CREATING A HIGHLY AVAILABLE 3-TIER ARCHITECTURE IN AWS

Browser ↔ Load balancer + Web server ↔App server

(application) ↓↑

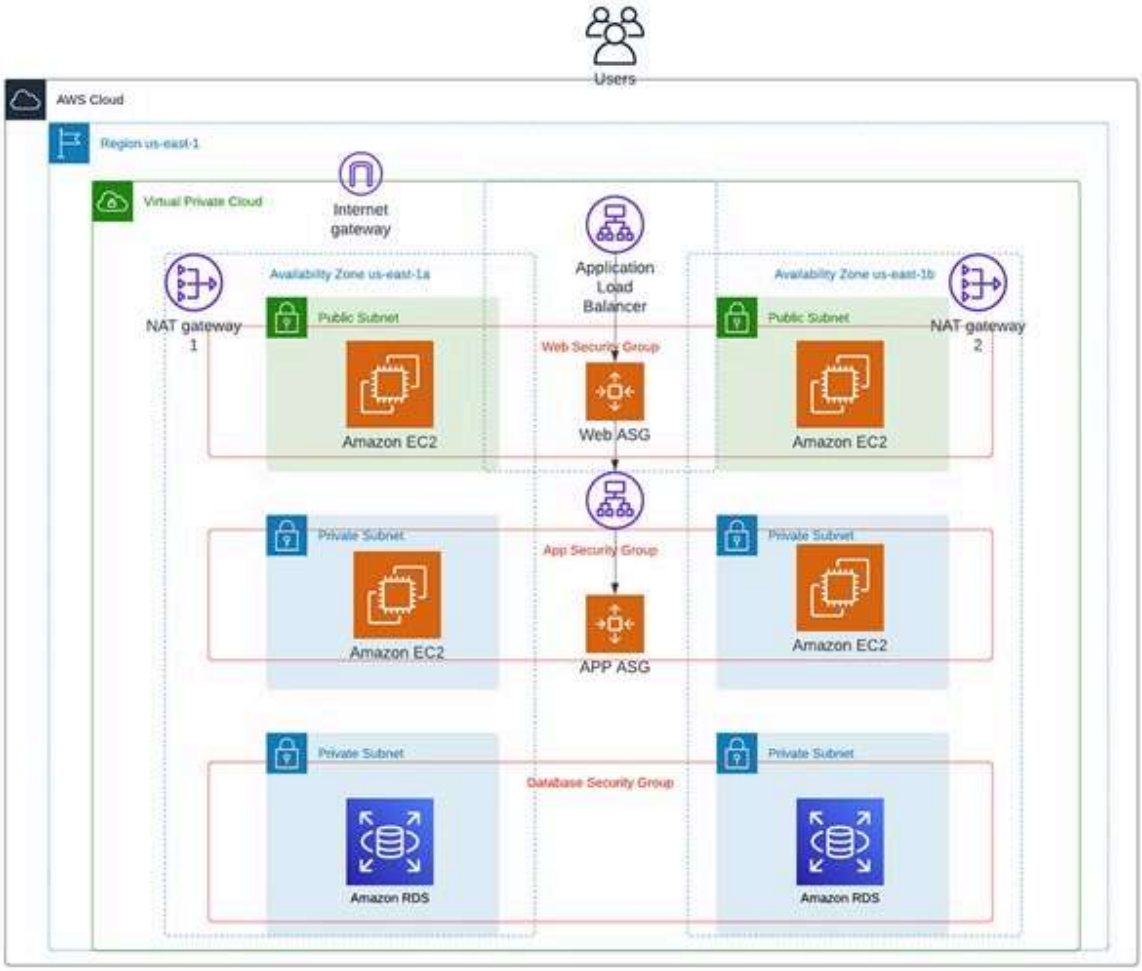
Database Server

1.Load balancer+web server = web tier
2.App server = app tier
3.Data base server = database tier

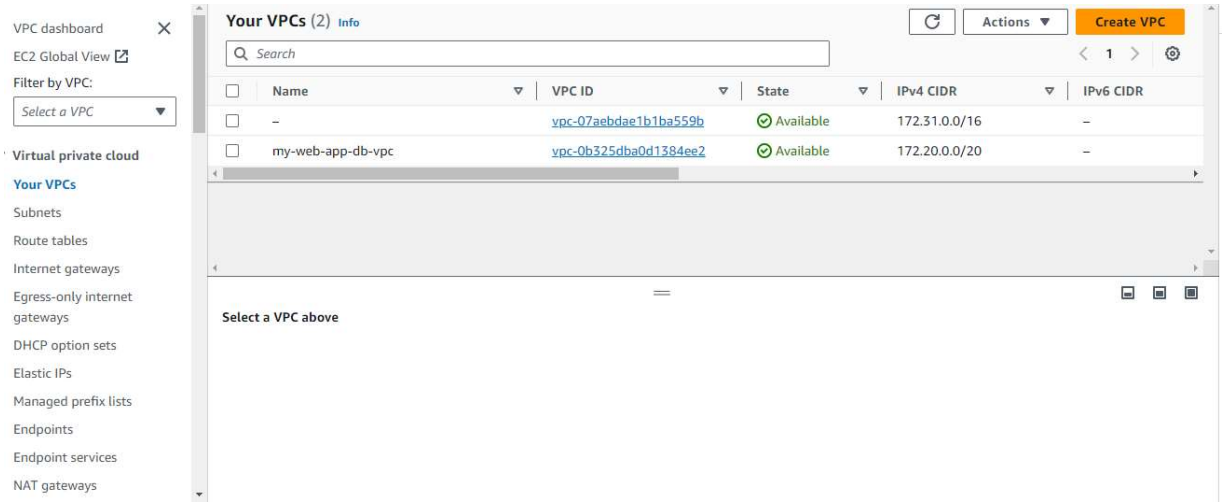
Whenever we open an application in browser it will go to load balancer there are multiple web servers and load balancer hits the empty web server it will send that request to application server and application server connects with the database server ,it will retrieve the data and convert to proper understanding format and send to app server .
App server sends to web server ,it will send to load balancer and load balancer will send to the browser.

Advantages of 3-tier architecture :

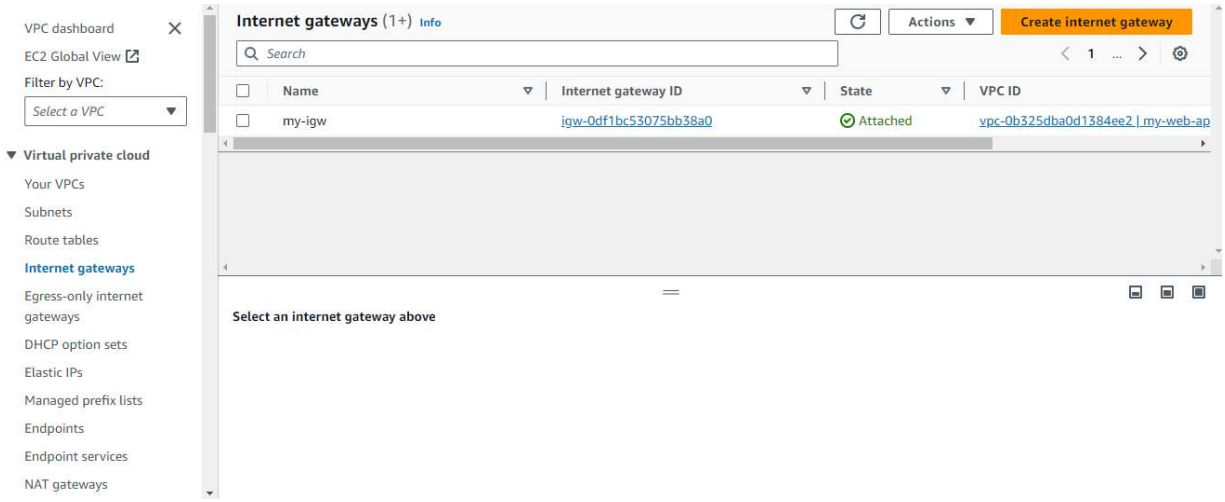
- Security
- Queue management
- Speed
- Quality of data
- Scalability and flexibility


Users
AWS Cloud
Region us-east-1
Virtual Private Cloud
Internet gateway
Application Load Balancer
Availability Zone us-east-1a
Availability Zone us-east-1b
NAT gateway 1
NAT gateway 2
Public Subnet
Public Subnet
Web Security Group
Amazon EC2
Web ASG
Amazon EC2
Private Subnet
Private Subnet
App Security Group
Amazon EC2
APP ASG
Amazon EC2
Private Subnet
Private Subnet
Database Security Group
Amazon RDS
Amazon RDS

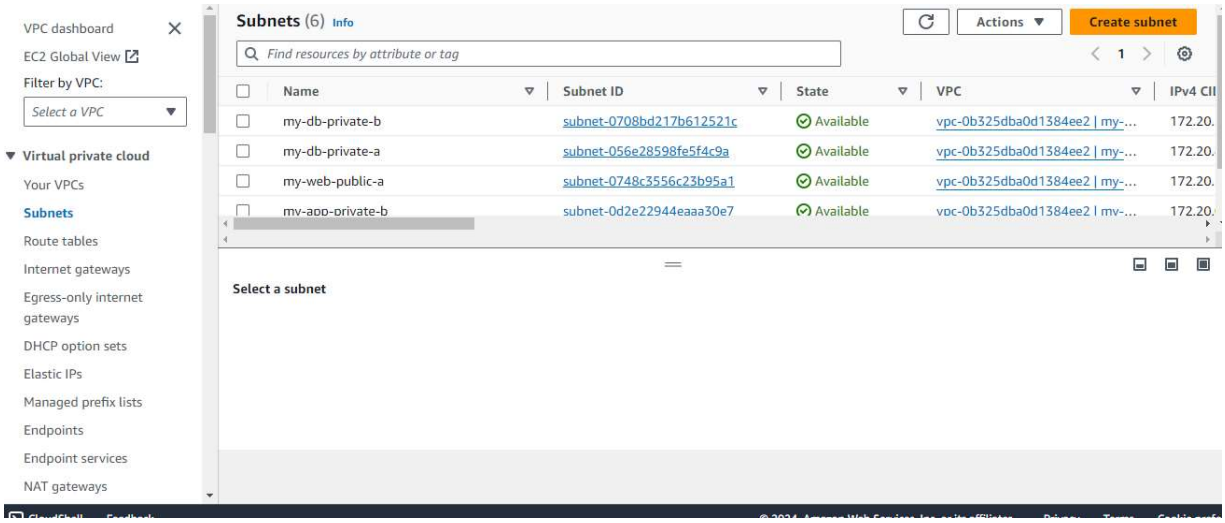
Step 1 : Create a VPC (my-web-app-db-vpc) .

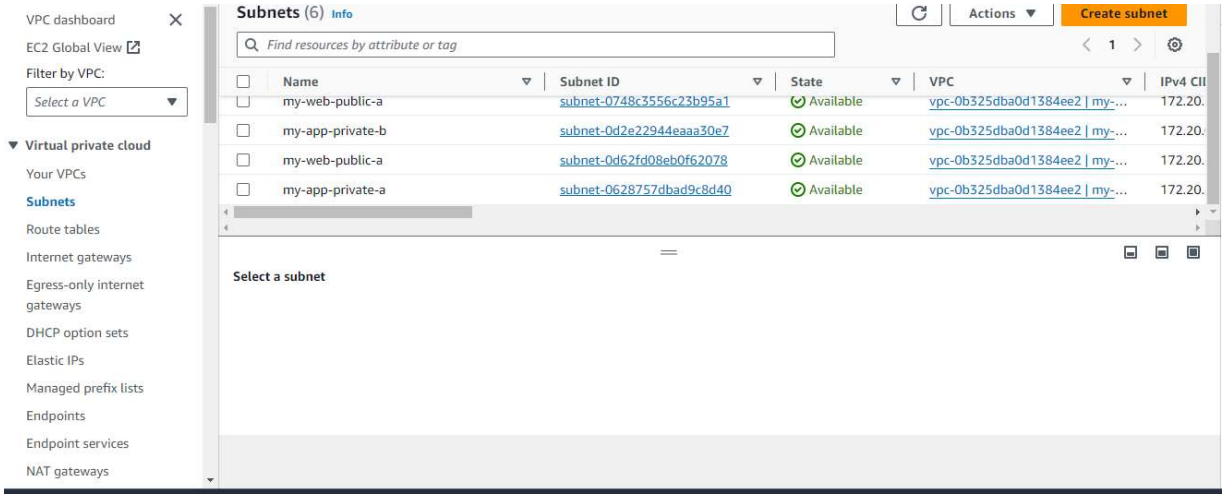


Step 2 : Create an internet gateway (my-igw) and attach to VPC (my-wed-app-db-vpc)

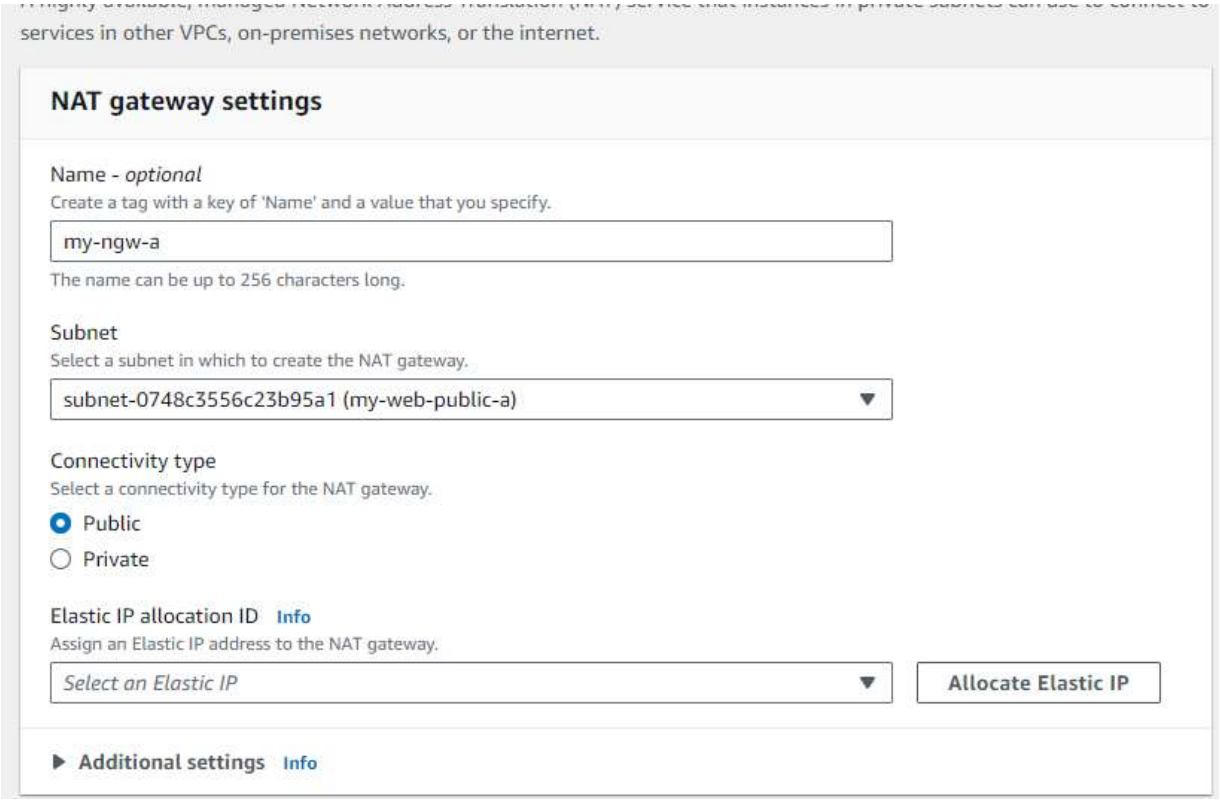


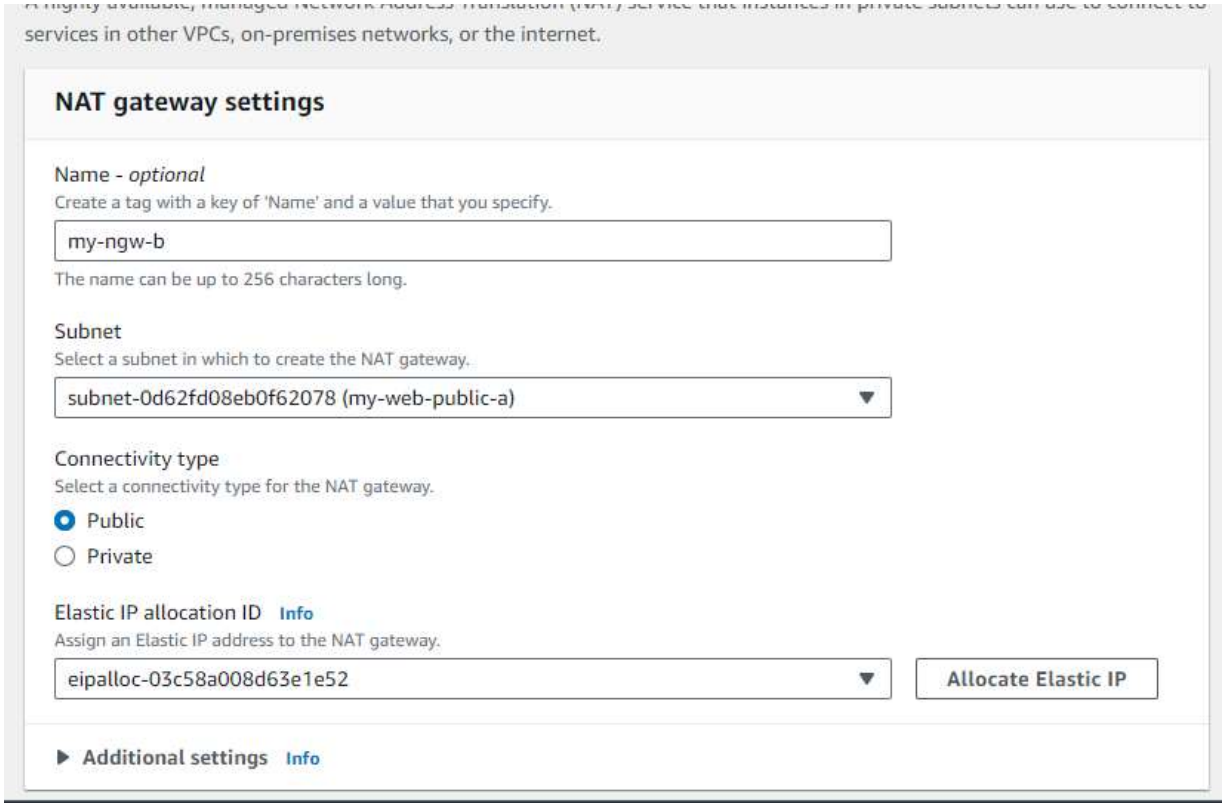
Step 3 : Create two public subnets and four private Subnets (my-web-public-a,my-web-public-b,My-app-private-a,my-app-private-b (my-web-public-a,my-web-public-b,My-app-private-a,my-app-private-b my-db-private-a,my-db-private-b) in two different availability zones .

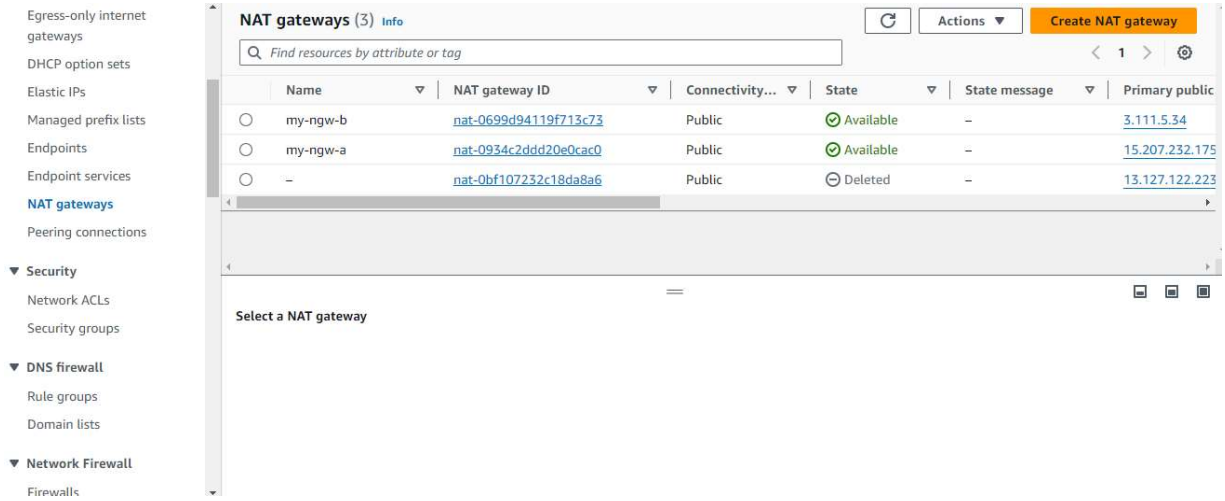




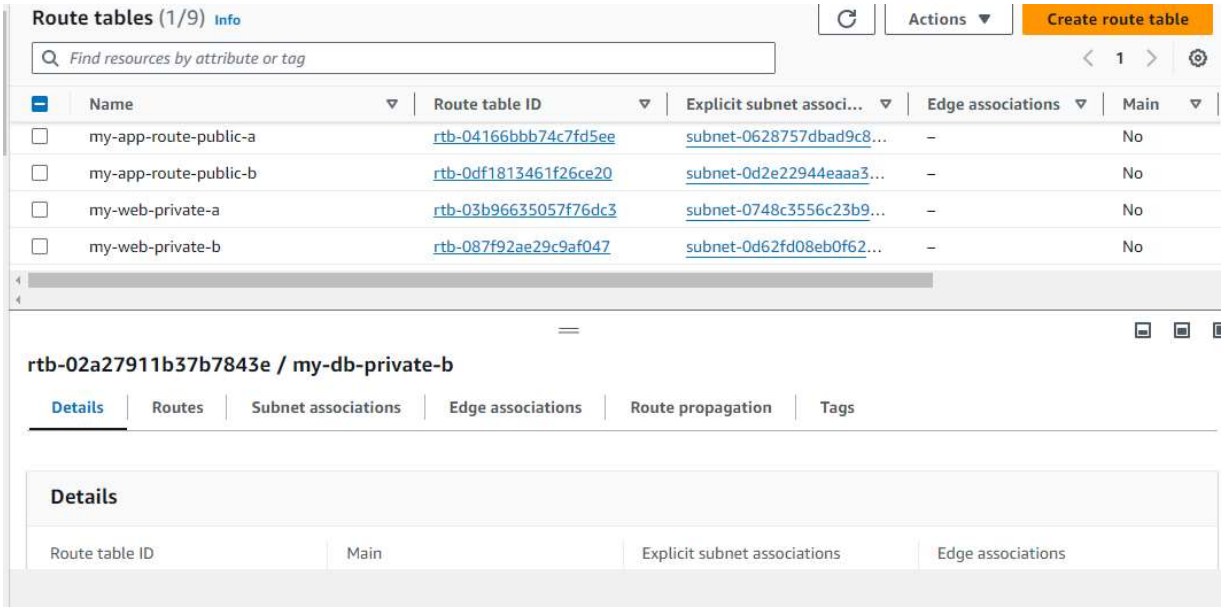
Step 4 : Create two nat gateways (my-ngw-a,my-ngw-b) and connect to public subnets (my-web-public-a,my-web-public-b) so that ec2 instance in public subnets can connect to internet and no one can connect to it because nat gateway is one way traffic either incoming (or) outgoing .

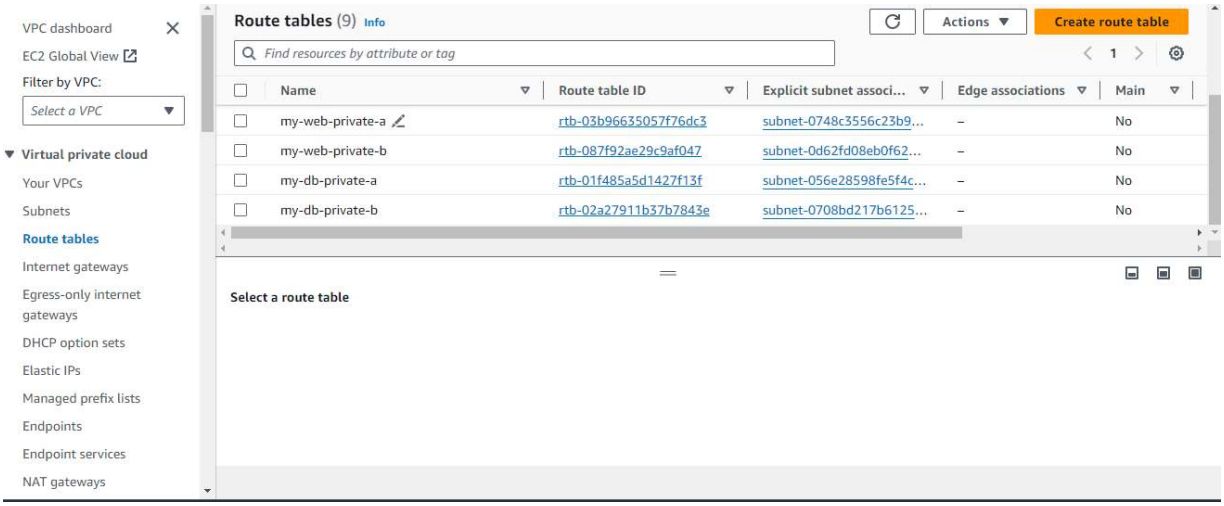



services in other VPCs, on-premises networks, or the internet.
NAT gateway settings
Name - optional
Create a tag with a key of 'Name' and a value that you specify.
my-ngw-b
The name can be up to 256 characters long.
Subnet
Select a subnet in which to create the NAT gateway.
subnet-0d62fd08eb0f62078 (my-web-public-a)
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-03c58a008d63e1e52
Allocate Elastic IP
Additional settings Info

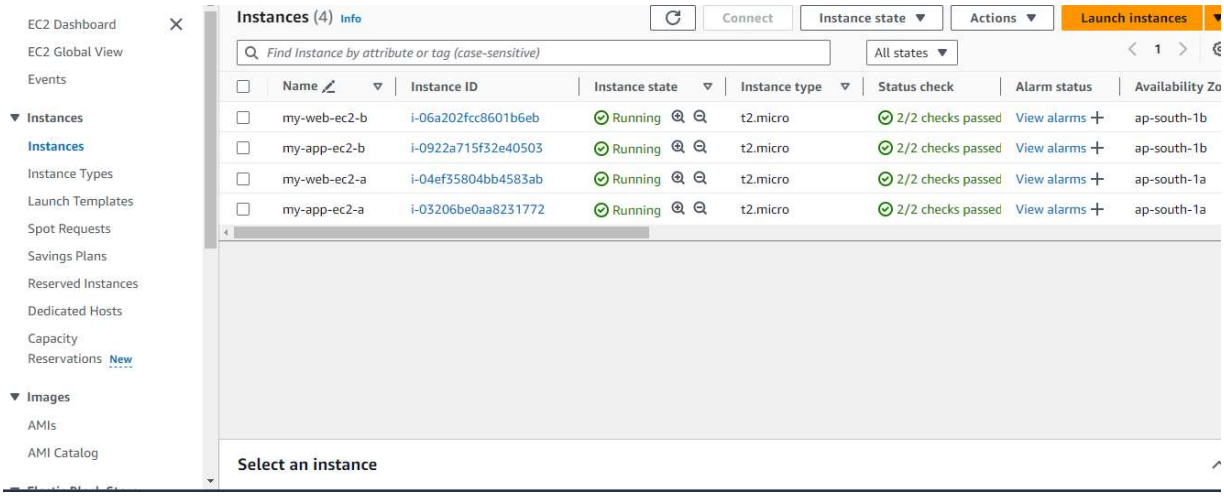

Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections
Security
Network ACLs
Security groups
DNS firewall
Rule groups
Domain lists
Network Firewall
Firewalls
NAT gateways (3) Info
Actions
Create NAT gateway
Find resources by attribute or tag
Name
NAT gateway ID
Connectivity...
State
State message
Primary public
my-ngw-b
nat-0699d94119f713c73
Public
Available
3.111.5.34
my-ngw-a
nat-0934c2ddd20e0cac0
Public
Available
15.207.232.175
–
nat-0bf107232c18da8a6
Public
Deleted
13.127.122.223
Select a NAT gateway

Step 5 : Now create six route tables (my-app-public-route-a,my-app-public-route-b,my-web-private-a,my-web-private-b,my-db-private-a,my-db-private-b) and click on save association ,so that route tables can configure routing to subnets in VPC .



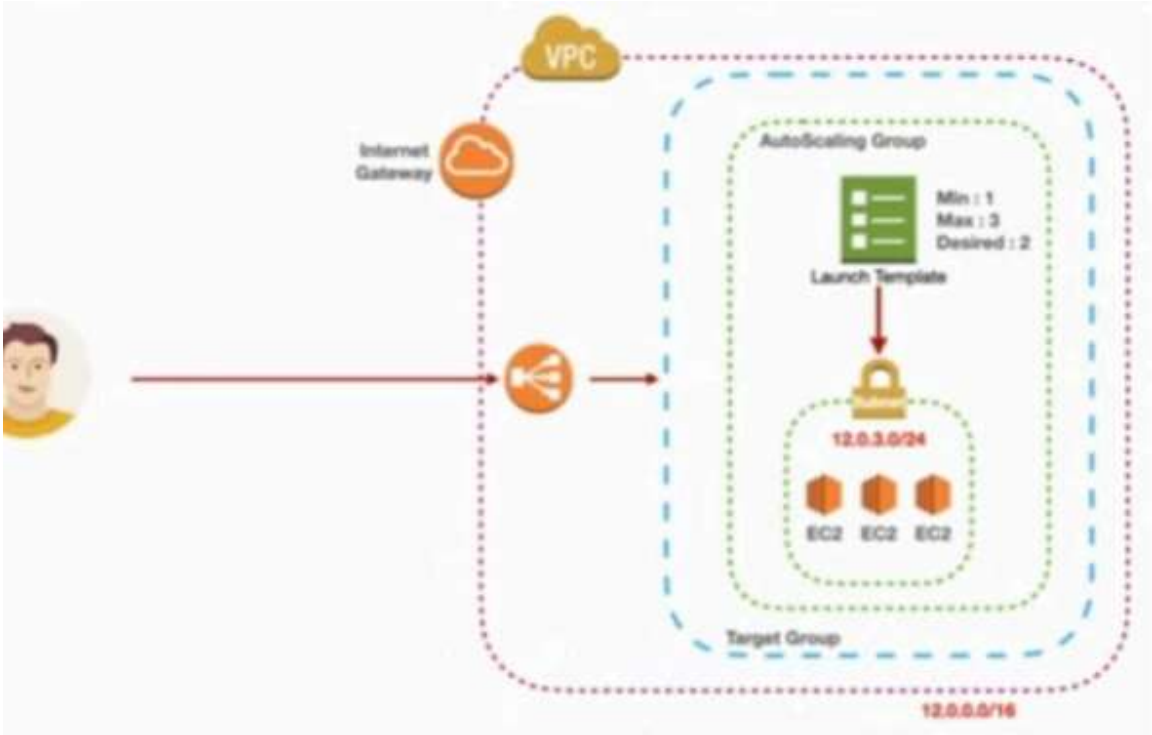


Step 6 : create four ec2 instances (my-web-ec2-a,my-web-ec2-b,my-app-ec2-a,my-app-ec2-b) in two different availability zones and connect to VPC and also for subnets my-web-public-a,my-web-public-b,my-app-private-a,my-app-private-b.



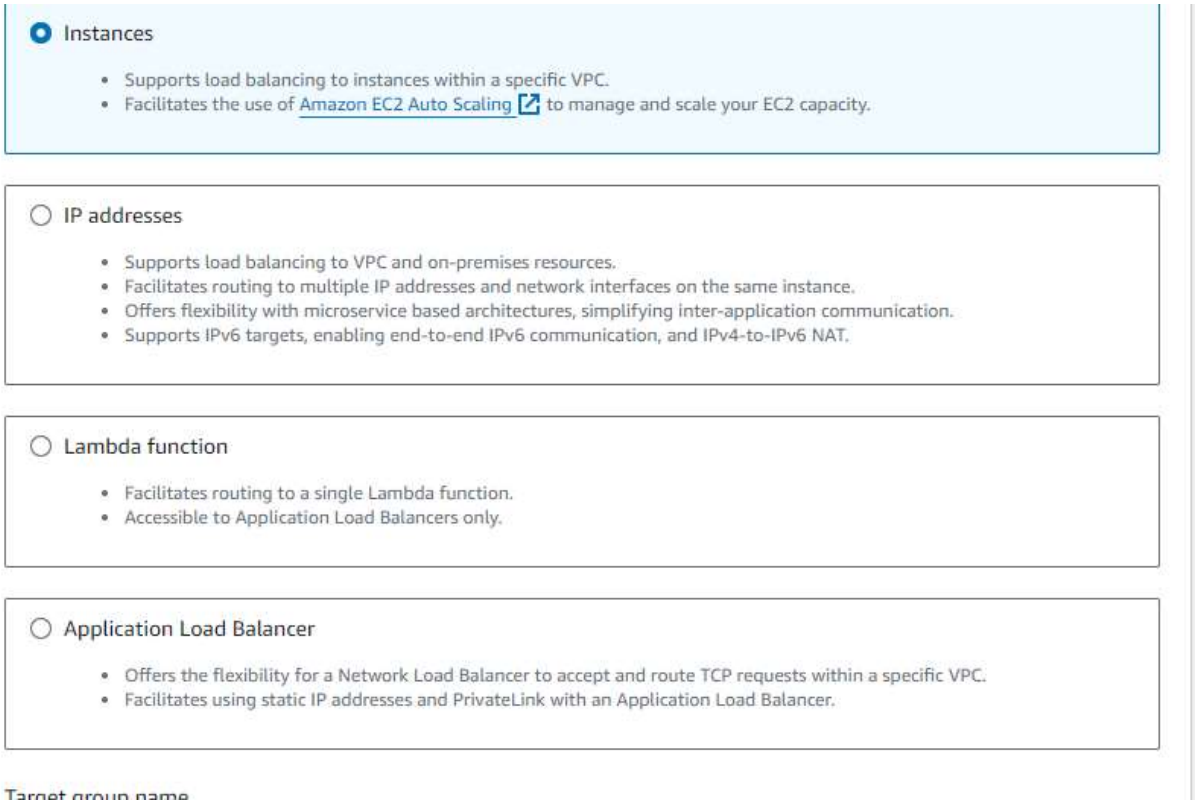
> Created vpc,subnets,route tables,internet gateway,nat gateways and attached four ec2 instances successfully .

CREATE APPLICATION LOAD BALANCER AND INSTALLING WEBSERVER,APPSERVER



- **Load balancer is a service provided by amazon in which the incoming traffic is automatically distributed across a group of servers or targets .**
- **Load balancer increases the speed and performance .**
- **Application load balancer is a type of load balancer it works on application layer and also works on http and https protocol .**

Step 1 : Create target group and we are creating target group inside the subnets



Target group name

tg-a

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

Protocol : Port

Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation

HTTP

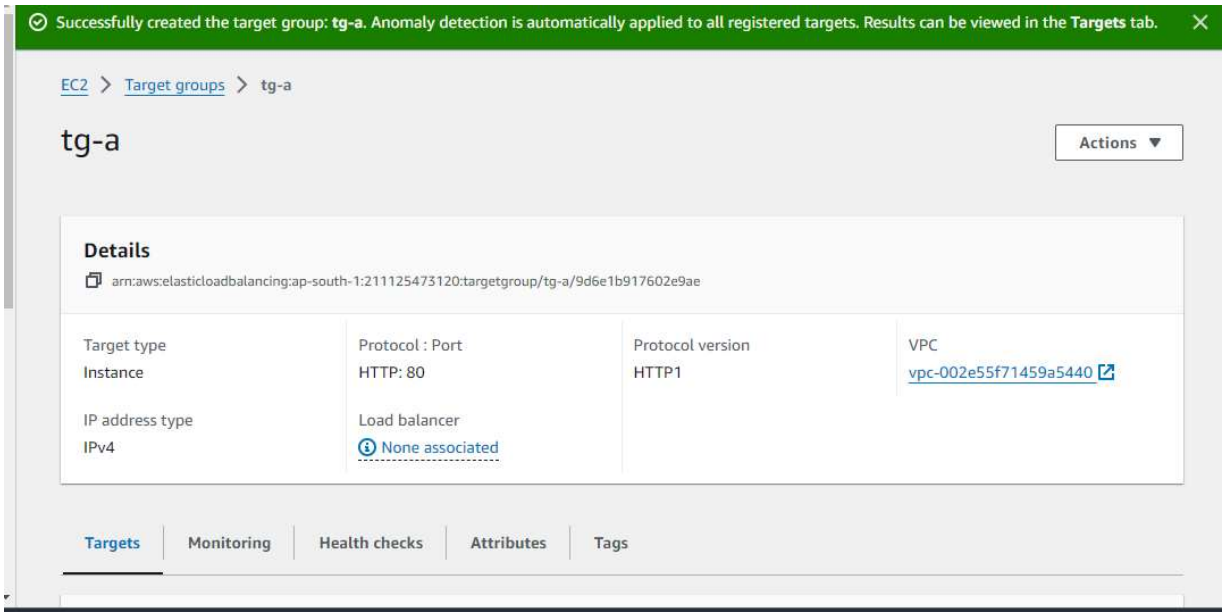
80

1-65535

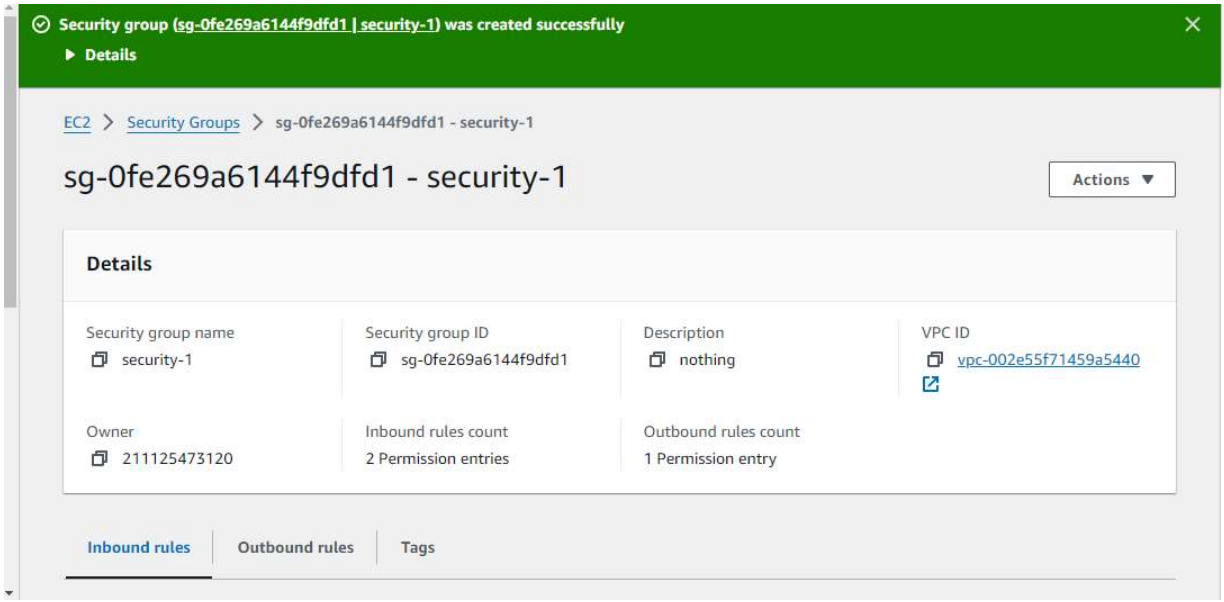
IP address type

Only targets with the indicated IP address type can be registered to this target group.

- (●) IPv4
  Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.
- ( ) IPv6
  Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more

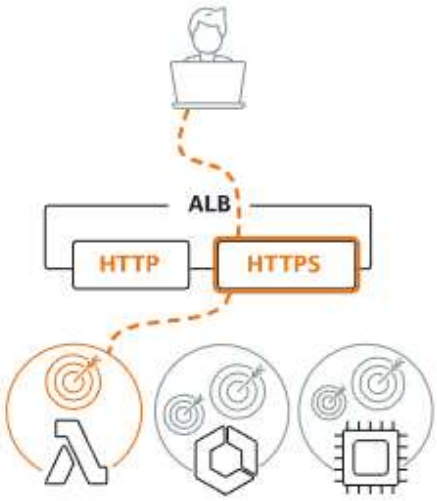


Step 2: create a security group

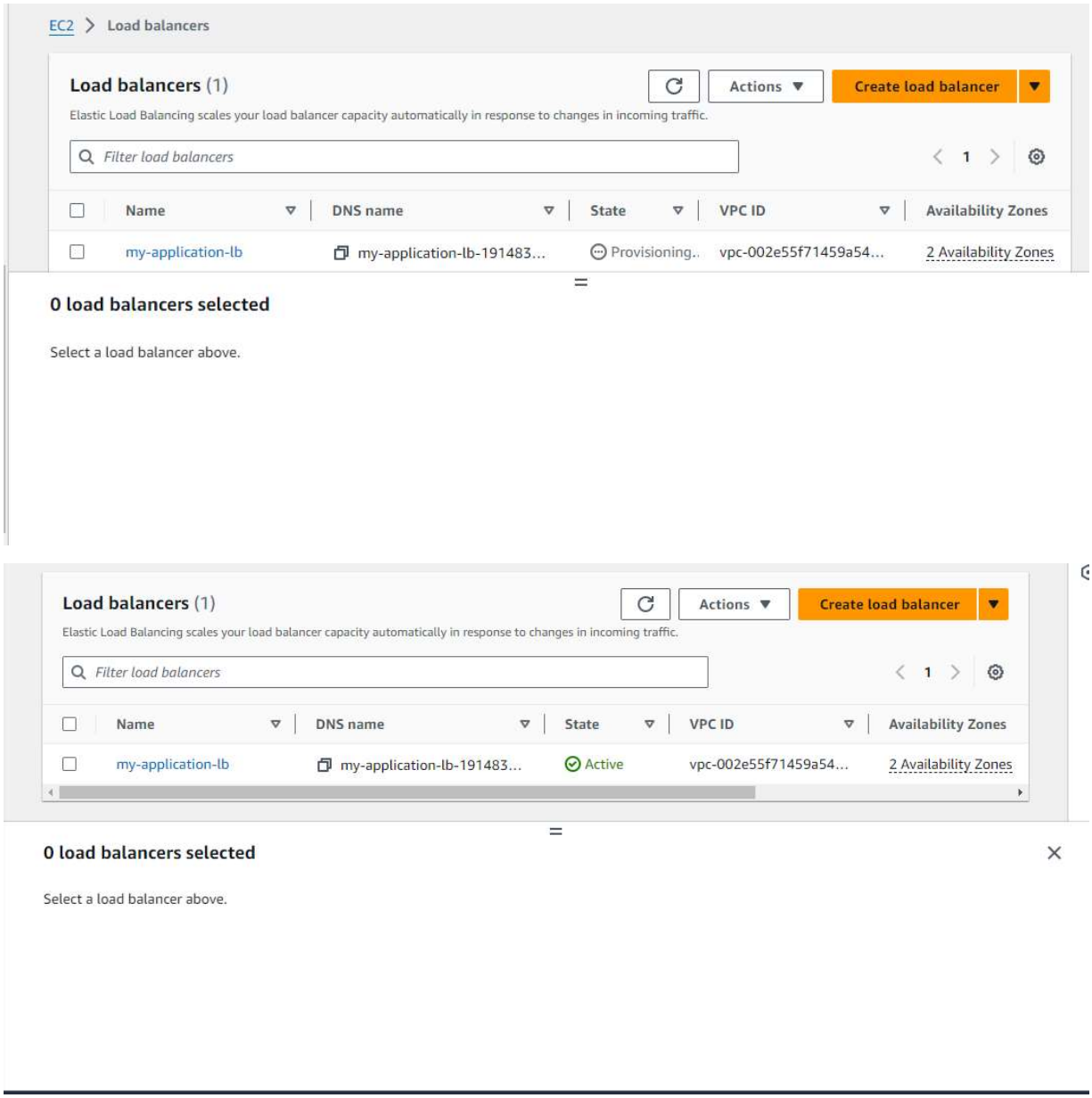


Step 3: create a application load balancer

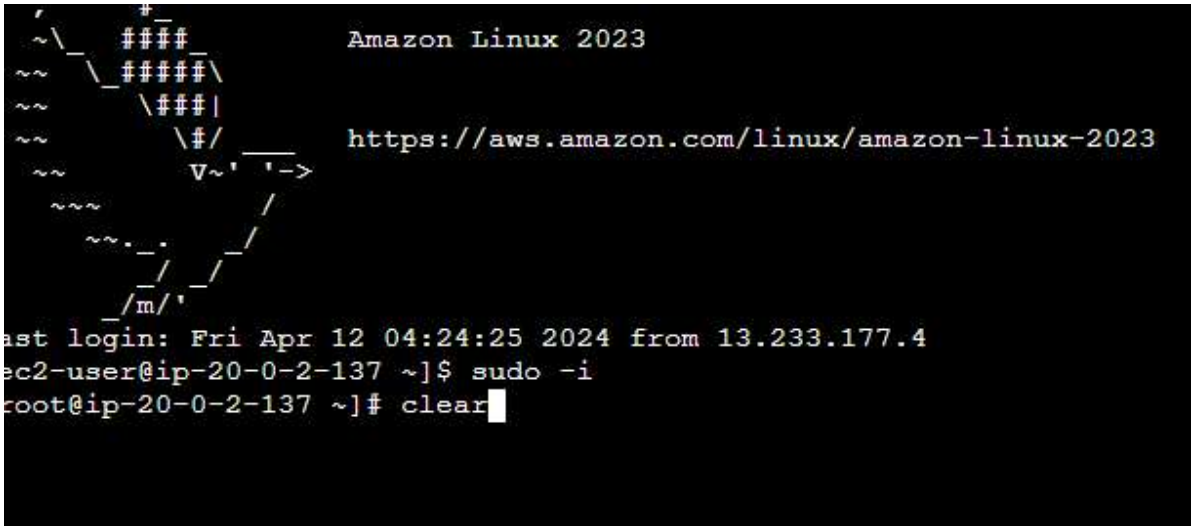
Application Load Balancer Info



Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level,

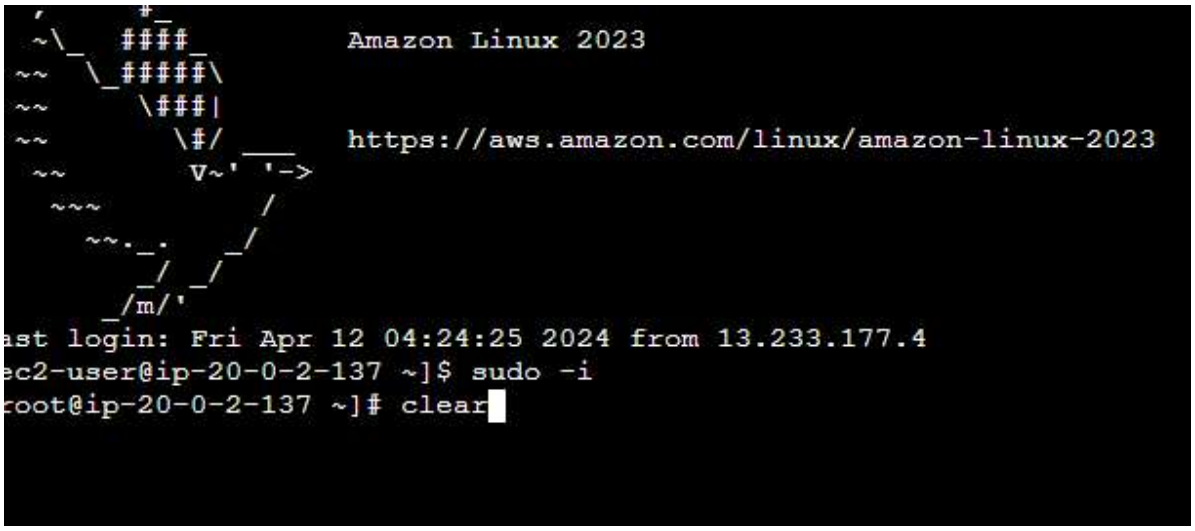


Created application load balancer now have to create a web server .
Now click on the public instance (my-web-ec2-a) and click on connect .



Now click on the instance (my-web-ec2-b) and click on connect .

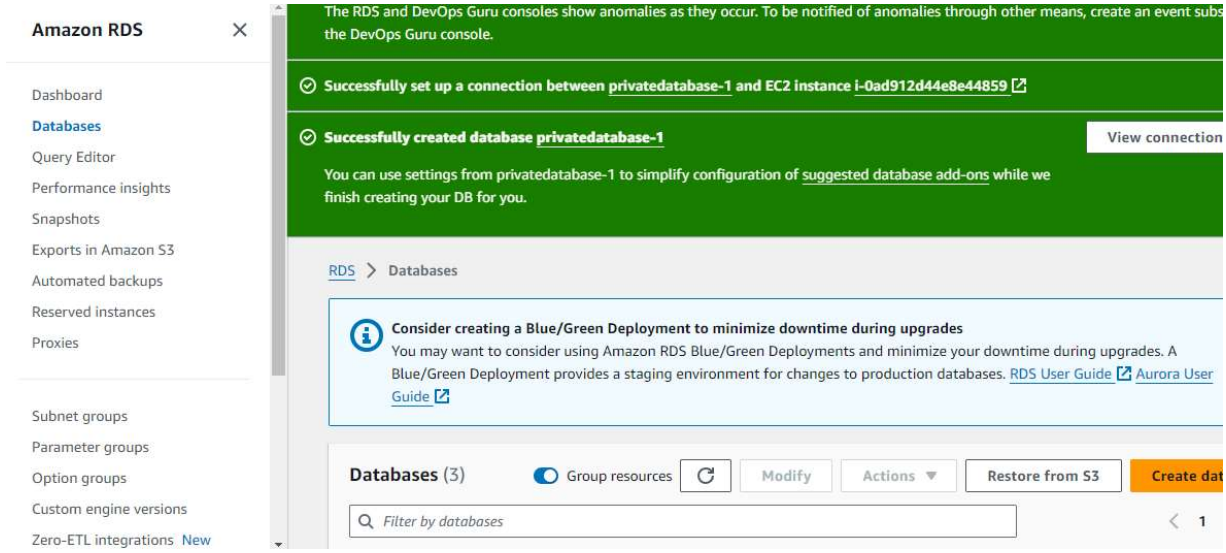
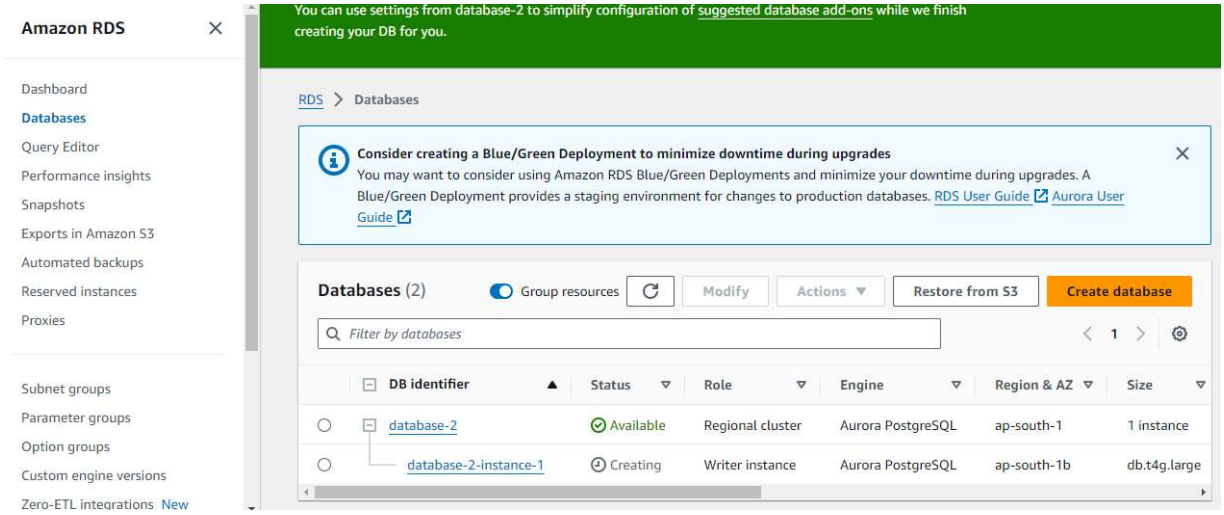
Now click on private ec2 instances and create app servers and click on connect .
Connect private with the public instances .



Now create a database and connect to private subnets (my-db-private-a,my-db-private-b).

- **Database is used to store the data in the form of tables that are in rows and columns .**

First create an instance with a private subnet and attach to the database,password is mandatory while creating the database .



Creating another ec2 instance with private subnet and connecting to the database .