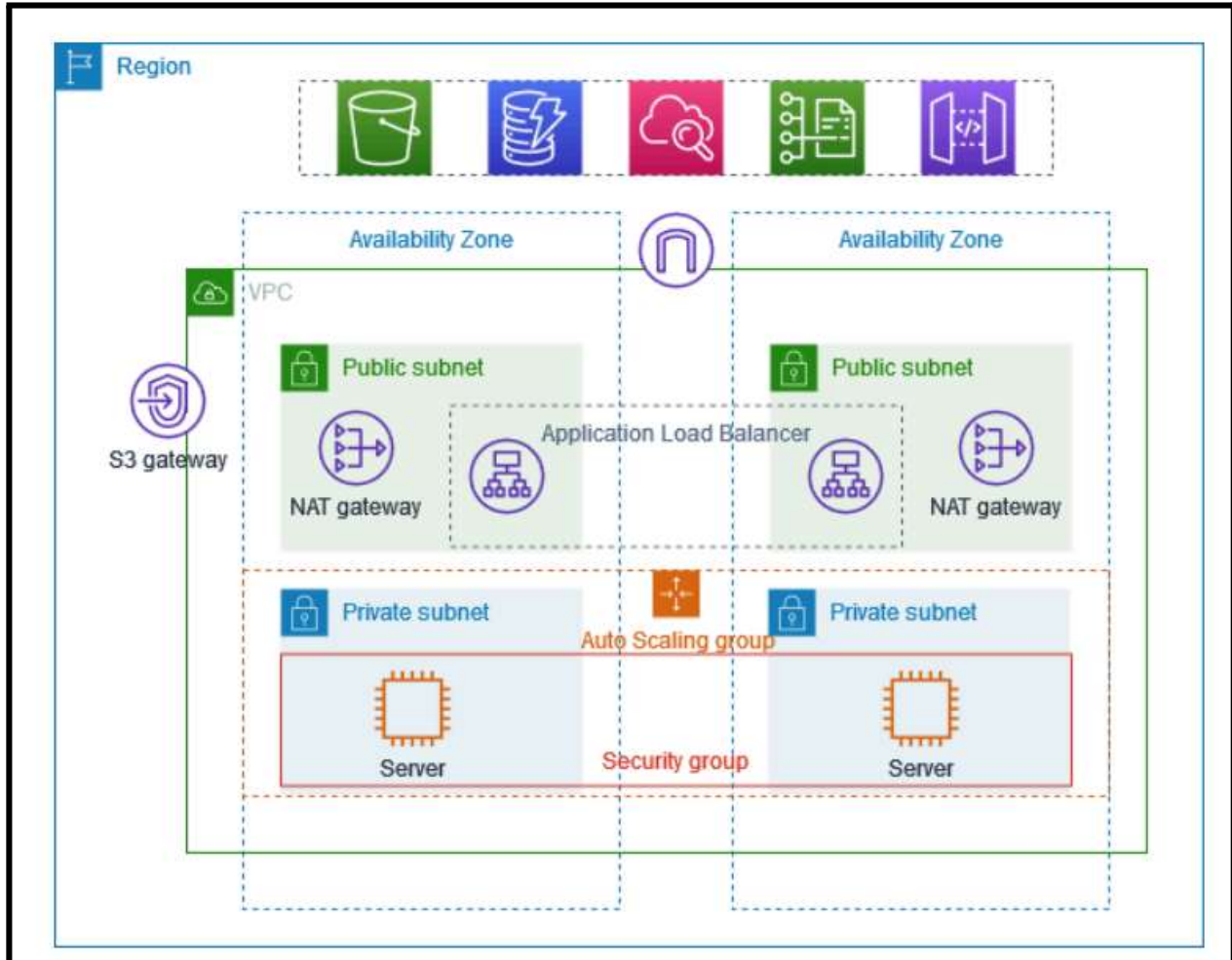


A DevOps project for a 2-Tier Application.

VPC with Private and Public Subnet for the production.



Region - N. Virginia

VPC -

Availability Zone

Subnet - Private / Public

Security Group

NAT GATEWAY

Internet Gateway

Autoscaling

LoadBalancer

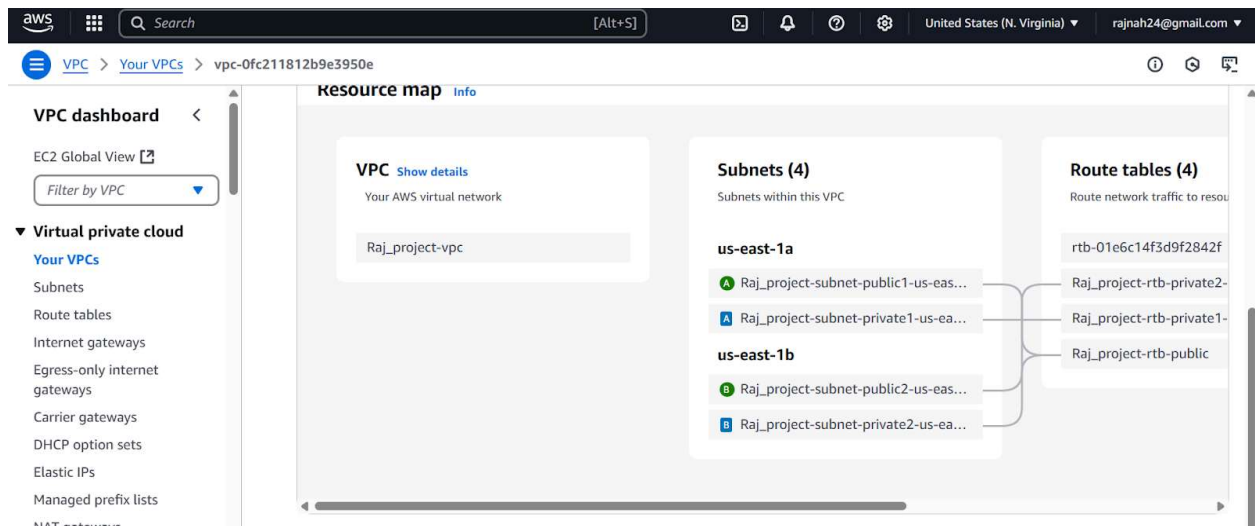
Bastion Host / Jump Server

Motive: demonstration to create a vpc that can be used for the server in a production environment.

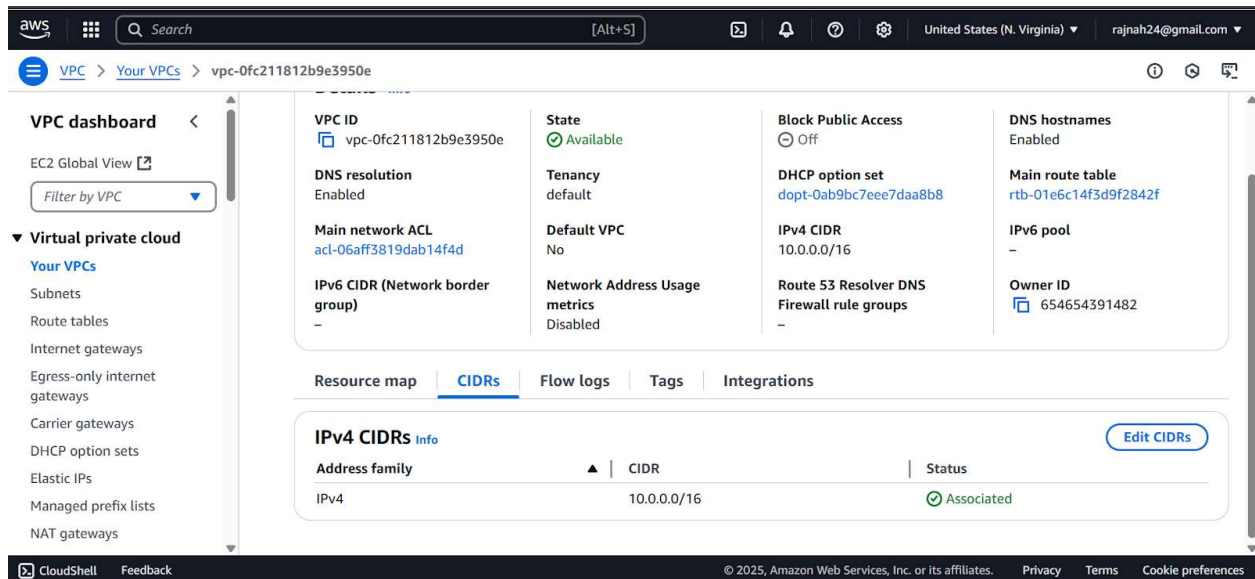
About the project:

- To improve resiliency, deploying the server in two availability zones, by using an autoscaling group and an application load balancer.
-
- For security, you serve the server in private subnets.
-
- The server receives requests through the load balancer.
-
- The servers can connect to the internet by using NAT gateway.
-
- To improve resiliency, deploying the NAT gateway in both availability zones.

VPC:



[A]

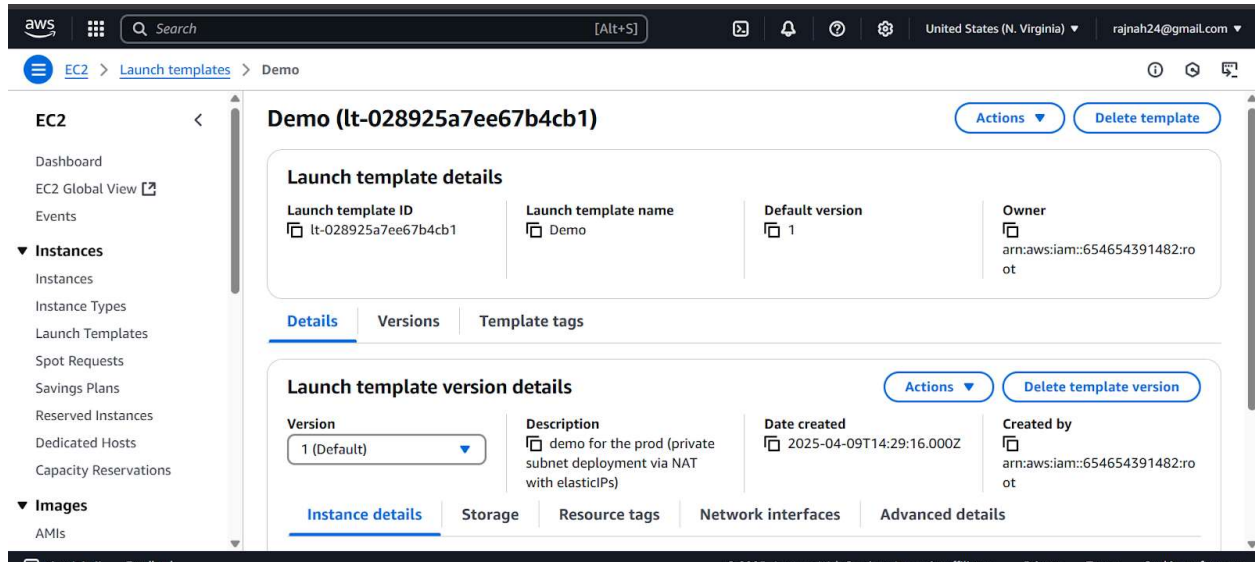


[B] CIRD Configuration

Post VPC creation, Need to create the Autoscaling / template for the following prod deployment.

AUTOSCALING

Creating a Launch template:



Desire prerequisite:

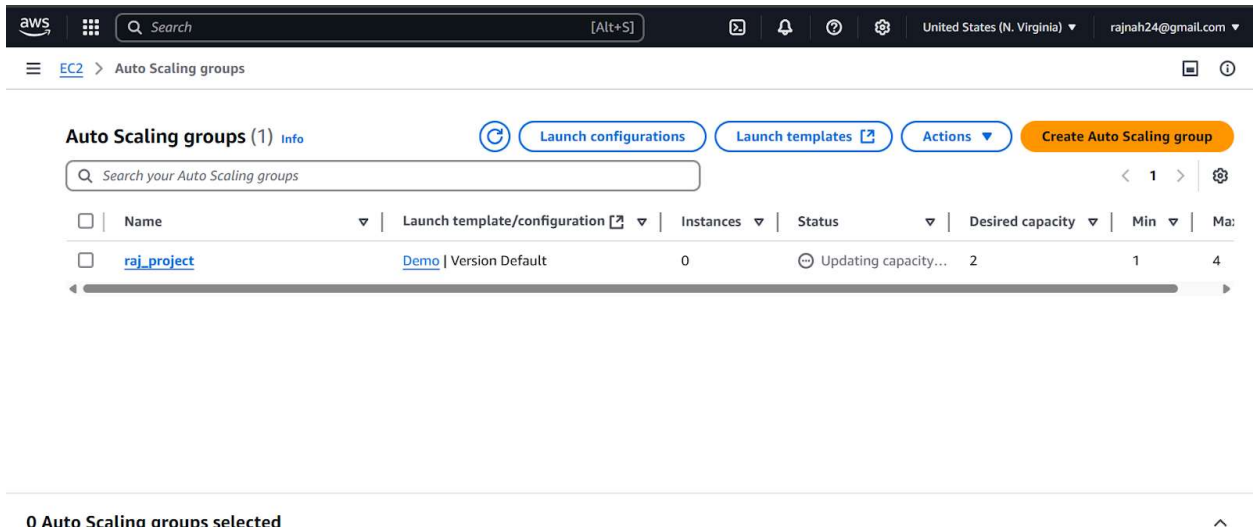
AMI

Type: Free tier

Security group - demo

Key_name - raj_project

VPC - New created VPC (private subnet / public subnet)



Auto Scaling groups (1) Info

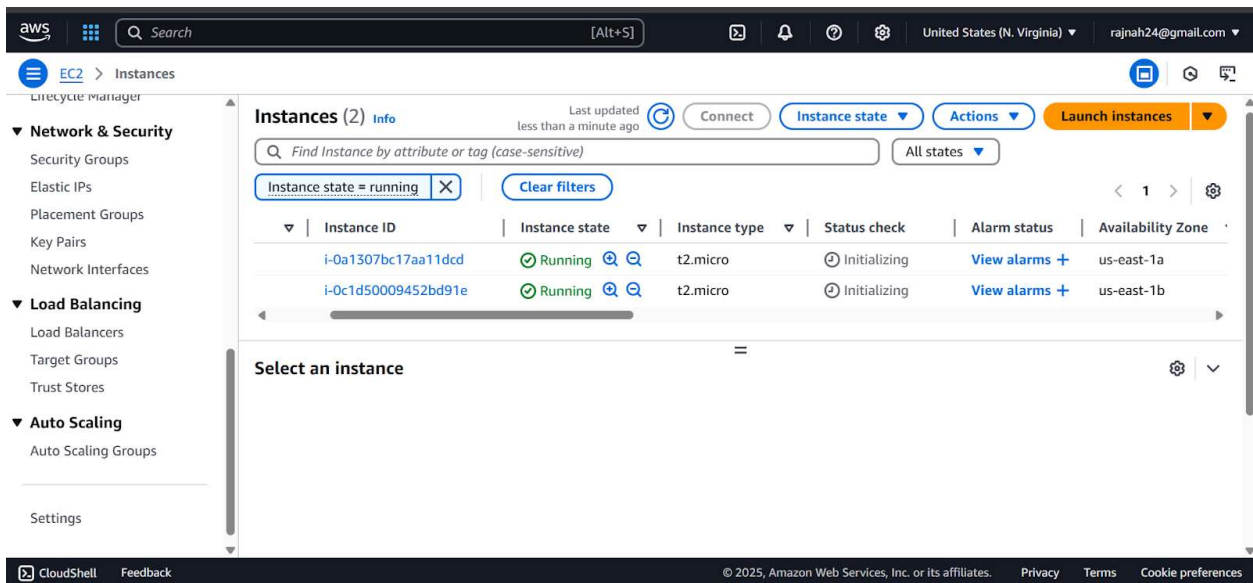
Launch configurations Launch templates Actions Create Auto Scaling group

Search your Auto Scaling groups

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max
raj_project	Demo Version Default	0	Updating capacity...	2	1	4

0 Auto Scaling groups selected

Instance creation: launch through auto-scaling



Instances (2) Info

Find Instance by attribute or tag (case-sensitive)

Instance state = running Clear filters

Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
i-0a1307bc17aa11dcd	Running	t2.micro	Initializing	View alarms +	us-east-1a
i-0c1d50009452bd91e	Running	t2.micro	Initializing	View alarms +	us-east-1b

Select an instance

Creating a bastion host to access the privateIP instance.

LOAD BALANCER FOR PUBLIC SUBNET AND APPLICATION ACCESS FROM IG APPLICATION LOAD BALANCER:

The screenshot displays the AWS Management Console interface for an Application Load Balancer. The top navigation bar includes the AWS logo, a search bar, and the user's account information (United States (N. Virginia), rajnah24@gmail.com). The left sidebar shows the navigation menu with categories like Network & Security, Load Balancing, and Auto Scaling. The main content area shows the 'Project' details for the newly created load balancer.

Successfully created load balancer: Project
It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

Application Load Balancers now support public IPv4 IP Address Management (IPAM)
You can get started with this feature by configuring IP pools in the **Network mapping** section. [Edit IP pools](#)

Project

Details

Load balancer type Application	Status ⏸ Provisioning	VPC vpc-067e36ee68e2efdb0	Load balancer IP address type IPv4
Scheme Internet-facing	Hosted zone Z35SXDOTRQ7X7K	Availability Zones subnet-04e088da57bf486ff us-east-1b (use1-az2) subnet-	Date created April 15, 2025, 20:32 (UTC+05:30)

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

LAUNCHING THE DNS FOR THE ACCESS OF PYTHON APPLICATION