

# Multi-Region Active-Active Global Survey Application on AWS

## Project Description:

Designed and deployed a **multi-region, fault-tolerant, and globally available web application** using AWS services. The project demonstrates a real-world **active-active architecture** where traffic and data are synchronized between the **Mumbai (ap-south-1)** and **Virginia (us-east-1)** regions.

## Business Problem & Goal

Modern enterprises require:

- Global website availability
- Zero downtime during regional failures
- Low-latency user experience
- Automatic failover
- Multi-region database replication
- Scalable frontend + backend

This project solves all of the above with real AWS services.

## AWS Services Used

### Compute

- Amazon EC2
- Auto Scaling Groups

- AMI Golden Image

## **Database**

- Amazon Aurora MySQL
- Aurora Global Database (cross-region replication)

## **Networking**

- Amazon VPC
- Subnets (Public + Private)
- Internet Gateway
- NAT Gateway
- VPC Peering
- Route Tables
- Security Groups

## **Load Balancing & Traffic Routing**

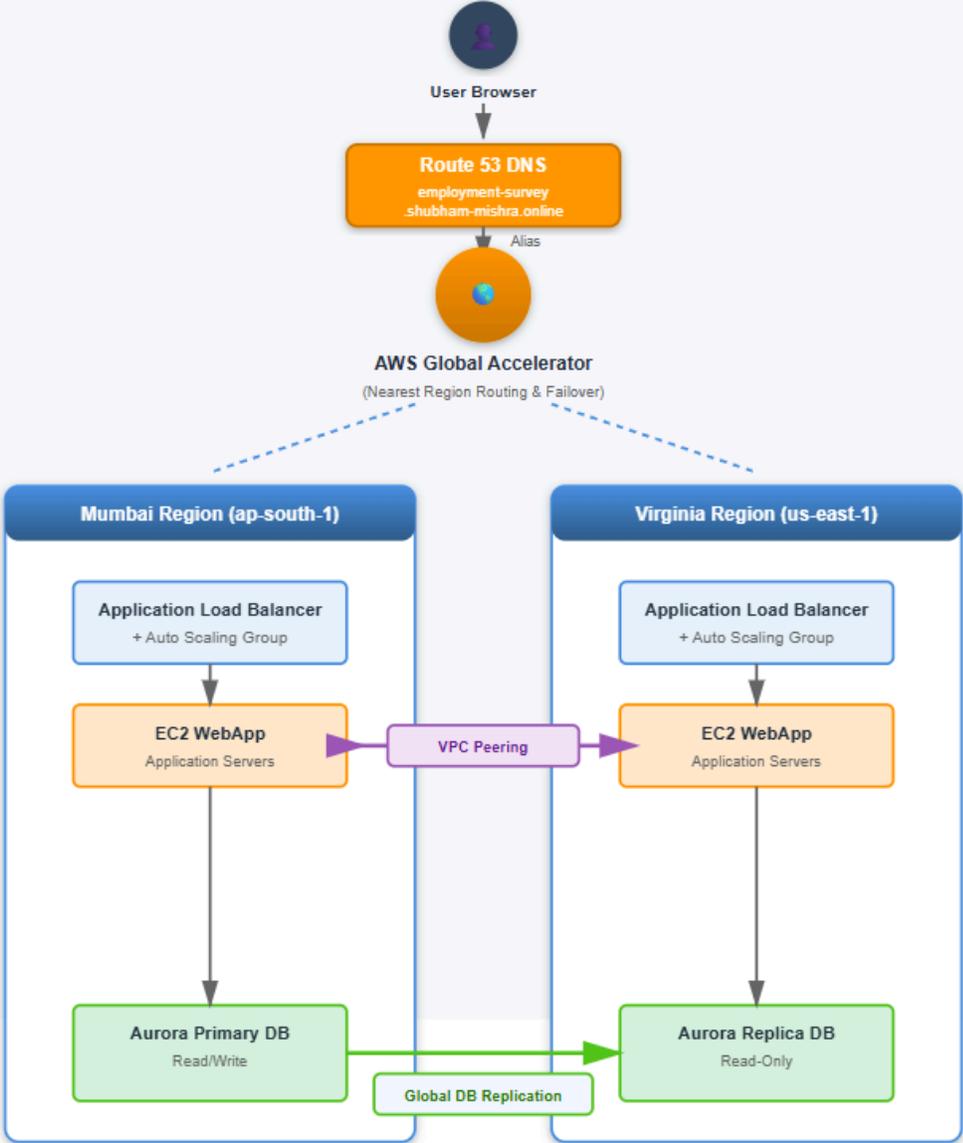
- Application Load Balancer (ALB)
- AWS Global Accelerator
- Amazon Route 53

## **Storage**

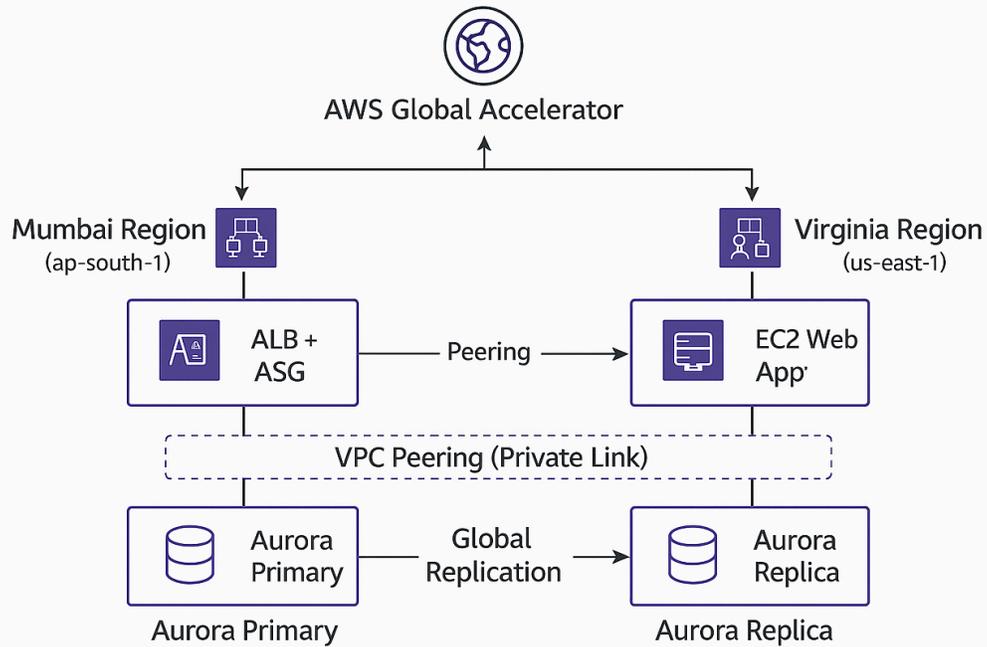
- Amazon S3 (optional for logs)

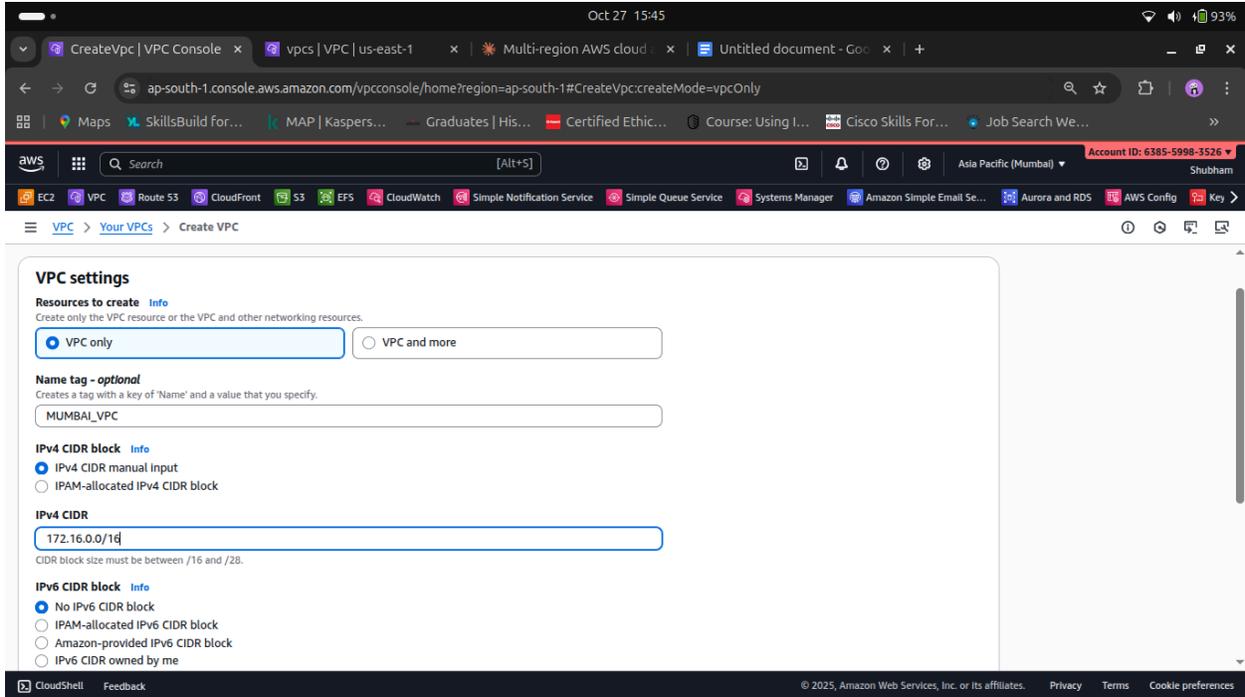
## **Architecture Diagram**

# AWS Global Accelerator Architecture

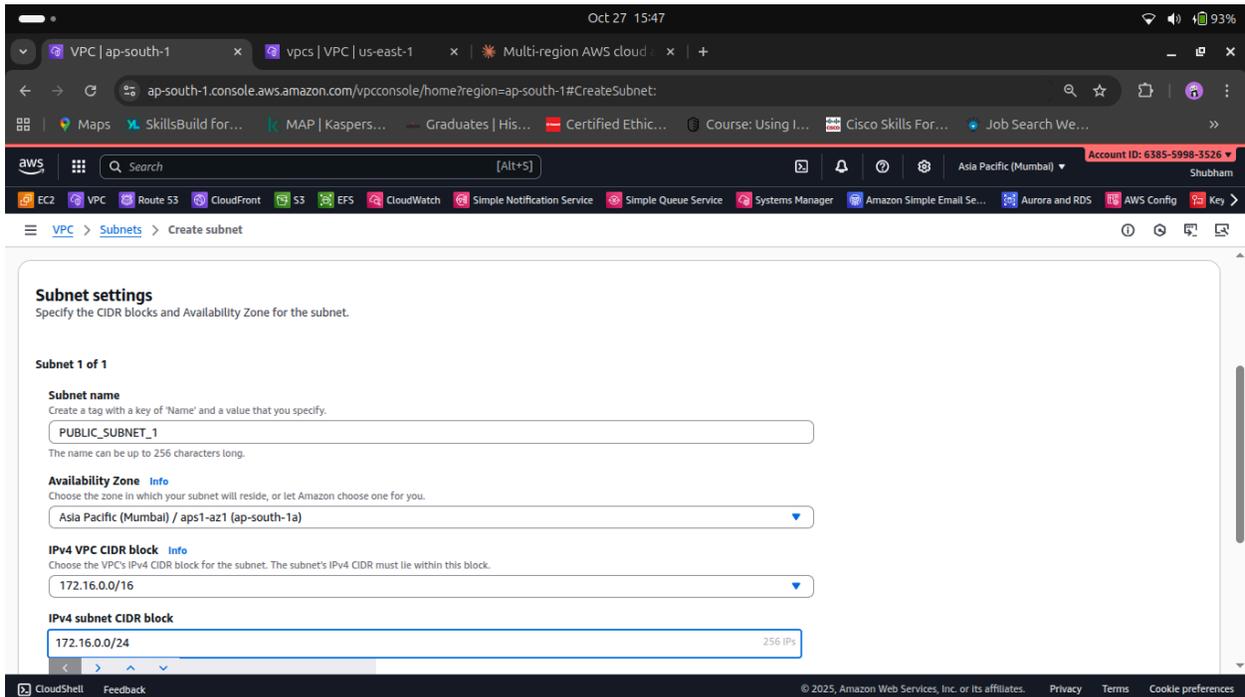


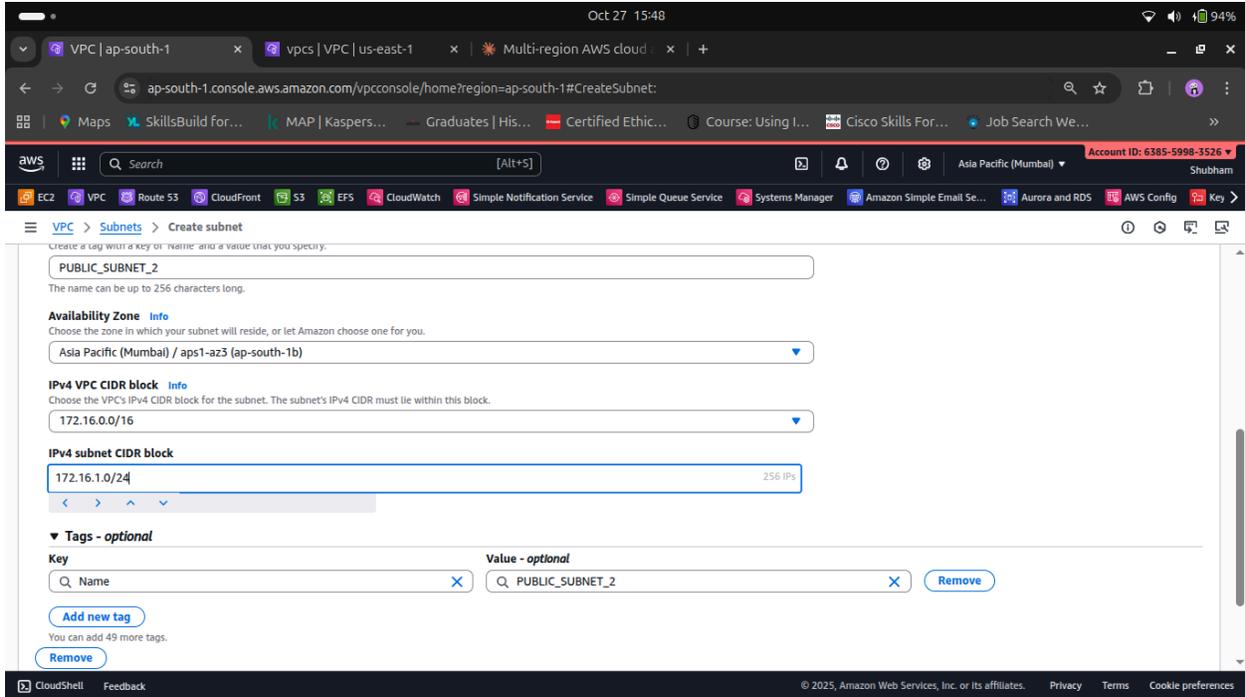
# Multi-Region Active-Active Global Survey Application on AWS



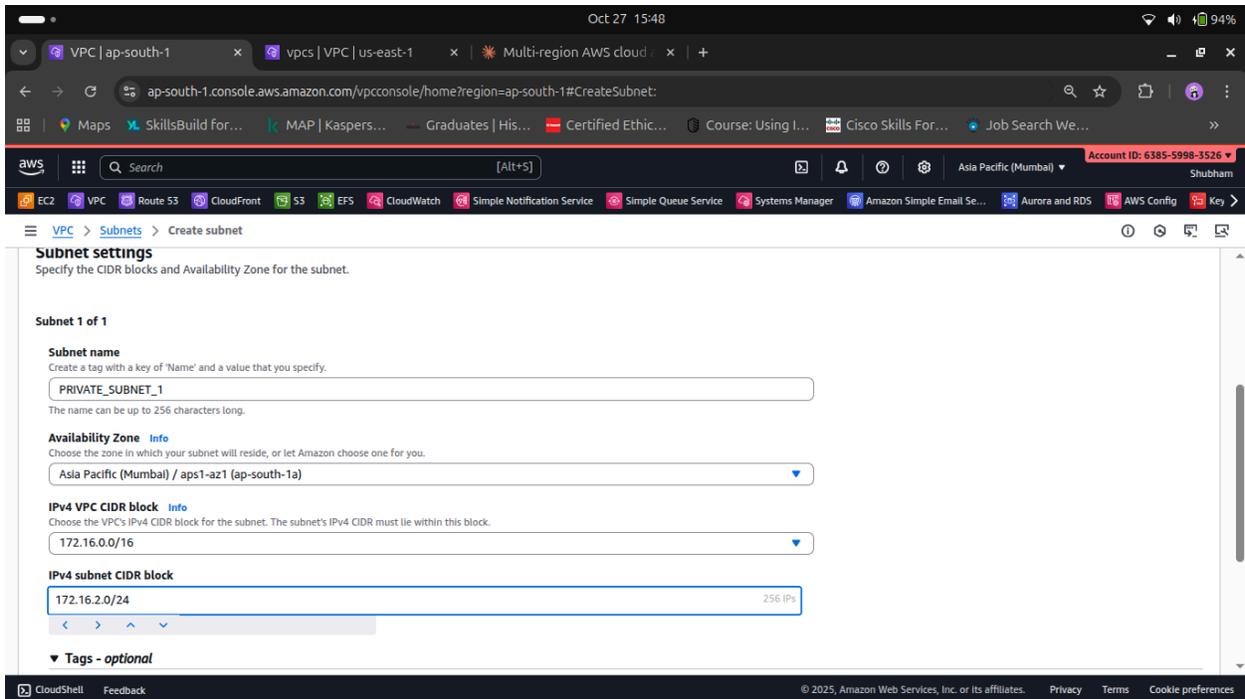


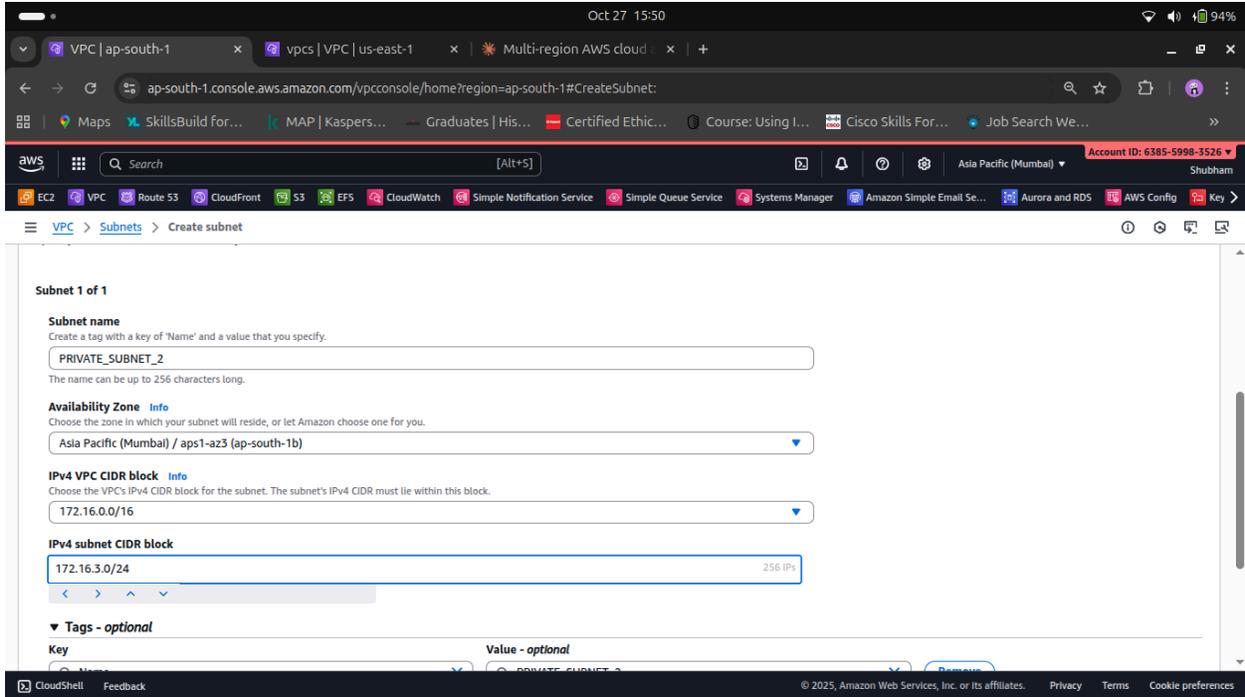
## Step 1) We will create VPC with cidr block 172.16.0.0/16



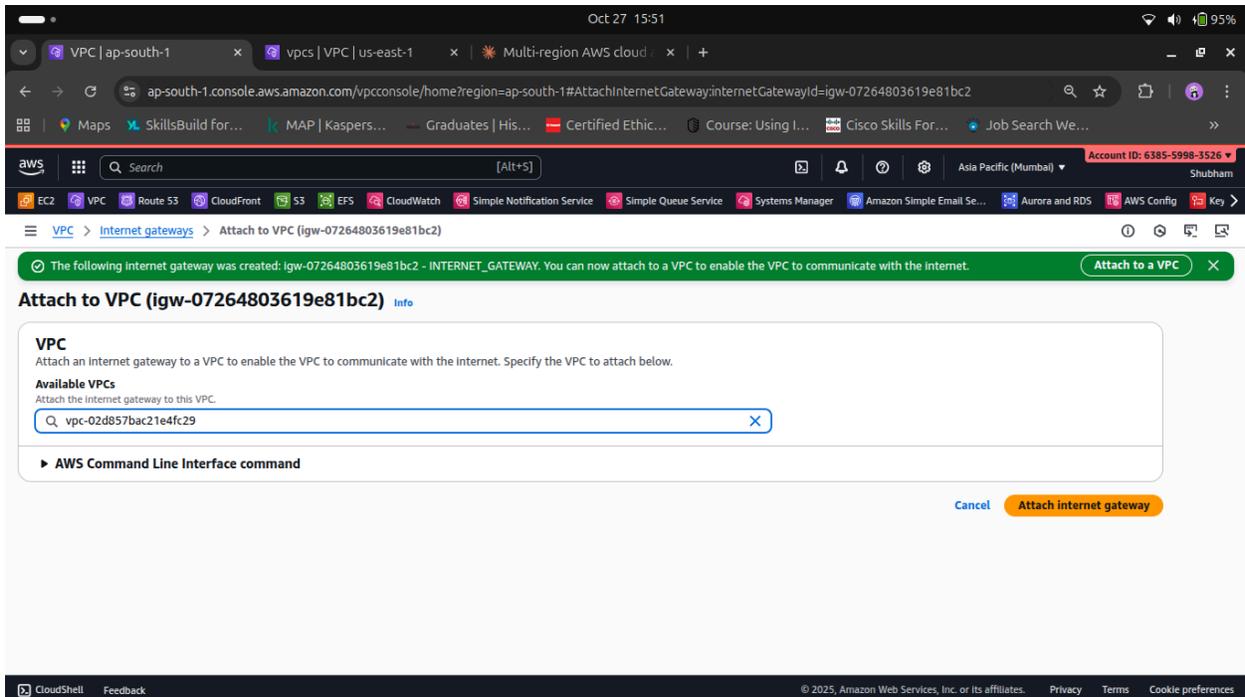


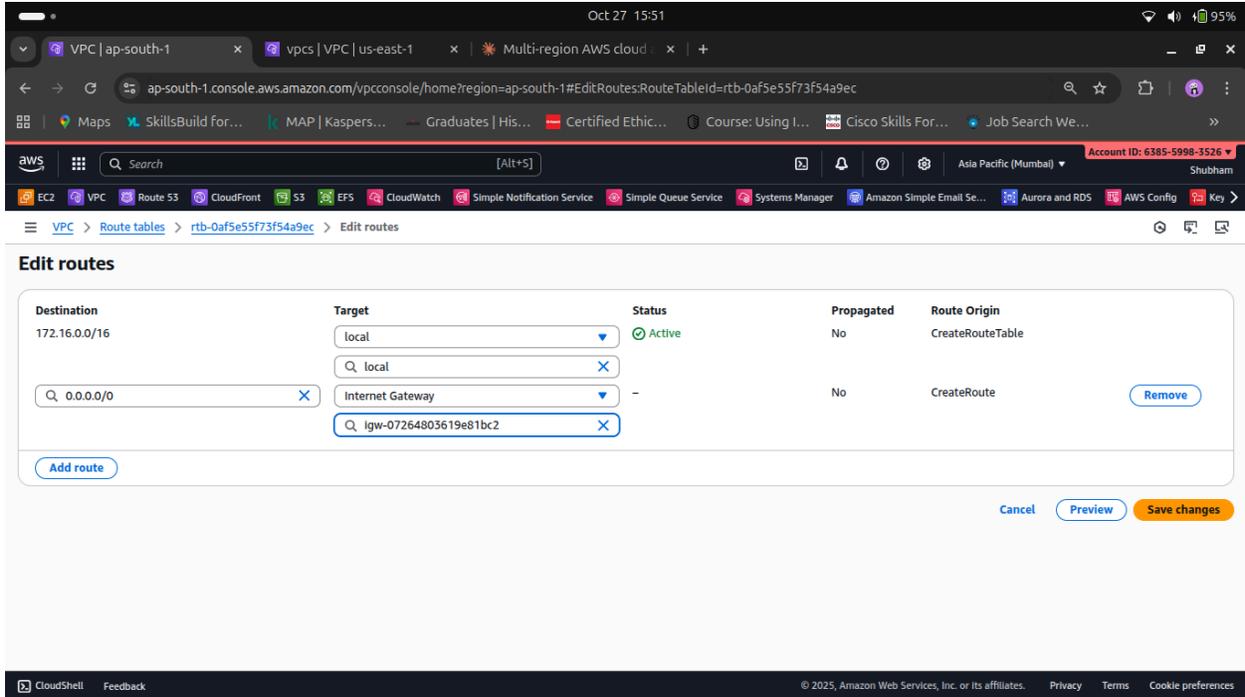
**Step 2) We will create two public subnets with cidr 172.16.0.0/24 and 172.16.1.0/24**



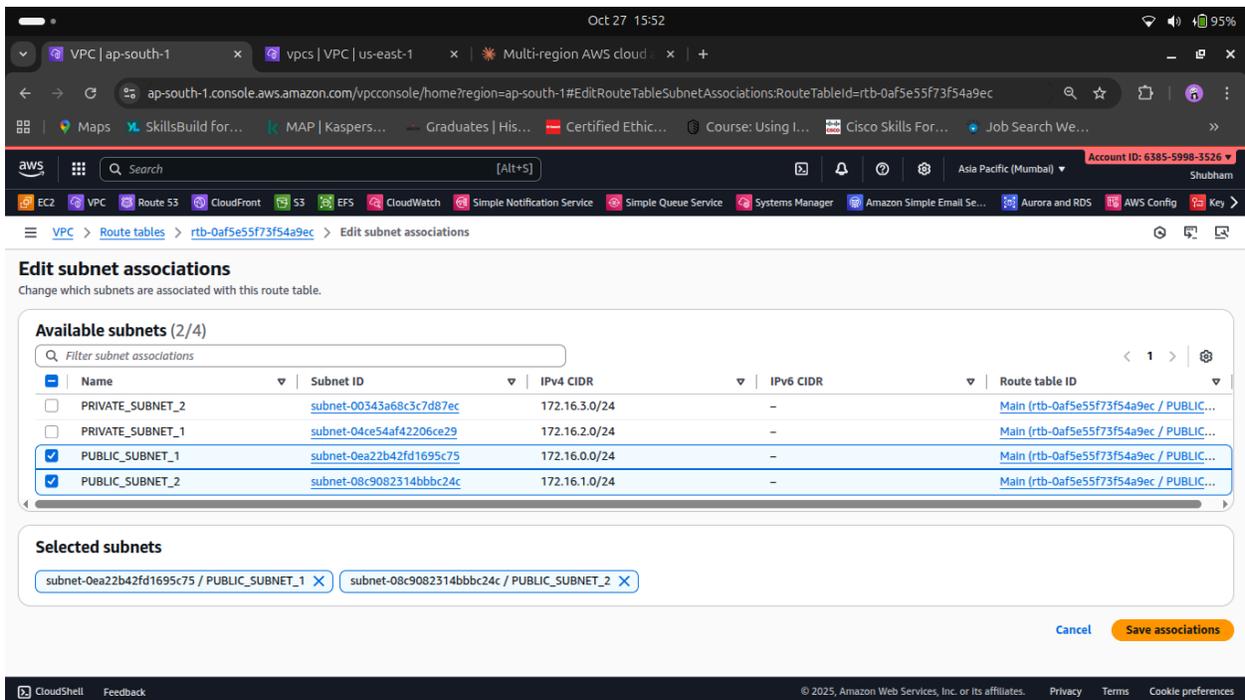


**Step 3) We will create two private subnets with cidr block as 172.16.2.0/24 and 172.16.3.0/24**





**Step 4) We will create internet gateway and attach to vpc also we will edit route in main route table to allow internet connectivity for public subnets**



## Step 5) Attach public subnet to public route table where internet connectivity is established

The screenshot shows the AWS Management Console interface for editing subnet associations. The breadcrumb navigation is VPC > Route tables > rtb-08f6ec21716b6fe81 > Edit subnet associations. The page title is 'Edit subnet associations' with a subtitle 'Change which subnets are associated with this route table.'

**Available subnets (2/4)**

<input type="checkbox"/>	Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/>	PRIVATE_SUBNET_2	subnet-00343a68c3c7d87ec	172.16.3.0/24	-	Main (rtb-0af5e5f73f54a9ec / PUBLIC...
<input checked="" type="checkbox"/>	PRIVATE_SUBNET_1	subnet-04ce54af42206ce29	172.16.2.0/24	-	Main (rtb-0af5e5f73f54a9ec / PUBLIC...
<input type="checkbox"/>	PUBLIC_SUBNET_1	subnet-0ea22b42fd1695c75	172.16.0.0/24	-	rtb-0af5e5f73f54a9ec / PUBLIC_RT
<input type="checkbox"/>	PUBLIC_SUBNET_2	subnet-08c9082314bbbc24c	172.16.1.0/24	-	rtb-0af5e5f73f54a9ec / PUBLIC_RT

**Selected subnets**

subnet-00343a68c3c7d87ec / PRIVATE\_SUBNET\_2 X    subnet-04ce54af42206ce29 / PRIVATE\_SUBNET\_1 X

Buttons: Cancel, Save associations

## Step 6) Attach private subnets to private route table

The screenshot shows the 'Create NAT gateway' page in the AWS Management Console. A green notification banner at the top states: 'Elastic IP address 13.200.57.77 (elpalloc-0b4b5c89c49dcb6f4) allocated.'

**Create NAT gateway** [Info](#)

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the Internet.

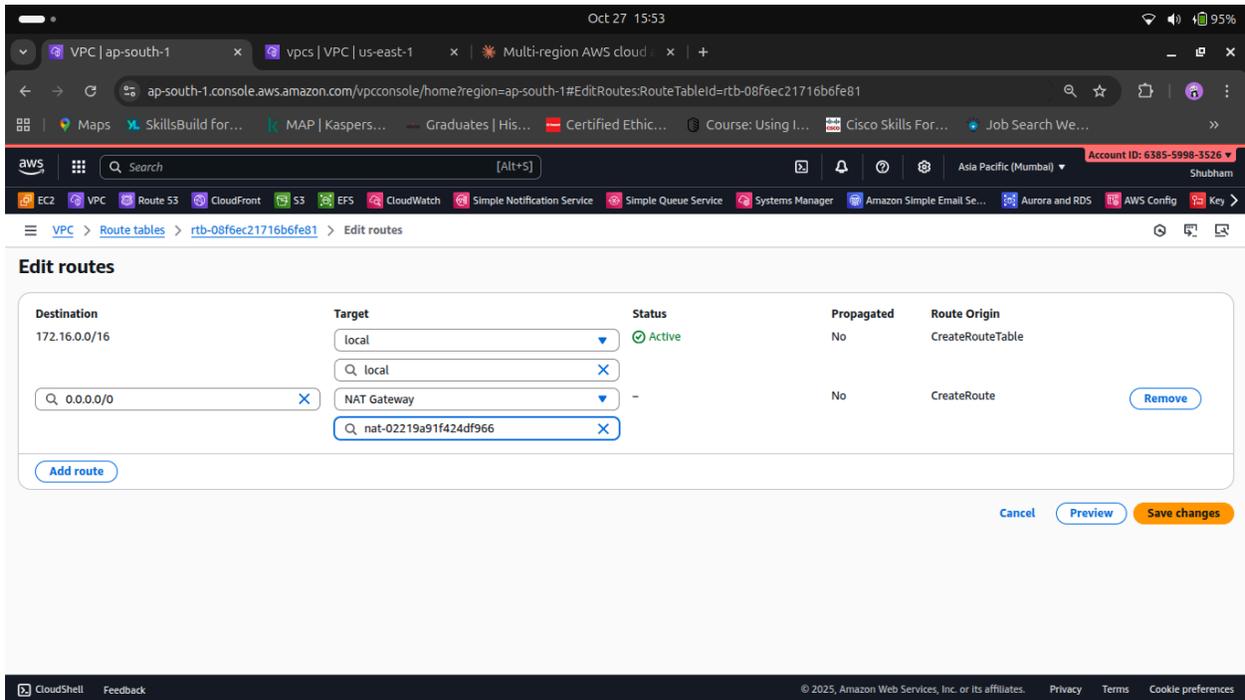
**NAT gateway settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.  
NAT\_GATEWAY  
The name can be up to 256 characters long.

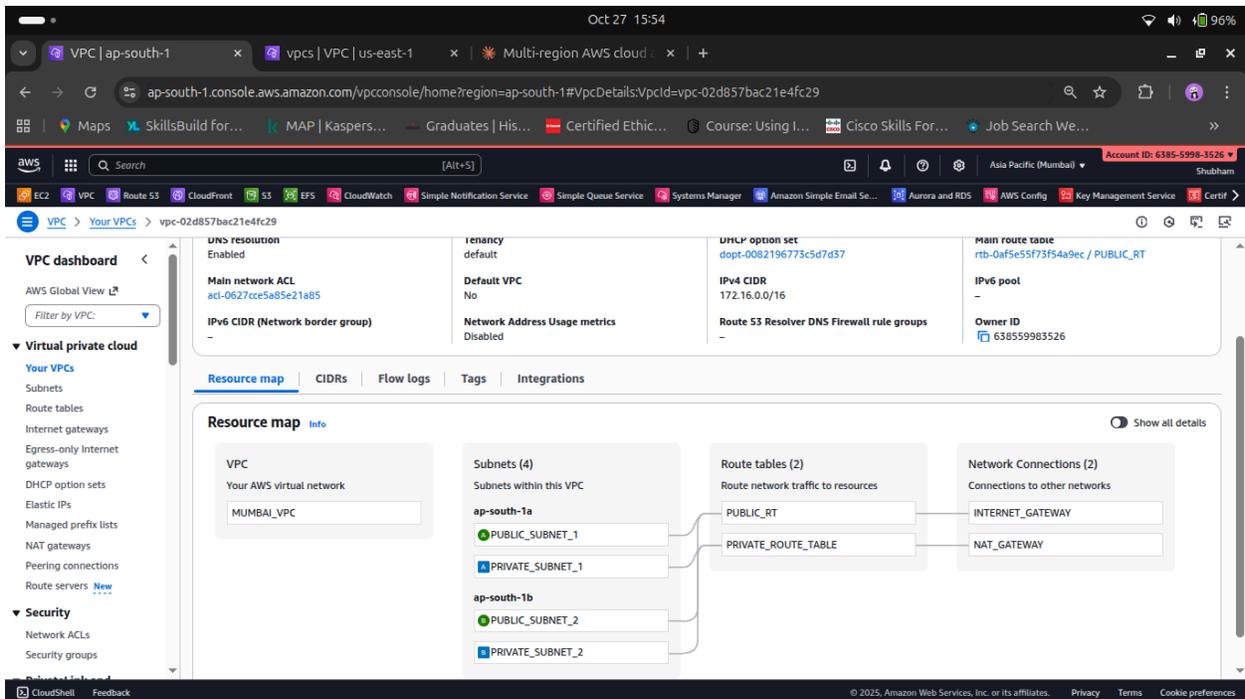
**Subnet**  
Select a subnet in which to create the NAT gateway.  
subnet-0ea22b42fd1695c75 (PUBLIC\_SUBNET\_1)

**Connectivity type**  
Select a connectivity type for the NAT gateway.  
 Public  
 Private

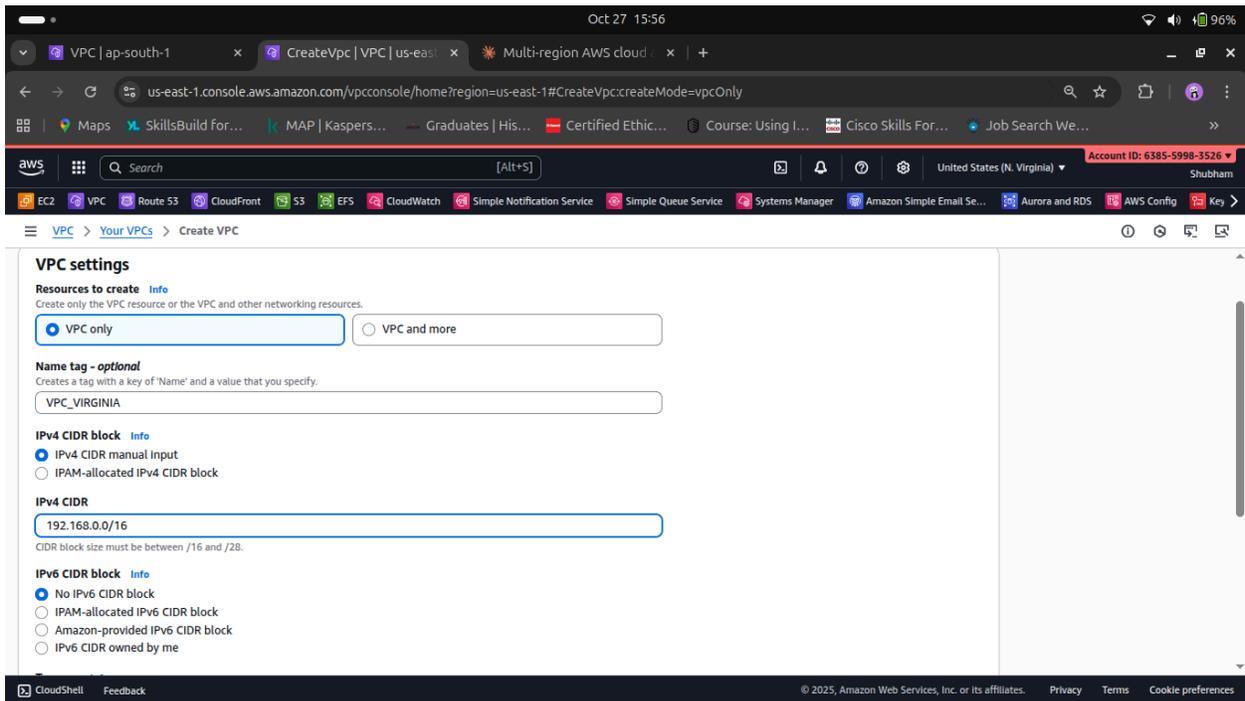
**Elastic IP allocation ID** [Info](#)  
Assign an Elastic IP address to the NAT gateway.  
elpalloc-0b4b5c89c49dcb6f4 [Allocate Elastic IP](#)



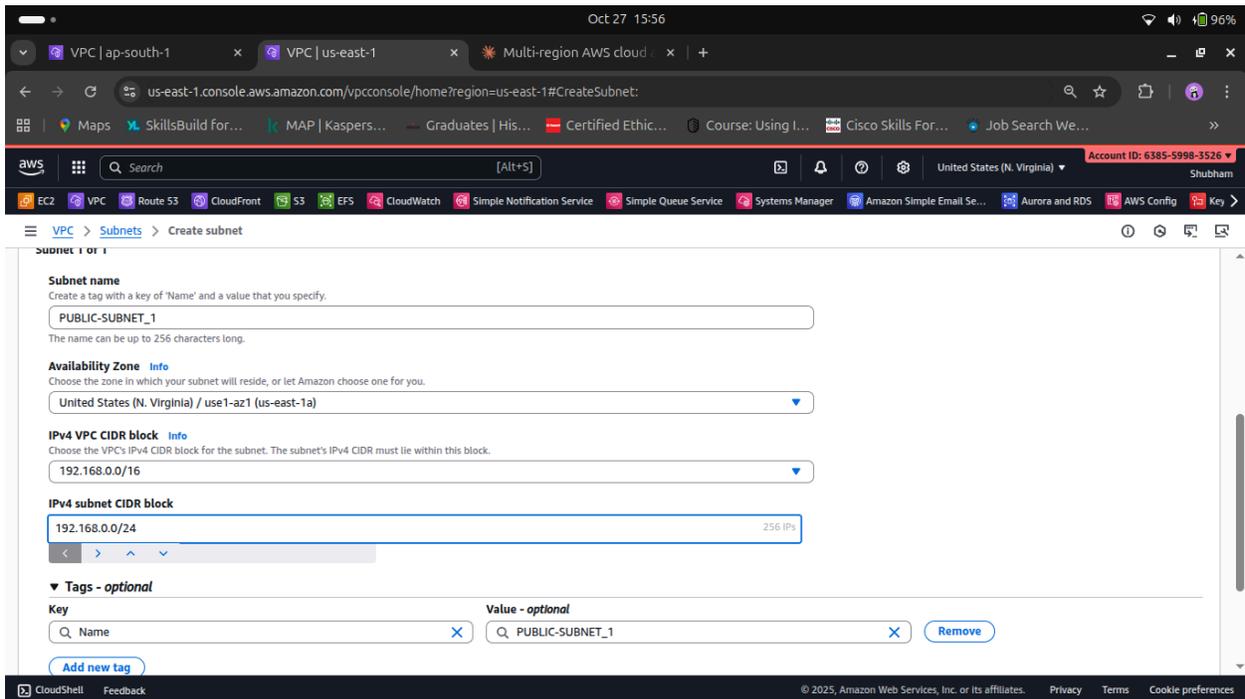
**Step 7) We will create nat gateway to provide outbound internet access to the resources in private subnet**

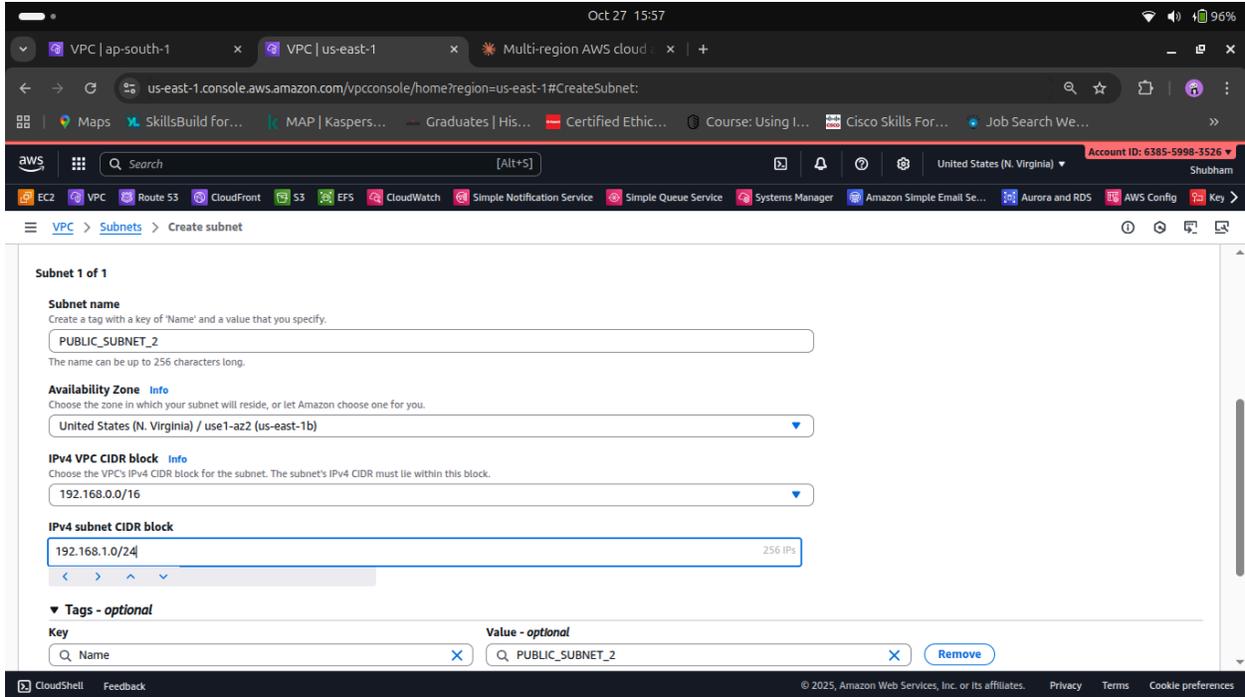


**Step 8) We can visualize all the resources we created in vpc**

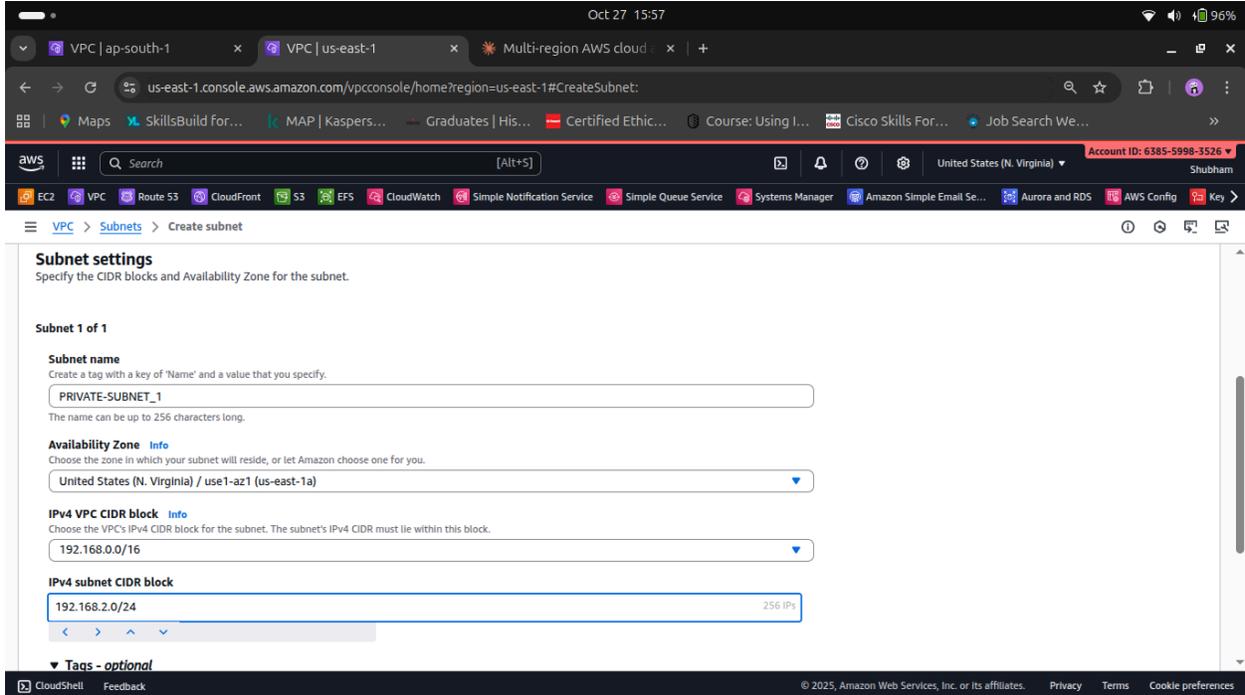


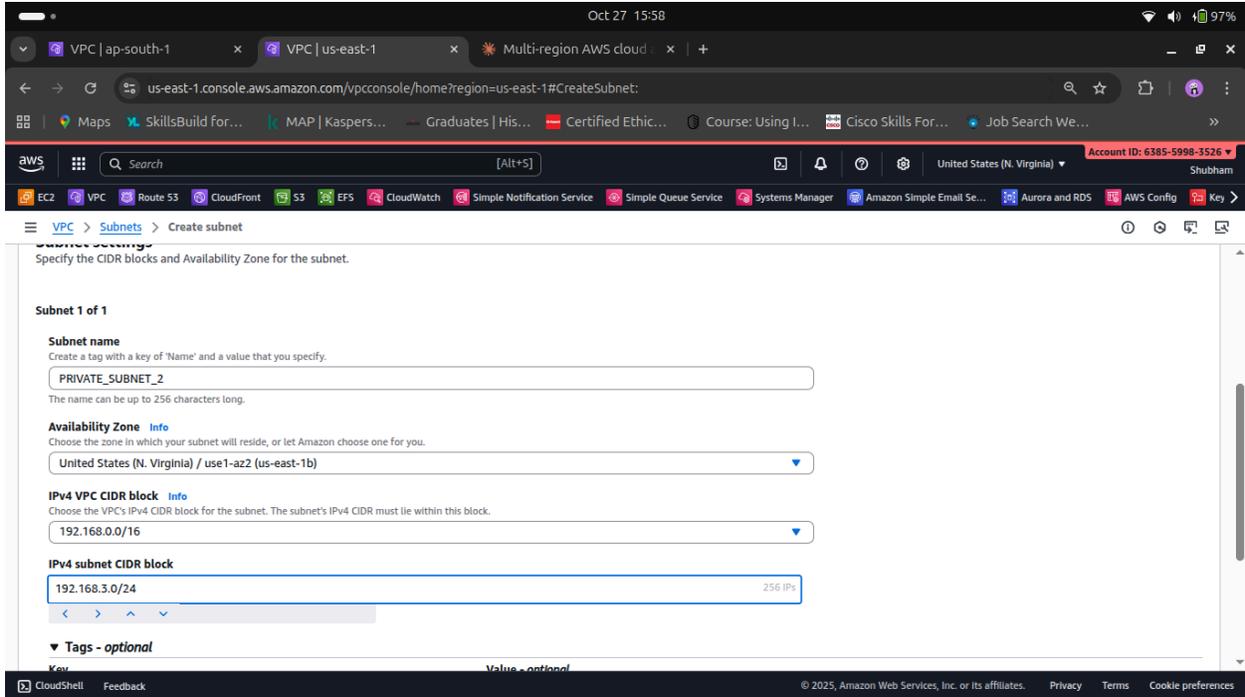
**Step 9) Similarly will create another vpc in north virginia region with cidr block 192.168.0.0/16**



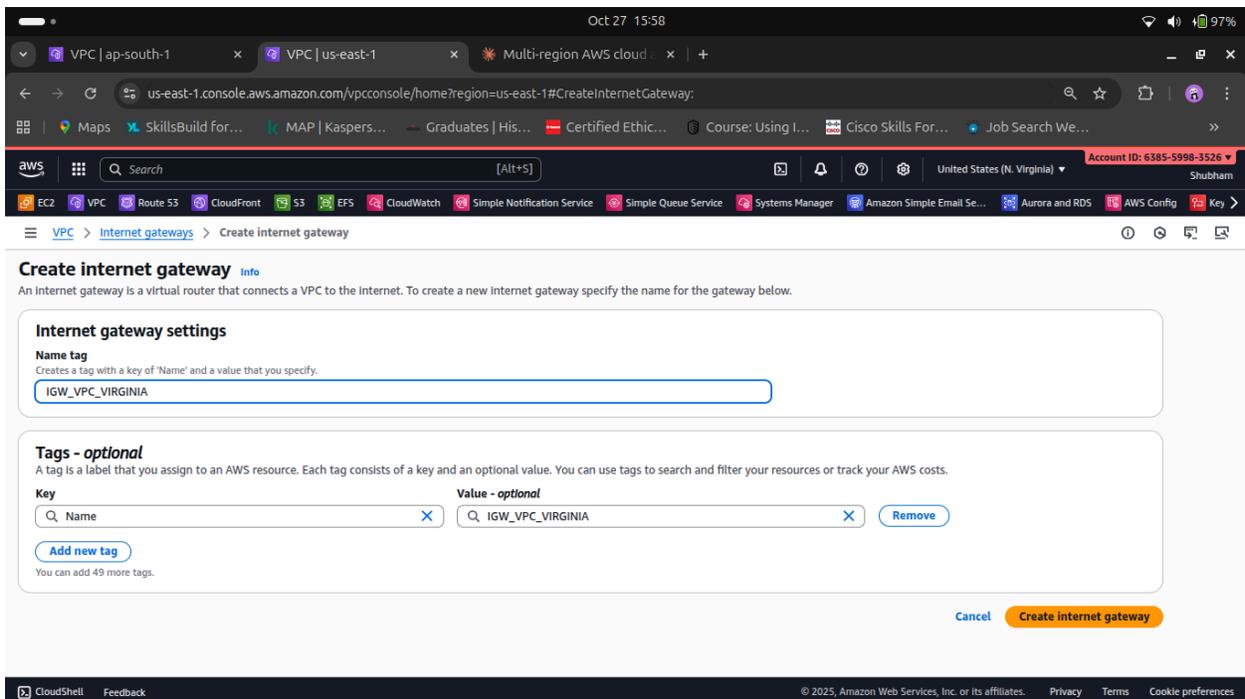


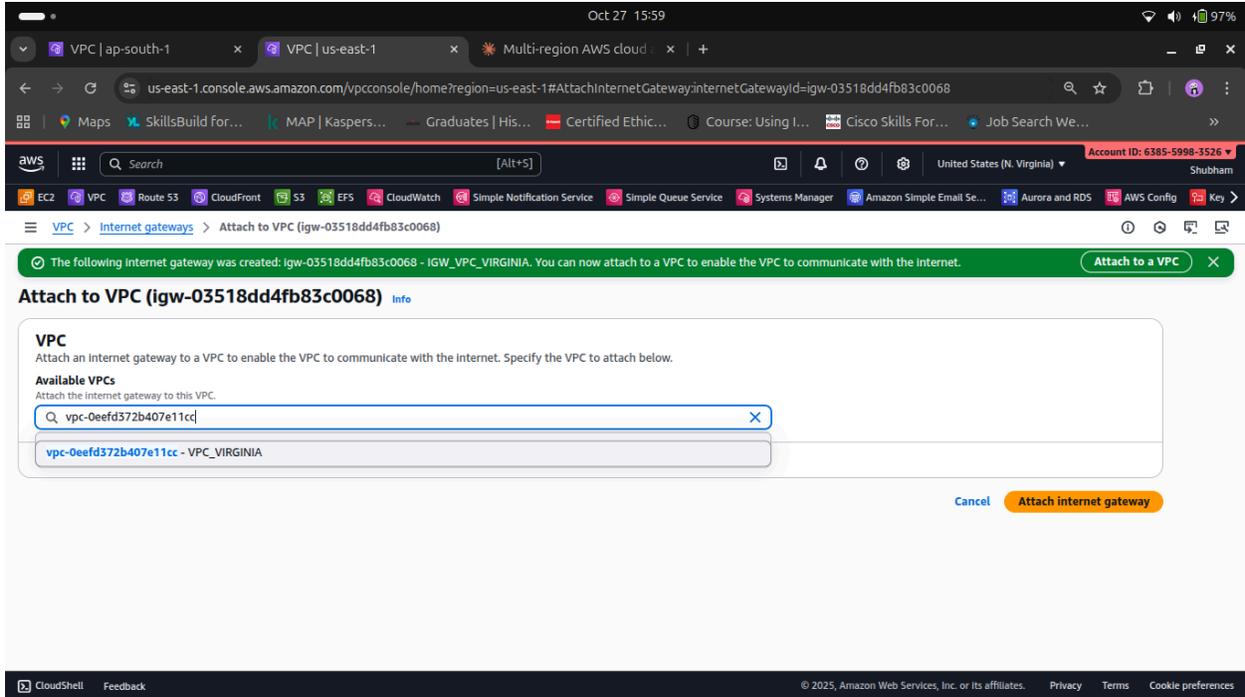
**Step 10) We will create two public subnets with cidr block 192.168.0.0/24 and cidr block 192.168.1.0/24**



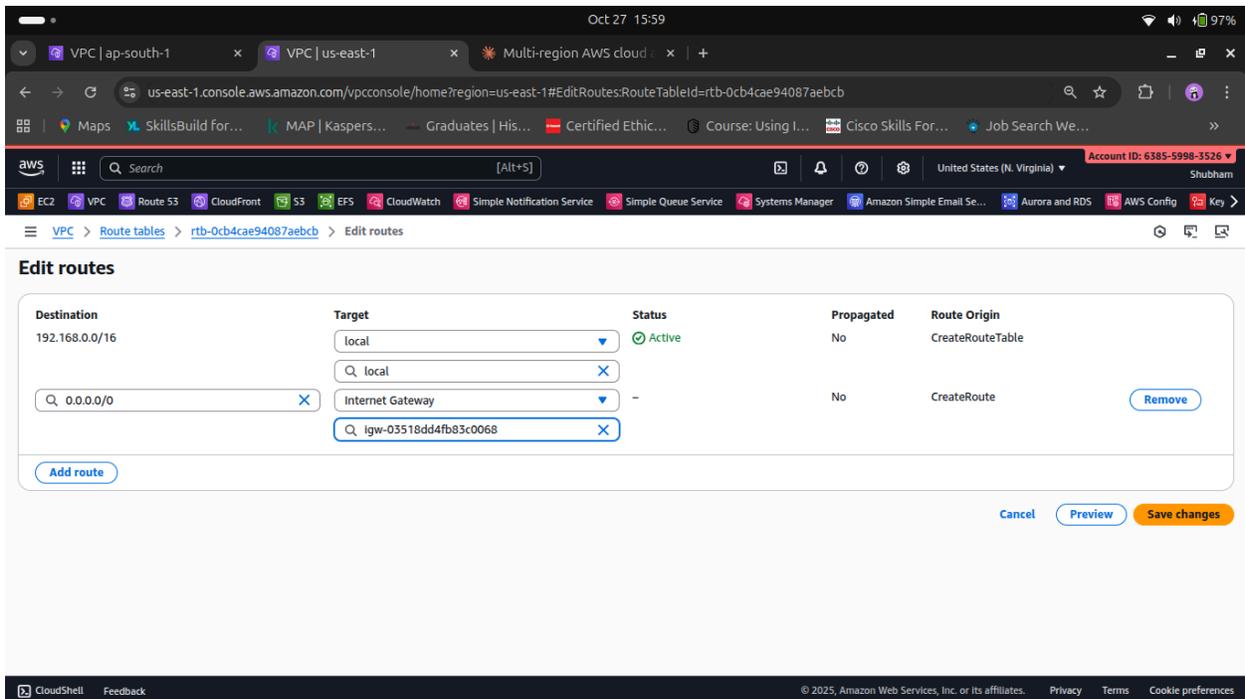


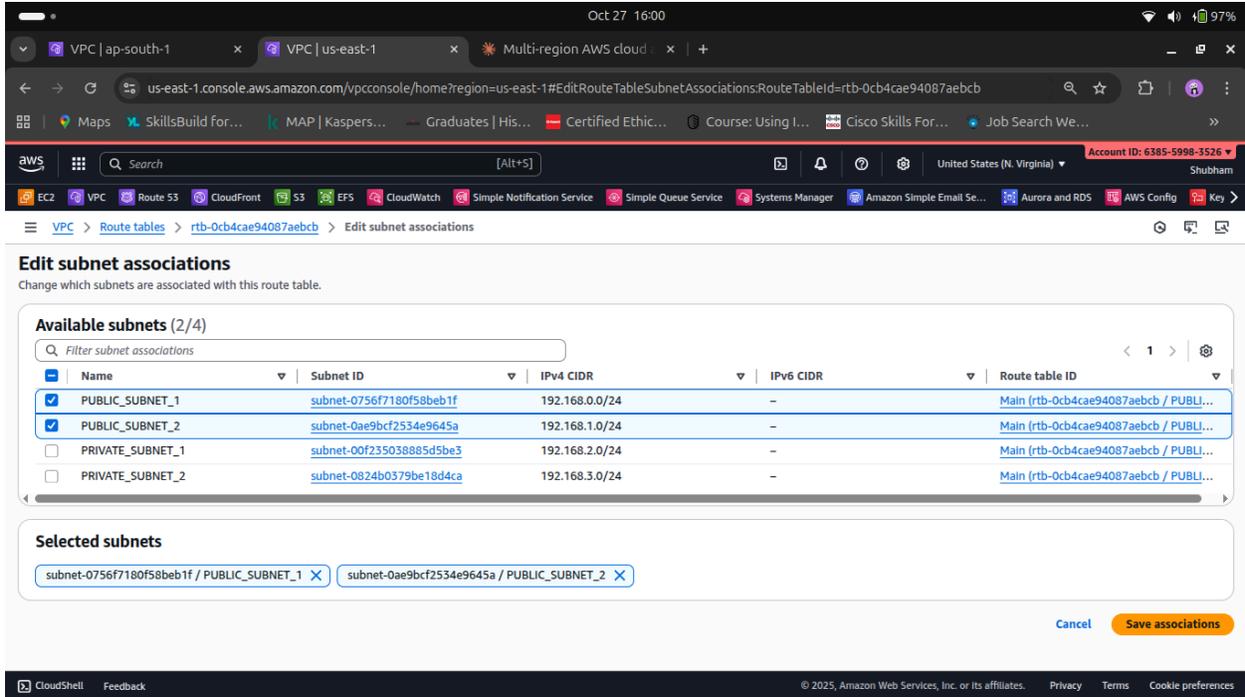
**Step 11) We will create two private subnets now with cidr block 192.168.2.0/24 and 192.168.3.0/24**



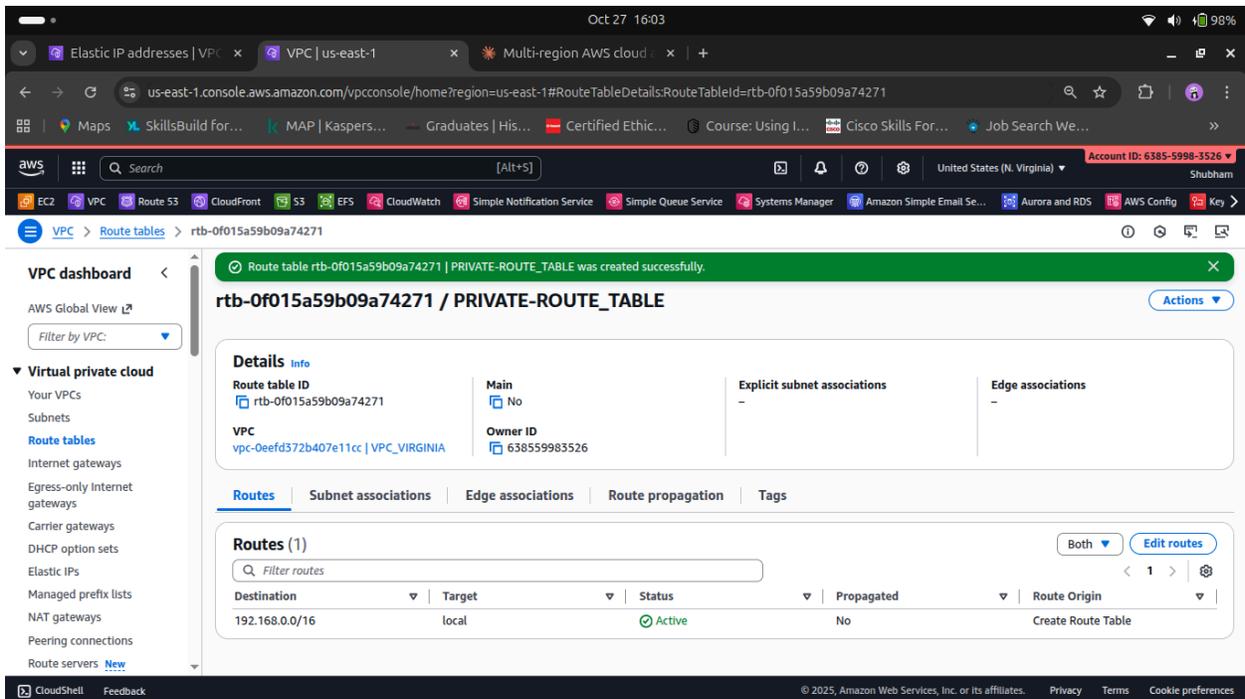


**Step 12) Again we will create internet gateway and attach it to north virginia vpc**





## Step 13) Edit route table to add route for internet gateway and also associate public subnet with public route table



**Edit subnet associations**  
Change which subnets are associated with this route table.

**Available subnets (2/4)**

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/> PUBLIC_SUBNET_1	subnet-0756f7180f58beb1f	192.168.0.0/24	-	rtb-0cb4cae94087aebcb / PUBLIC-ROU...
<input type="checkbox"/> PUBLIC_SUBNET_2	subnet-0ae9bcf2534e9645a	192.168.1.0/24	-	rtb-0cb4cae94087aebcb / PUBLIC-ROU...
<input checked="" type="checkbox"/> PRIVATE_SUBNET_1	subnet-00f235038885d5be3	192.168.2.0/24	-	Main (rtb-0cb4cae94087aebcb / PUBLI...
<input checked="" type="checkbox"/> PRIVATE_SUBNET_2	subnet-0824b0379be18d4ca	192.168.3.0/24	-	Main (rtb-0cb4cae94087aebcb / PUBLI...

**Selected subnets**

subnet-00f235038885d5be3 / PRIVATE\_SUBNET\_1 x subnet-0824b0379be18d4ca / PRIVATE\_SUBNET\_2 x

Cancel Save associations

## Step 14) Associate private subnet with private route table

**Create NAT gateway**

Elastic IP address 34.229.137.49 (eipalloc-0b2d824de22710635) allocated.

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the Internet.

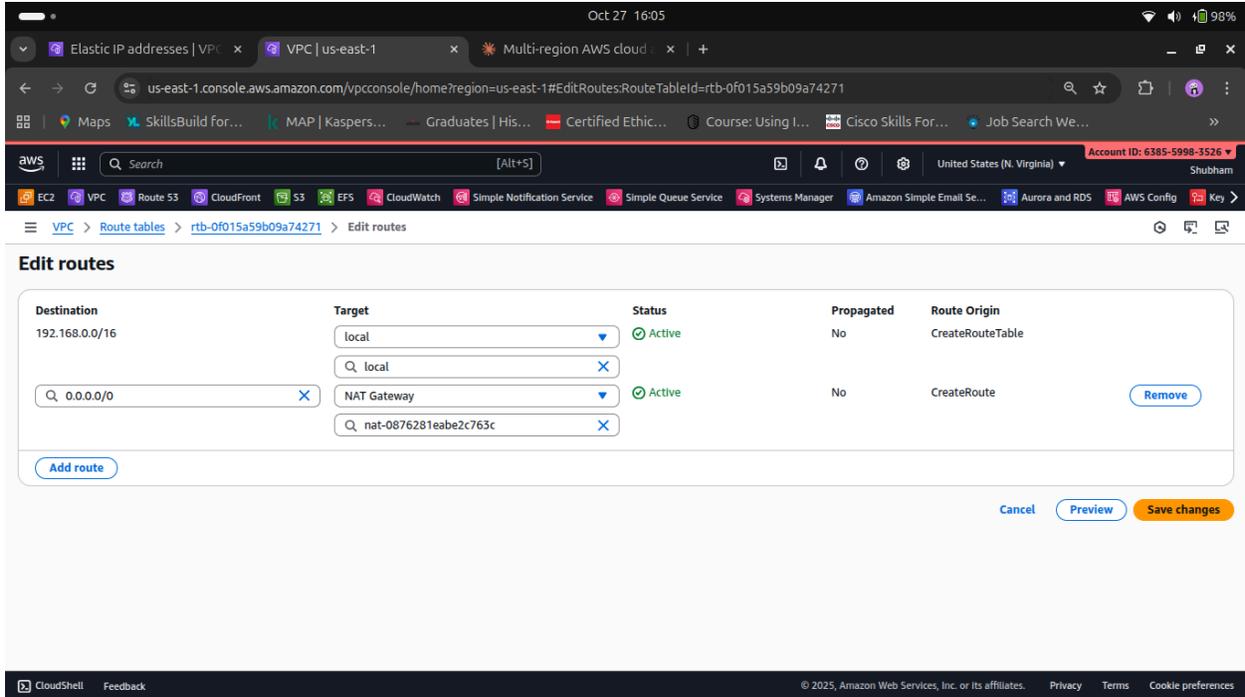
**NAT gateway settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.  
NAT\_GATEWAY

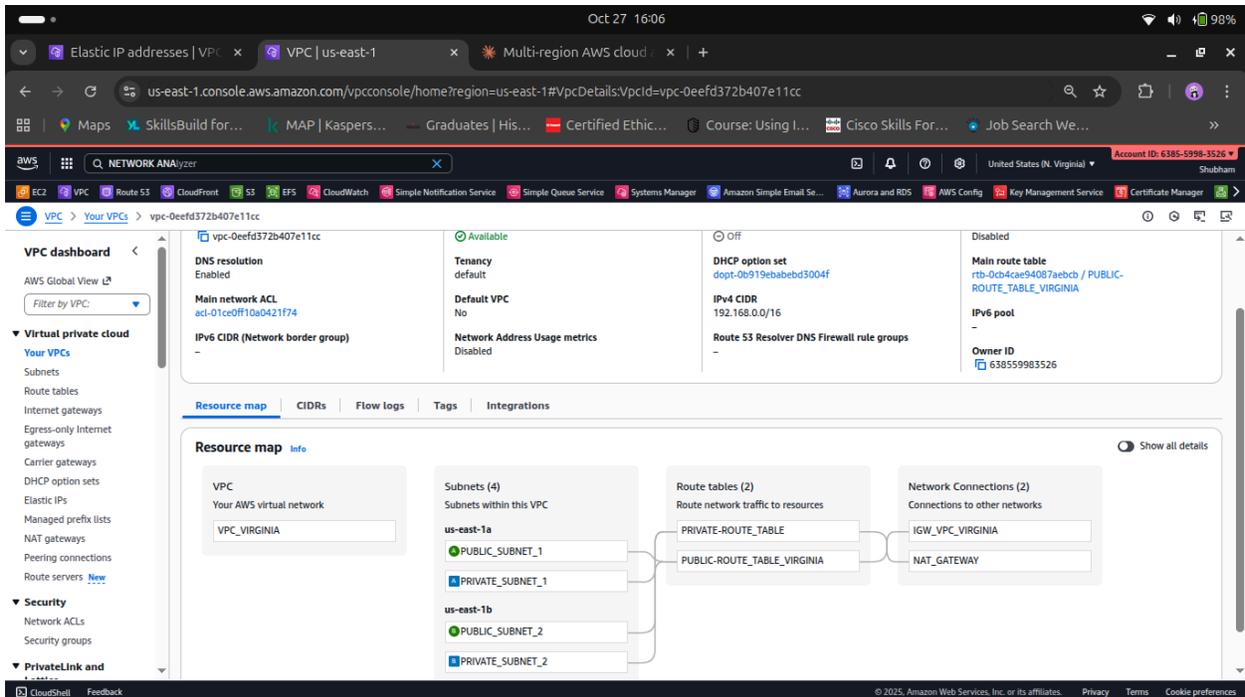
**Subnet**  
Select a subnet in which to create the NAT gateway.  
subnet-0ae9bcf2534e9645a (PUBLIC\_SUBNET\_2)

**Connectivity type**  
Select a connectivity type for the NAT gateway.  
 Public  
 Private

**Elastic IP allocation ID** Info  
Assign an Elastic IP address to the NAT gateway.  
eipalloc-0b2d824de22710635 Allocate Elastic IP



**Step 15) Will create Natgateway to provide internet in private subnet and also add the route in private route table**



**Step 16) We can verify the all resources in vpc are connected and configured correctly**

Oct 27 16:07

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

Search [Alt+S]

Account ID: 6385-5998-3526 Shubham

### Create security group

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

#### Basic details

**Security group name** info

  
Name cannot be edited after creation.

**Description** info

**VPC** info

vpc-02d857bac21e4fc29 (MUMBAL\_VPC)

#### Inbound rules

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22	Anywhere... 0.0.0.0/0	

[Add rule](#) [Delete](#)

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

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

Oct 27 16:08

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

Search [Alt+S]

Account ID: 6385-5998-3526 Shubham

### Create security group

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

#### Basic details

**Security group name** info

  
Name cannot be edited after creation.

**Description** info

**VPC** info

vpc-02d857bac21e4fc29 (MUMBAL\_VPC)

#### Inbound rules

Type	Protocol	Port range	Source	Description - optional
All traffic	All	All	Custom	

[Add rule](#) [Delete](#)

#### Outbound rules

Type	Protocol	Port range	Destination	Description - optional
------	----------	------------	-------------	------------------------

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

Oct 27 16:10

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

Account ID: 6385-5998-3526 Shubham

### Create security group

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

#### Basic details

**Security group name** Info  
EC2\_SG  
Name cannot be edited after creation.

**Description** Info  
allows all traffic to ec2 only through loadbalancer security group

**VPC** Info  
vpc-02d857bac21e4fc29 (MUMBAL\_VPC)

#### Inbound rules

Type	Protocol	Port range	Source	Description - optional
All traffic	All	All	Custom sg-045d2d062ac4c2cdb	

[Add rule](#)

#### Outbound rules

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

Oct 27 16:11

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

Account ID: 6385-5998-3526 Shubham

### Create security group

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

#### Basic details

**Security group name** Info  
Http  
Name cannot be edited after creation.

**Description** Info  
ALLOWS HTTP ON PORT 80

**VPC** Info  
vpc-02d857bac21e4fc29 (MUMBAL\_VPC)

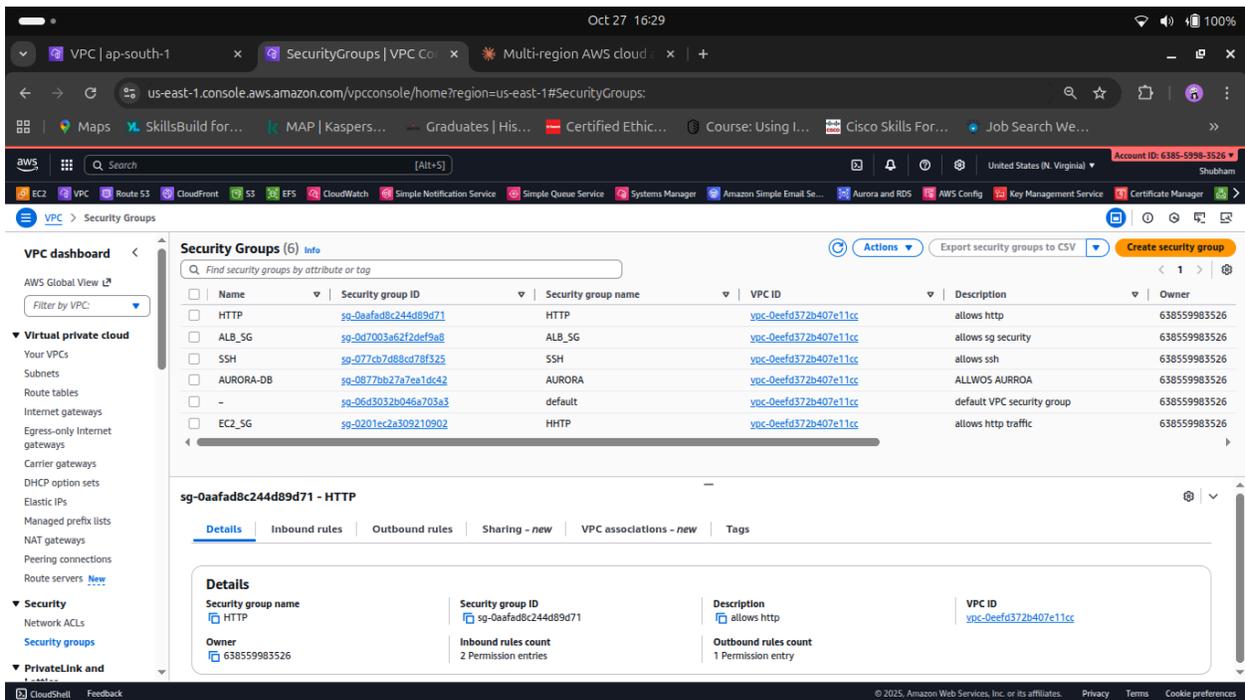
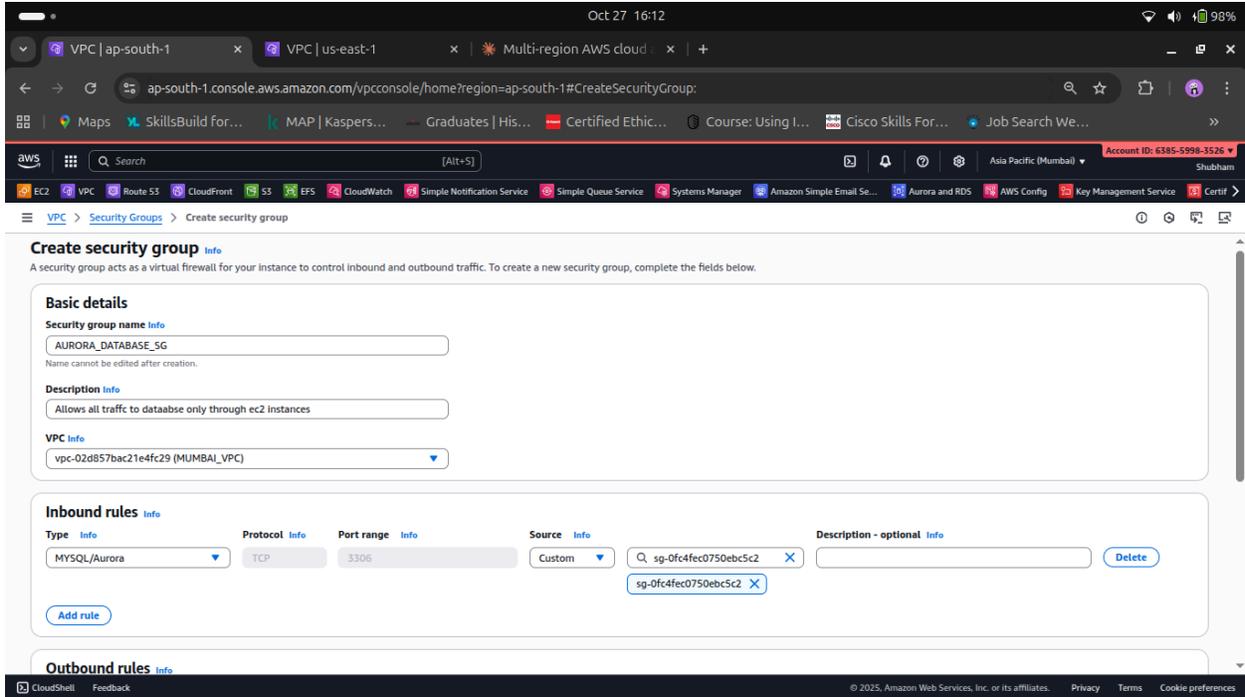
#### Inbound rules

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Anywhere... 0.0.0.0/0	
HTTP	TCP	80	Anywhere... ::/0	

[Add rule](#)

#### Outbound rules

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



**Step 17) We will create security groups**  
**SSH SG only for ssh**  
**HTTP SG for allowing port 80**  
**ALB\_SG allowing all traffic on it**

# EC2\_SG allowing all traffic on ec2 only through ALB\_SG security group AURORA\_DB SG allowing 3306 on database only through EC2\_SG

Oct 27 16:31

Launch an instance | EC2 x SecurityGroups | VPC Co x Multi-region AWS cloud x +

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

Search [Alt+S]

EC2 > Instances > Launch an instance

It seems like you may be new to launching instances in EC2. Take a walkthrough to learn about EC2, how to launch instances and about best practices. [Take a walkthrough](#) [Do not show me this message again.](#)

### Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags**

Name:  [Add additional tags](#)

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

Search:

Recents: [Quick Start](#)

- Amazon Linux
- macOS
- Ubuntu
- Windows
- Red Hat
- SUSE Linux
- Debian

[Browse more AMIs](#)

**Summary**

Number of instances:

Software image (AMI): Canonical, Ubuntu, 24.04, amd64... [read more](#)  
ami-02d26659f82cf299

Virtual server type (instance type): t2.micro

Firewall (security group): 2 security groups

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

**Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when

[Cancel](#) [Launch Instance](#)

Oct 27 16:32

Launch an instance | EC2 x SecurityGroups | VPC Co x Multi-region AWS cloud x +

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

Search [Alt+S]

EC2 > Instances > Launch an instance

2

**Allow tags in metadata**

Enable

**User data - optional**

Upload a file with your user data or enter it in the field.

[Choose file](#)

```
#!/bin/bash
sudo apt update -y
sudo apt install nginx -y
systemctl enable nginx
systemctl start nginx
```

User data has already been base64 encoded

**Summary**

Number of instances:

Software image (AMI): Canonical, Ubuntu, 24.04, amd64... [read more](#)  
ami-02d26659f82cf299

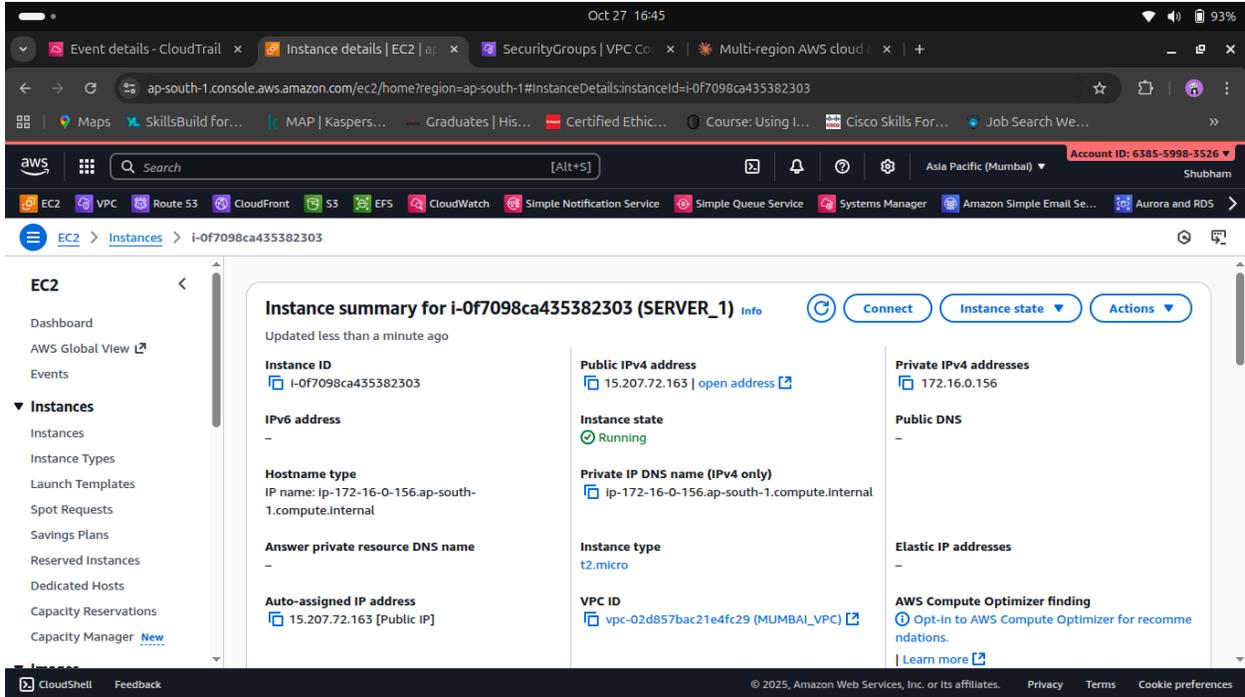
Virtual server type (instance type): t2.micro

Firewall (security group): 2 security groups

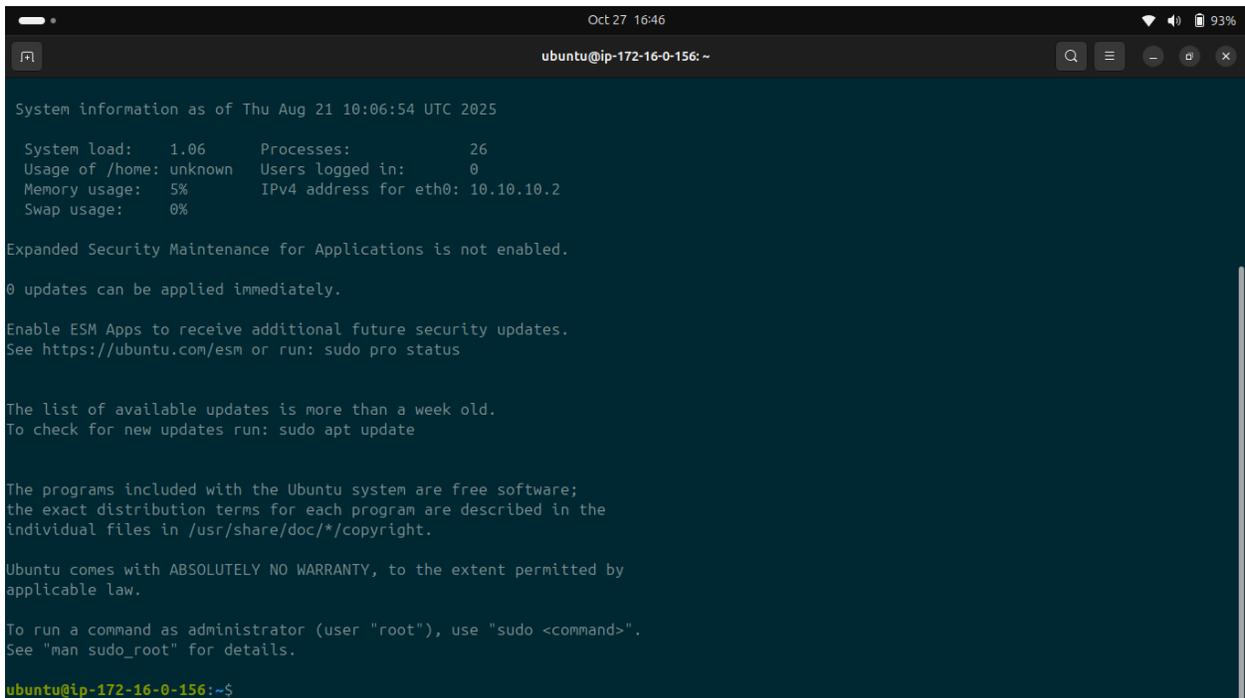
Storage (volumes): 1 volume(s) - 8 GiB

**Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when

[Cancel](#) [Launch Instance](#) [Preview code](#)



## Step 18) We will launch nginx server on ec2 in mumbai region



## Step 19) We will ssh into the server

```
Oct 27 16:51
root@ip-172-16-0-156: ~
root@ip-172-16-0-156:~# systemctl status nginx.service
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-10-27 11:18:23 UTC; 3min 12s ago
     Docs: man:nginx(8)
    Main PID: 1827 (nginx)
      Tasks: 2 (limit: 1121)
   Memory: 1.7M (peak: 3.7M)
      CPU: 18ms
   CGroup: /system.slice/nginx.service
           └─1827 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─1830 "nginx: worker process"

Oct 27 11:18:23 ip-172-16-0-156 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Oct 27 11:18:23 ip-172-16-0-156 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
root@ip-172-16-0-156:~#
```

## Step 20) We will check nginx server status

```
Oct 27 16:54
root@ip-172-16-0-156: /home/ubuntu/survey-app
root@ip-172-16-0-156:/home/ubuntu# ls
root@ip-172-16-0-156:/home/ubuntu# mkdir survey-app
cd survey-app
root@ip-172-16-0-156:/home/ubuntu/survey-app# # Backend files
touch server.js
touch .env
touch package.json

# Frontend folder
mkdir public
cd public
touch index.html styles.css script.js
cd ..
root@ip-172-16-0-156:/home/ubuntu/survey-app#
```

## Step 21) we will create directory for survey-app

```
Oct 27 16:55
root@ip-172-16-0-156: ~
root@ip-172-16-0-156:/home/ubuntu/survey-app# cd
root@ip-172-16-0-156:~# sudo apt update && sudo apt upgrade -y
sudo apt install nodejs npm -y
node -v
npm -v
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... 0%
```

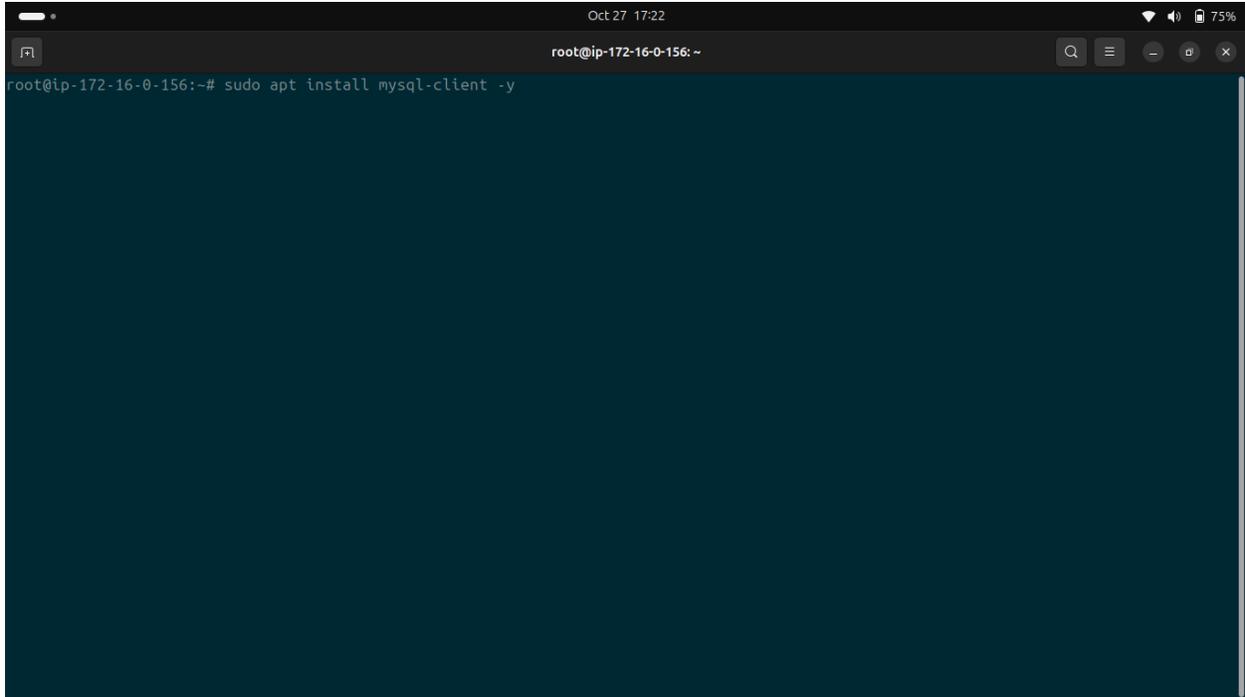
**Step 22) We will install node js package manager**

```
Oct 27 17:11
root@ip-172-16-0-156:/home/ubuntu/survey-app
root@ip-172-16-0-156:/home/ubuntu/survey-app# pm2 start server.js --name survey-app -f
[PM2] Starting /home/ubuntu/survey-app/server.js in fork_mode (1 instance)
[PM2] Done.

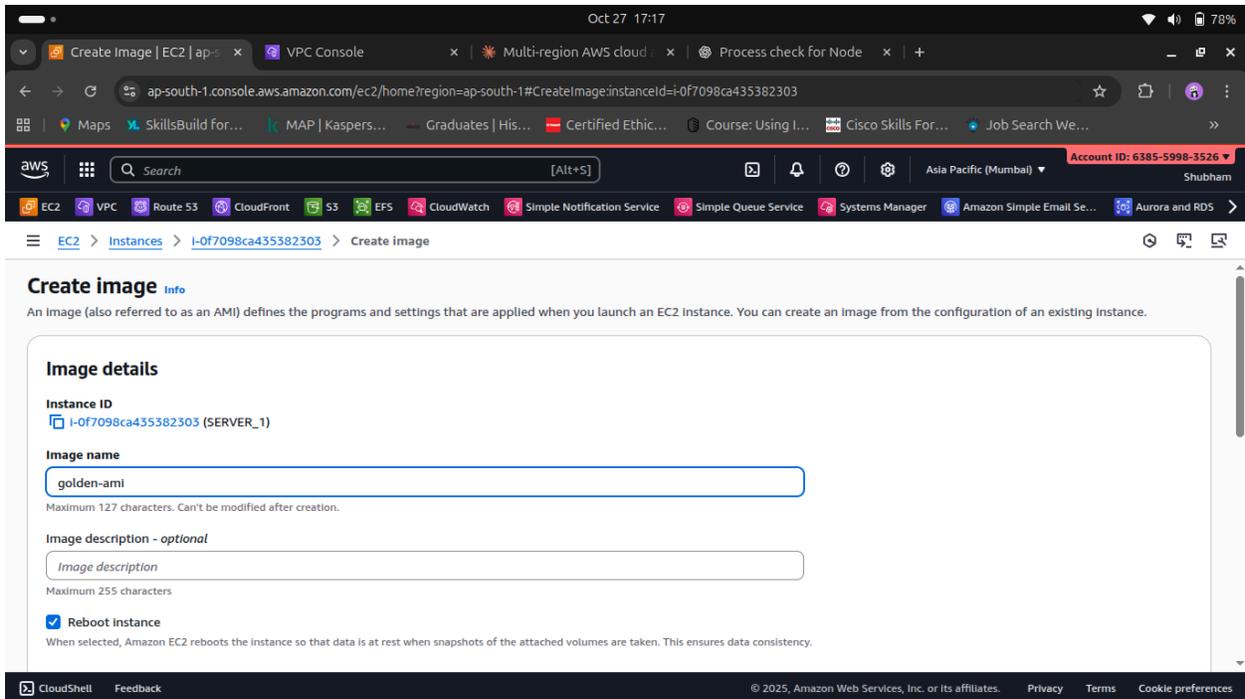
  id   name         namespace   version  mode    pid     uptime  ⌵   status  cpu    mem    user    watchi
ng
  --  -
  0    survey-app    default     1.0.0    fork    22500   40s    0    online  0%    67.3mb root
  1    survey-app    default     1.0.0    fork    22536   27s    0    online  0%    65.2mb root
  2    survey-app    default     1.0.0    fork    22562   0s     0    online  0%    9.8mb  root

root@ip-172-16-0-156:/home/ubuntu/survey-app#
```

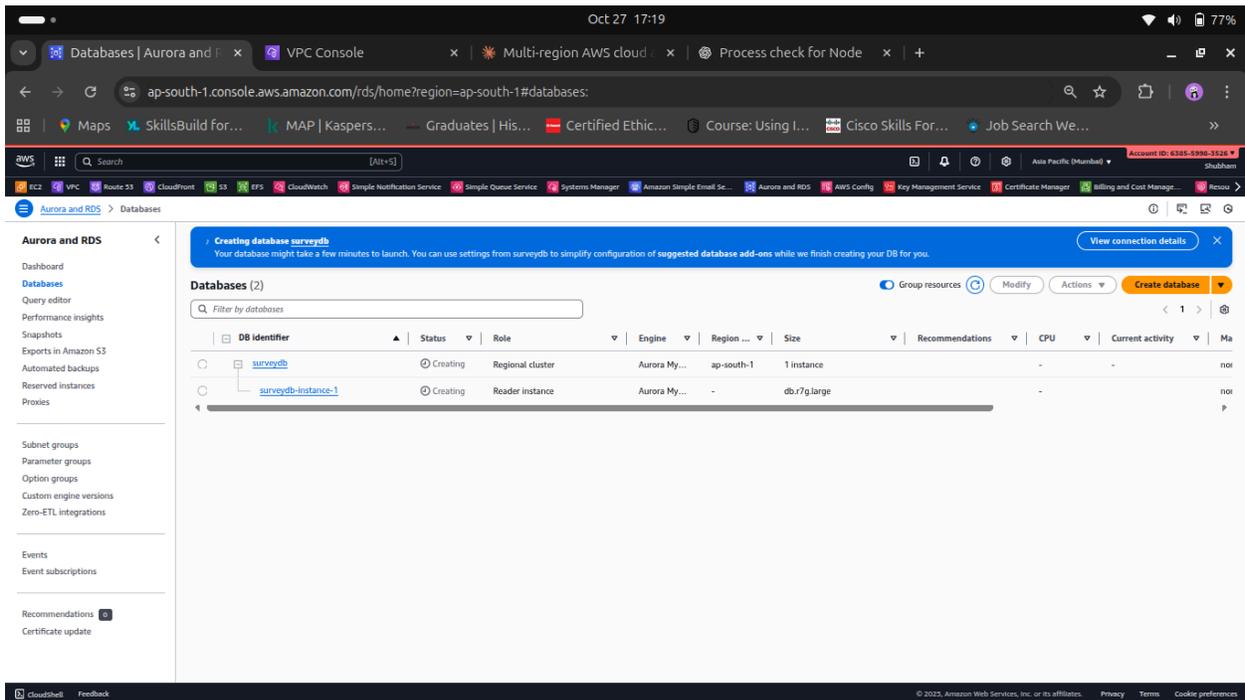
**Step 23) we will start our app now**



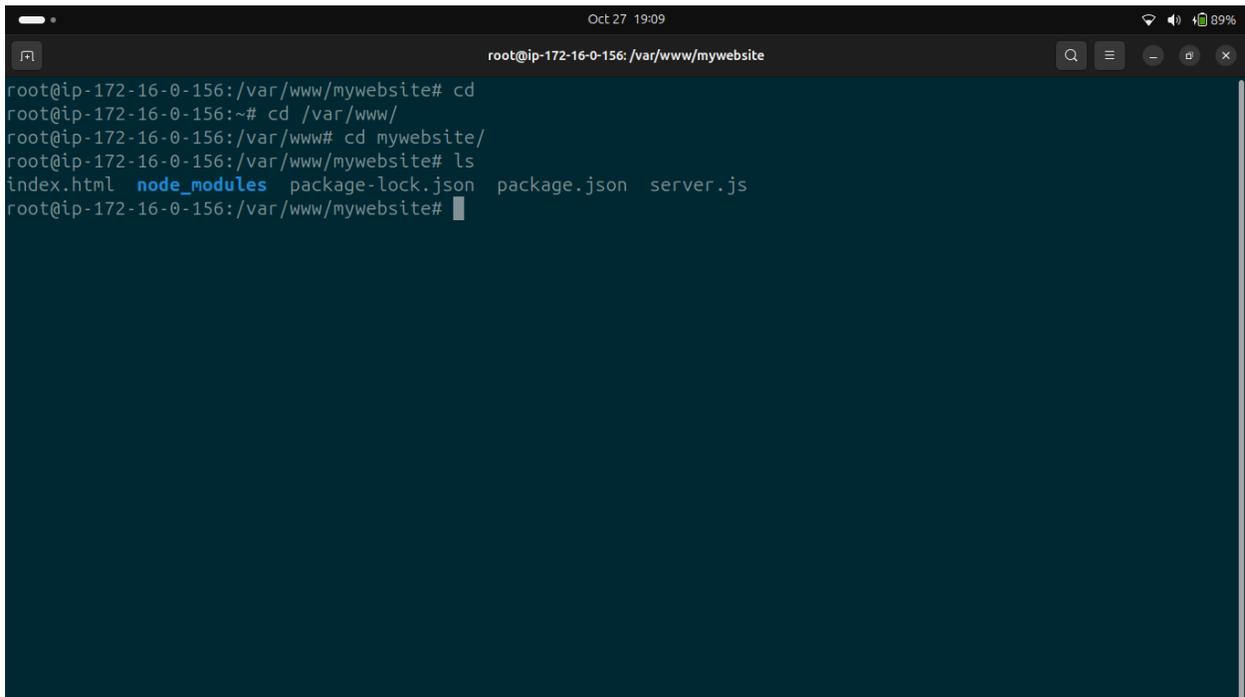
**Step 24) We will install mysql-client on ec2**

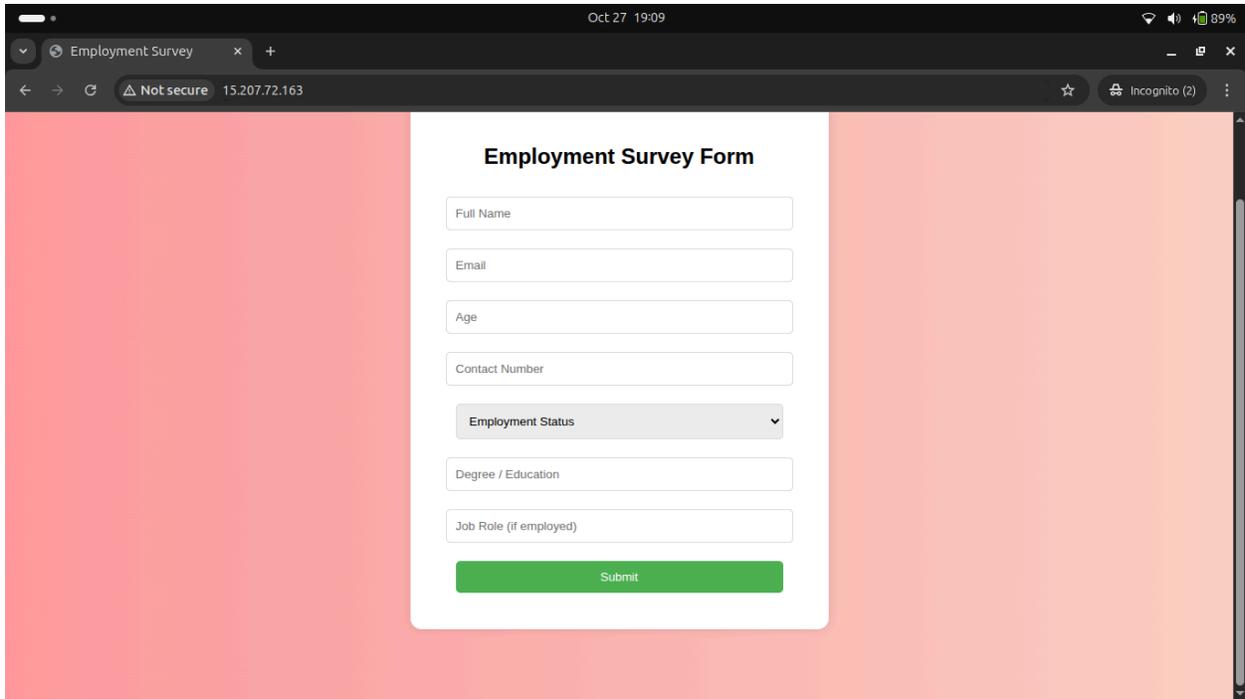


**Step 25) After all installation we will create golden image of our server**

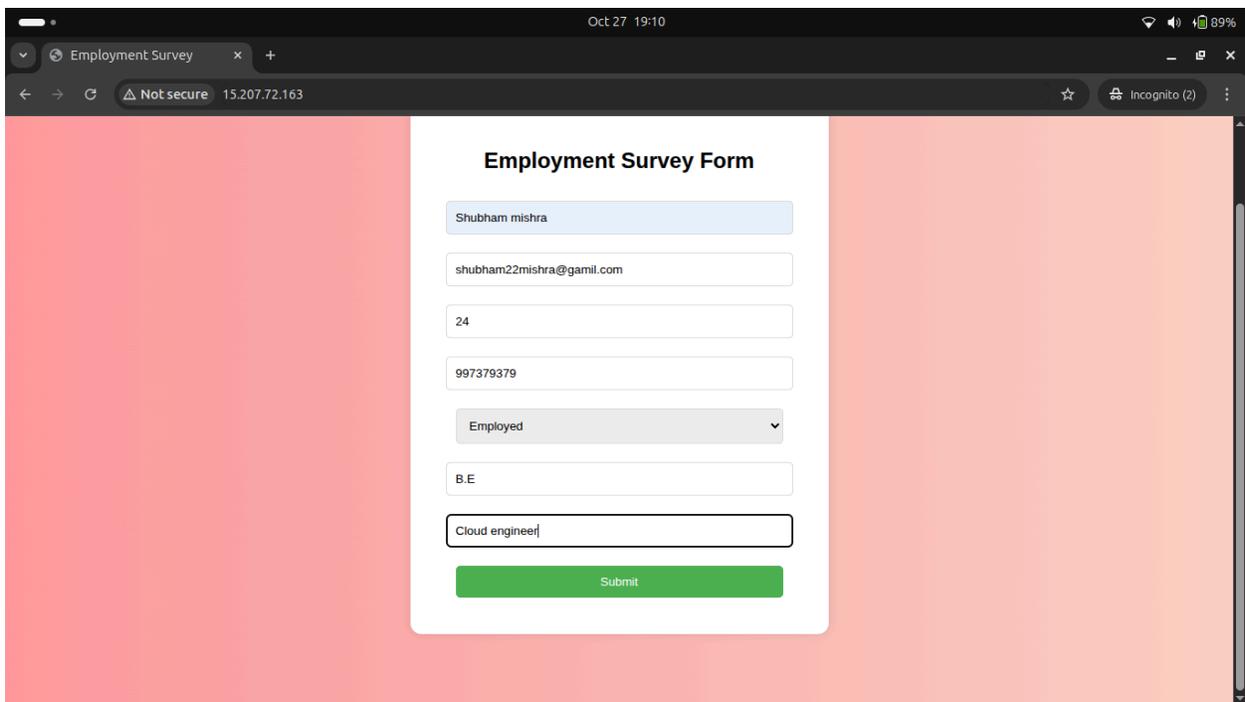


**Step 26) We will create Aurora mysql database in mumbai region**





**Step 27) We can access our website on public ip of ec2 instance**



**Step 28) We will fill data on our employment survey form**

```

Oct 27 19:10
root@ip-172-16-0-156: /var/www/mywebsite

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

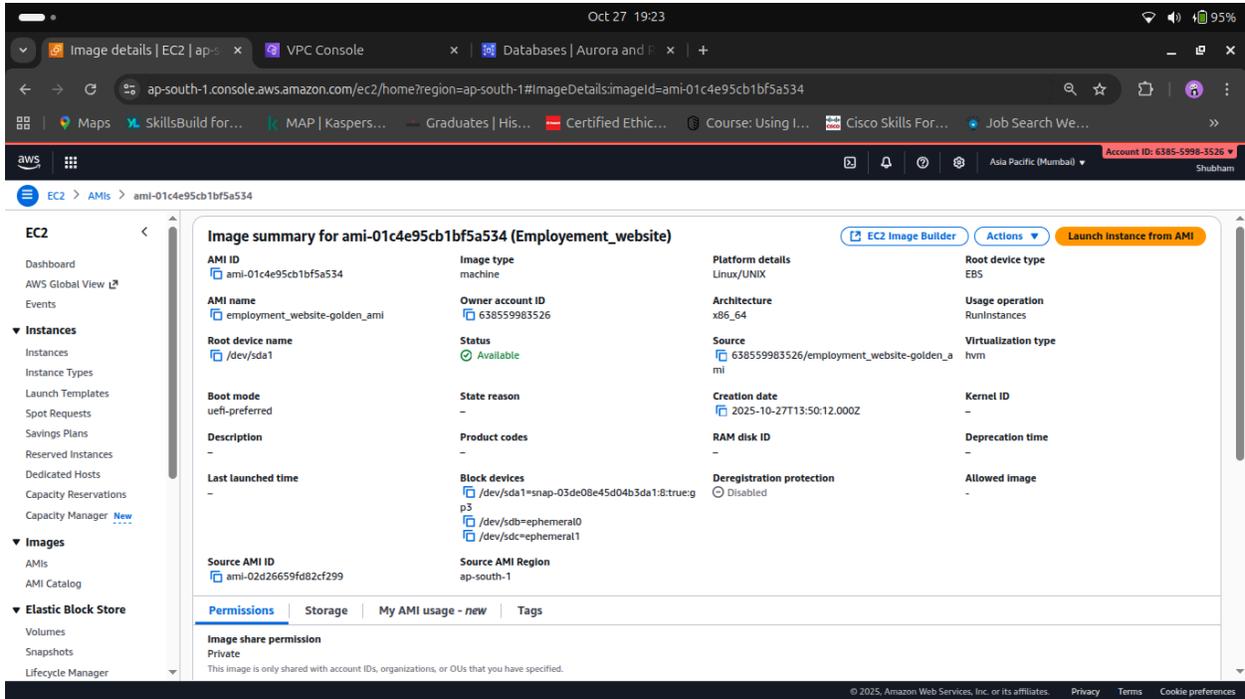
Database changed
mysql> show tables;
+-----+
| Tables_in_surveydb |
+-----+
| survey              |
+-----+
1 row in set (0.00 sec)

mysql> slect * from survey;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'slect * from survey' at line 1
mysql> select * from survey;
+-----+-----+-----+-----+-----+-----+-----+-----+
| id | name          | email                               | age | contact   | status  | degree | jobrole   |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  | Shubham mishra | shubhammishra4227@gmail.com        | 22  | 9619676859 | Employed | B.E    | Cloud engineer |
| 2  | Shubham mishra | shubhammishra4227@gmail.com        | 22  | 9619676859 | Employed | B.E    | Cloud engineer |
| 3  | Shubham mishra | shubham22mishra@gamil.com          | 24  | 997379379  | Employed | B.E    | Cloud engineer |
+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

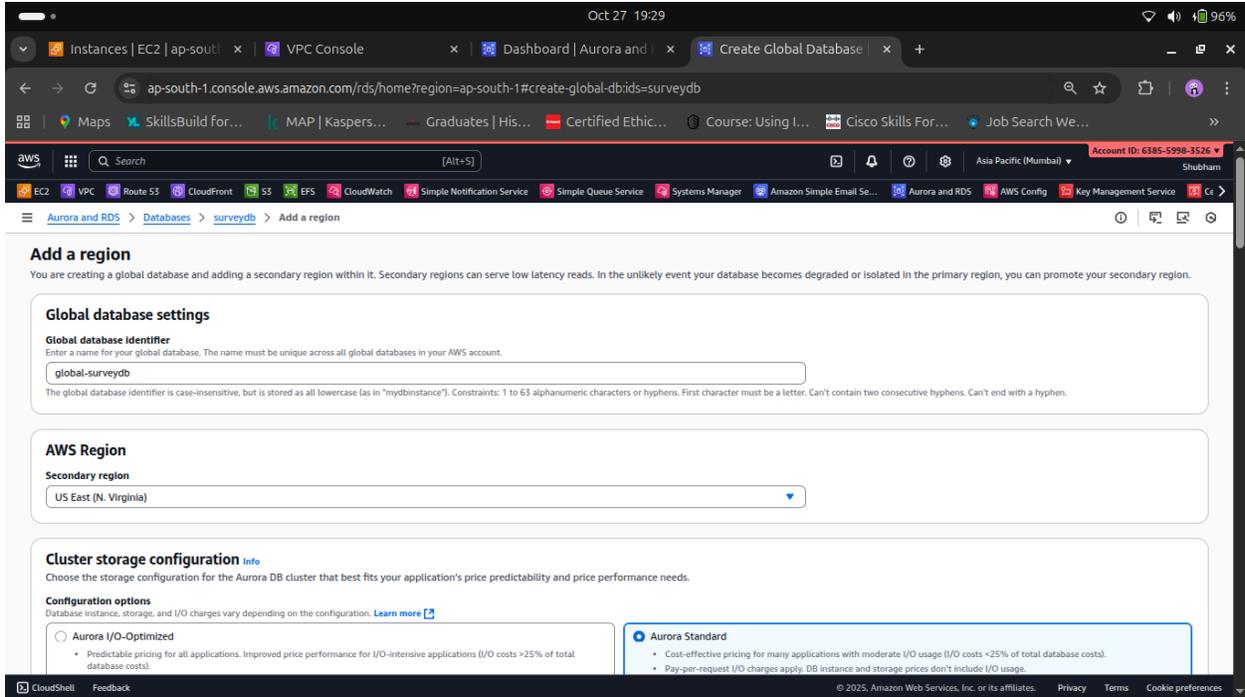
mysql>

```

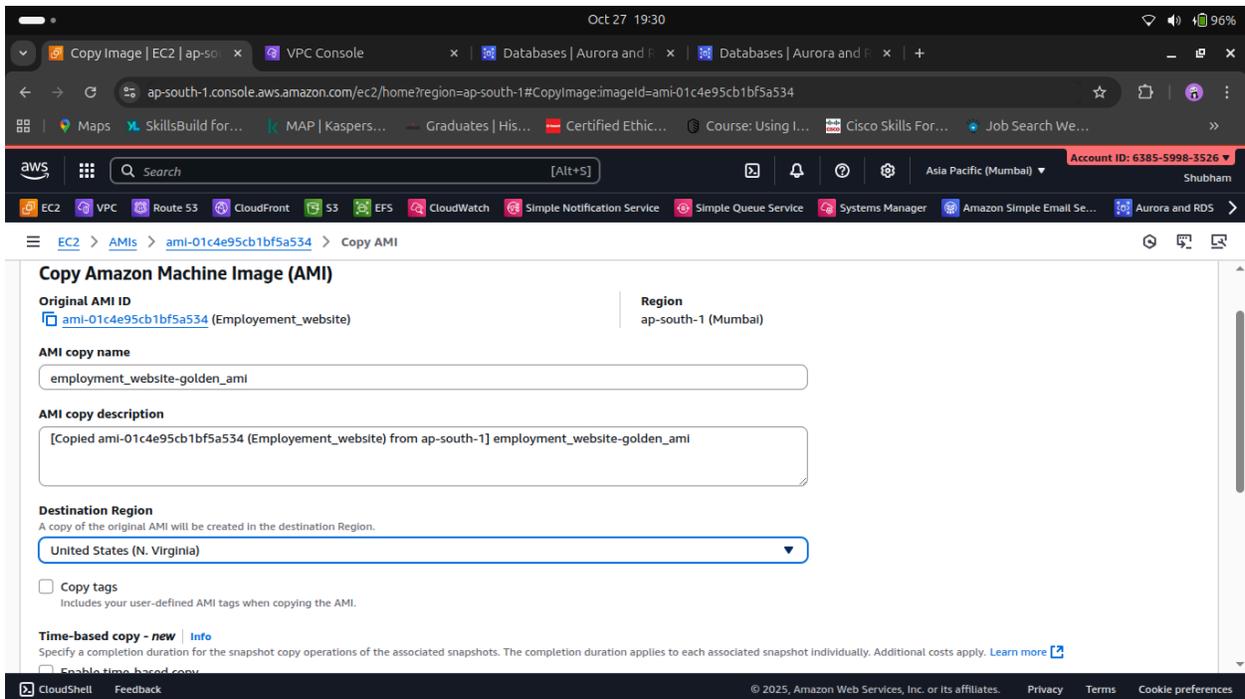
**Step 29) We can see that data from website is stored in our global aurora database**



**Step 30) Till then our ami was created successfully in mumbai region**



**Step 31) We will create aurora replica in north virgina region now**



**Step 32) We will now copy our golden ami in to north virginia region**

Oct 27 19:31

Step 1 Create target group

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

Search [Alt+S]

Account ID: 6385-5998-3526 Shubham

EC2 > Target groups > Create target group

### Step 1 Create target group

Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

#### Basic configuration

Settings in this section can't be changed after the target group is created.

#### Choose a target type

**Instances**

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

**IP addresses**

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

**Lambda function**

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

**Application Load Balancer**

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

Oct 27 19:32

Create application load

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

Search [Alt+S]

Account ID: 6385-5998-3526 Shubham

EC2 > Load balancers > Create Application Load Balancer

### Create Application Load Balancer Info

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

#### How Application Load Balancers work

#### Basic configuration

**Load balancer name**

Name must be unique within your AWS account and can't be changed after the load balancer is created.

ALB

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

**Scheme** Info

Scheme can't be changed after the load balancer is created.

**Internet-facing**

- Serves internet-facing traffic.
- Has public IP addresses.
- DNS name resolves to public IPs.
- Requires a public subnet.

**Internal**

- Serves internal traffic.
- Has private IP addresses.
- DNS name resolves to private IPs.
- Compatible with the IPv4 and Dualstack IP address types.

**Load balancer IP address type** Info

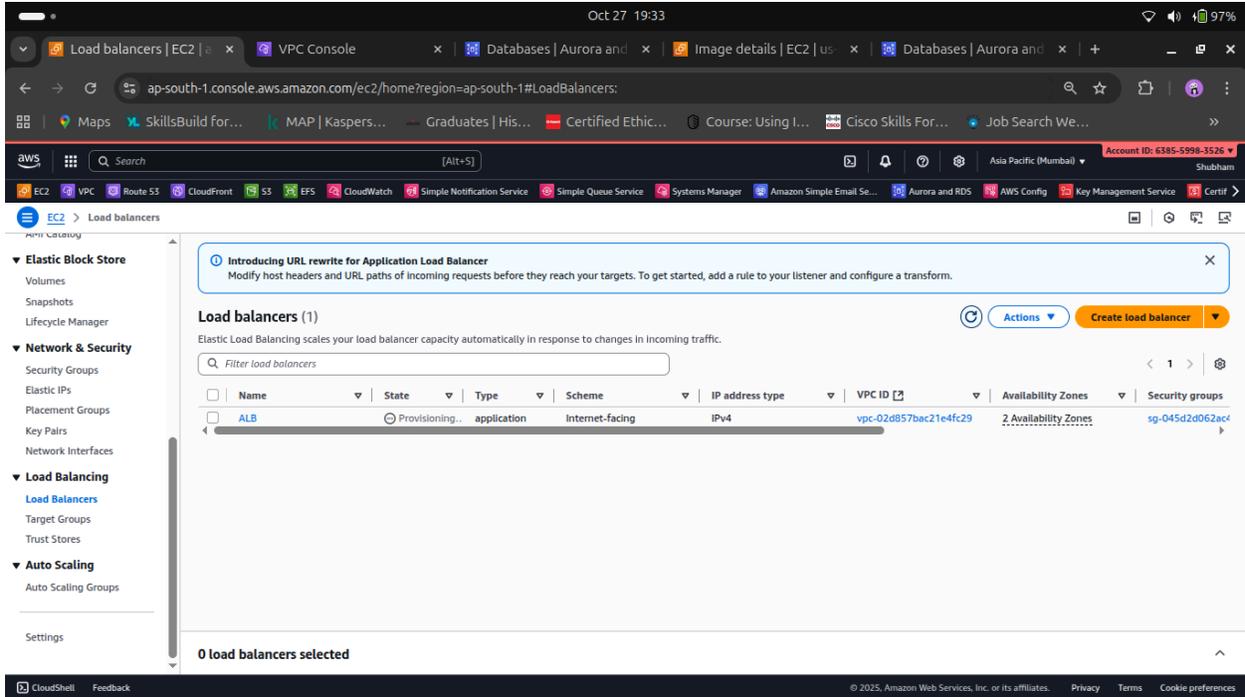
Select the front-end IP address type to assign to the load balancer. The VPC and subnets mapped to this load balancer must include the selected IP address types. Public IPv4 addresses have an additional cost.

**IPv4**

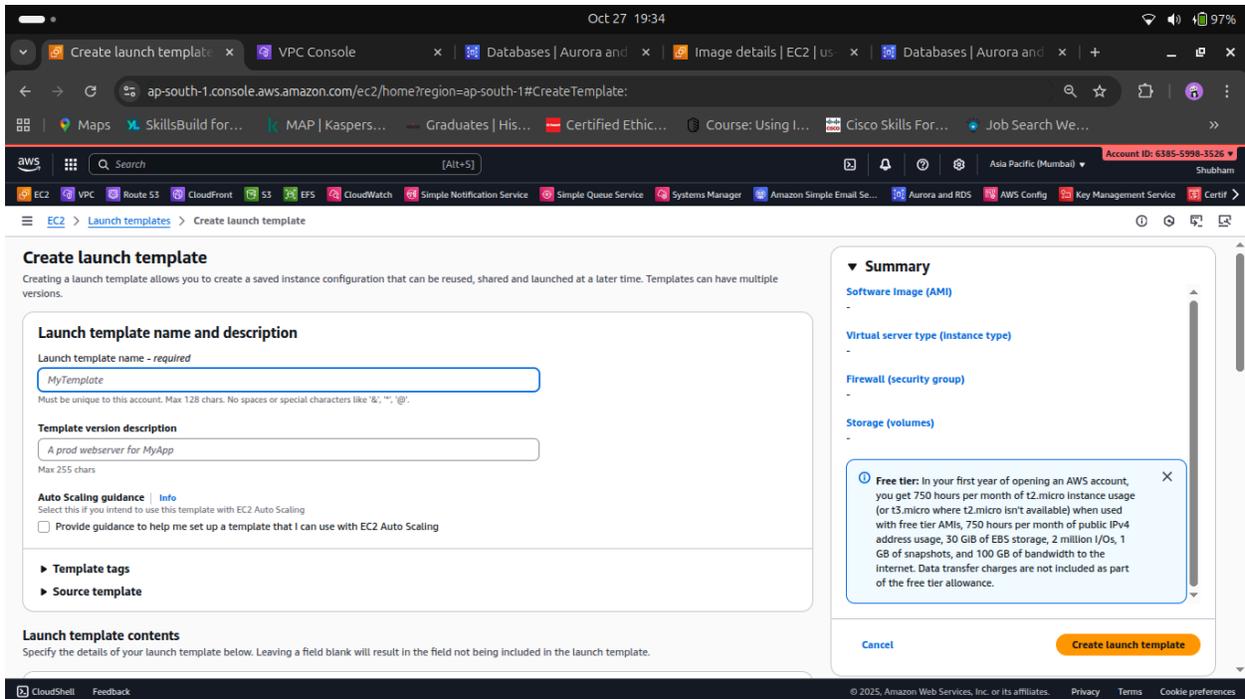
- Includes only IPv4 addresses.

**Dualstack**

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



**Step 33) We will create Target group, Application Load balancer in mumbai region**



**Step 34) We will now create launch template for ASG**

Oct 27 19:36

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

EC2 > Auto Scaling groups > Create Auto Scaling group

### Step 1 Choose launch template

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

**Name**

**Auto Scaling group name**  
Enter a name to identify the group.  
ASG-1  
Must be unique to this account in the current Region and no more than 255 characters.

**Launch template**

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

**Launch template**  
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.  
ASG-1

**Version**  
Default (1)

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

Oct 27 19:36

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

EC2 > Auto Scaling groups > Create Auto Scaling group

### Review

**Network**

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

**VPC**  
Choose the VPC that defines the virtual network for your Auto Scaling group.  
vpc-02d857bac21e4fc29 (MUMBAL\_VPC)  
172.16.0.0/16

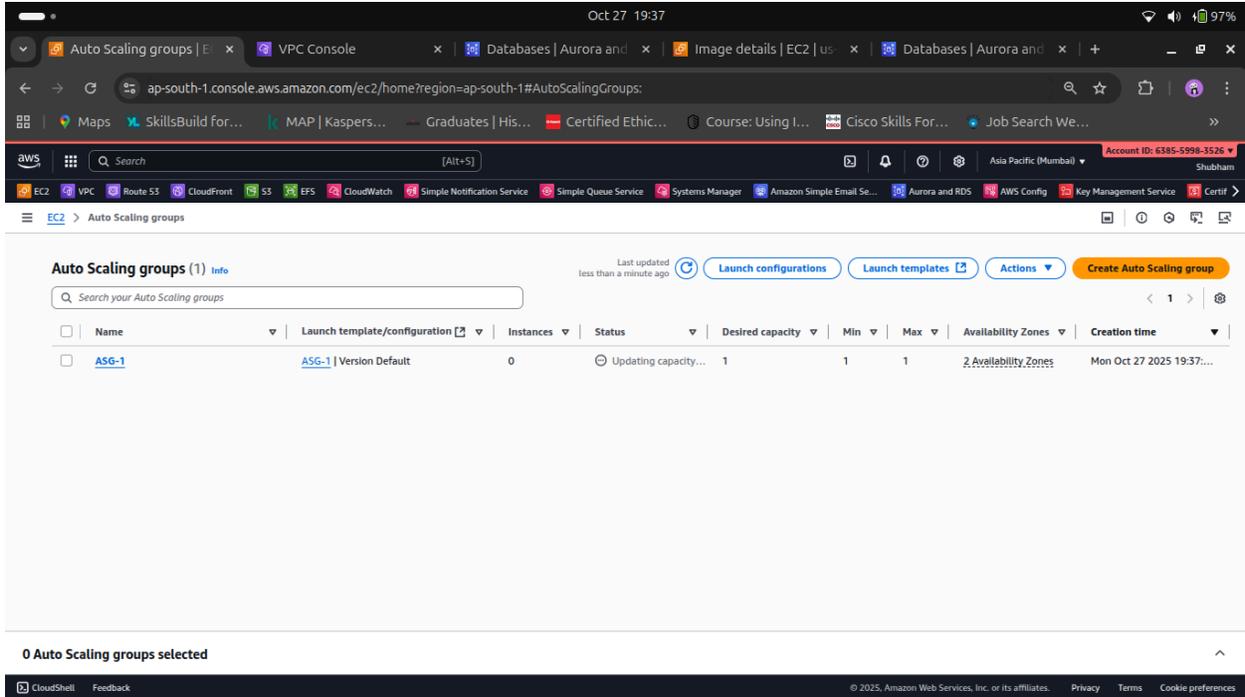
**Availability Zones and subnets**  
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.  
Select Availability Zones and subnets

- aps1-az1 (ap-south-1a) | subnet-04ce54af42206ce29 (PRIVATE\_SUBNET\_1) | 172.16.2.0/24
- aps1-az2 (ap-south-1b) | subnet-00343a68c3c7d87ec (PRIVATE\_SUBNET\_2) | 172.16.3.0/24

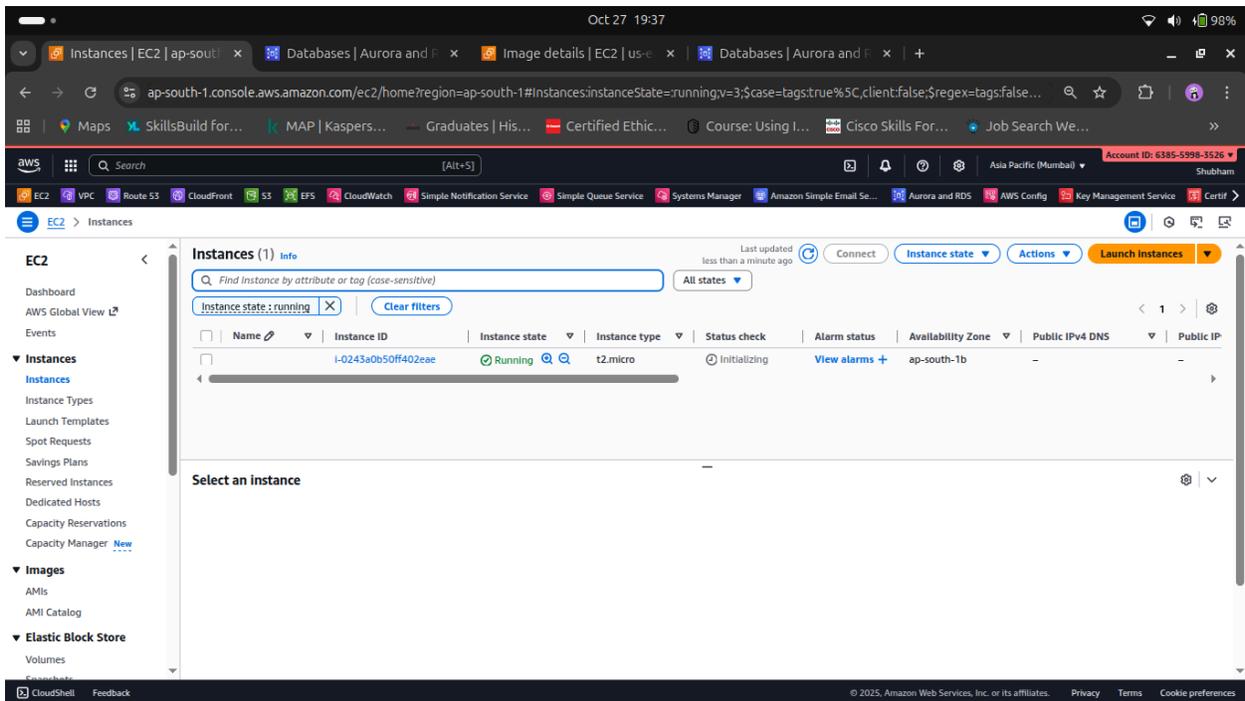
**Availability Zone distribution - new**  
Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.

- Balanced best effort**  
If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.
- Balanced only**  
If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

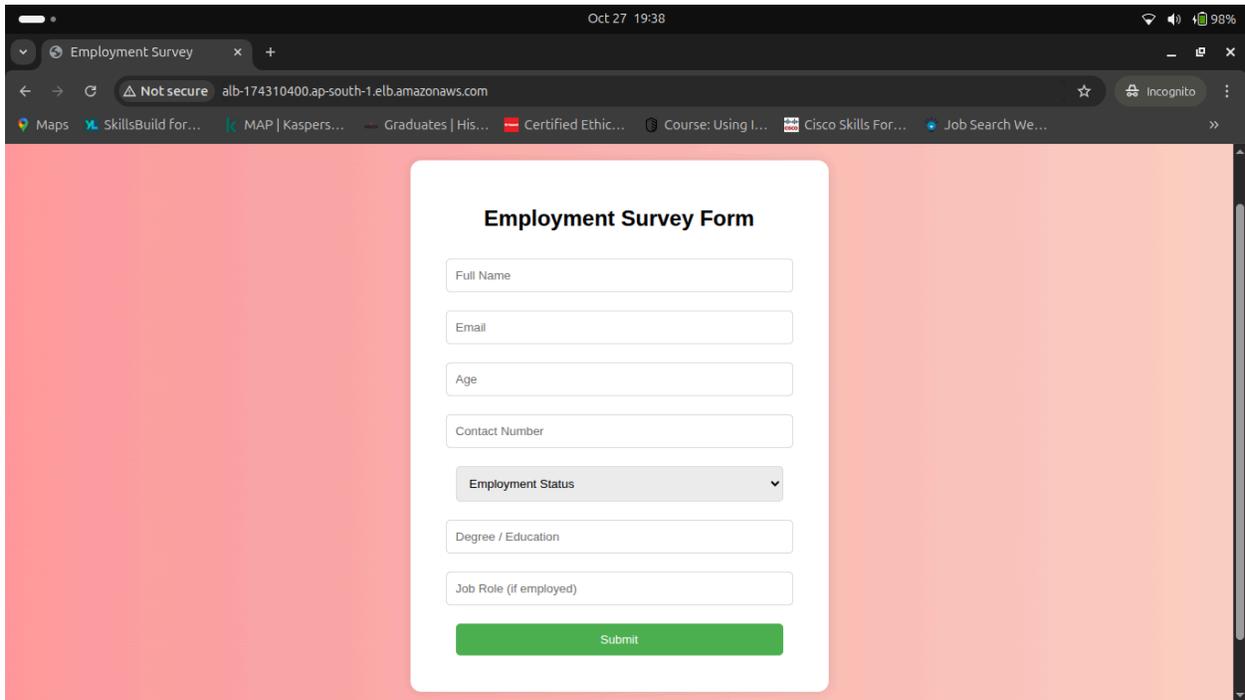
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



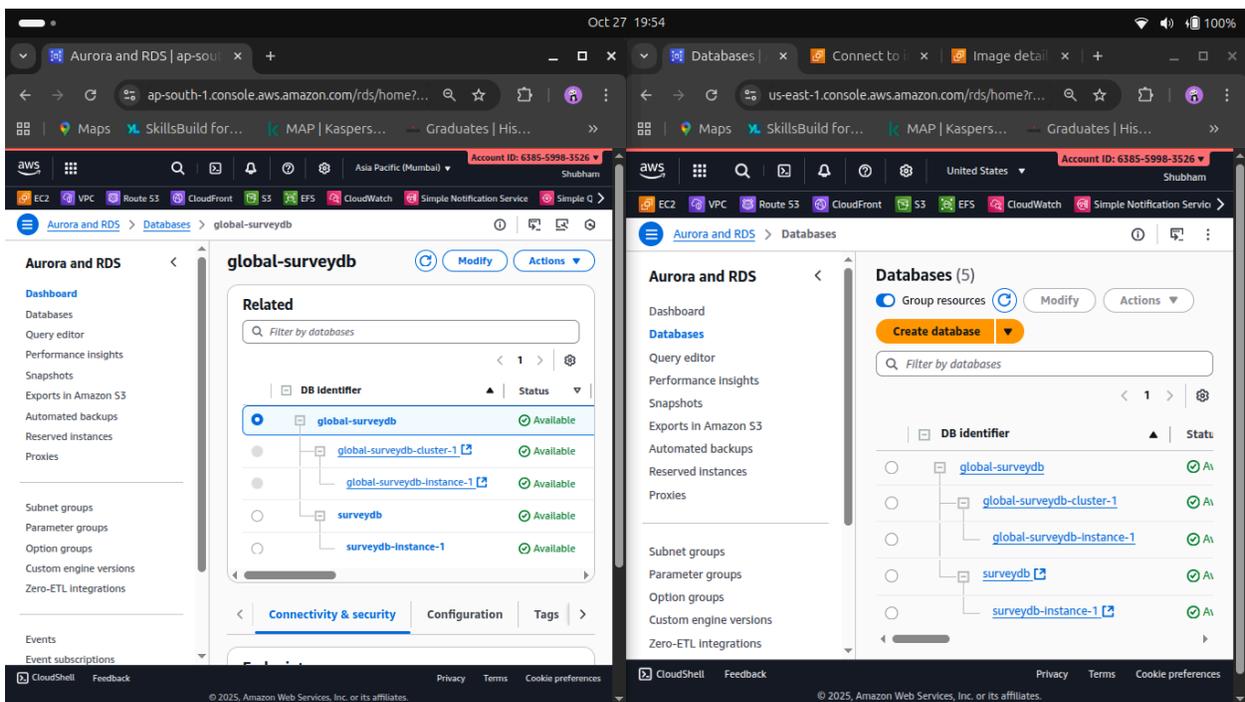
**Step 35 ) We have created now Auto Scaling Group above successfully**



**Step 36) Now we can see 1 instance was launched by ASG using our launch template where golden ami was used.**



**Step 37) We can now verify that our load balancer is working and we are able to access our employment survey website**



## Step 38) We can see global aurora database is replicated successfully in north virginia region

The screenshot displays the AWS Management Console interface for a Target Group named 'TG-VIRGINIA'. A green notification banner at the top states: 'Successfully created the target group: TG-VIRGINIA. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the Targets tab.' The main content area shows the 'Details' tab for the target group, with the following information:

- Target type:** Instance
- Protocol : Port:** HTTP: 80
- Protocol version:** HTTP1
- VPC:** [vpc-0ee372b407e11cc](#)
- IP address type:** IPv4
- Load balancer:** [None associated](#)

Below the details, a summary row shows the following status:

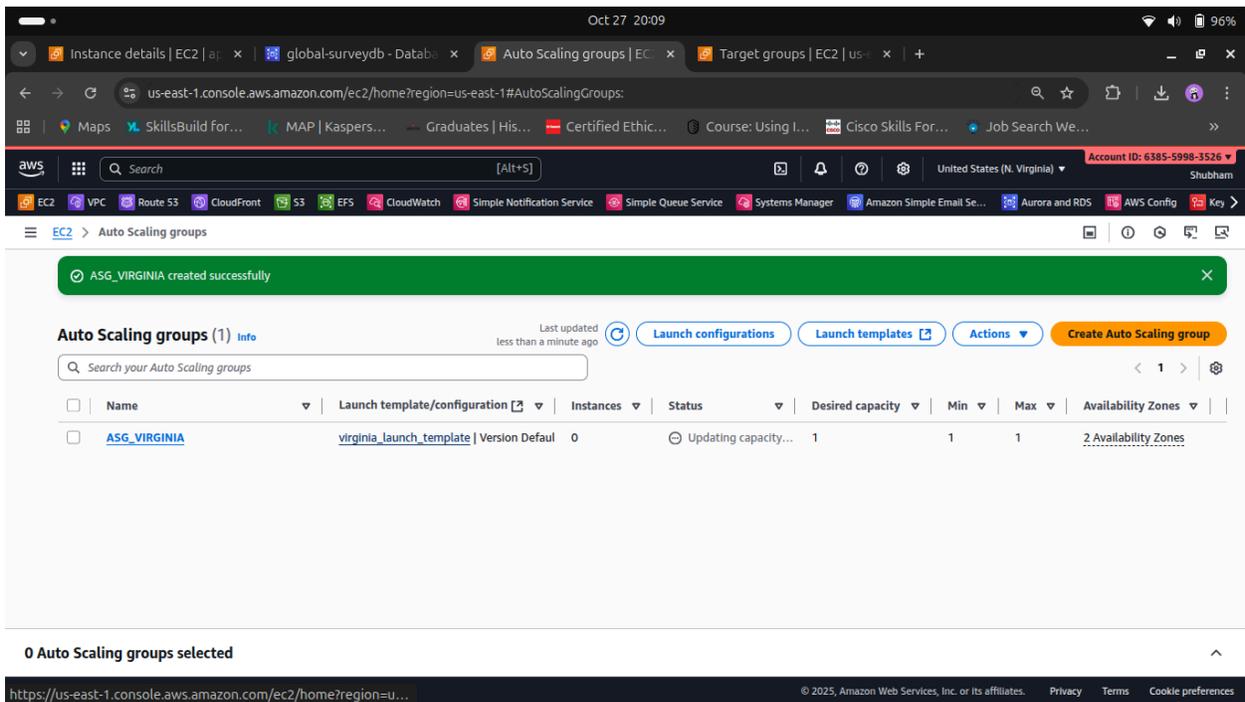
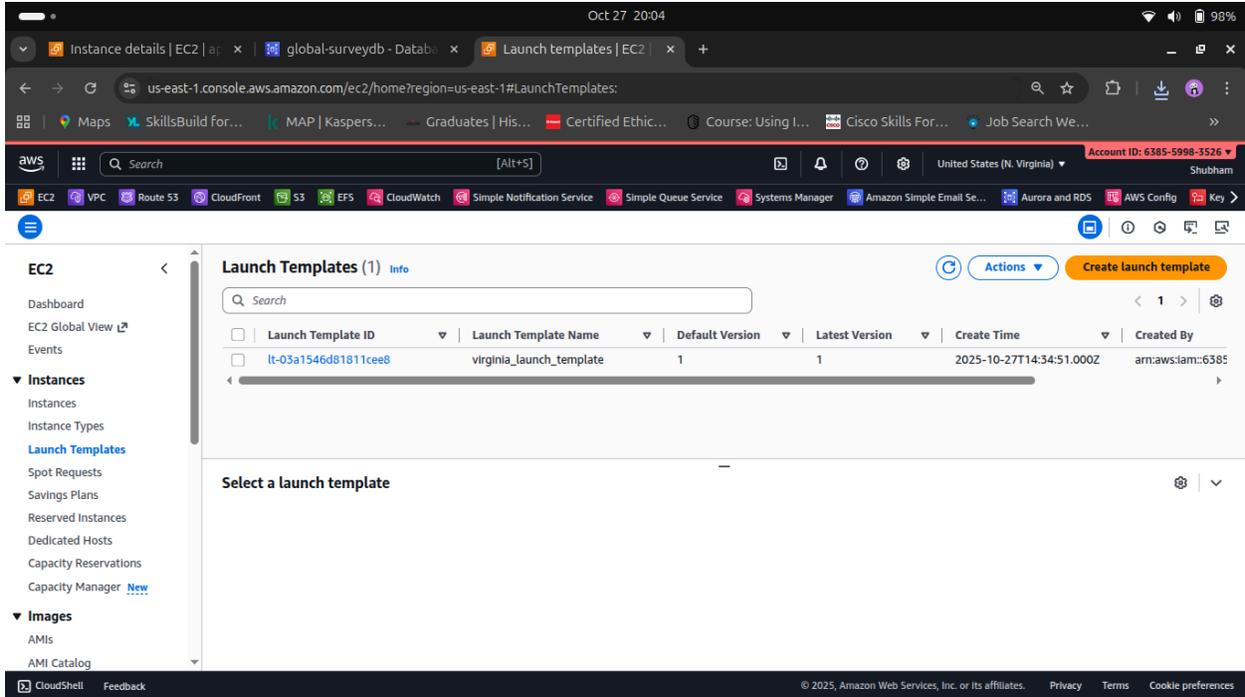
- Total targets:** 0
- Healthy:** 0
- Unhealthy:** 0
- Unused:** 0
- Initial:** 0
- Draining:** 0

There is also a note: '0 Anomalous'. The navigation tabs at the bottom include 'Targets', 'Monitoring', 'Health checks', 'Attributes', and 'Tags'.

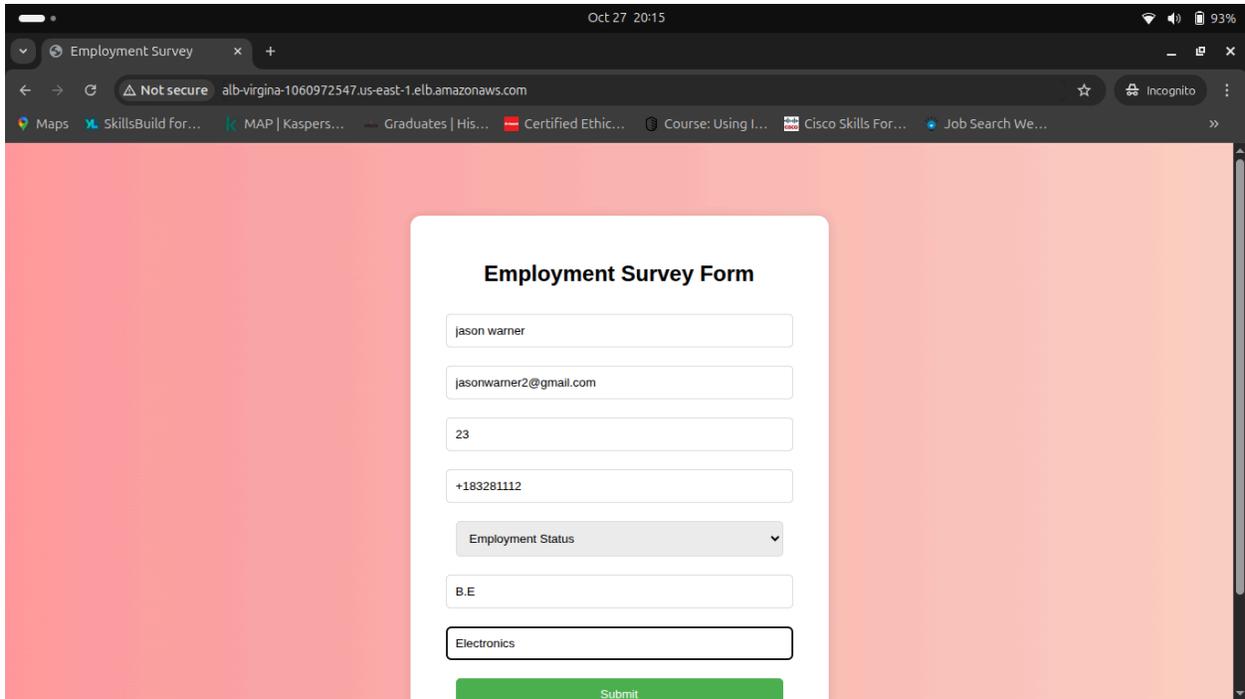
The screenshot displays the AWS Management Console interface for an Application Load Balancer named 'alb-virgina'. A blue notification banner at the top states: 'Introducing URL rewrite for Application Load Balancer. Modify host headers and URL paths of incoming requests before they reach your targets. To get started, add a rule to your listener and configure a transform.' The main content area shows the 'Details' tab for the load balancer, with the following information:

- Load balancer type:** Application
- Status:** Provisioning
- VPC:** [vpc-0ee372b407e11cc](#)
- Load balancer IP address type:** IPv4
- Scheme:** Internet-facing
- Hosted zone:** Z35XDTRQ7X7K
- Availability Zones:** [subnet-00f23503885d5be3](#) us-east-1a (use1-az1), [subnet-0824b0379be18d4ca](#) us-east-1b (use1-az2)
- Date created:** October 27, 2025, 20:03 (UTC+05:30)
- Load balancer ARN:** [arn:aws:elasticloadbalancing:us-east-1:638559983526:loadbalancer/app/alb-virgina/2fd8b9cdea2fd87d](#)
- DNS name info:** [alb-virgina-1060972547.us-east-1.elb.amazonaws.com](#) (A Record)

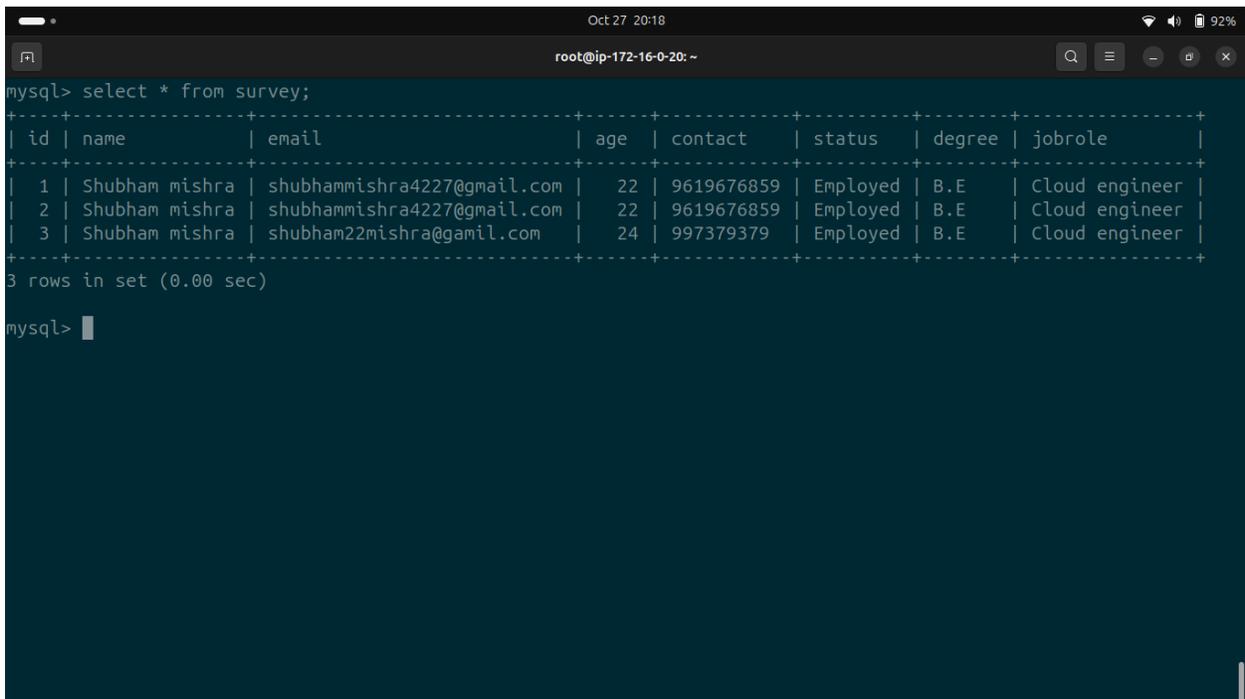
The navigation tabs at the bottom include 'Listeners and rules', 'Network mapping', 'Resource map', 'Security', 'Monitoring', 'Interactions', 'Attributes', 'Capacity', and 'Tags'.

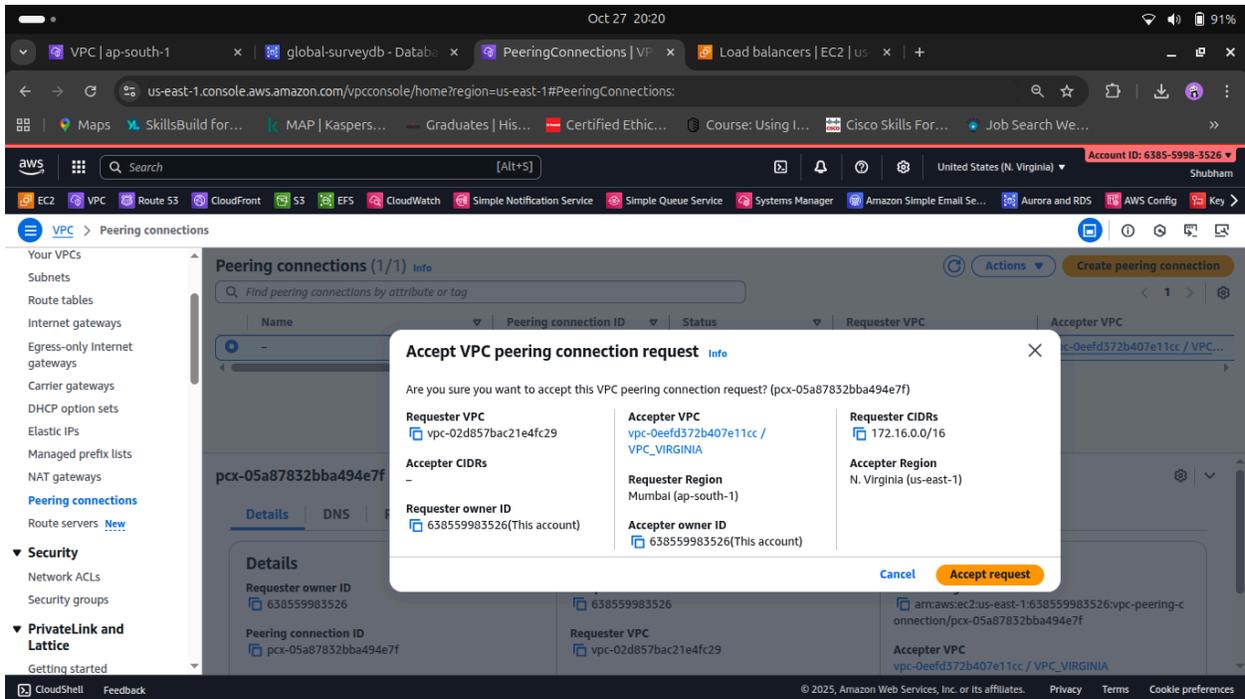
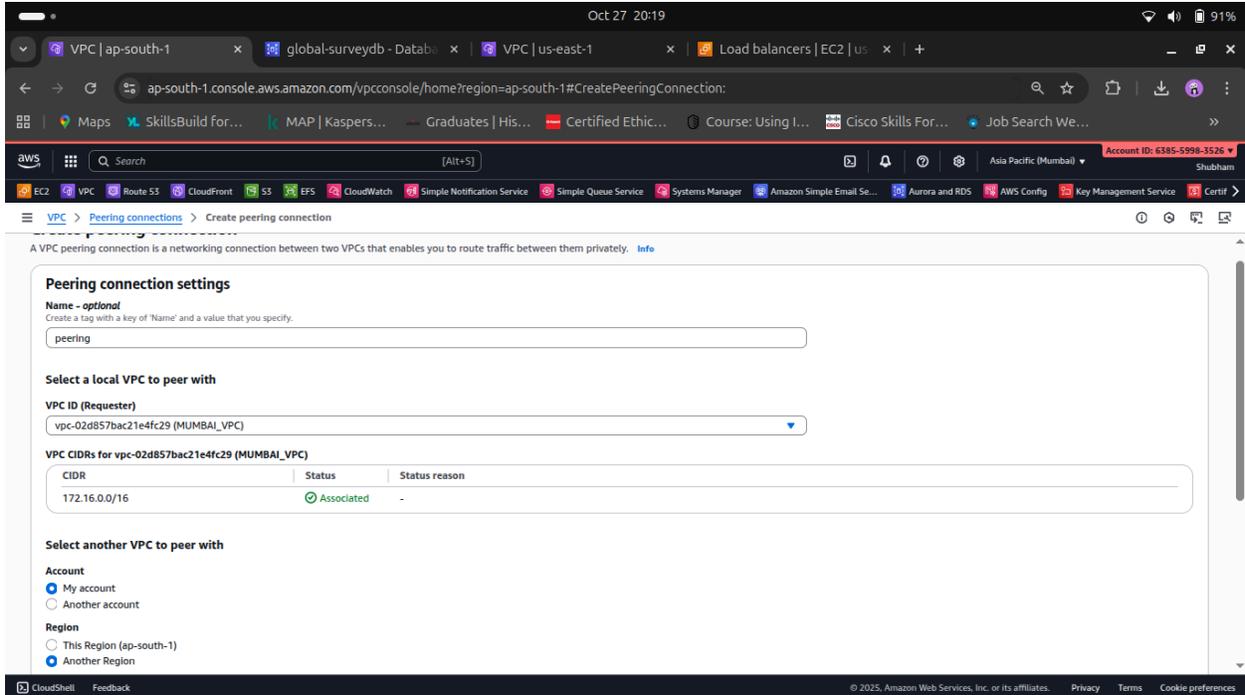


**Step 39) Similarly we have created Target group ALB and ASG in north virginia region as well**



**Step 40) We can verify that same ami is used by north virginia ALB and setup is successful in north virginia region as well**





**Step 41) We will establish peering connection between Mumbai and Virginia vpc as database and ec2 are in private subnet to enable communication**

Oct 27 20:20

ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#EditRoutes:RouteTableId=rtb-0af5e5f73f54a9ec

Search [Alt+S]

Asia Pacific (Mumbai) Account ID: 6385-5998-3526 Shubham

VPC > Route tables > rtb-0af5e5f73f54a9ec > Edit routes

### Edit routes

Destination	Target	Status	Propagated	Route Origin
172.16.0.0/16	local	Active	No	CreateRouteTable
0.0.0.0/0	Internet Gateway lgw-07264803619e81bc2	Active	No	CreateRoute
192.168.0.0/16	Peering Connection pcx-05a87832bba494e7f	-	No	CreateRoute

Buttons: Add route, Cancel, Preview, Save changes

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

Oct 27 20:20

ap-south-1.console.aws.amazon.com/vpconsole/home?region=ap-south-1#RouteTableDetails:RouteTableId=rtb-08f6ec21716b6fe81

Search [Alt+S]

Asia Pacific (Mumbai) Account ID: 6385-5998-3526 Shubham

VPC > Route tables > rtb-08f6ec21716b6fe81

Updated routes for rtb-08f6ec21716b6fe81 / PRIVATE\_ROUTE\_TABLE successfully

### rtb-08f6ec21716b6fe81 / PRIVATE\_ROUTE\_TABLE

Details Info

Route table ID rtb-08f6ec21716b6fe81	Main No	Explicit subnet associations 2 subnets	Edge associations -
VPC vpc-02d857bac21e4fc29   MUMBAL_VPC	Owner ID 638559983526		

Routes (2)

Destination	Target	Status	Propagated	Route Origin
172.16.0.0/16	local	Active	No	Create Route Table
192.168.0.0/16	pcx-05a87832bba494e7f	Active	No	Create Route

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

Oct 27 20:21

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0f015a59b09a74271

Account ID: 6385-5998-3526

Shubham

VPC > Route tables > rtb-0f015a59b09a74271 > Edit routes

### Edit routes

Destination	Target	Status	Propagated	Route Origin	
192.168.0.0/16	local	Active	No	CreateRouteTable	
172.16.0.0/16	Peering Connection	-	No	CreateRoute	<a href="#">Remove</a>

[Add route](#) [Cancel](#) [Preview](#) [Save changes](#)

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

Oct 27 20:21

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditRoutes:RouteTableId=rtb-0cb4cae94087aebcb

Account ID: 6385-5998-3526

Shubham

VPC > Route tables > rtb-0cb4cae94087aebcb > Edit routes

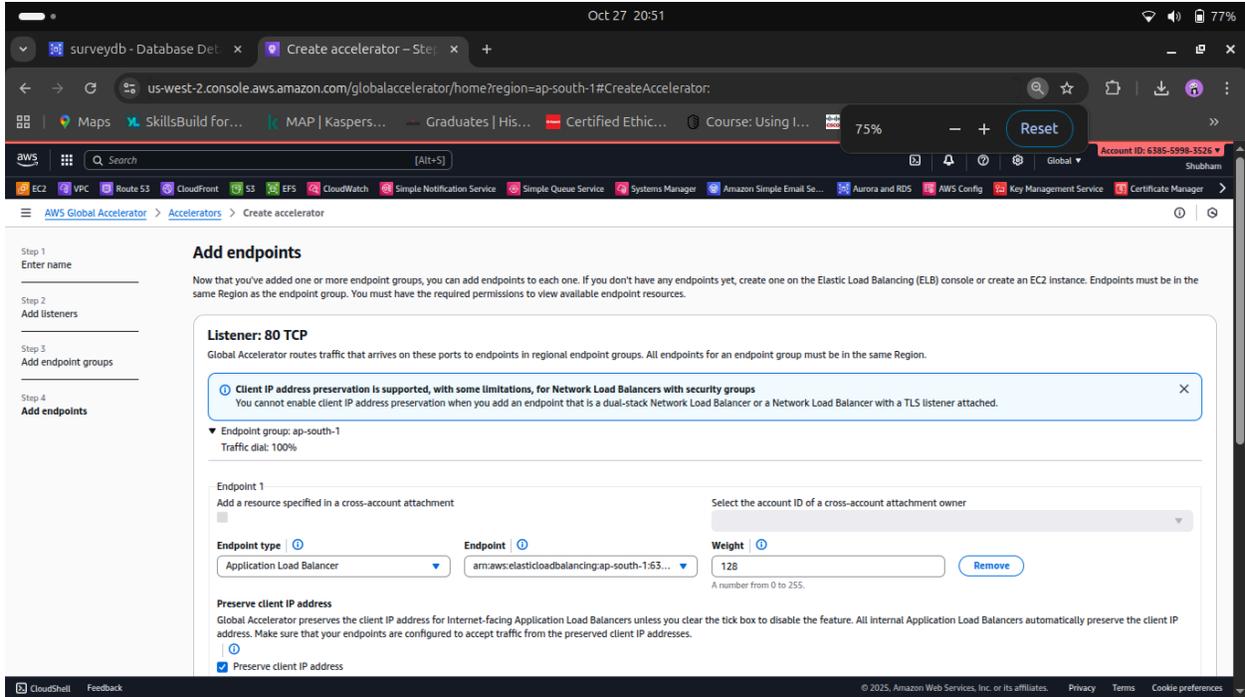
### Edit routes

Destination	Target	Status	Propagated	Route Origin	
192.168.0.0/16	local	Active	No	CreateRouteTable	
0.0.0.0/0	Internet Gateway	Active	No	CreateRoute	<a href="#">Remove</a>
172.16.0.0/16	Peering Connection	-	No	CreateRoute	<a href="#">Remove</a>

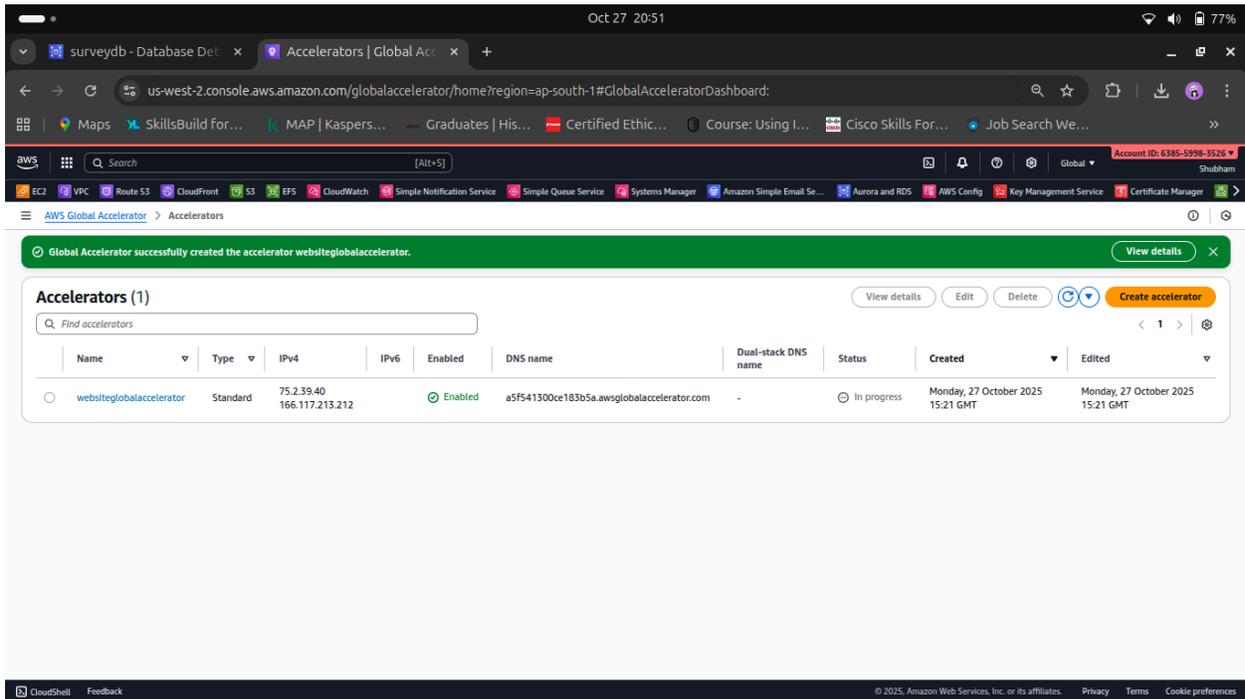
[Add route](#) [Cancel](#) [Preview](#) [Save changes](#)

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

**Step 42) Update route table entries for proper routing to both vpc**



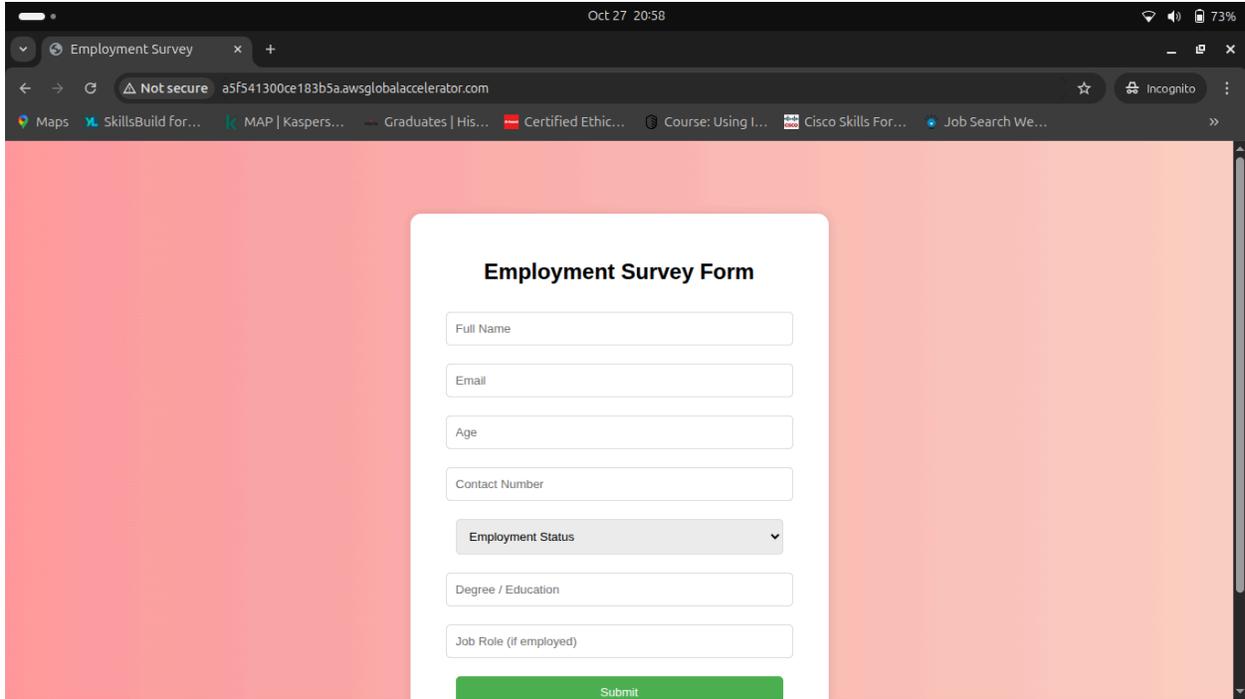
## Step 45) We will create Global Accelerator



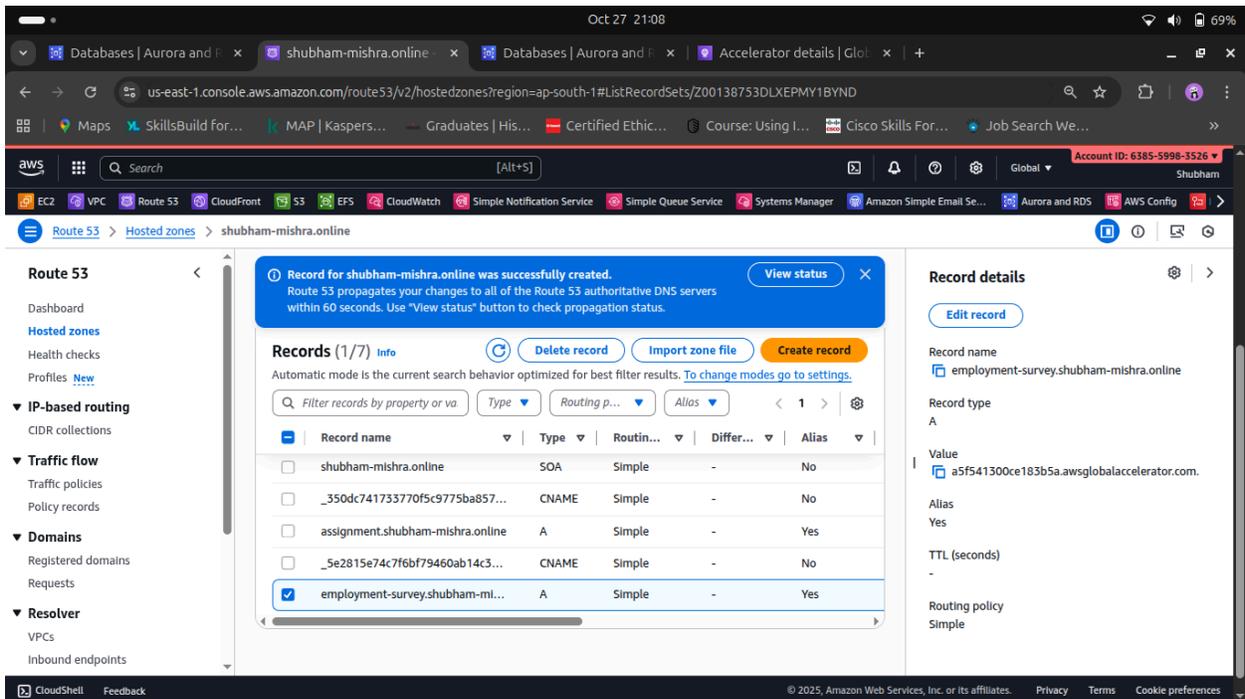
## STEP 46) Global accelerator is available now

```
Oct 27 21:00
shubham@shubham-Inspiron-14-3467: ~/Downloads
root@ip-172-16-0-20: ~
root@ip-192-168-3-212: /var/www/mywebsite
shubham@shubham-Inspiron-14-3467: ~/Downloads
shubham@shubham-Inspiron-14-3467:~/Downloads$ curl -I a5f541300ce183b5a.awsglobalaccelerator.com
HTTP/1.1 200 OK
Date: Mon, 27 Oct 2025 15:30:18 GMT
Content-Type: text/html
Content-Length: 2438
Connection: keep-alive
Server: nginx/1.24.0 (Ubuntu)
Last-Modified: Mon, 27 Oct 2025 13:31:49 GMT
ETag: "68ff7445-986"
Accept-Ranges: bytes

shubham@shubham-Inspiron-14-3467:~/Downloads$
```



## Step 47) Global Accelerator is reachable



## Step 48) We will add this global accelerator as alias in Route53

The screenshot shows a web browser window with the address bar displaying "employment-survey.shubham-mishra.online". The page content is a white form titled "Employment Survey Form" centered on a light red background. The form contains the following fields: "Full Name", "Email", "Age", "Contact Number", "Employment Status" (a dropdown menu), "Degree / Education", and "Job Role (if employed)". A green "Submit" button is located at the bottom of the form.

**Step 49) We can access our website at [employment-survey.shubham-mishra.online](https://employment-survey.shubham-mishra.online)**

## Key Highlights

- **Multi-Region Active-Active Setup:** Application stack deployed in two regions with independent ALB + ASG for high availability.
- **Aurora Global Database:** Sub-second cross-region replication between primary (Mumbai) and global replica (Virginia).
- **Private Connectivity:** Inter-region VPC peering enabled secure private communication between EC2 and Aurora.
- **Global Traffic Routing:** AWS Global Accelerator directs users to the nearest region with automatic regional failover.

- **Scalable & Secure:** Stateless app servers, private subnets, least-privilege security groups, NAT for outbound internet, and fully automated ASG deployments.
- **Disaster Recovery Ready:** If the primary region fails, the secondary region becomes fully operational with DB promotion and traffic rerouting.

## **Outcome**

**A fully functional, enterprise-grade, globally available survey application delivering high performance, resilience, and near-zero downtime—ideal for real-world multi-region use cases.**