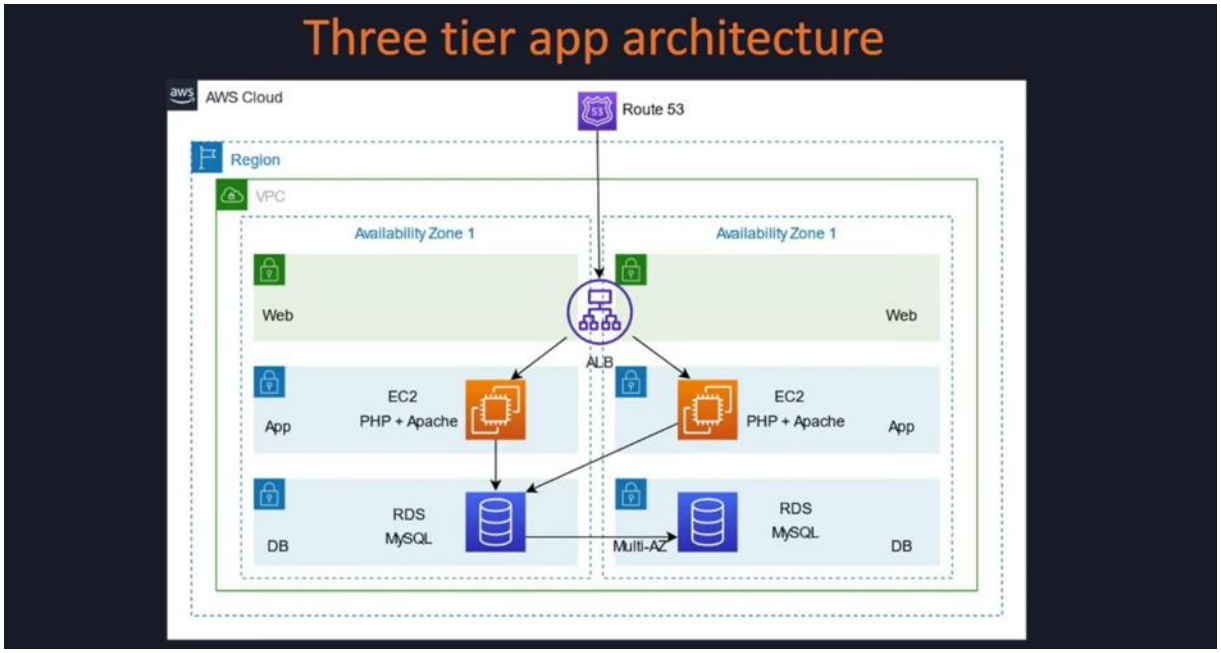
AUTHOR : ISRAR AHEMAD KHAN

MOB NO: 8888337050

EMAIL: ik427094@gmail.com

AWS Challenge (Designing Robust Web Application on AWS)

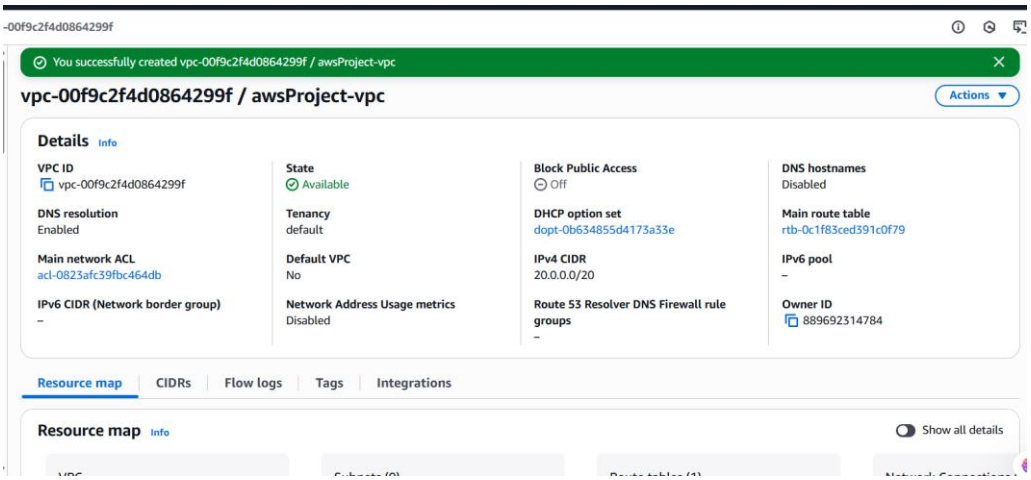


Summary: This AWS architecture diagram shows a reliable web application built with three parts. Users find the application through Route 53, a DNS service. The application is protected in a virtual private network (VPC) on AWS, spread across different availability zones to ensure it keeps working even if one zone fails. The user-facing part of the application uses multiple servers(EC2 Instances) for better performance, with an Application Load Balancer (ALB) sharing the traffic evenly. The business logic is handled in another part, possibly using a database service. Data is securely stored in a separate part, using a database service RDS. This setup provides a strong base for web applications on AWS, with the flexibility to add more services for advanced features.

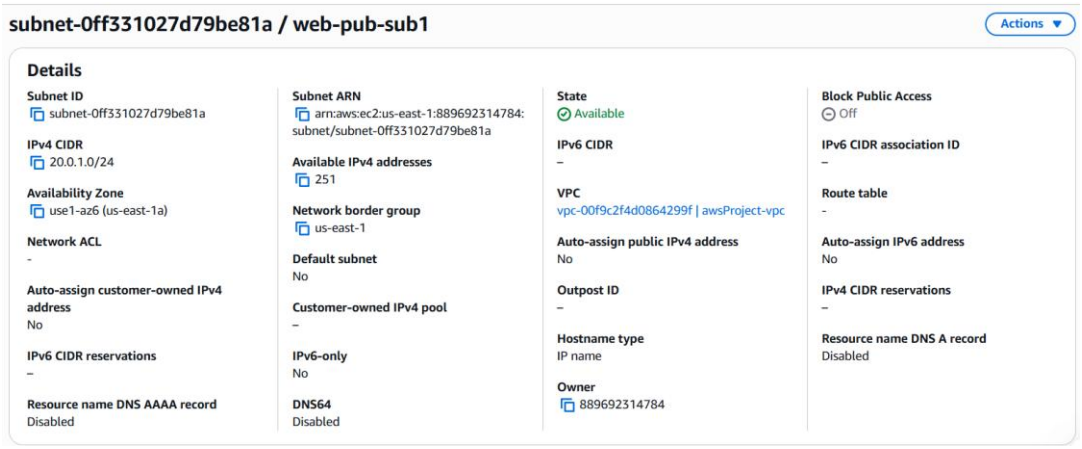
Lets Proceed:

Part 1

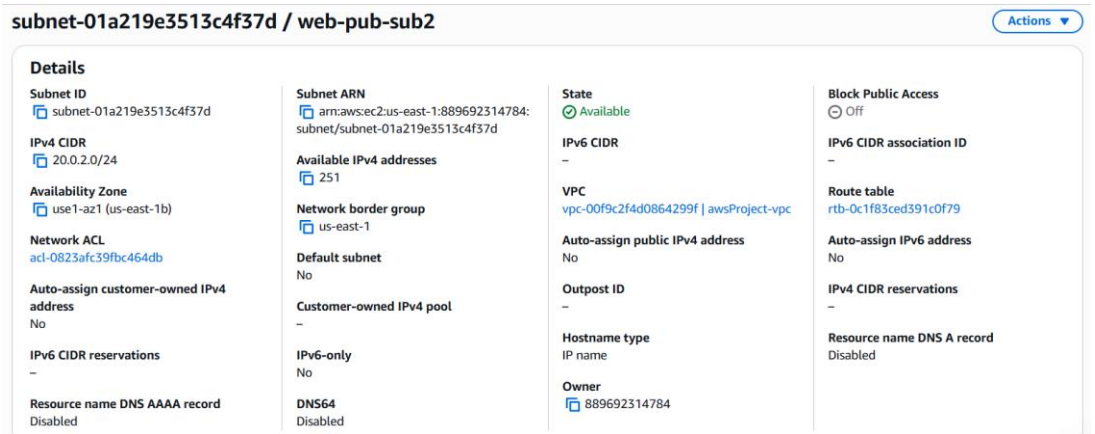
- Create a VPC named **awsProject-vpc** with a CIDR block of **20.0.0.0/20.**



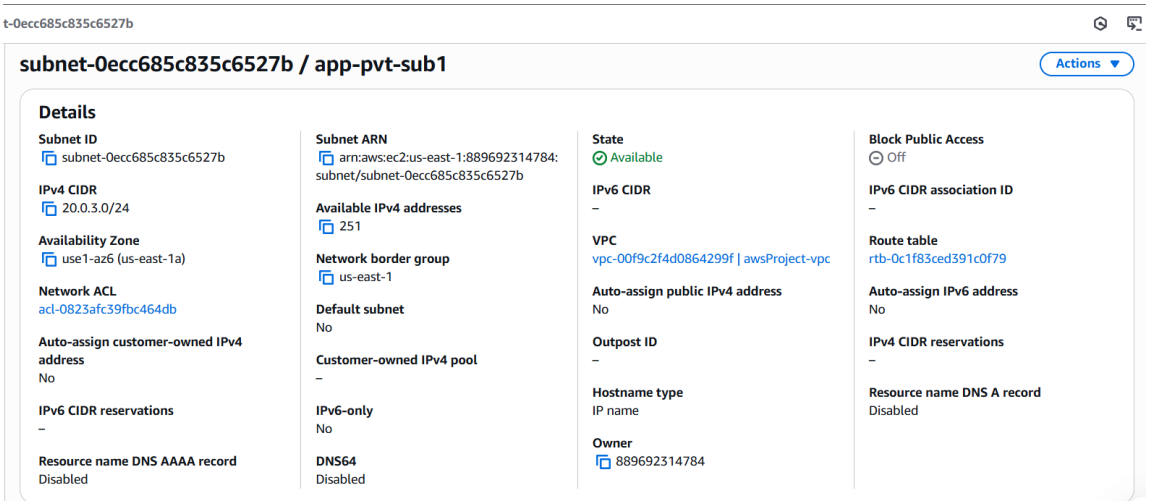
- Create 2 public subnets:
  - web-pub-sub1 with CIDR block **20.0.1.0/24 in us-east-1a**



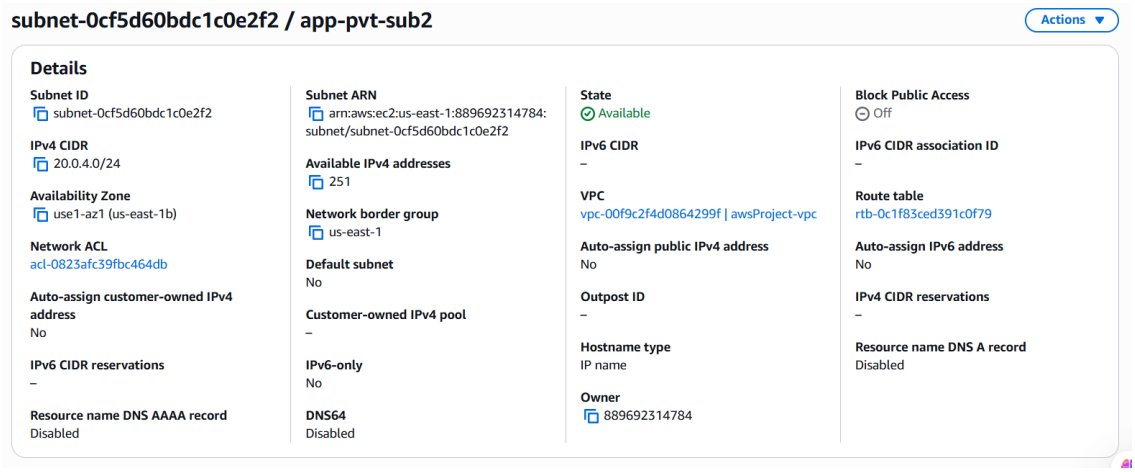
  - web-pub-sub2 with CIDR block **20.0.2.0/24 in us-east-1b**



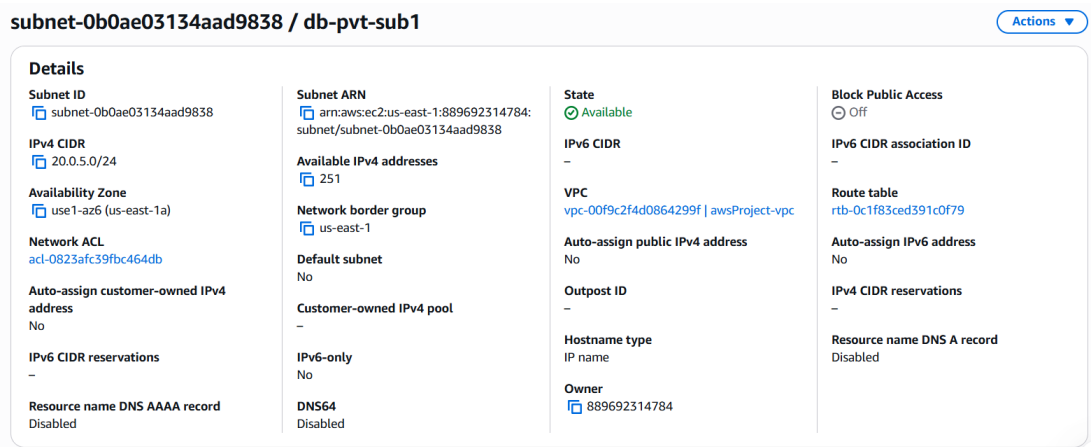
- Create **2 private subnets** for applications:
  - app-pvt-sub1 with **CIDR block 20.0.3.0/24 in us-east-1a**



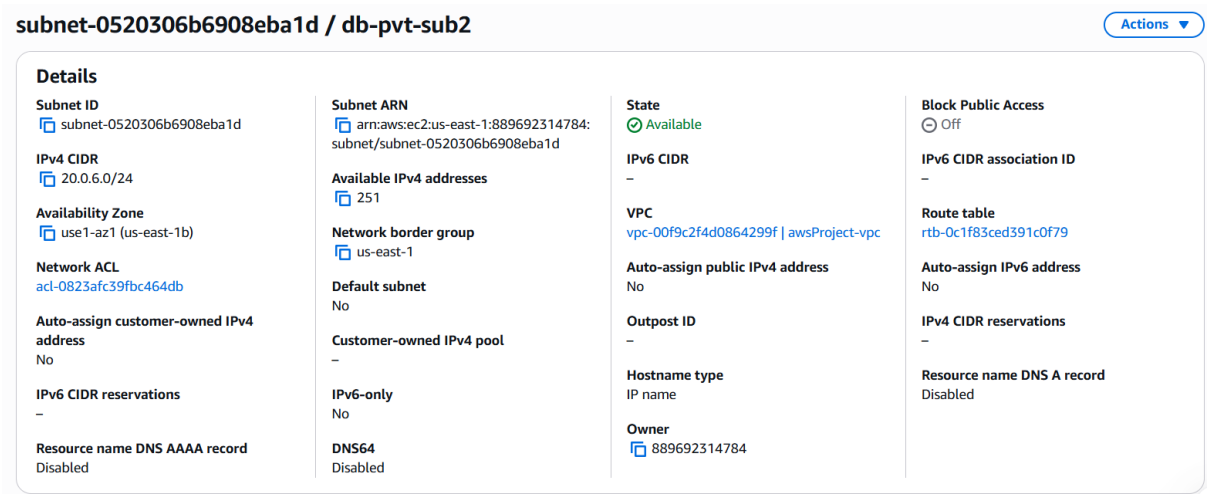
  - app-pvt-sub2 with **CIDR block 20.0.4.0/24 in us-east-1b**



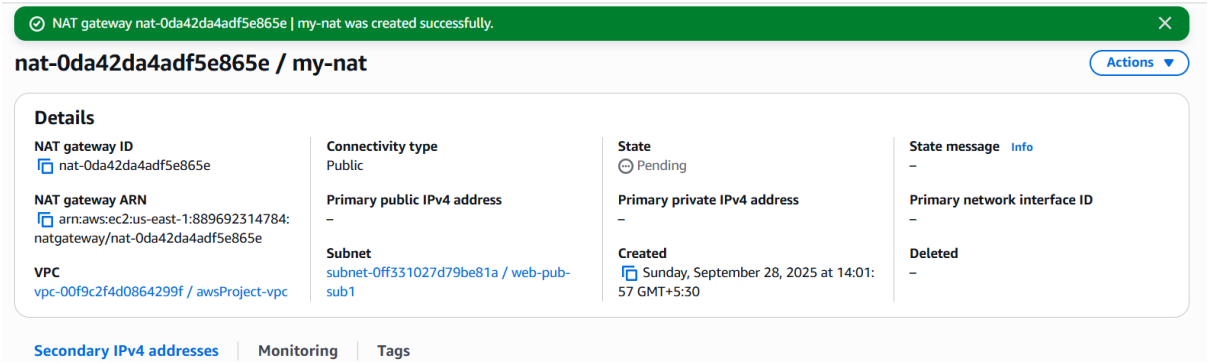
- Create **2 private subnets for databases**:
  - db-pvt-sub1 with **CIDR block 20.0.5.0/24 in us-east-1a**



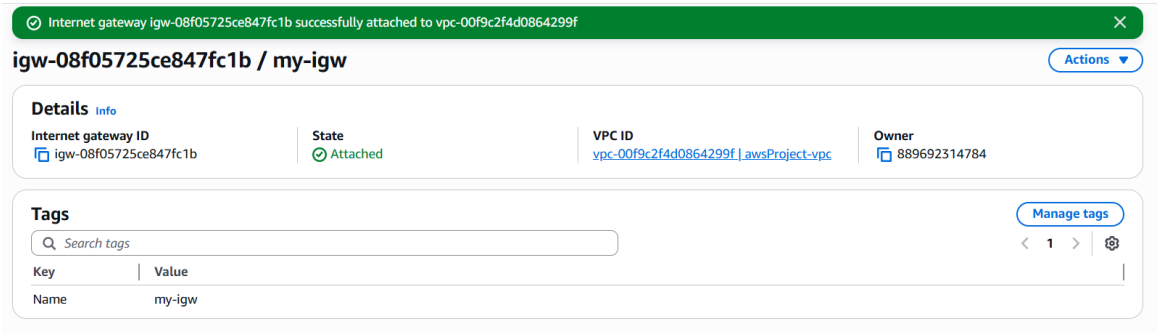
- db-pvt-sub2 with **CIDR block 20.0.6.0/24 in us-east-1b**



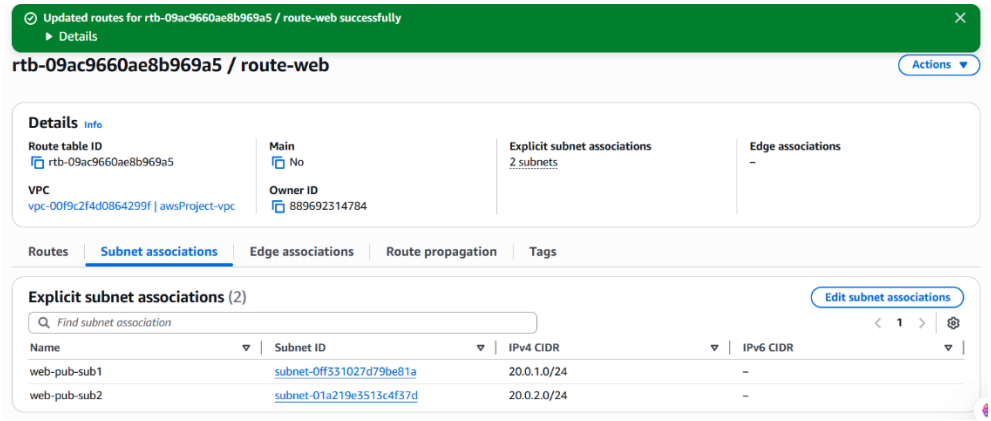
- Create a **NAT Gateway named my-nat**, provide the subnet as web-pub-sub1, **and allocate an Elastic IP**.



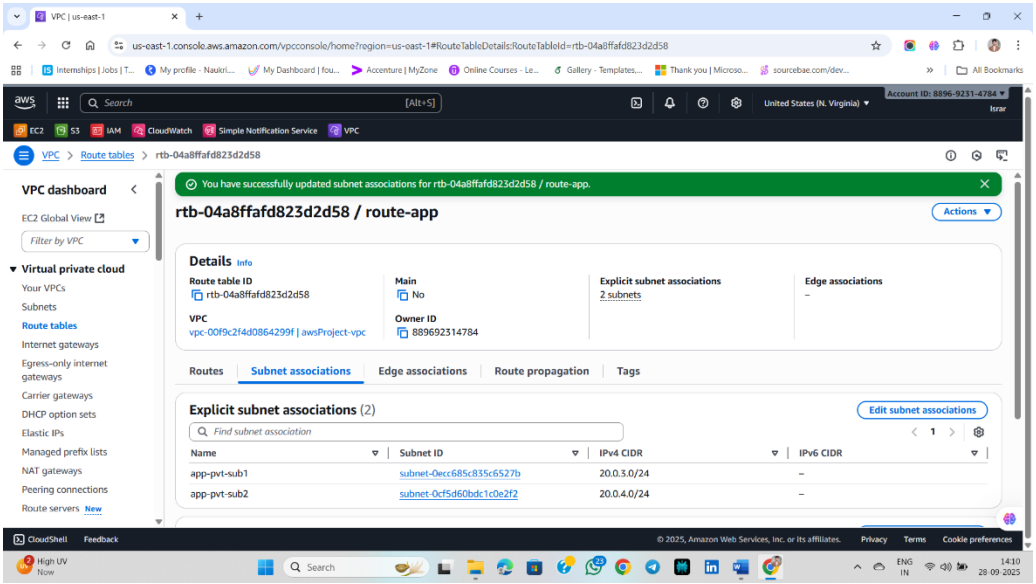
- Create an **Internet Gateway named my-igw** and **attach it to the VPC awsProject-vpc**.



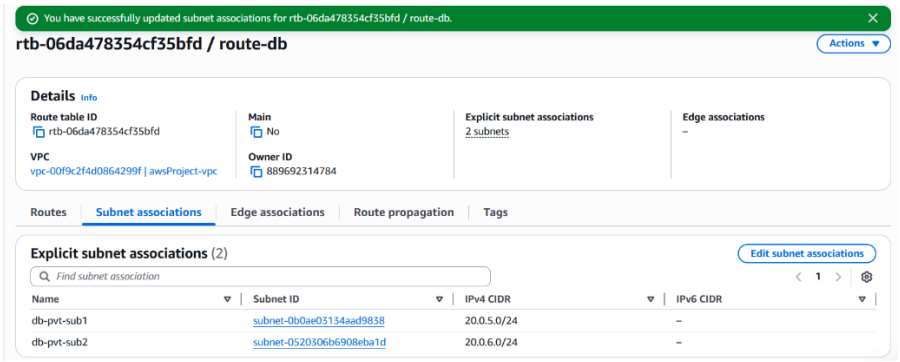
- Create **3 route tables** named **route-web, route-app, and route-db**.
- Associate the subnets with the route tables as follows:
  - **Select** route-web and **associate it with the subnets** web-pub-sub1 **and** web-pubsub2.



  - **Select** route-app and **associate it with the subnets** app-pvt-sub1 **and** app-pvtsub2.

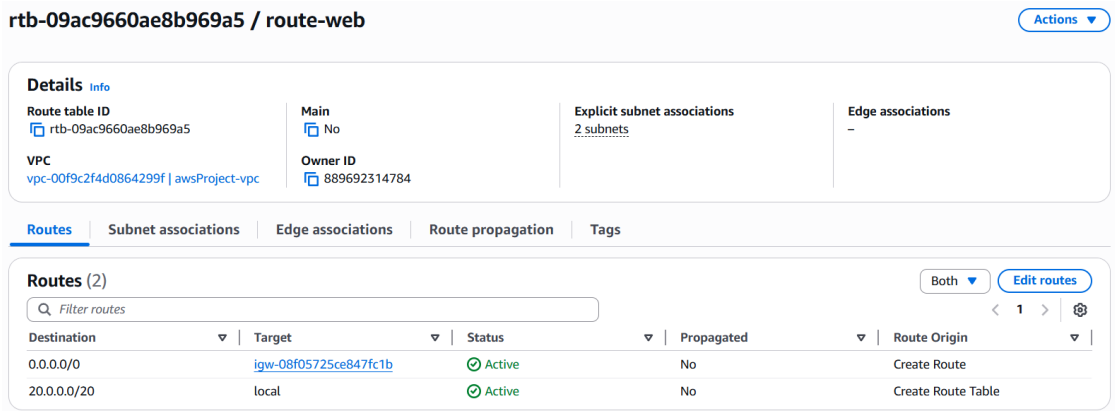


  - **Select** route-db and **associate it with the subnets** db-pvt-sub1 **and** db-pvt-sub2.

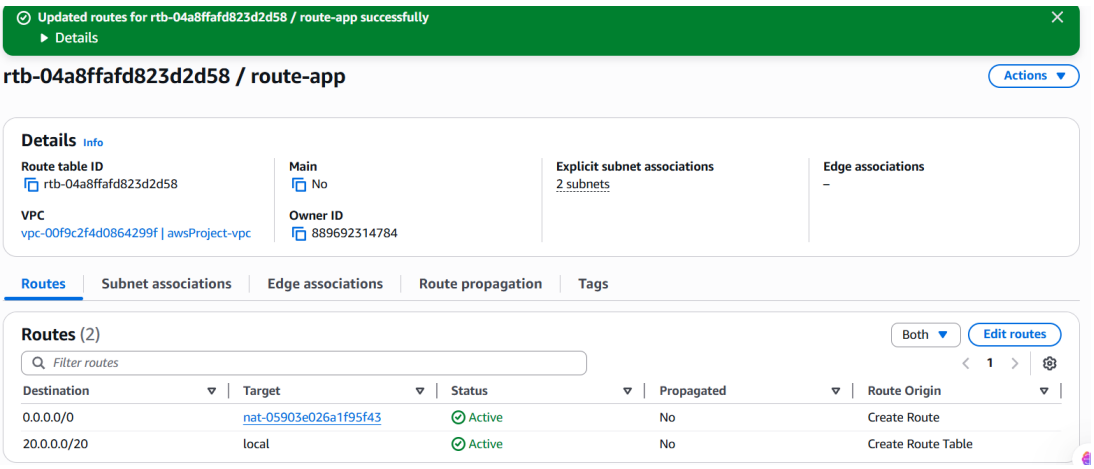


Now add the routes

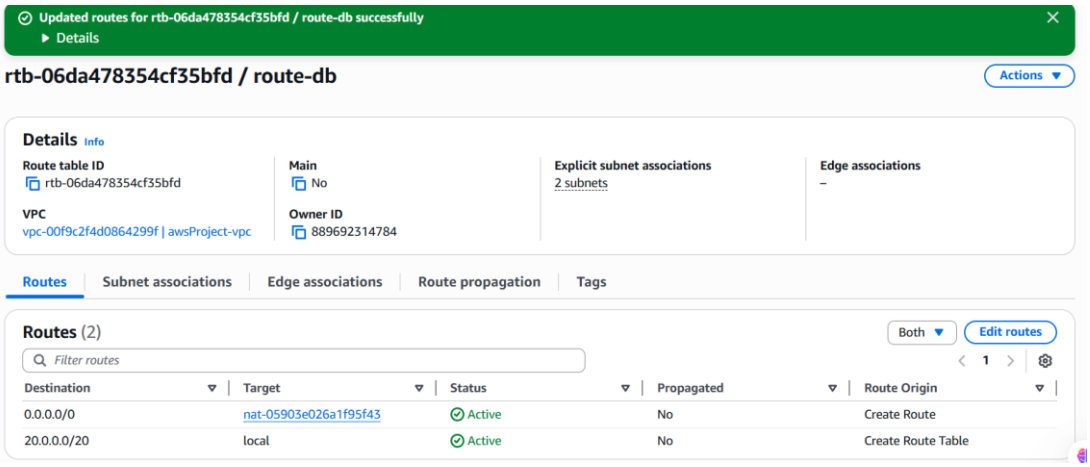
Select route-web, go to routes and click edit routes, add route select the Destination as 0.0.0.0/0, Target Internet Gateway(my-igw) and click on save changes

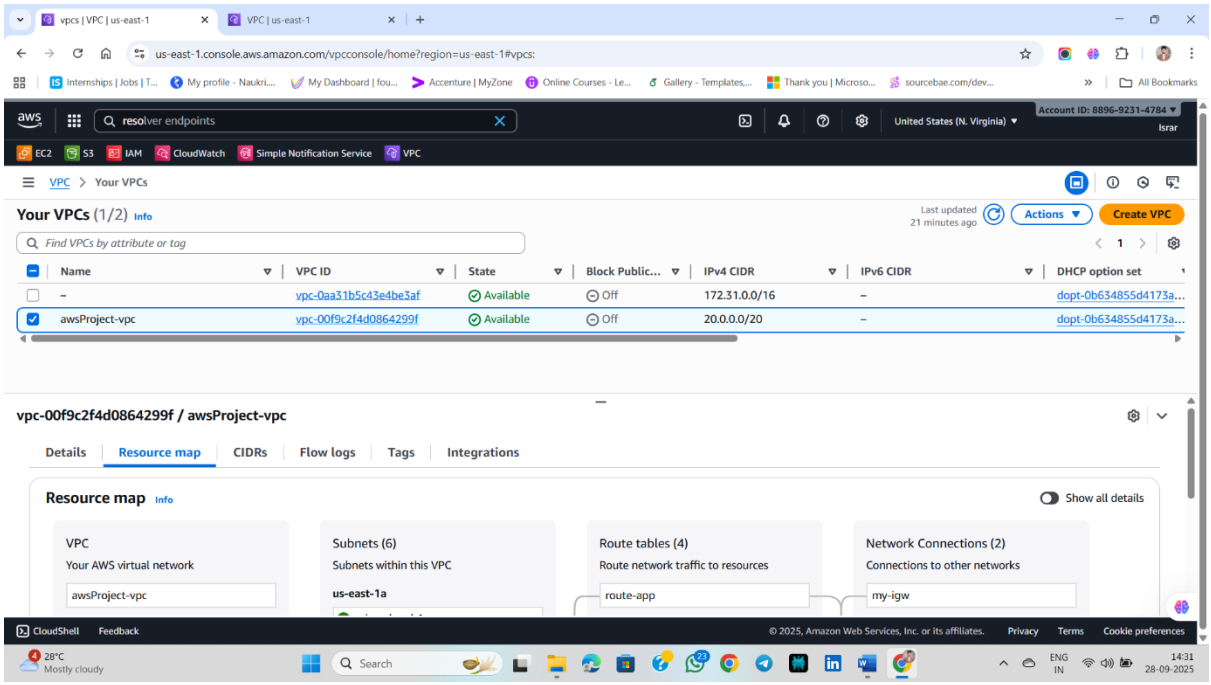


Select route-app, go to routes and click edit routes, add route select the Destination as 0.0.0.0/0, Target NAT Gateway(my-nat) and click on save changes



Select route-db, go to routes and click edit routes, add route select the Destination as 0.0.0.0/0, Target NAT Gateway(my-nat) and click on save changes

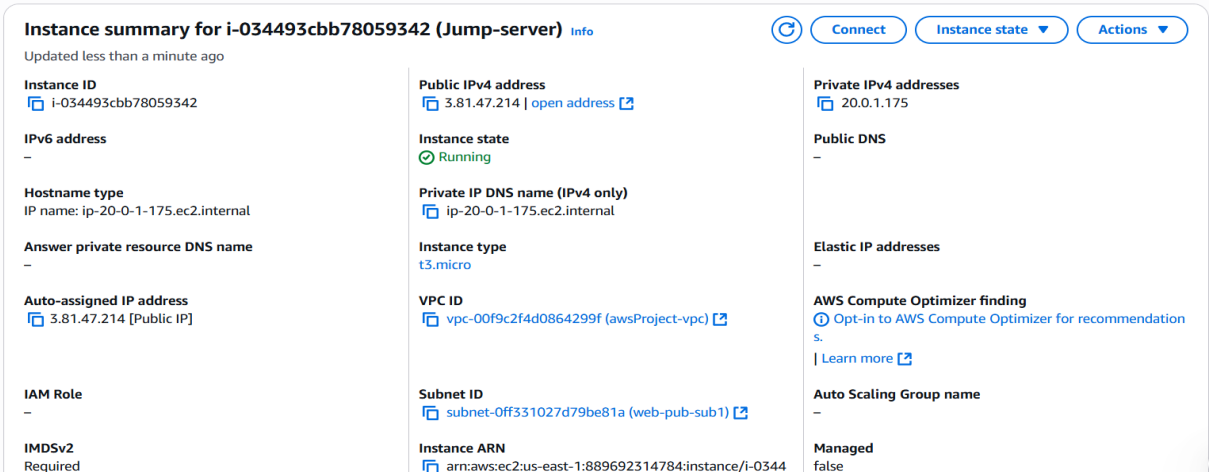




Part 2

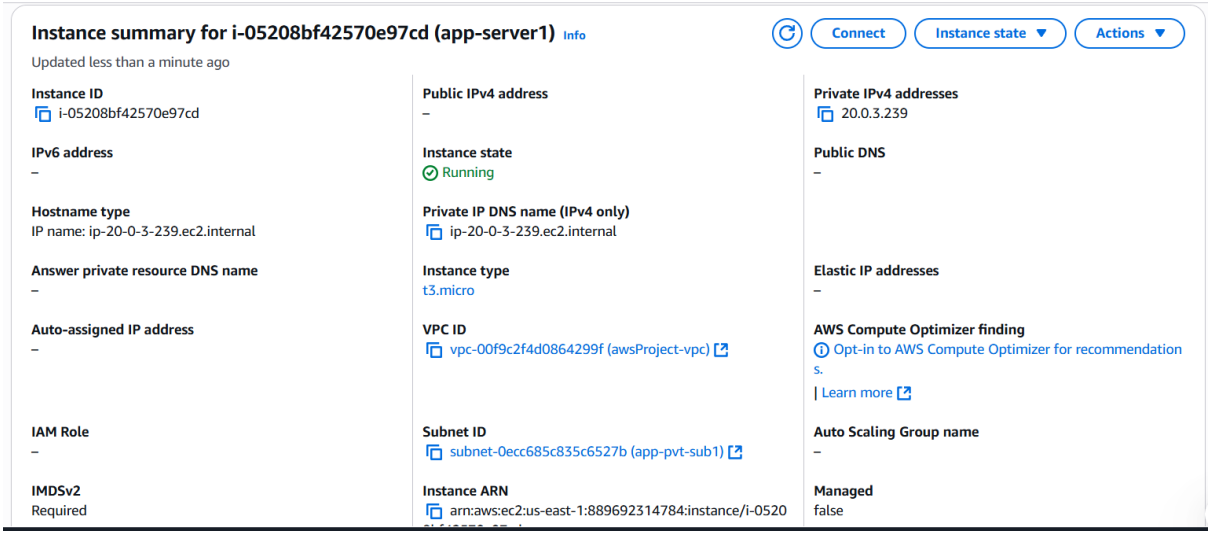
Create 3 EC2 instances in which 1 in public subnet with publicIP enable which acts as Jump server or bastion host and 2 private subnet with publicIP disable in which we will download phpMyAdmin and apache server

1) Launch an Instance with name jump-server, AMI (Amazon Linux), Instance type (t2.micro), Create a new keypair as (projectkey) , click on edit button on right side of Network settings select vpc(awsProject-vpc), Subnet (web-pub-sub1), Auto-assign IP (Enable), Create security group [Security group name (jump-sg)], allow port SSH (22) and HTTP (80) now Launch instance

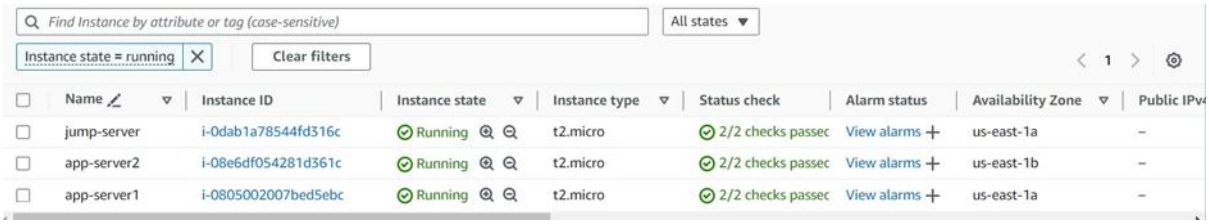


Create a security group as alb-sg and allow port HTTP (80)

2) Launch an Instance with name app-server1, AMI (Amazon Linux), Instance type (t2.micro), Selectthe keypair as (projectkey) , click on edit button on right side of Network settings select vpc (awsProject-vpc), Subnet (app-pvt-sub1), Auto-assign IP (Disable), Create security group [Security group name (app-sg)], allow port SSH (22) and Click on Add security group rule select Type (All traffic) Source type (Custom) and Source (here select your [alb-sg] you created) now Launch instance



3) Launch an Instance with name app-server2, AMI (Amazon Linux), Instance type (t2.micro), Select the keypair as (projectkey) , click on edit button on right side of Network settings select vpc (awsProject-vpc), Subnet (app-pvt-sub2), Auto-assign IP (Disable), Select existing security group as (app-sg), now Launch instance



Copy the private key in your jump-server instance

>> scp -i <your key> <yourkey path> ec2-user@<private ip>:~

>> chmod 400 <your key>

Now you can ssh into your app-server1 and app-server2 instance

ssh into your jump-server and your **private key** should be present in home directory

```
[ec2-user@ip-20-0-1-175 ~]$ ls
projectkey.pem
[ec2-user@ip-20-0-1-175 ~]$ sudo chmod 400 projectkey.pem
[ec2-user@ip-20-0-1-175 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.3.239
The authenticity of host '20.0.3.239 (20.0.3.239)' can't be established.
ED25519 key fingerprint is SHA256:Z44eQdSIZQi9a91jBdHPw+Yi5ea04AUWQyY0aDoCoiA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.0.3.239' (ED25519) to the list of known hosts.
Load key "projectkey.pem": Permission denied
ec2-user@20.0.3.239: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-20-0-1-175 ~]$ sudo chmod 700 projectkey.pem
[ec2-user@ip-20-0-1-175 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.3.239
Load key "projectkey.pem": Permission denied
ec2-user@20.0.3.239: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-20-0-1-175 ~]$ sudo chmod 777 projectkey.pem
[ec2-user@ip-20-0-1-175 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.3.239
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
```

Now ssh into app-server1 and run the below commands

Update the system

>> sudo yum update -y

```
[ec2-user@ip-20-0-3-239 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository                    152 kB/s |  23 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

Install PHP 8.2

>> sudo dnf install php8.2

```
[ec2-user@ip-20-0-3-239 ~]$ sudo dnf install php8.2
Last metadata expiration check: 0:00:51 ago on Sun Sep 28 09:42:35 2025.
Dependencies resolved.
================================================================================
 Package                 Architecture  Version                  Repository  Size
================================================================================
Installing:
 php8.2                  x86_64        8.2.29-1.amzn2023.0.1    amazonlinux 9.7 k
Installing dependencies:
 apr                     x86_64        1.7.5-1.amzn2023.0.4     amazonlinux 129 k
 apr-util                x86_64        1.6.3-1.amzn2023.0.1     amazonlinux  98 k
Installed:
  apr-1.7.5-1.amzn2023.0.4.x86_64
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.65-1.amzn2023.0.1.x86_64
  httpd-core-2.4.65-1.amzn2023.0.1.x86_64
  httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
  httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  libsodium-1.0.19-4.amzn2023.x86_64
  libxslt-1.1.43-1.amzn2023.0.2.x86_64
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  mod_lua-2.4.65-1.amzn2023.0.1.x86_64
  nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch
  php8.2-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-cli-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-common-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-opcache-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-pdo-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-process-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-sodium-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

>> sudo yum install php8.2-mysqlnd

```
Installed:
  apr-1.7.5-1.amzn2023.0.4.x86_64
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.65-1.amzn2023.0.1.x86_64
  httpd-core-2.4.65-1.amzn2023.0.1.x86_64
  httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
  httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  libsodium-1.0.19-4.amzn2023.x86_64
  libxslt-1.1.43-1.amzn2023.0.2.x86_64
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_http2-2.0.27-1.amzn2023.0.3.x86_64
  mod_lua-2.4.65-1.amzn2023.0.1.x86_64
  nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch
  php8.2-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-cli-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-common-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-opcache-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-pdo-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-process-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-sodium-8.2.29-1.amzn2023.0.1.x86_64
  php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

Install Apache Web Server

>> sudo yum install -y httpd

```
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :
  Installing       : php8.2-mysqlnd-8.2.29-1.amzn2023.0.1.x86_64
  Running scriptlet: php8.2-mysqlnd-8.2.29-1.amzn2023.0.1.x86_64
  Verifying        : php8.2-mysqlnd-8.2.29-1.amzn2023.0.1.x86_64

Installed:
  php8.2-mysqlnd-8.2.29-1.amzn2023.0.1.x86_64

Complete!
[ec2-user@ip-20-0-3-239 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:05:44 ago on Sun Sep 28 09:42:35 2025.
Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$ sudo yum install -y httpd
Last metadata expiration check: 0:06:23 ago on Sun Sep 28 09:42:35 2025.
Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

Start and Enable Apache

>> sudo systemctl start httpd

>> sudo systemctl enable httpd

>> sudo systemctl is-enabled httpd

```
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-3-239 ~]$
```

```
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.ser
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl is-enable httpd
Unknown command verb is-enable.
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-20-0-3-239 ~]$
```

Add User to Apache Group

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

```
[ec2-user@ip-20-0-3-239 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-3-239 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
chmod: cannot access 'var/www': No such file or directory
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-3-239 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-3-239 ~]$
```

Change Ownership and Permissions for Web Directory

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

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

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

```
[ec2-user@ip-20-0-3-239 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-3-239 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
chmod: cannot access 'var/www': No such file or directory
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {}\;
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec
find: missing argument to `-exec'
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-3-239 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-3-239 ~]$
```

Install Additional PHP Modules

>> sudo yum install php-mbstring php-xml -y

```
[ec2-user@ip-20-0-3-239 ~]$ sudo yum install php-mbstring php-xml -y
Last metadata expiration check: 0:17:59 ago on Sun Sep 28 09:42:35 2025.
Package php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

>> sudo yum install php-fpm

```
[ec2-user@ip-20-0-3-239 ~]$ sudo yum install php-mbstring php-xml -y
Last metadata expiration check: 0:17:59 ago on Sun Sep 28 09:42:35 2025.
Package php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$ sudo yum install php-fpm
Last metadata expiration check: 0:18:26 ago on Sun Sep 28 09:42:35 2025.
Package php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-239 ~]$
```

Restart Apache and PHP-FPM

>> sudo systemctl restart httpd

>> sudo systemctl restart php-fpm

```
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-3-239 ~]$ cd /var/www/htl
-bash: cd: /var/www/htl: No such file or directory
[ec2-user@ip-20-0-3-239 ~]$ cd /var/www/html
[ec2-user@ip-20-0-3-239 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-09-28 10:03:12--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 79.127.206.234, 79.127.206.207, 2a02:6ea0:c400::54, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|79.127.206.234|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz [following]
--2025-09-28 10:03:12--  https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 109.61.91.231, 109.61.91.198, 109.61.91.195, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.231|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13764534 (13M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

phpMyAdmin-latest-all-languages.tar.gz    100%[=====================================================================================>]  13.13M  --.-KB/s    in 0.04s

2025-09-28 10:03:13 (292 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]

[ec2-user@ip-20-0-3-239 html]$
```

Download and Set Up phpMyAdmin

>> cd /var/www/html

>> wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz

>> mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1

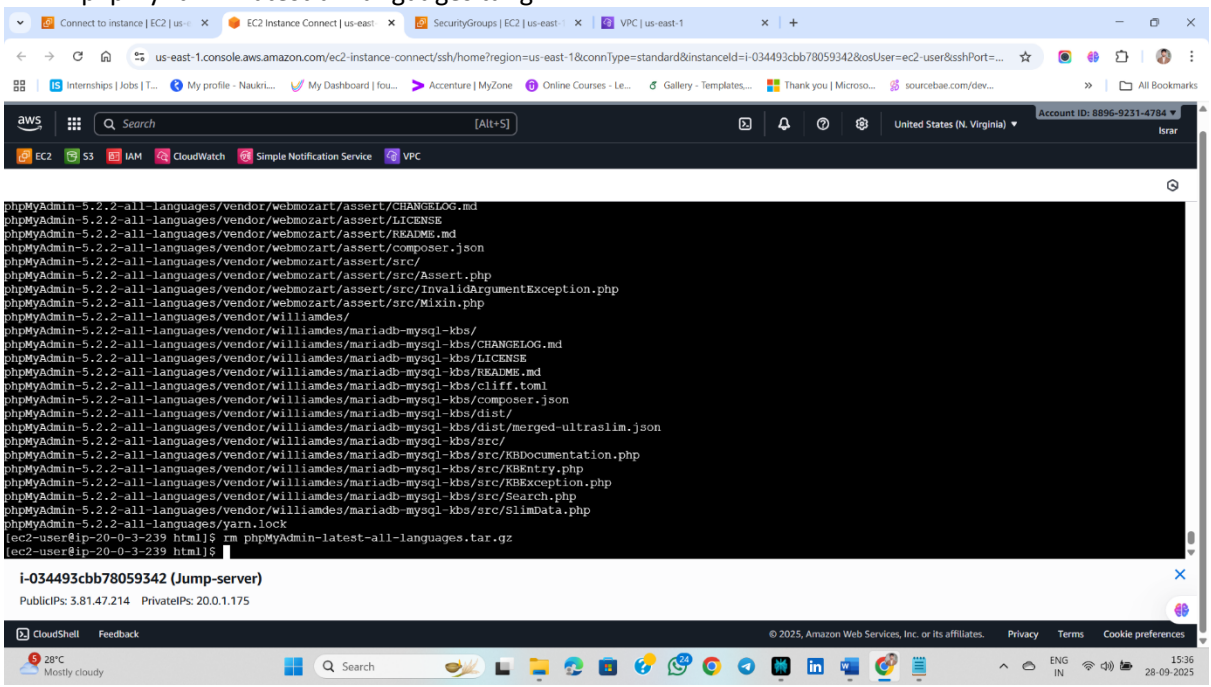
```
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-3-239 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-3-239 ~]$ cd /var/www/htl
-bash: cd: /var/www/htl: No such file or directory
[ec2-user@ip-20-0-3-239 ~]$ cd /var/www/html
[ec2-user@ip-20-0-3-239 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-09-28 10:03:12--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 79.127.206.234, 79.127.206.207, 2a02:6ea0:c400::54, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|79.127.206.234|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz [following]
--2025-09-28 10:03:12--  https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 109.61.91.231, 109.61.91.198, 109.61.91.195, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.231|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13764534 (13M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

phpMyAdmin-latest-all-languages.tar.gz    100%[=====================================================================================>]  13.13M  --.-KB/s    in 0.04s

2025-09-28 10:03:13 (292 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]

[ec2-user@ip-20-0-3-239 html]$
```

>> rm phpMyAdmin-latest-all-languages.tar.gz



Create a Test Page and Test the Server

>> echo "PHP server 1" > /var/www/html/index.html

>> curl http://localhost

```
cgi-bin/ html/
[ec2-user@ip-20-0-3-239 html]$ echo "PHP server 1 " > /var/www/html/index.html
[ec2-user@ip-20-0-3-239 html]$ curl http://localhost
PHP server 1
[ec2-user@ip-20-0-3-239 html]$
```

Now ssh into app-server2 and run the below commands

```
[ec2-user@ip-20-0-3-239 ~]$ sudo nano projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 777 projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.4.125
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-20-0-4-125 ~]$
```

Update the system

>> sudo yum update -y

Install PHP 8.2

>> sudo dnf install php8.2

>> sudo yum install php8.2-mysqlnd

```
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 777 projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.4.125
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-20-0-4-125 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository                    189 kB/s |  23 kB     00:00
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-125 ~]$ sudo dnf install php8.2
Last metadata expiration check: 0:01:08 ago on Sun Sep 28 10:11:59 2025.
Dependencies resolved.
================================================================================
 Package                 Architecture          Version                  Repository                Size
================================================================================
Installing:
 php8.2                  x86_64                8.2.29-1.amzn2023.0.1    amazonlinux               9.7 k
Installing dependencies:
 apr                     x86_64                1.7.5-1.amzn2023.0.4     amazonlinux               129 k
```

Install Apache Web Server

>> sudo yum install -y httpd

Start and Enable Apache

>> sudo systemctl start httpd

>> sudo systemctl enable httpd

>> sudo systemctl is-enabled httpd

```
Complete!
[ec2-user@ip-20-0-4-125 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:03:25 ago on Sun Sep 28 10:11:59 2025.
Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-125 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-4-125 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-4-125 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-20-0-4-125 ~]$
```

Add User to Apache Group

>> sudo usermod -a -G apache ec2-user
Change Ownership and Permissions for Web Directory
>> sudo chown -R ec2-user:apache /var/www
>> sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
>> find /var/www -type f -exec sudo chmod 0664 {} \;

```
[ec2-user@ip-20-0-4-125 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-4-125 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-4-125 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-4-125 ~]$ find /var/www -type f -exec sudo chmod 0664{} \;
[ec2-user@ip-20-0-4-125 ~]$
```

Install Additional PHP Modules
>> sudo yum install php-mbstring php-xml -y
>> sudo yum install php-fpm

```
[ec2-user@ip-20-0-4-125 ~]$ sudo yum install php-mbstring php-xml -y
Last metadata expiration check: 0:09:15 ago on Sun Sep 28 10:11:59 2025.
Package php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-125 ~]$ sudo yum install php-fpm'
> ^C
[ec2-user@ip-20-0-4-125 ~]$ sudo yum install php-fpm
Last metadata expiration check: 0:09:41 ago on Sun Sep 28 10:11:59 2025.
Package php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-125 ~]$
```

Restart Apache and PHP-FPM
>> sudo systemctl restart httpd
>> sudo systemctl restart php-fpm

```
[ec2-user@ip-20-0-4-125 ~]$ sudo systemctl restart httpd
s[ec2-user@ip-20-0-4-125 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-4-125 ~]$
```

Download and Set Up phpMyAdmin
>> cd /var/www/html
>> wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
>> mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1

```
[ec2-user@ip-20-0-4-125 ~]$ cd /var/www/html
[ec2-user@ip-20-0-4-125 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-09-28 10:24:05--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 79.127.206.208, 79.127.206.234, 2a02:6ea0:c400::53, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|79.127.206.208|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz [following]
--2025-09-28 10:24:06--  https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 109.61.91.194, 109.61.91.198, 109.61.91.230, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.194|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13764534 (13M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

phpMyAdmin-latest-all-languages.tar.gz    100%[=============================================================================>]  13.13M  --.-KB/s    in 0.05s

2025-09-28 10:24:06 (284 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]

[ec2-user@ip-20-0-4-125 html]$
```

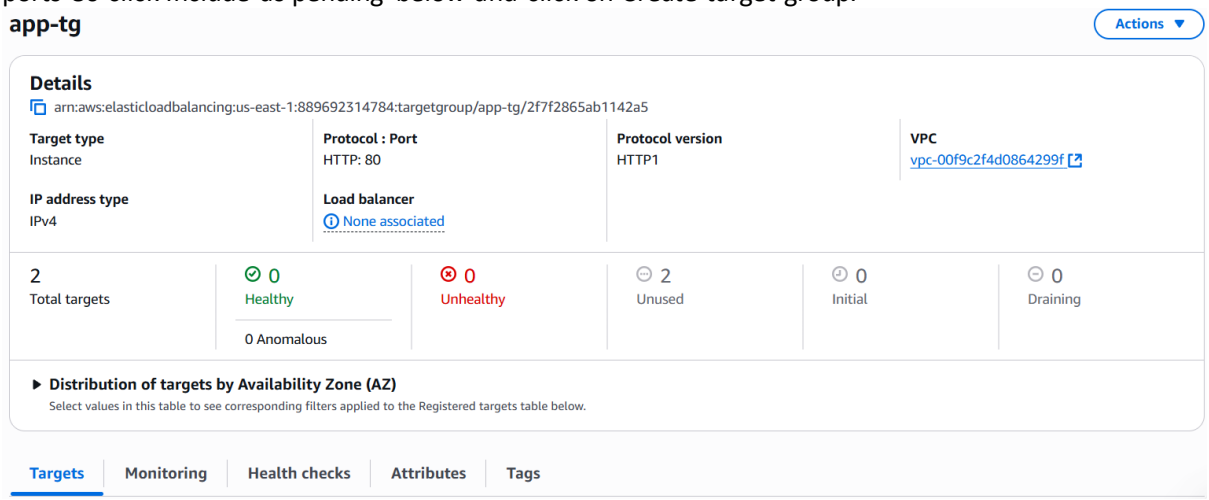
>> rm phpMyAdmin-latest-all-languages.tar.gz

```
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/SlimData.php
phpMyAdmin-5.2.2-all-languages/yarn.lock
[ec2-user@ip-20-0-4-125 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-4-125 html]$
```

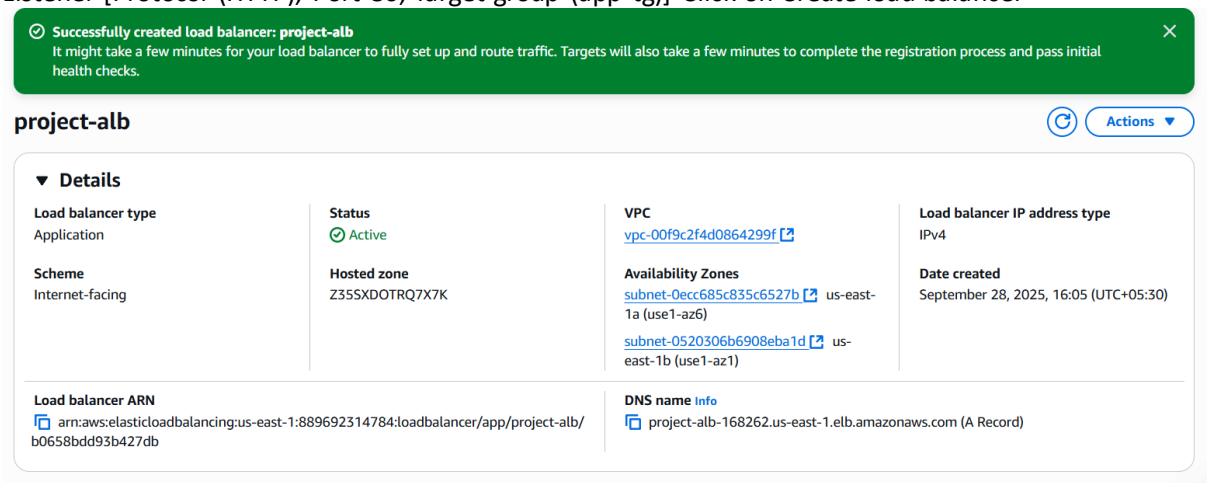
Create a Test Page and Test the Server
>> echo "PHP server 2" > /var/www/html/index.html
>> curl http://localhost

```
phpMyAdmin-5.2.2-all-languages/yarn.lock
[ec2-user@ip-20-0-4-125 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-4-125 html]$ echo "PHP server 2 " > /var/www/html/index.html
[ec2-user@ip-20-0-4-125 html]$ curl http://localhost
PHP server 2
[ec2-user@ip-20-0-4-125 html]$
```

As you have downloaded Apache and phpMyAdmin to access them, you now need to:
Create the Target groups:
Go to target groups, create a target group with target type (Instance), name (app-tg), Port HTTP 80, IP address type IPv4, VPC (awsProject-vpc), Protocol version HTTP1, Health checks as HTTP, Health check path / and click on next, now select the app-server1 and app-server2 in Available instances, ports 80 click Include as pending below and click on Create target group.



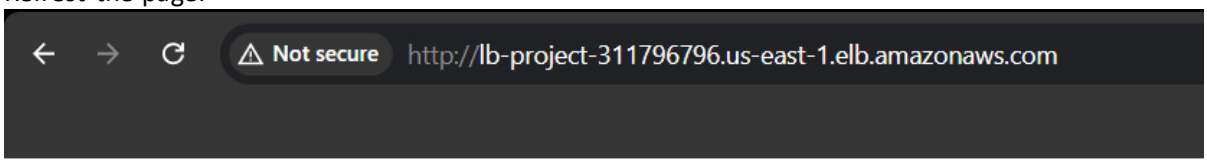
Now go to Load Balancer:
Create Application Load Balancer with name project-alb, Scheme (Internet-facing), Load balancer IP (IPv4), VPC (awsProject-vpc), Mappings us-east-a1, us-east-1b, select the security groups as (alb-sg), Listener [Protocol (HTTP), Port 80, Target group (app-tg)] Click on Create load balancer



Now take the load balancer DNS and past it on chrome browser you should see PHP Server 1 and when you refresh the page it should show PHP Server 2 that means your load balancing is working as expected

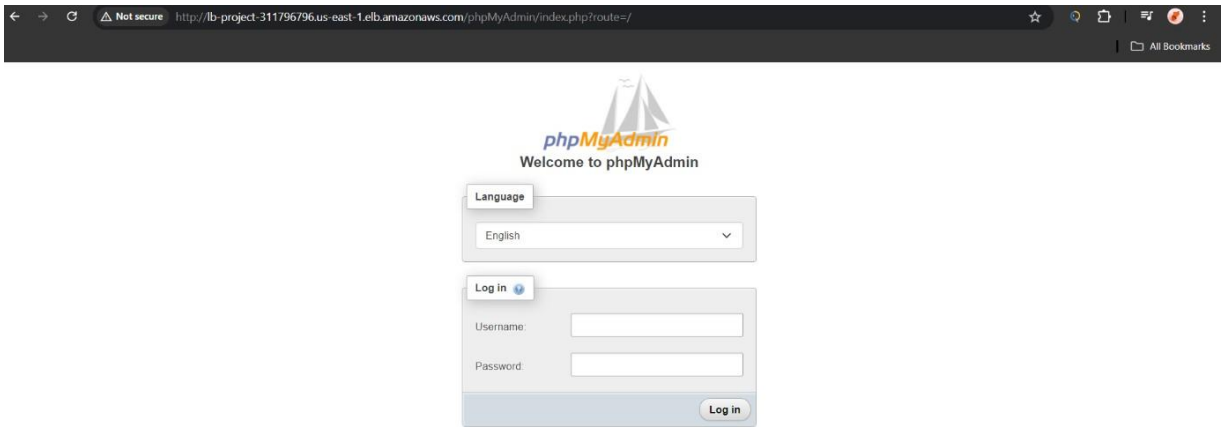
PHP server 1

Refrest the page:



PHP server 2

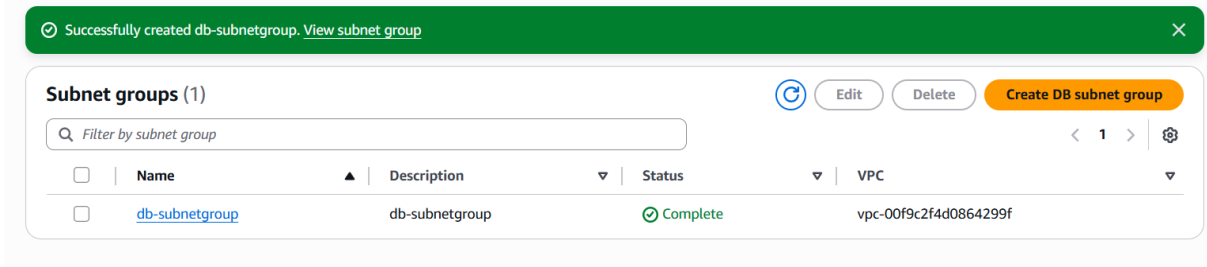
And when you write phpMyAdmin after your domain name it should navigate to



Part 3

Go to RDS

Select Subnet groups, Create a Subnet group and name it as db-subnetgroup, give description as db- subnetgroup, VPC (awsProject-vpc), Select the subnets that were created for db as (db-pvt-sub1 and db-pvt-sub2) and click on create



Now go to Databases,Click on Create database, select Standard create, **Engine options** (MySQL), **Templates** (Free tier), **Settings** DB instance identifier (mydb-project), Master username (admin), Credentials management (Self managed), Master password (admin), Confirm password (admin), **Instance configuration** DB instance class (db.t3.micro), **Storage** Click on **Storage autoscaling** and uncheck the (Enable storage autoscaling) **Connectivity** Compute resource (Don't connect to an EC2 compute resource), VPC (awsProject-vpc), DB subnet group (db-subnetgroup), Public access (No), VPC security group (Create new)… New VPC security group name (db-sg), Availability Zone (us-east-1a), Click on Create database.

Now go to security group (db-sg) click on edit inbound rules, Add rule, select Type (CustomIP),Port 3306, Source select the (app-sg) and remove the above MYSQL/Aurora rule and save changes

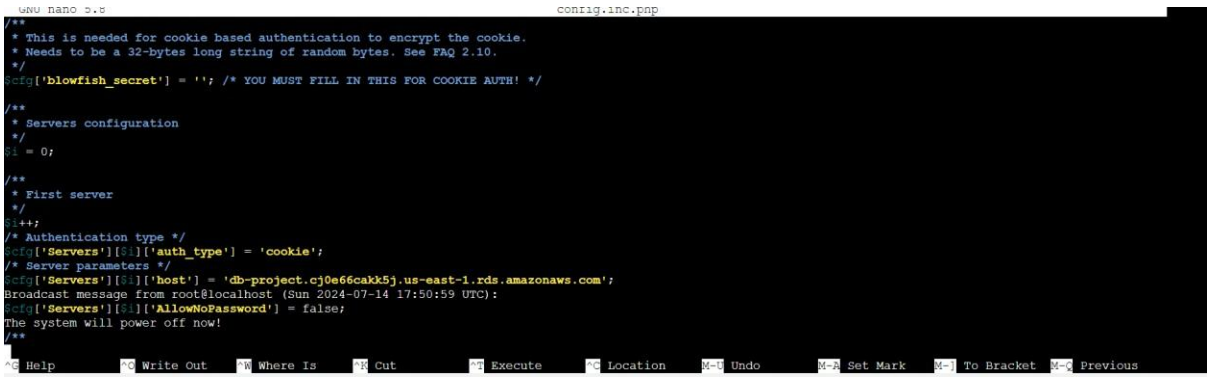
Now copy the Endpoint of RDS cluster and ssh into app-server1 instance and app-server2 instance

>>cd /var/www/html/phpMyAdmin

>> mv config.sample.inc.php config.inc.php

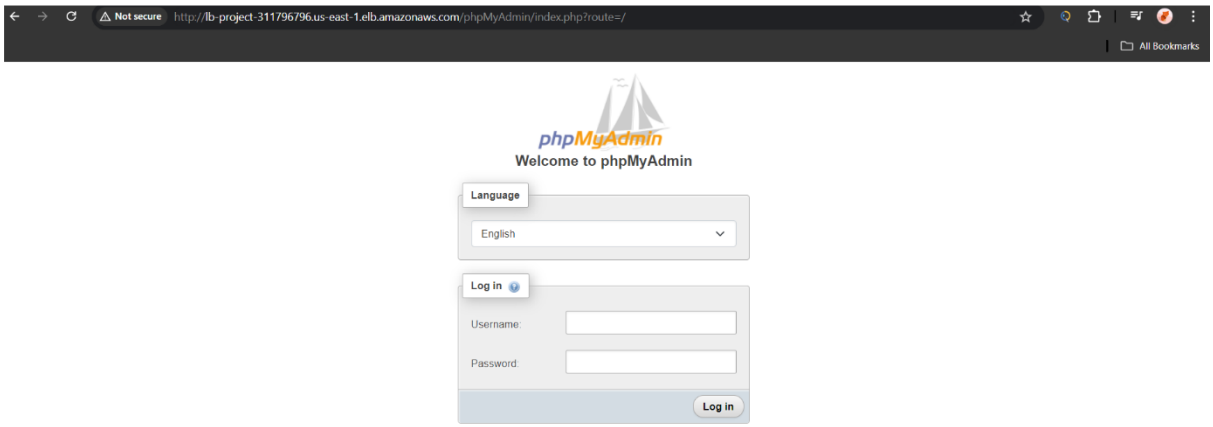
>> nano config.inc.php

And search for host and remove 'localhost' and replace it with cluster endpoint, now save the file



One last thing , go to target groups, select the (app-tg), go to Attributes and click on edit, scroll down and click on Turn on stickiness (Load balancer generated cookie) and save changes

Go to chrome paste the loadbalancer DNS with /myPhpAdmin at the last and give the username, and password you will be able to login.



Once you enter your username and password: