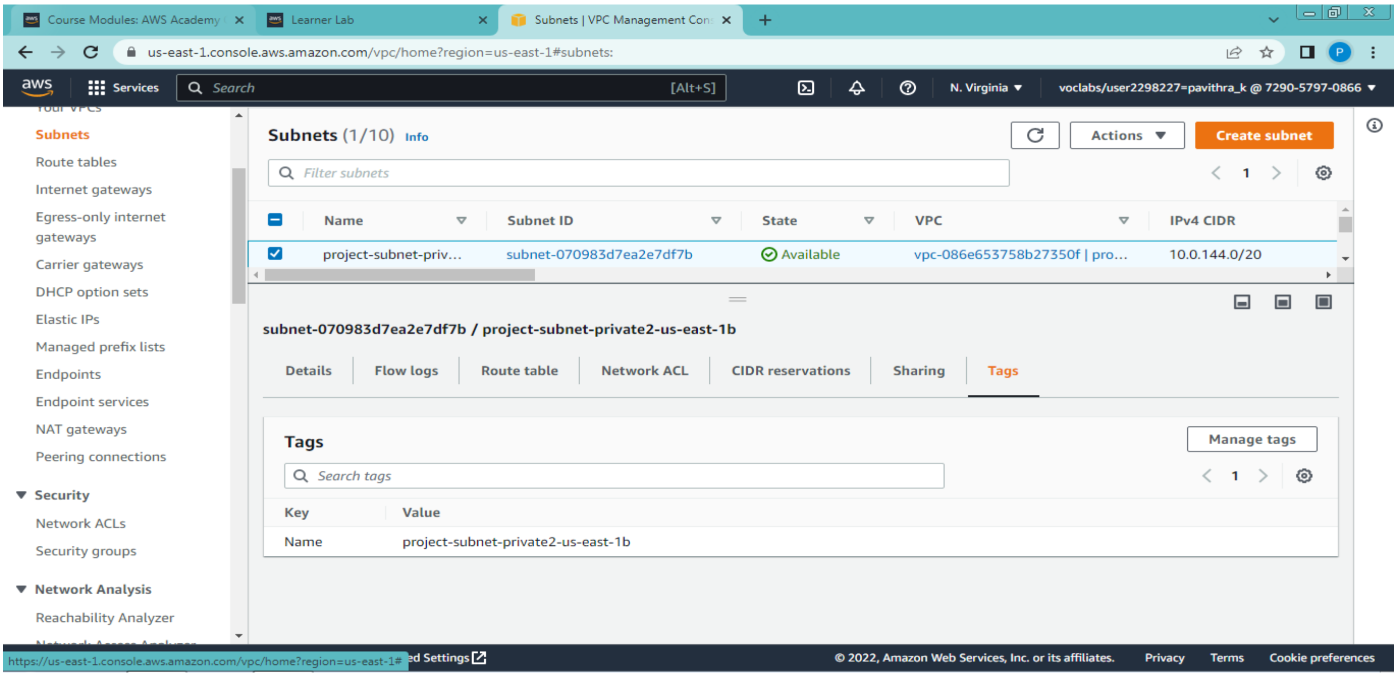
Subnets (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| ☑ | project-subnet-priv… | subnet-070983d7ea2e7df7b | ✓ Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |

| Rule number | Type | Protocol | Port range | Source | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | ✓ Allow |
| * | All traffic | All | All | 0.0.0.0/0 | ⊗ Deny |

Outbound rules (2)

| Rule number | Type | Protocol | Port range | Destination | Allow/Deny |
|---|---|---|---|---|---|
| 100 | All traffic | All | All | 0.0.0.0/0 | ✓ Allow |
| * | All traffic | All | All | 0.0.0.0/0 | ⊗ Deny |



Subnets (1/10) Info

| | Name | Subnet ID | State | VPC | IPv4 CIDR |
|---|---|---|---|---|---|
| ☑ | project-subnet-priv… | subnet-070983d7ea2e7df7b | ✓ Available | vpc-086e653758b27350f \| pro… | 10.0.144.0/20 |

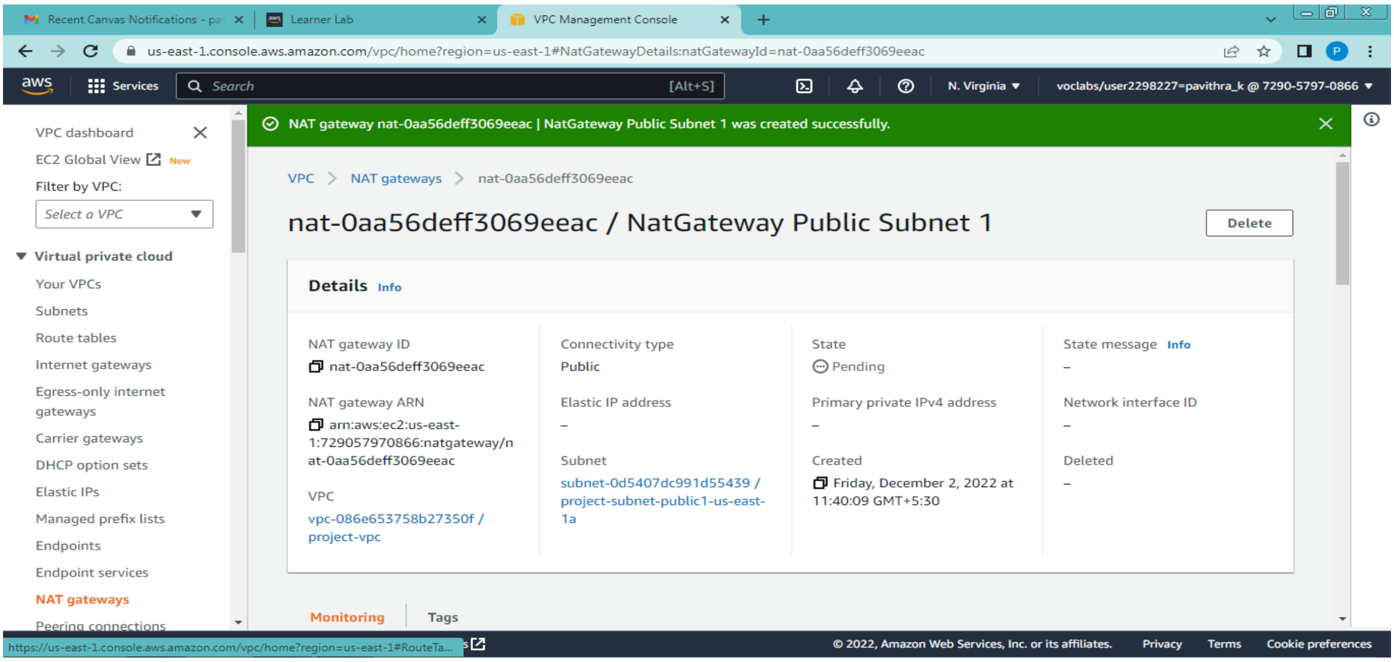
subnet-070983d7ea2e7df7b / project-subnet-private2-us-east-1b

Tags

| Key | Value |
|---|---|
| Name | project-subnet-private2-us-east-1b |

- **Launch a NAT Gateway so that internet access is provided to private resources**

## STEP – 19:


Recent Canvas Notifications - pa
Learner Lab
VPC Management Console
us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#NatGatewayDetails:natGatewayId=nat-0aa56deff3069eeac
aws
Services
Search
[Alt+S]
N. Virginia ▾
voclabs/user2298227=pavithra_k @ 7290-5797-0866 ▾
VPC dashboard
EC2 Global View
New
Filter by VPC:
Select a VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
VPC > NAT gateways > nat-0aa56deff3069eeac
Delete
Monitoring
Tags
https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#RouteTa...
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

NAT gateway nat-0aa56deff3069eeac | NatGateway Public Subnet 1 was created successfully.

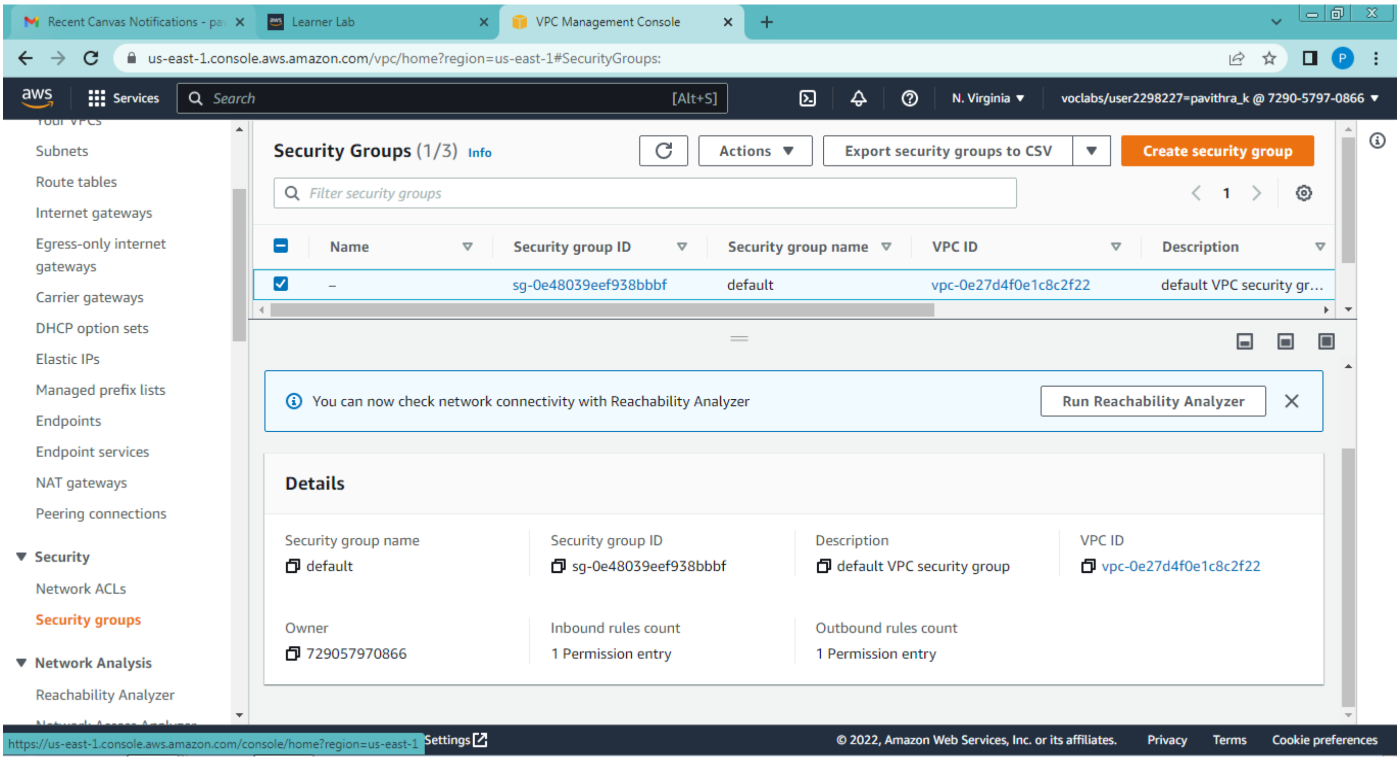
nat-0aa56deff3069eeac / NatGateway Public Subnet 1

Details Info

| | | | |
|---|---|---|---|
| NAT gateway ID<br>nat-0aa56deff3069eeac | Connectivity type<br>Public | State<br>Pending | State message Info<br>– |
| NAT gateway ARN<br>arn:aws:ec2:us-east-1:729057970866:natgateway/nat-0aa56deff3069eeac | Elastic IP address<br>– | Primary private IPv4 address<br>– | Network interface ID<br>– |
| VPC<br>vpc-086e653758b27350f / project-vpc | Subnet<br>subnet-0d5407dc991d55439 / project-subnet-public1-us-east-1a | Created<br>Friday, December 2, 2022 at 11:40:09 GMT+5:30 | Deleted<br>– |

- **Define security groups with appropriate inbound rules**

## STEP – 20:



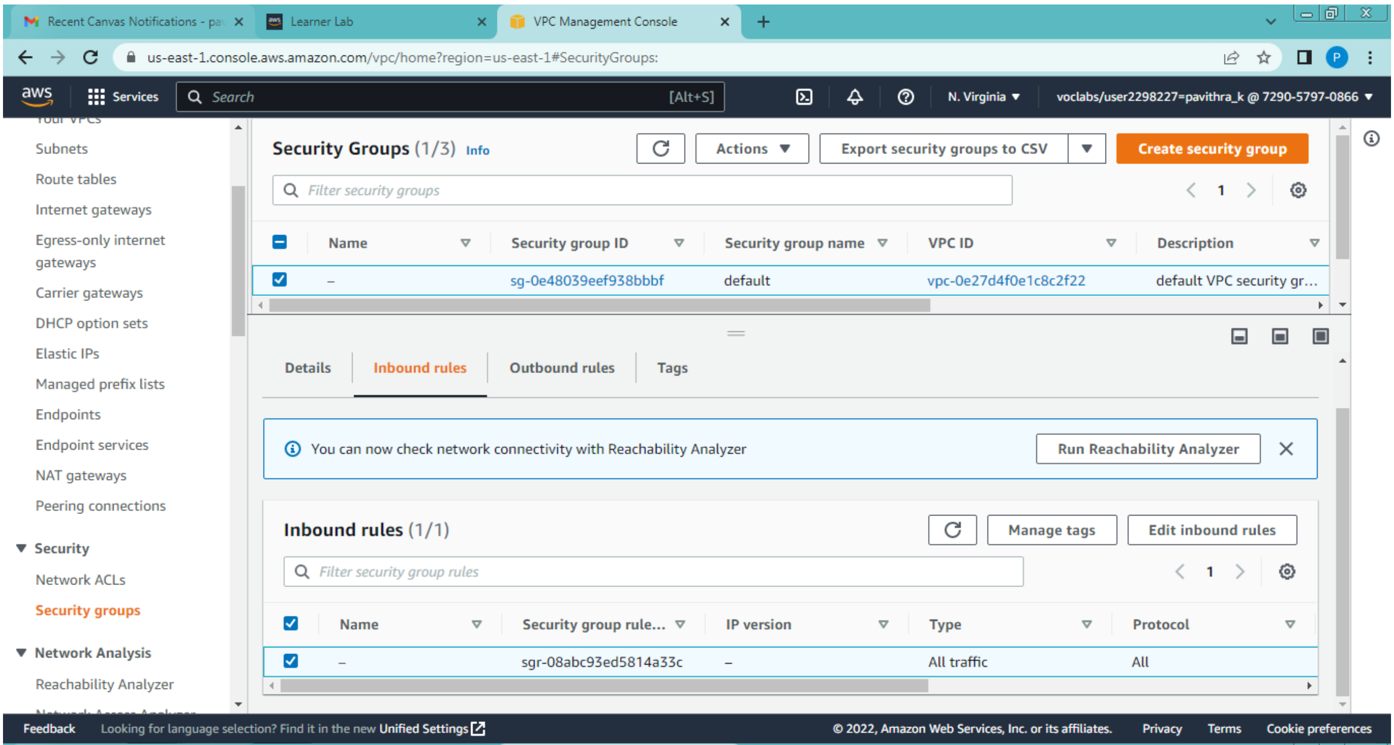
Security Groups (1/3) Info

| ☑ | Name | Security group ID | Security group name | VPC ID | Description |
|---|---|---|---|---|---|
| ☑ | – | sg-0e48039eef938bbbf | default | vpc-0e27d4f0e1c8c2f22 | default VPC security gr... |

You can now check network connectivity with Reachability Analyzer

Details

| | | | |
|---|---|---|---|
| Security group name<br>default | Security group ID<br>sg-0e48039eef938bbbf | Description<br>default VPC security group | VPC ID<br>vpc-0e27d4f0e1c8c2f22 |
| Owner<br>729057970866 | Inbound rules count<br>1 Permission entry | Outbound rules count<br>1 Permission entry | |



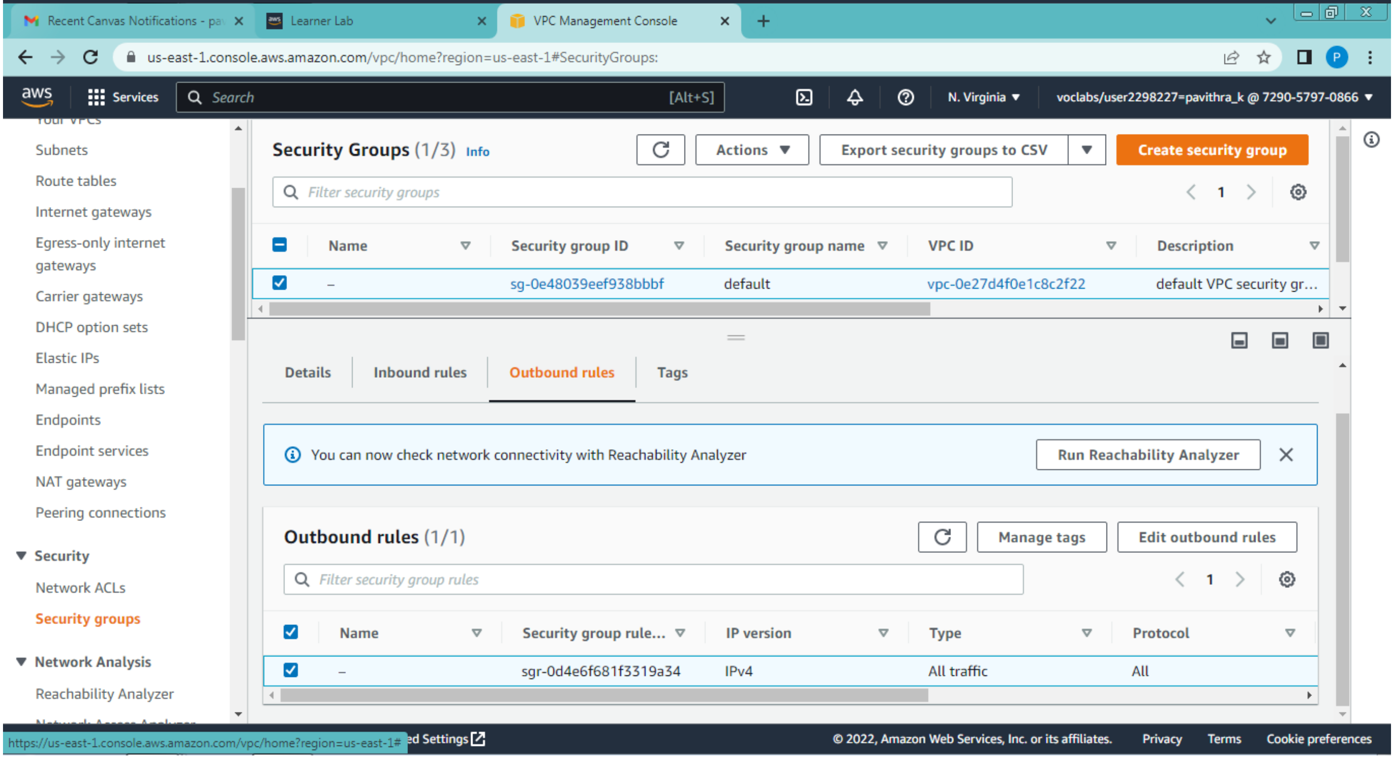
Security Groups (1/3) Info

| | Name | Security group ID | Security group name | VPC ID | Description |
|---|---|---|---|---|---|
| ☑ | – | sg-0e48039eef938bbbf | default | vpc-0e27d4f0e1c8c2f22 | default VPC security gr… |

You can now check network connectivity with Reachability Analyzer

Inbound rules (1/1)

| | Name | Security group rule... | IP version | Type | Protocol |
|---|---|---|---|---|---|
| ☑ | – | sgr-08abc93ed5814a33c | – | All traffic | All |



Security Groups (1/3) Info

| | Name | Security group ID | Security group name | VPC ID | Description |
|---|---|---|---|---|---|
| ☑ | – | sg-0e48039eef938bbbf | default | vpc-0e27d4f0e1c8c2f22 | default VPC security gr… |

You can now check network connectivity with Reachability Analyzer

Outbound rules (1/1)

| | Name | Security group rule... | IP version | Type | Protocol |
|---|---|---|---|---|---|
| ☑ | – | sgr-0d4e6f681f3319a34 | IPv4 | All traffic | All |