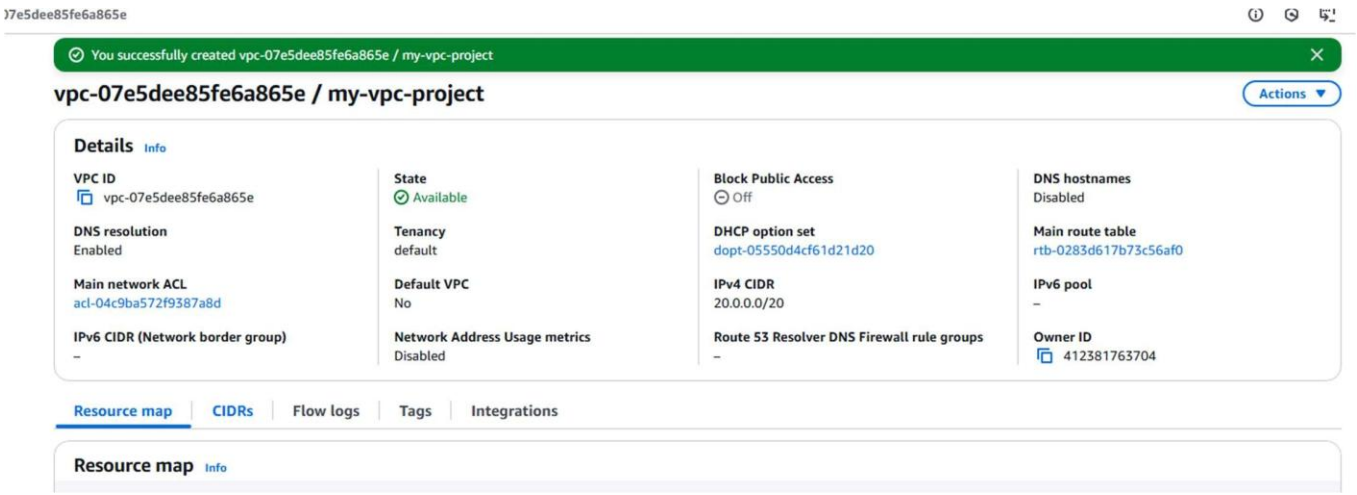
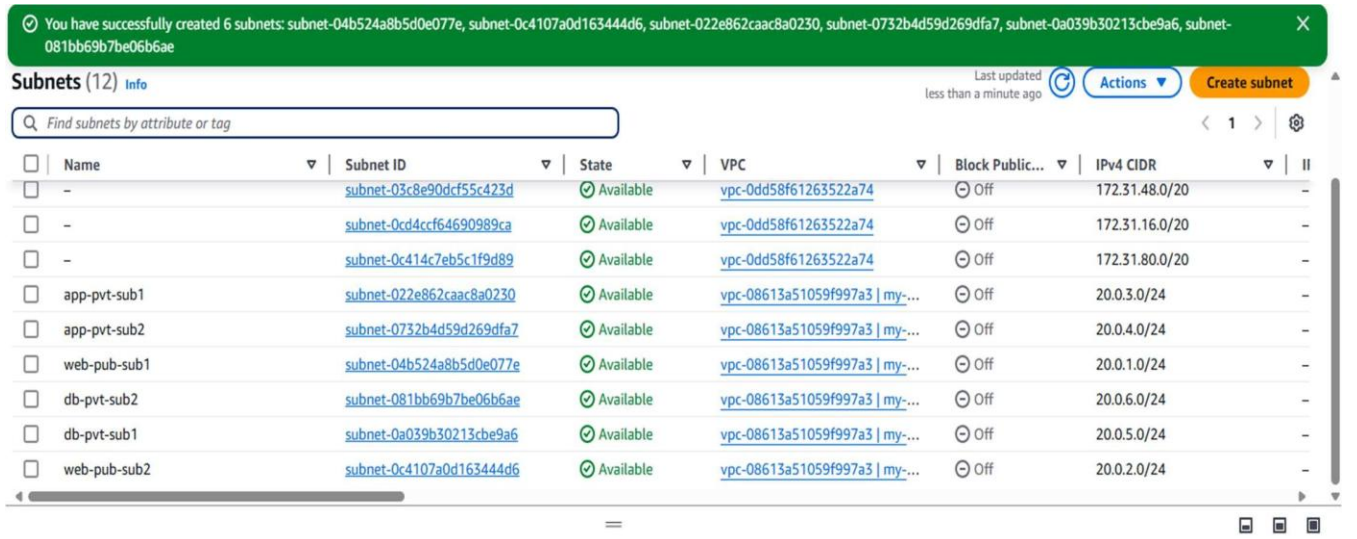
Name: Arafat Khaled

Part 1

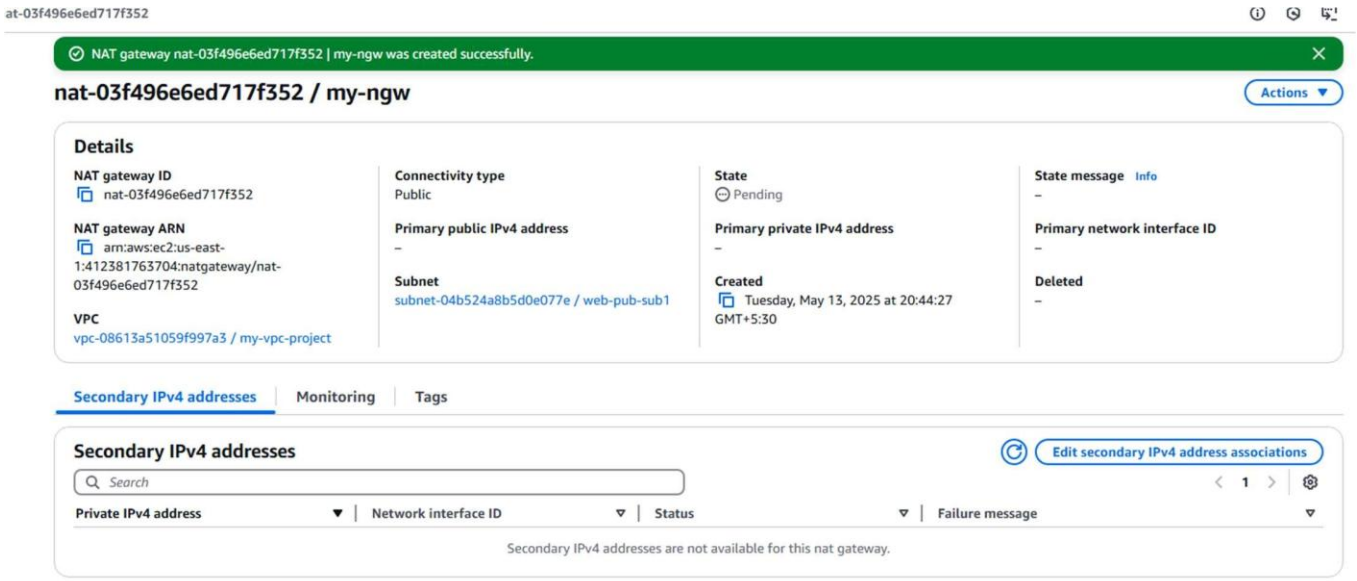
- Create a VPC named awsProject-vpc with a CIDR block of 20.0.0.0/20.



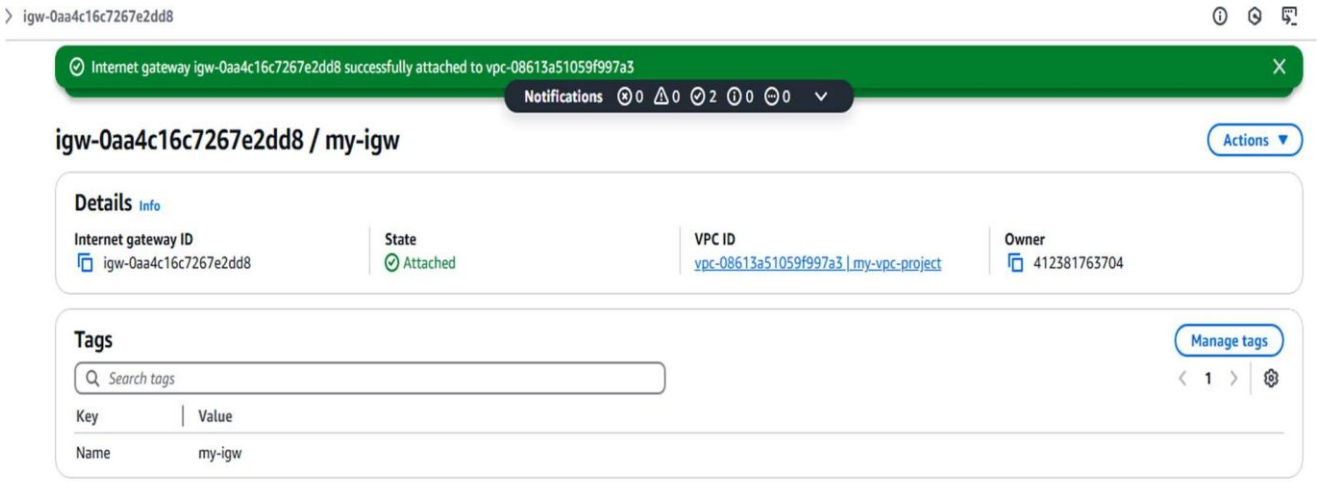
- Create 2 public subnets:
- web-pub-sub1 with CIDR block 20.0.1.0/24 in us-east-1a web-pub-sub2 with CIDR block 20.0.2.0/24 in us-east-1b
- Create 2 private subnets for applications:
- app-pvt-sub1 with CIDR block 20.0.3.0/24 in us-east-1a
- app-pvt-sub2 with CIDR block 20.0.4.0/24 in us-east-1b



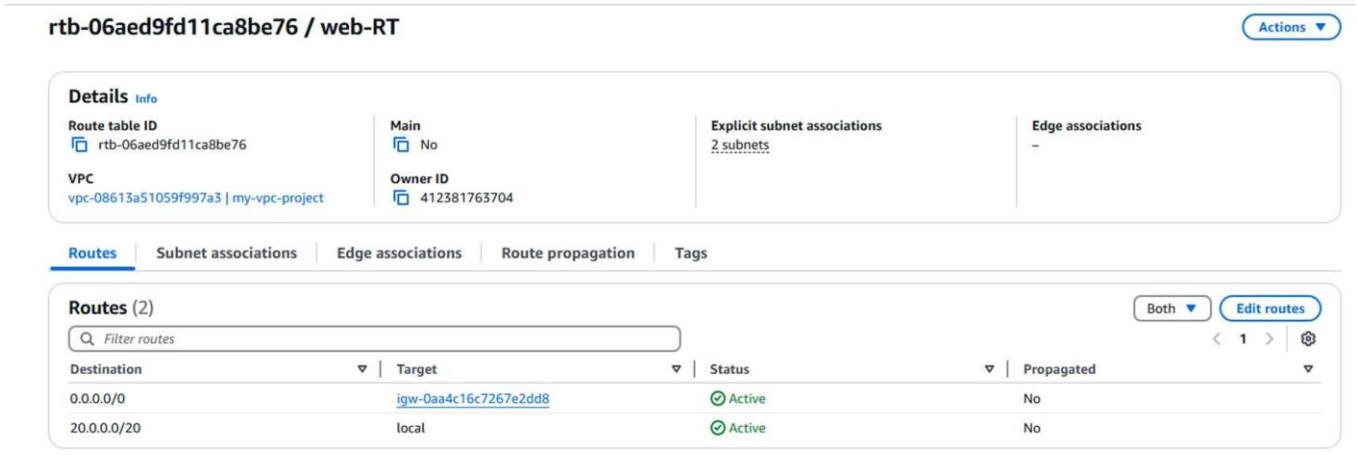
- Create 2 private subnets for databases:
- db-pvt-sub1 with CIDR block 20.0.5.0/24 in us-east-1a
- db-pvt-sub2 with CIDR block 20.0.6.0/24 in us-east-1b
- Create a NAT Gateway named my-nat, provide the subnet as web-pub-sub1, and allocate an Elastic IP.



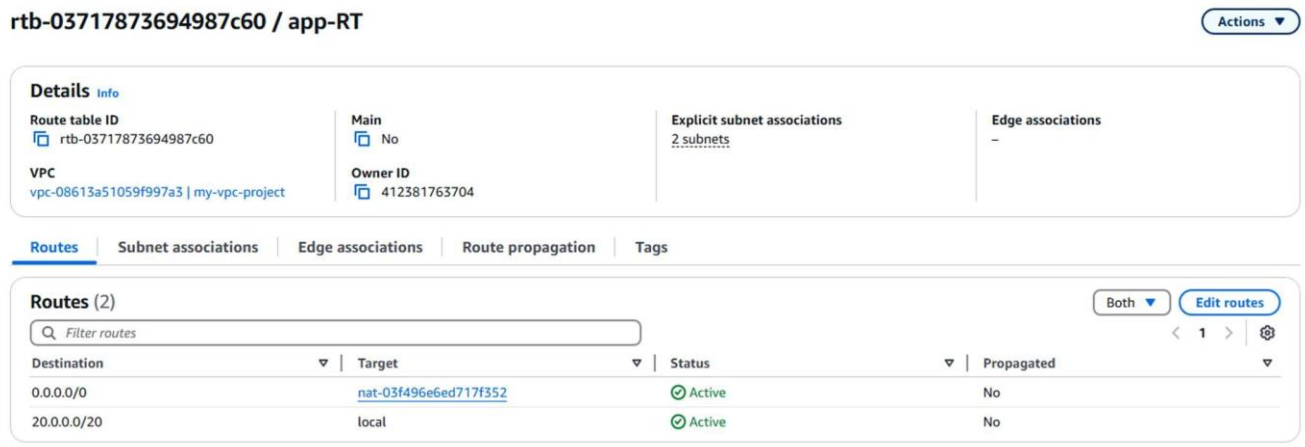
- Create an Internet Gateway named my-igw and attach it to the VPC awsProject-vpc.



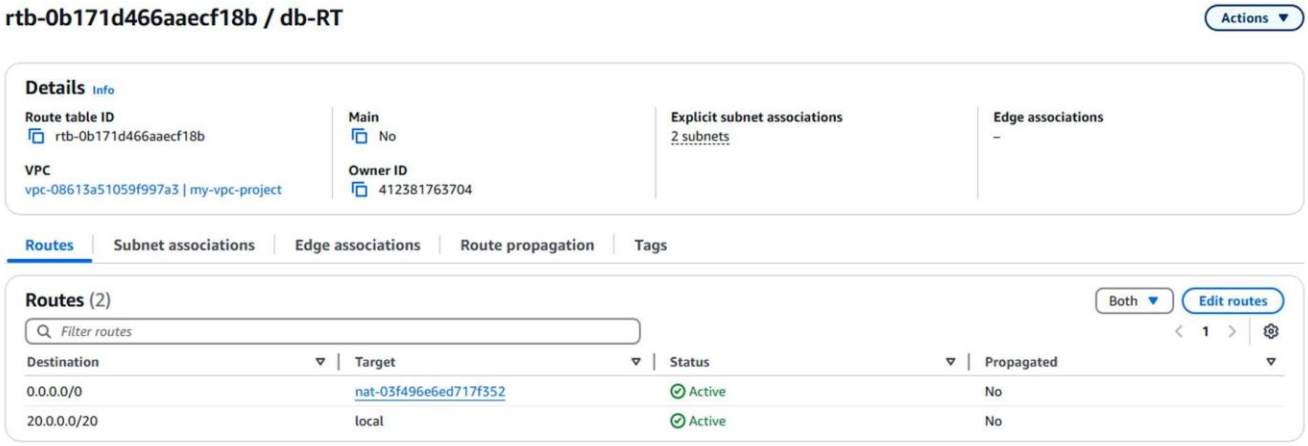
- Create 3 route tables named route-web, route-app, and route-db.
- Associate the subnets with the route tables as follows:
- Select route-web and associate it with the subnets web-pub-sub1 and web-pub- sub2.



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



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

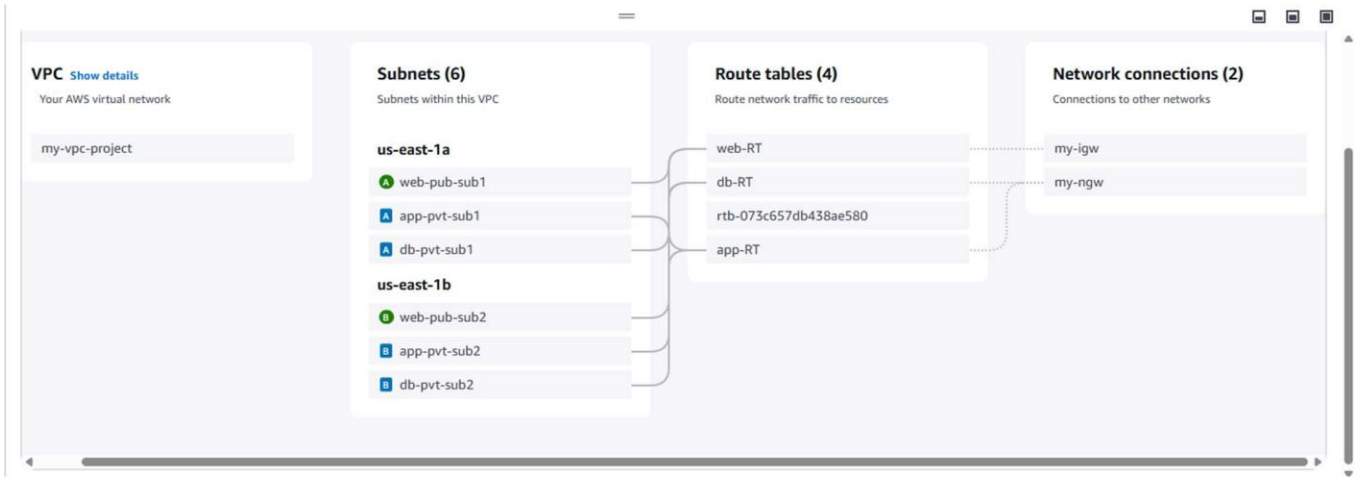


Now add the routes

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 changesPart 2



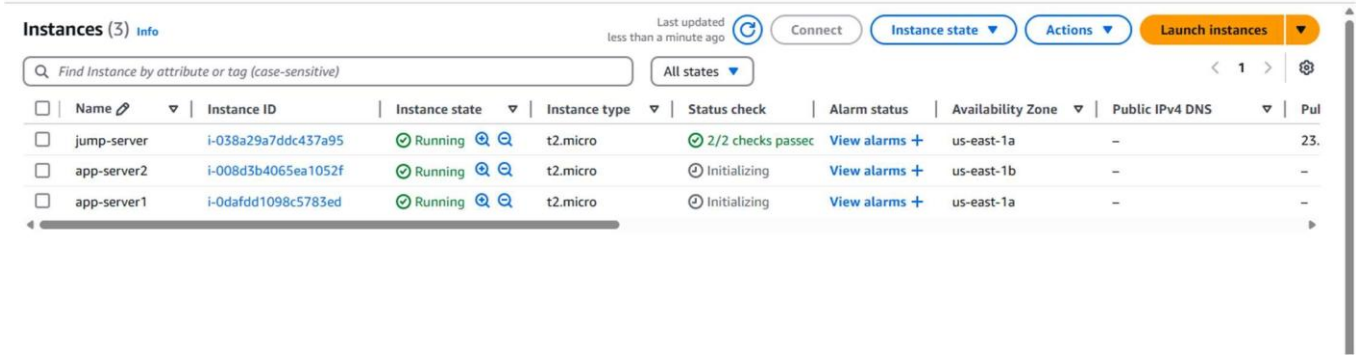
Create 3 EC2 instances in which 1 in public subnet with publicIP enable which acts as Jump server or bastion host and 2 private subnet with publicIP disable in which we will download phpMyAdmin and apache server

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), Select the 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 instanceCopy 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 Now ssh into app-server1 and run the below commands # Update the system

>> sudo yum update –y

```
[ec2-user@ip-20-0-1-247 ~]$ ll
total 4
-r-------- 1 ec2-user ec2-user 1679 May 12 17:19 project.pem
[ec2-user@ip-20-0-1-247 ~]$ ssh -i "project.pem" ec2-user@20.0.5.51
The authenticity of host '20.0.5.51 (20.0.5.51)' can't be established.
ECDSA key fingerprint is SHA256:ovD39OKnh5w7hUVabjcHkgO2bpZlSf4xVpdgoOM5Pjs.
ECDSA key fingerprint is MD5:94:b9:bd:1b:a3:bf:70:23:8c:9c:e2:bf:b1:dd:5a:06.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '20.0.5.51' (ECDSA) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

No packages needed for security; 1 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-20-0-5-51 ~]$ sudo yum update -y
```

```
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-5-51 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-20-0-5-51 ~]$
```

>> 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-20-0-5-51 ~]$  sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-5-51 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-5-51 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-5-51 ~]$
```

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

>> sudo yum install php-fpm

```
[ec2-user@ip-20-0-5-51 ~]$ sudo yum install php-fpm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package php-fpm.x86_64 0:5.4.16-46.amzn2.0.6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch          Version                  Repository       Size
================================================================================
Installing:
 php-fpm          x86_64        5.4.16-46.amzn2.0.6      amzn2-core       1.4 M

Transaction Summary
================================================================================
Install  1 Package

Total download size: 1.4 M
Installed size: 4.5 M
Is this ok [y/d/N]: y
```

Restart Apache and PHP-FPM

>> sudo systemctl restart httpd

>> sudo systemctl restart php-fpm

```
[ec2-user@ip-20-0-5-51 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-5-51 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-5-51 ~]$
```

Download and Set Up phpMyAdmin

>> cd /var/www/html

```
[ec2-user@ip-20-0-5-51 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-5-51 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-5-51 ~]$ cd /var/www/html
[ec2-user@ip-20-0-5-51 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-05-12 17:32:01--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 79.127.206.208, 79.127.206.235, 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]
--2025-05-12 17:32:02--  https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 109.61.91.230, 109.61.91.198, 109.61.91.195, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.230|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13764534 (13M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

100%[==========================================================================================================

2025-05-12 17:32:02 (77.7 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]

[ec2-user@ip-20-0-5-51 html]$
```

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

```
[ec2-user@ip-20-0-5-51 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --stripcomponents
tar: unrecognized option '--stripcomponents'
Try `tar --help' or `tar --usage' for more information.
[ec2-user@ip-20-0-5-51 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --stripcomponents 1
mkdir: cannot create directory 'phpMyAdmin': File exists
[ec2-user@ip-20-0-5-51 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-5-51 html]$
```

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

>> 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

```
[ec2-user@ip-20-0-5-51 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --stripcomponents
tar: unrecognized option '--stripcomponents'
Try `tar --help' or `tar --usage' for more information.
[ec2-user@ip-20-0-5-51 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --stripcomponents 1
mkdir: cannot create directory 'phpMyAdmin': File exists
[ec2-user@ip-20-0-5-51 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-5-51 html]$ echo "PHP server 1" > /var/www/html/index.html
[ec2-user@ip-20-0-5-51 html]$ curl http://localhost
PHP server 1
[ec2-user@ip-20-0-5-51 html]$
```

Now ssh into app-server2 and run the below commands

```
ec2-user
[ec2-user@ip-20-0-5-51 /]$ uname
Linux
[ec2-user@ip-20-0-5-51 /]$ nano project.pem
[ec2-user@ip-20-0-5-51 /]$ ls
bin  boot  dev  etc  home  lib  lib64  local  media  mnt  opt  proc  root
[ec2-user@ip-20-0-5-51 /]$ sudo su
[root@ip-20-0-5-51 /]# exit
exit
[ec2-user@ip-20-0-5-51 /]$ touch project.pem
touch: cannot touch 'project.pem': Permission denied
[ec2-user@ip-20-0-5-51 /]$ sudo touch project.pem
[ec2-user@ip-20-0-5-51 /]$ sudo nano project.pem
[ec2-user@ip-20-0-5-51 /]$ ssh -i "project.pem" ec2-user@20.0.6.172
       #_
       ~\_  ####_        Amazon Linux 2
      ~~  \_#####\
      ~~     \###|       AL2 End of Life is 2026-06-30.
      ~~       \#/ ___
       ~~       V~' '->
        ~~~         /    A newer version of Amazon Linux is available!
          ~~._.   _/
             _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
           _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

No packages needed for security; 1 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-20-0-6-172 ~]$
```

Update the system

>> sudo yum update –y

```
[ec2-user@ip-20-0-6-172 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
No Match for argument: -y
No packages marked for update
[ec2-user@ip-20-0-6-172 ~]$
```

Install PHP 8.2

>> sudo dnf install php8.2

>> sudo yum install php8.2-mysqlnd

```
[ec2-user@ip-20-0-6-172 ~]$ sudo yum install php8.2-mysqlnd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No package php8.2-mysqlnd available.
Error: Nothing to do
[ec2-user@ip-20-0-6-172 ~]$
```

Install Apache Web Server

>> sudo yum install -y httpd

```
[ec2-user@ip-20-0-6-172 ~]$ sudo yum install -y httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.2-1.amzn2.0.1 will be installed
---> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
---> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
---> Package httpd-filesystem.noarch 0:2.4.62-1.amzn2.0.2 will be installed
---> Package httpd-tools.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
---> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
---> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.2 will be installed
--> Running transaction check
---> Package apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Finished Dependency Resolution
```

>> sudo systemctl enable httpd

>> sudo systemctl is-enabled httpd

```
Complete!
[ec2-user@ip-20-0-6-172 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-6-172 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-6-172 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-20-0-6-172 ~]$
```

Add User to Apache Group

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

```
[ec2-user@ip-20-0-6-172 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-6-172 ~]$
```

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-20-0-6-172 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-6-172 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-6-172 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-6-172 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-6-172 ~]$
```

Install Additional PHP Modules

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

>> sudo yum install php-fpm

```
[ec2-user@ip-20-0-6-172 ~]$ sudo yum install php-mbstring php-xml -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
---> Package php-mbstring.x86_64 0:5.4.16-46.amzn2.0.6 will be installed
--> Processing Dependency: php-common(x86-64) = 5.4.16-46.amzn2.0.6 for package: php-mbstring-5.4.16-46.amzn2.0.6.x86_64
---> Package php-xml.x86_64 0:5.4.16-46.amzn2.0.6 will be installed
--> Processing Dependency: libxslt.so.1(LIBXML2_1.0.11)(64bit) for package: php-xml-5.4.16-46.amzn2.0.6.x86_64
--> Processing Dependency: libxslt.so.1(LIBXML2_1.0.13)(64bit) for package: php-xml-5.4.16-46.amzn2.0.6.x86_64
```

```
[ec2-user@ip-20-0-6-172 ~]$ sudo yum install php-fpm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package php-fpm.x86_64 0:5.4.16-46.amzn2.0.6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch                   Version
================================================================================
Installing:
 php-fpm                x86_64                 5.4.16-46.amzn2.0.6

Transaction Summary
================================================================================
Install  1 Package

Total download size: 1.4 M
Installed size: 4.5 M
Is this ok [y/d/N]:
```

Restart Apache and PHP-FPM

>> sudo systemctl restart httpd

```
Is this ok [y/d/N]: sudo systemctl restart httpd
Is this ok [y/d/N]: y
Downloading packages:
php-fpm-5.4.16-46.amzn2.0.6.x86_64.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : php-fpm-5.4.16-46.amzn2.0.6.x86_64
  Verifying  : php-fpm-5.4.16-46.amzn2.0.6.x86_64

Installed:
  php-fpm.x86_64 0:5.4.16-46.amzn2.0.6
```

>> sudo systemctl restart php-fpm

```
[ec2-user@ip-20-0-6-172 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-6-172 ~]$
```

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

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

```
[ec2-user@ip-20-0-6-172 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-6-172 ~]$ cd /var/www/html
[ec2-user@ip-20-0-6-172 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-05-12 18:04:50--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 79.127.206.207, 79.127.206.234, 2a02:6ea0:c400::54, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|79.127.206.207|: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]
--2025-05-12 18:04:50--  https://files.phpmyadmin.net/phpMyAdmin/5.2.2/phpMyAdmin-5.2.2-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 109.61.91.230, 109.61.91.194, 109.61.91.198, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.230|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13764534 (13M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

100%[==========================================================================================

2025-05-12 18:04:51 (70.6 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]

[ec2-user@ip-20-0-6-172 html]$
```

```
[ec2-user@ip-20-0-6-172 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --stripcomponents 1
tar: unrecognized option '--stripcomponents'
Try `tar --help' or `tar --usage' for more information.
[ec2-user@ip-20-0-6-172 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-6-172 html]$
```

Create a Test Page and Test the Server

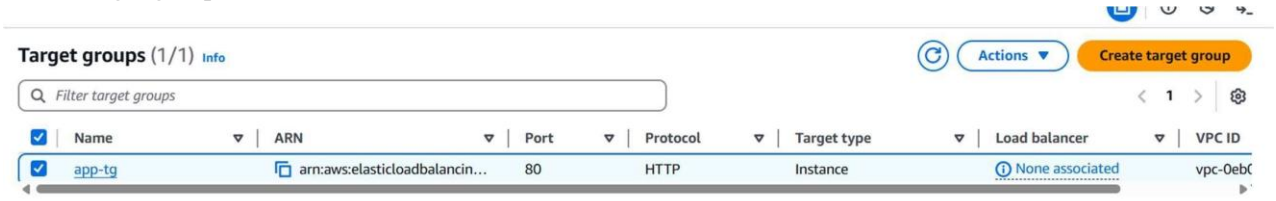
>> echo "PHP server 2" > /var/www/html/index.html

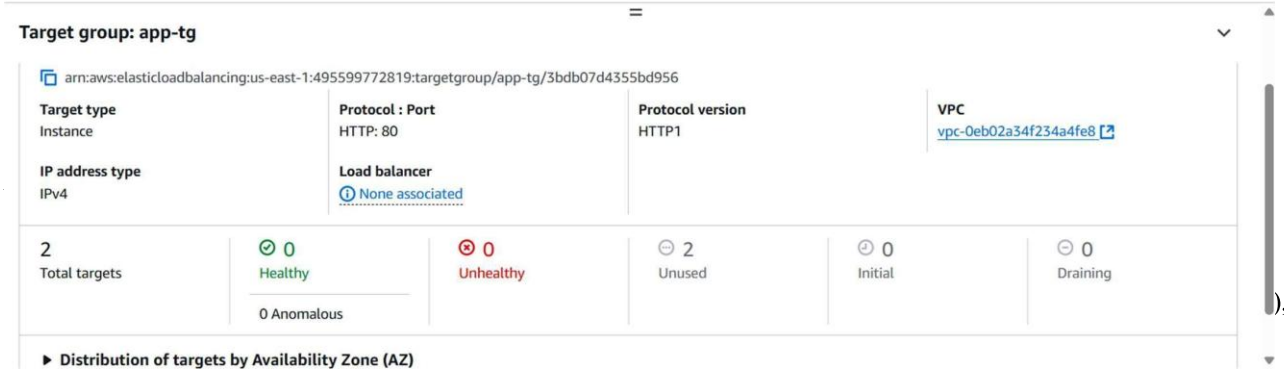
>> curl http://localhost

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.



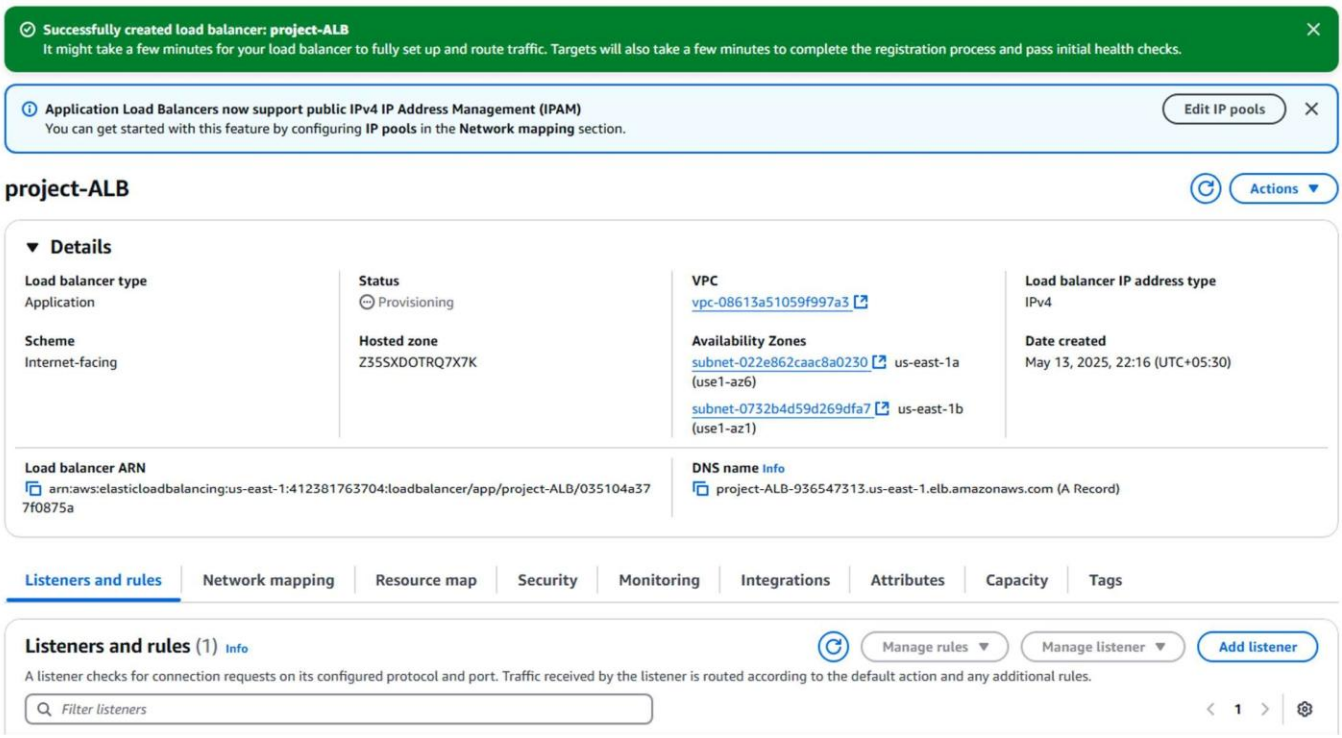


), Port

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

Refrest the page: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 Stancerd 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-

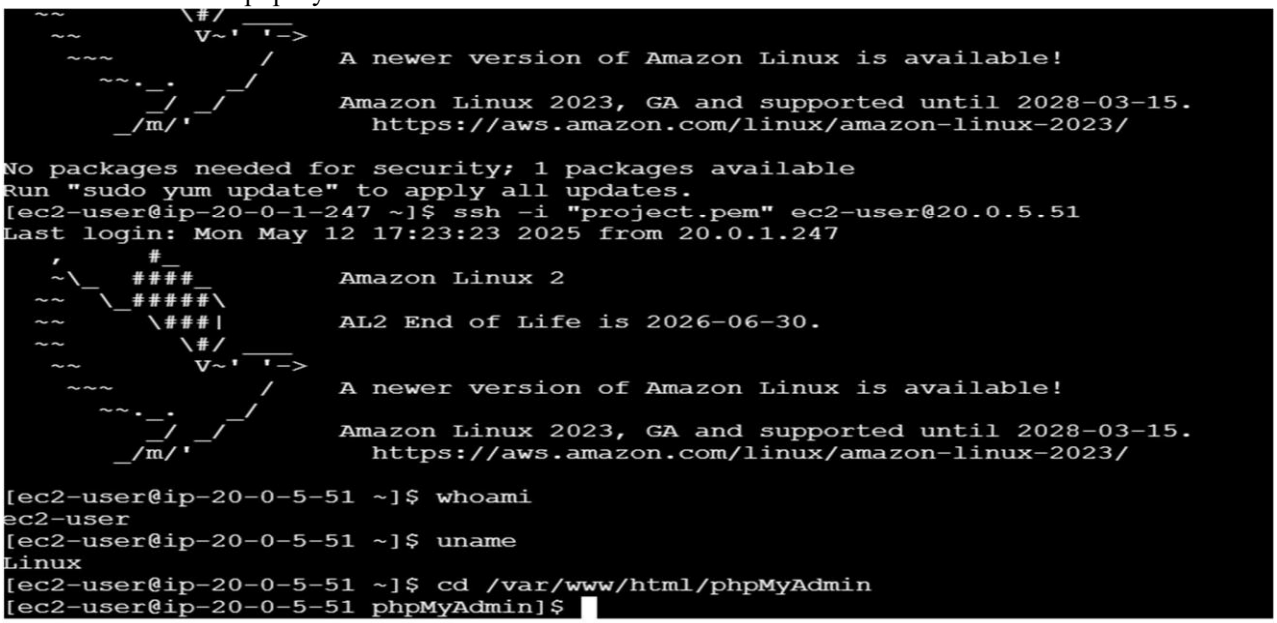
subnetgroup), Public access (No), VPC security group (Create new)… New VPC security group name (db-sg), Availability Zone (us-east-1a), Click on Create database.

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

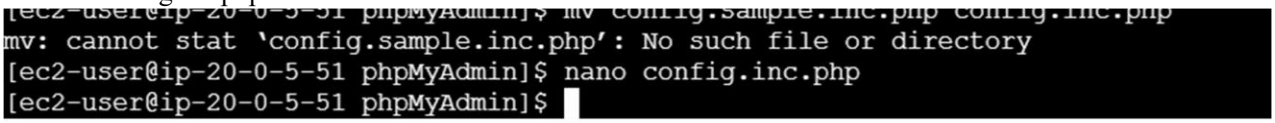
Now copy the Endpoint of RDS cluster and ssh into app-server1 instance and app-server2 instance

>>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: