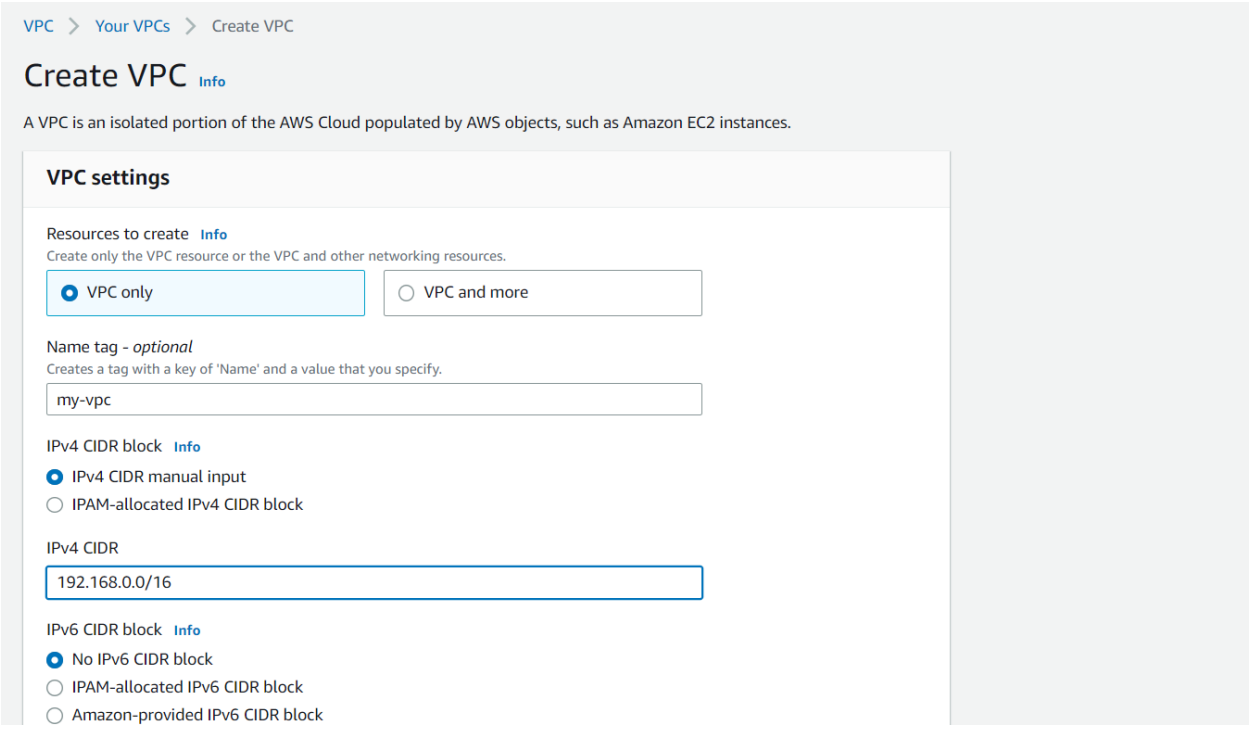
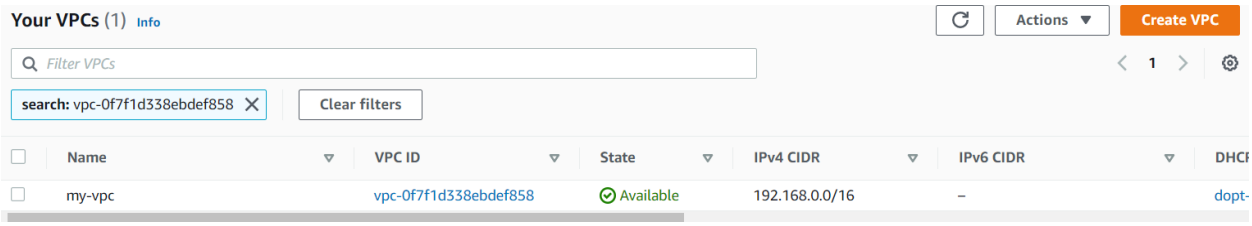
VPC

Create a VPC with two subnets private instance and public instance and launch two instances in both subnets. Configure myPhpAdmin in the public instance and MySQL server in the private instance where private instances only have access to the public instance, not to the outside world. Make sure to have connectivity between the myPhpAdmin and MySQL server.

Creating a VPC

Creating a VPC with the my-vpc with the CIDR block 192.168.0.0/16. In the CIDR block 16 represents the netmask 255.255.0.0 and the total number of IP addresses available in the VPC are 65536.

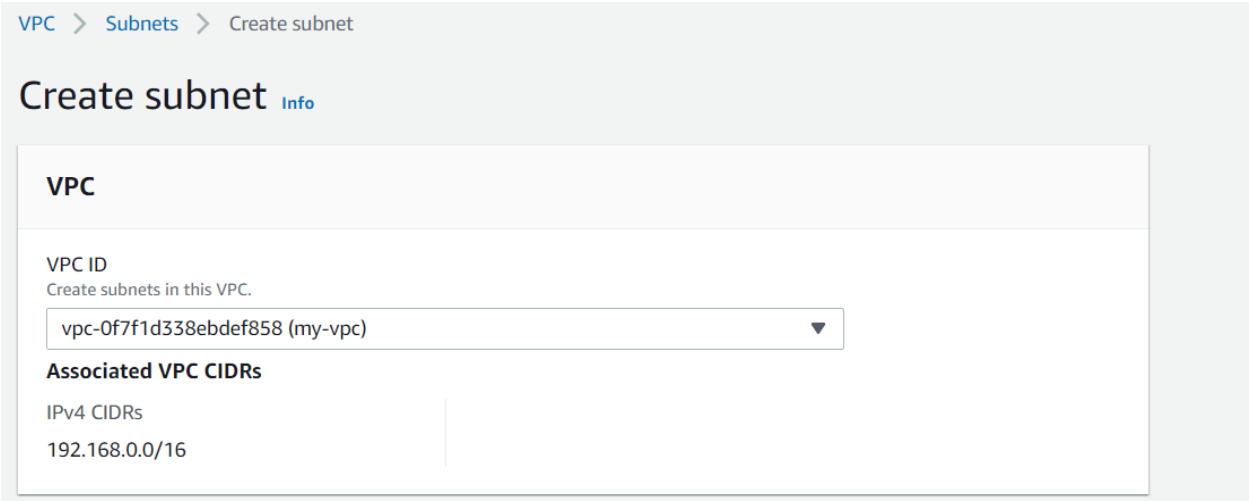


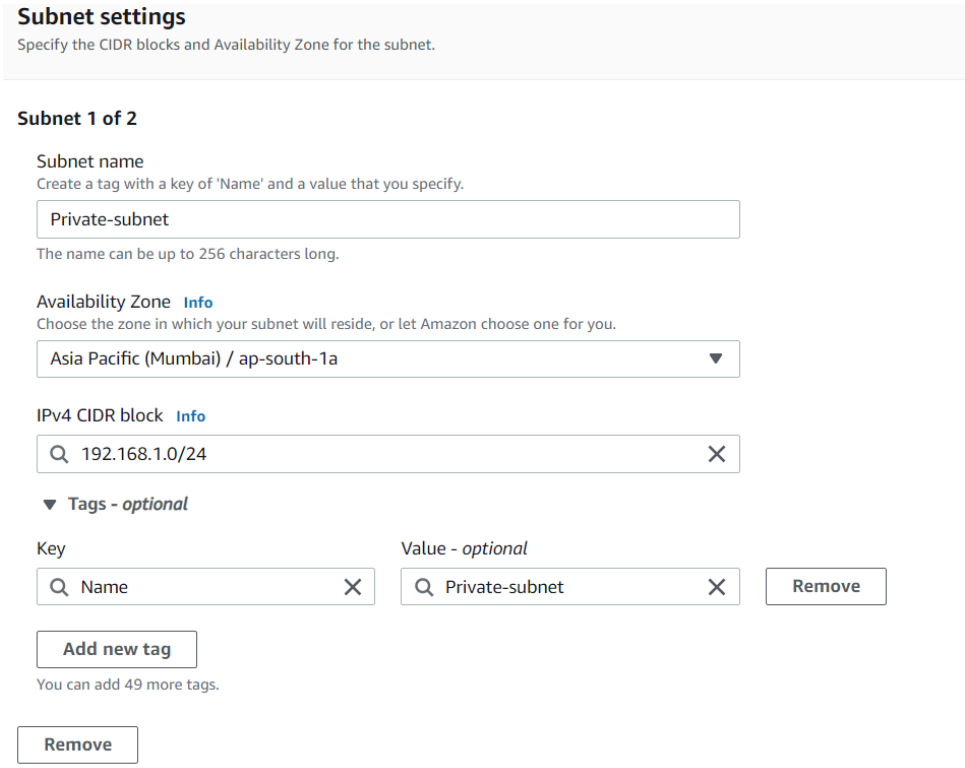


VPC created successfully

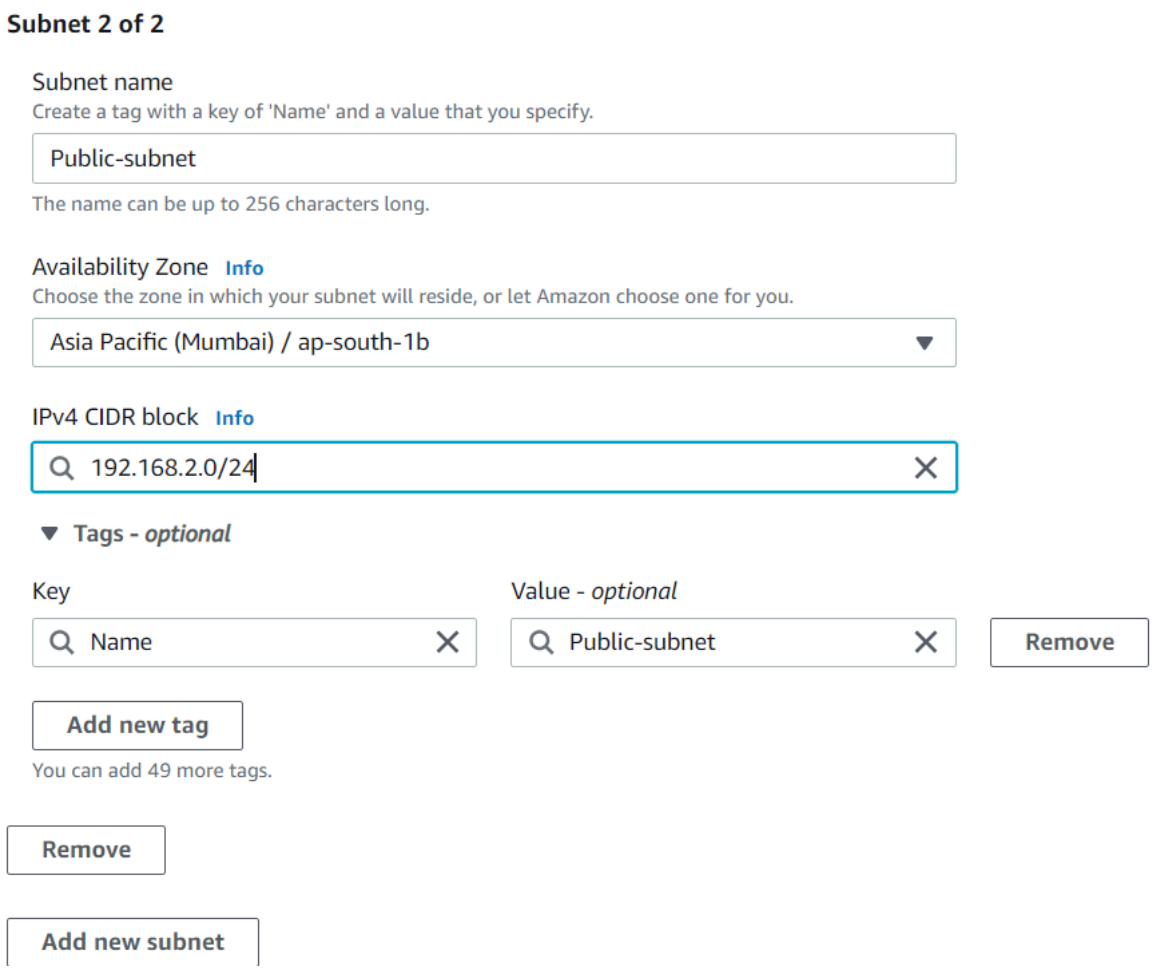
Creating Private and Public subnets

Creating a private subnet and public subnet in the previously created VPC with the CIDR block 192.168.1.0/24 and 192.168.2.0/24 totally consists of 256 IP addresses in each subnet.

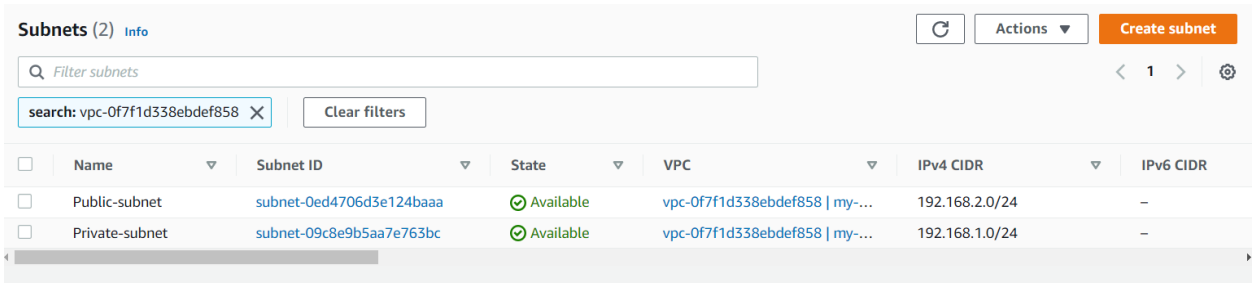




Creating Private subnet in the ap-south-1a availability zone



Creating Public subnet in the ap-south-1b availability zone



Two subnets are created successfully

Launching two instances in both Private and Public subnets

Launching one instance in the public subnet which has public access, same way launching another instance which doesn't have public access.

Launching a public instance

EC2 > Instances > Launch an instance

Launch an instance Info

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 Info

Name

Public-instance

Add additional tags

Launching an instance with the name of Publc-instance

Instance type Info

Instance type

t2.micro — Free tier eligible

Family: t2 1 vCPU 1 GiB Memory

On-Demand Linux pricing: 0.0124 USD per Hour

On-Demand Windows pricing: 0.017 USD per Hour

On-Demand RHEL pricing: 0.0724 USD per Hour

On-Demand SUSE pricing: 0.0124 USD per Hour

Compare instance types

Configuration of the instance 1 CPU and 1GB Memory

Network settings Info

VPC - *required* Info

vpc-0f7f1d338ebdef858 (my-vpc)

192.168.0.0/16

Subnet Info

subnet-0ed4706d3e124baaa — Public-subnet

VPC: vpc-0f7f1d338ebdef858 Owner: 818119396514

Availability Zone: ap-south-1b IP addresses available: 251 CIDR: 192.168.2.0/24)

Create new subnet

Auto-assign public IP Info

Enable

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - *required*

launch-wizard-8

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

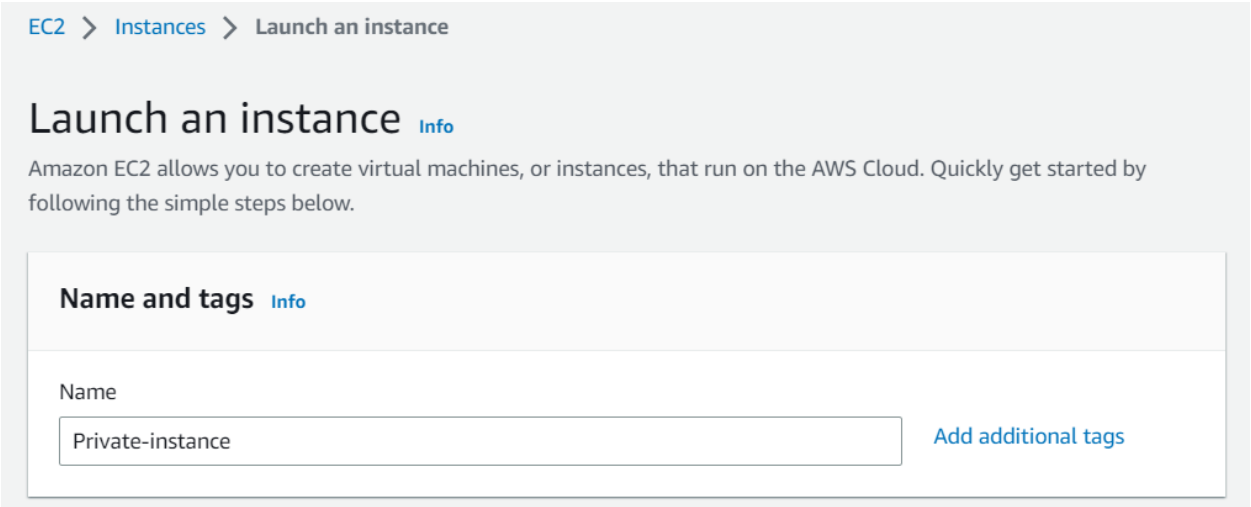
Description - *required* Info

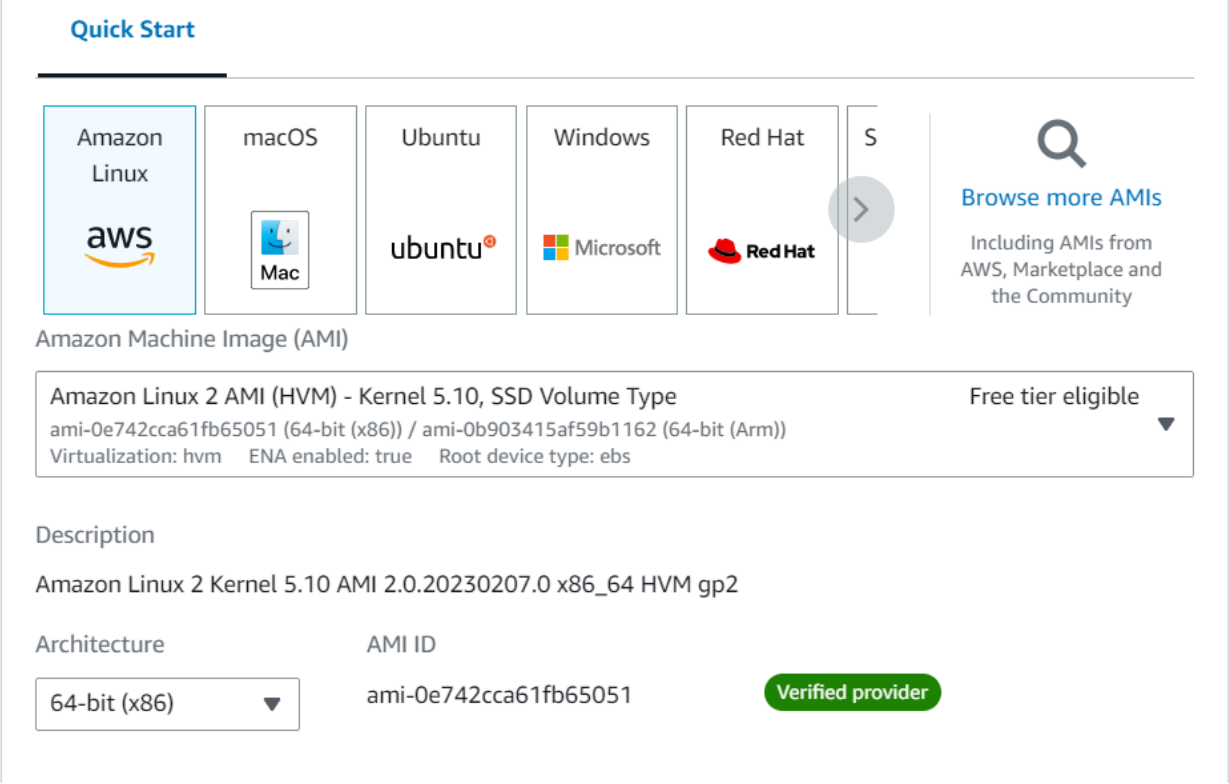
launch-wizard-8 created 2023-02-21T09:25:21.600Z

Launching instance in the public subnet which is created previously which have public access

Launching a private instance

Sameway launching another instance in the private subnet which does not have public access.





Instance type Info

Instance type

t2.micro — Free tier eligible

Family: t2 1 vCPU 1 GiB Memory
On-Demand Linux pricing: 0.0124 USD per Hour
On-Demand Windows pricing: 0.017 USD per Hour
On-Demand RHEL pricing: 0.0724 USD per Hour
On-Demand SUSE pricing: 0.0124 USD per Hour

Compare instance types

Network settings Info

VPC - *required* Info

vpc-0f7f1d338ebdef858 (my-vpc)
192.168.0.0/16

Subnet Info

subnet-09c8e9b5aa7e763bc — Private-subnet
VPC: vpc-0f7f1d338ebdef858 Owner: 818119396514
Availability Zone: ap-south-1a IP addresses available: 251 CIDR: 192.168.1.0/24)

Create new subnet

Auto-assign public IP Info

Disable

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

- (•) Create security group
- ( ) Select existing security group

Security group name - *required*

launch-wizard-7

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

Instances (2) Info

Connect | Instance state | Actions | Launch instances

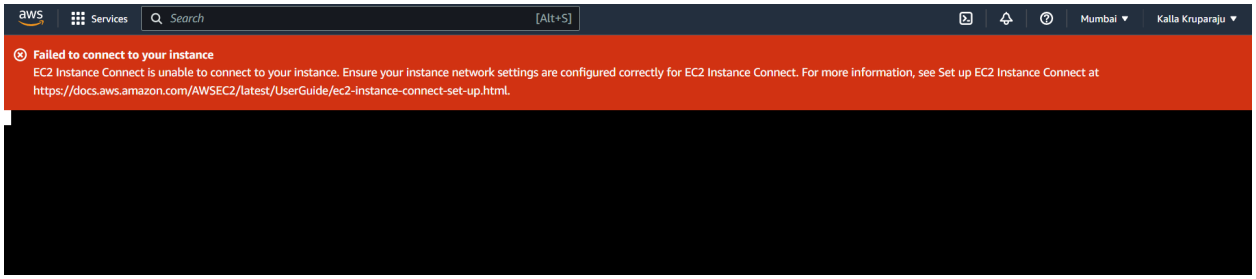
Find instance by attribute or tag (case-sensitive)

Instance state = running | Clear filters

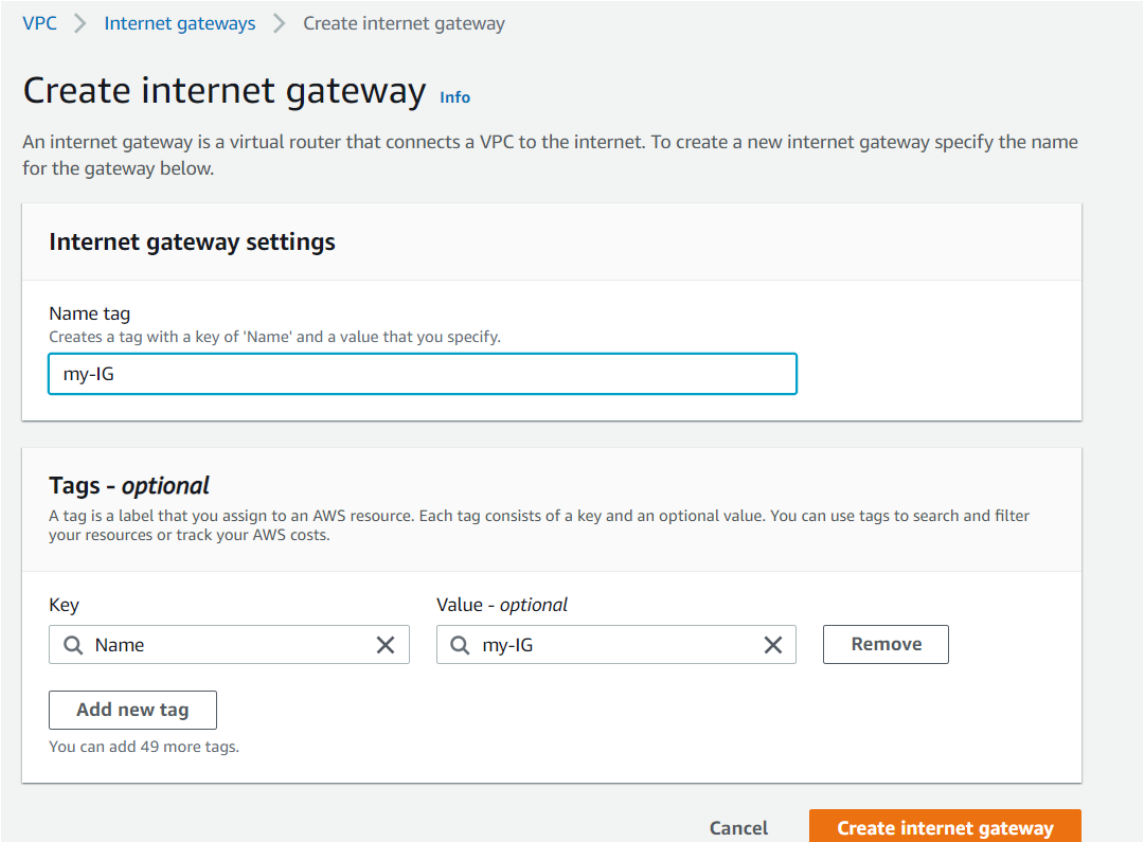
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
Private-instance	i-0e492315a859420f8	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	–
Public-instance	i-0fb7d2aa6b45afcde	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1b	–

Finally two instances are created successfully

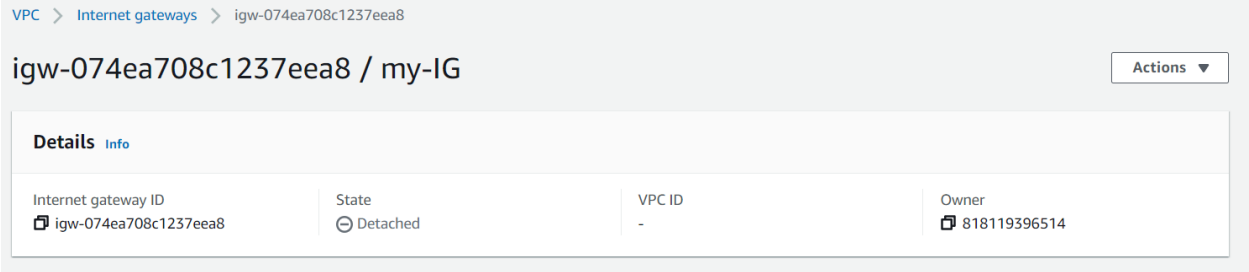
I Tried to connect a public instance from the console but it failed to connect because it doesn't have a proper network connection to the internet from VPC.



Creating Internet Gateway



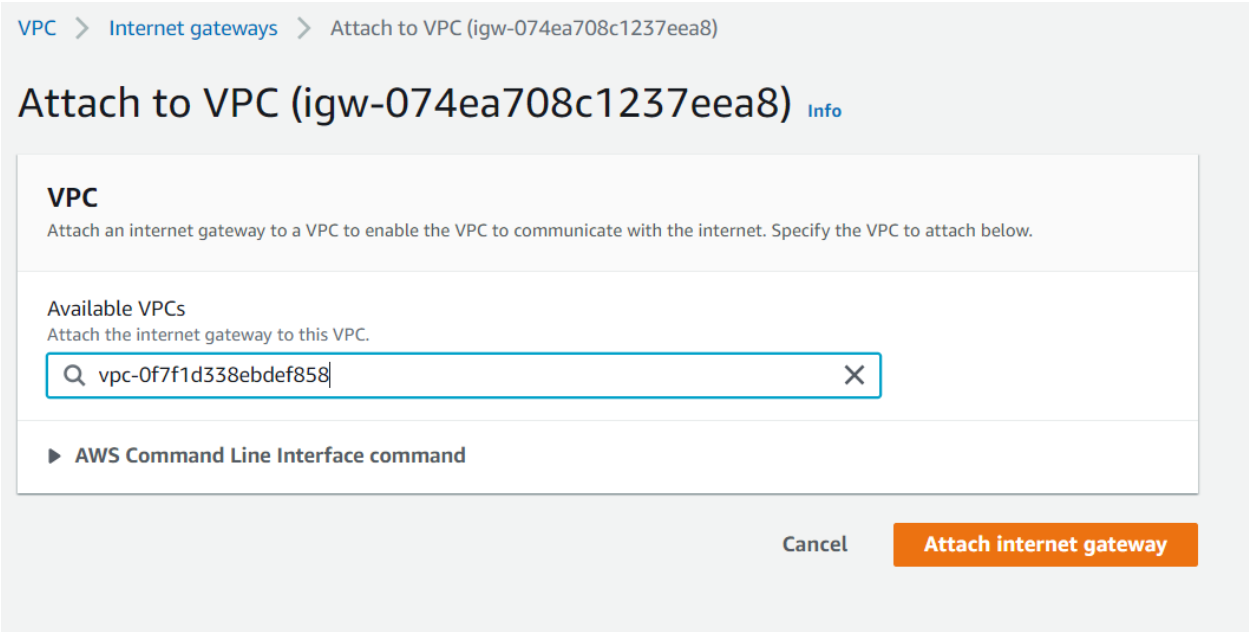
Creating a internet gateway to establish a public access to the VPC



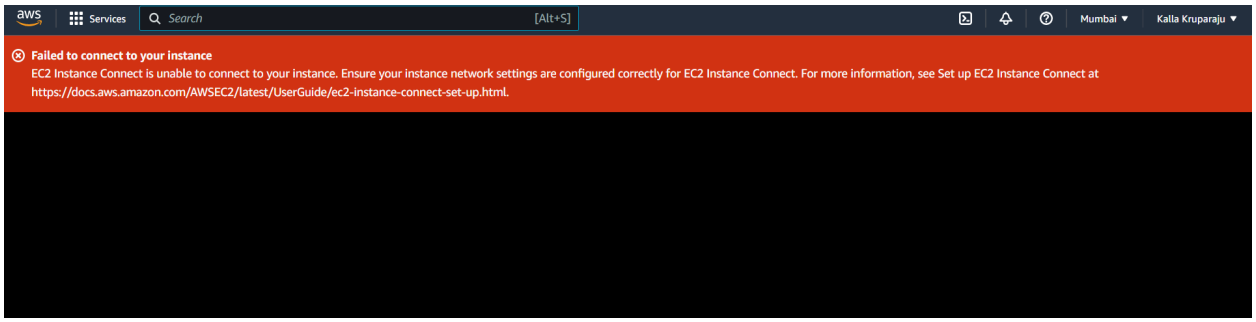
Internet gateway was created successfully

Attaching Internet Gateway to VPC

The Internet gateway was created successfully but it was in the detached mode so attaching the Internet gateway to the VPC.



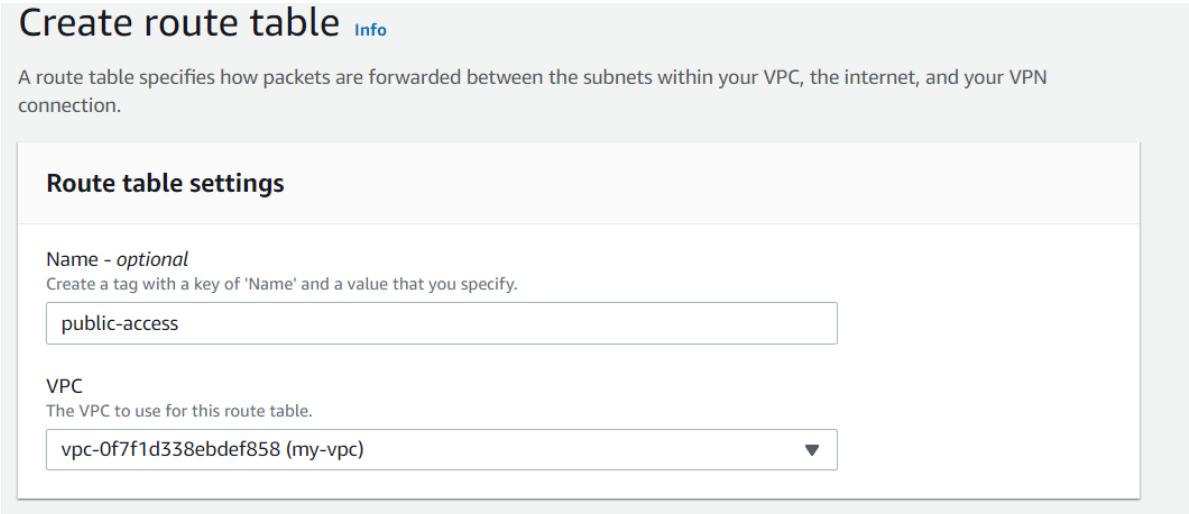
Again tried to connect to the instance but it failed again because the public access was established up to VPC only. it doesn't connect to the public instance.



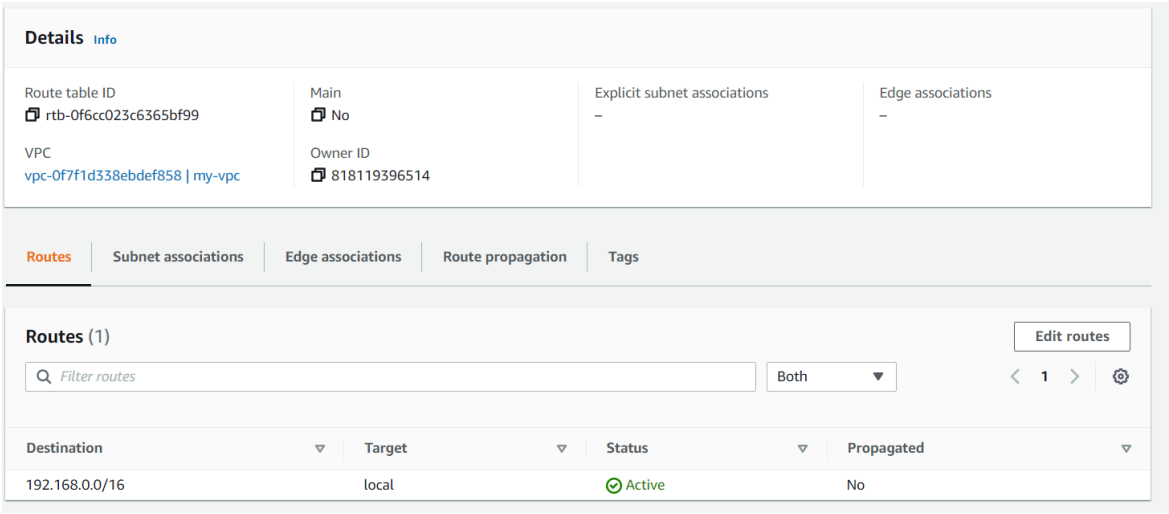
Establishing public access between VPC and public subnet

To establish connection between any two elements, routes will come and play a role for that we have to create routing table.

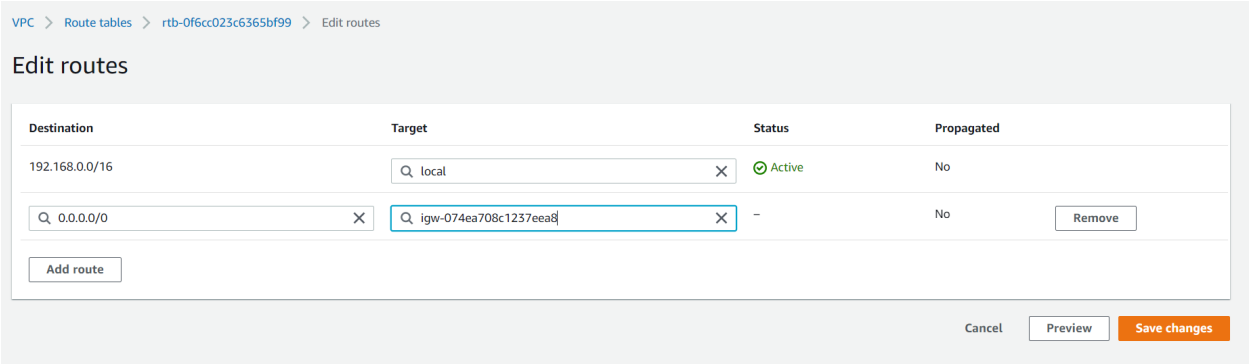
Creating a route table



Creating a routing table in the previously created VPC

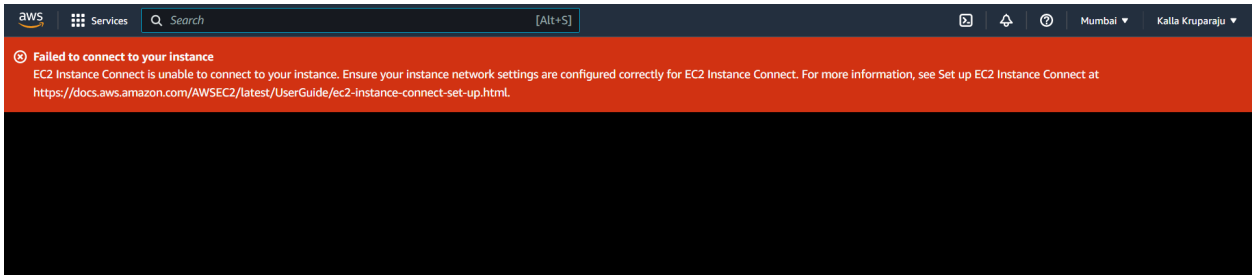


Routing table created successfully but it doesn't have routes
to establish connection between VPC and public subnet

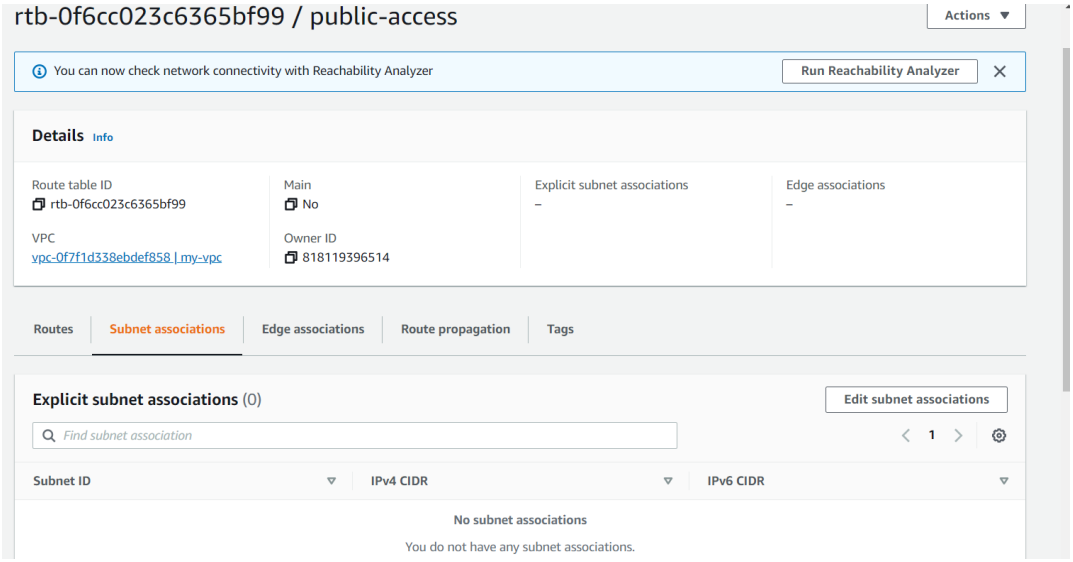


Adding the internet gateway route to the routing table

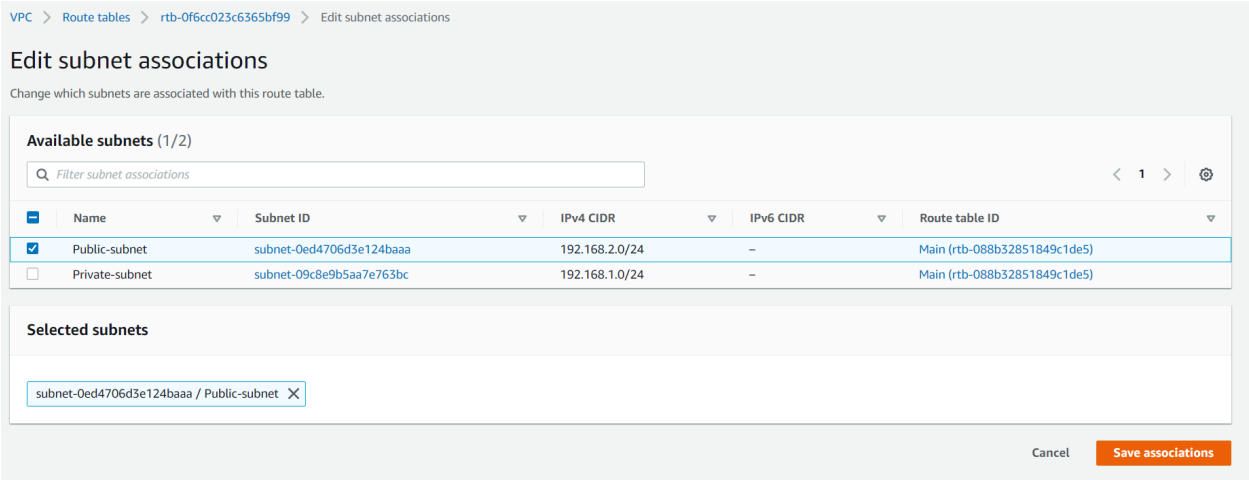
Again tried to connect to the instance but it failed again because the public access only established up to the routing table till now there was no connection between the routing table and the public subnet for that it is required to associate the routing table with the public subnet.



Subnet association of routing table



You can see there was no subnet association to the route table. adding the subnet by clicking the edit subnet associations.



Added the public subnet to the route table

Now tried to connect the public instance again but it was connected successfully.

```
aws  Services  Search [Alt+S]                                  Mumbai ▾  Kalla Kruparaju ▾

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-192-168-2-174 ~]$
```

i-0fb7d2aa6b45afcde (Public-instance)

PublicIPs: 43.204.114.27 PrivateIPs: 192.168.2.174

Installing the packages in the public instance

Installing httpd web server: **sudo yum install -y httpd**

```
[root@ip-192-168-2-174 ec2-user]# sudo yum install -y httpd
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.54-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.7.2-1.amzn2 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 0:2.4.54-1.amzn2 will be installed
---> Package httpd-tools.x86_64 0:2.4.54-1.amzn2 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.amzn2.0.1 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
```

Installing php : **amazon-linux-extras install -y php7.2**

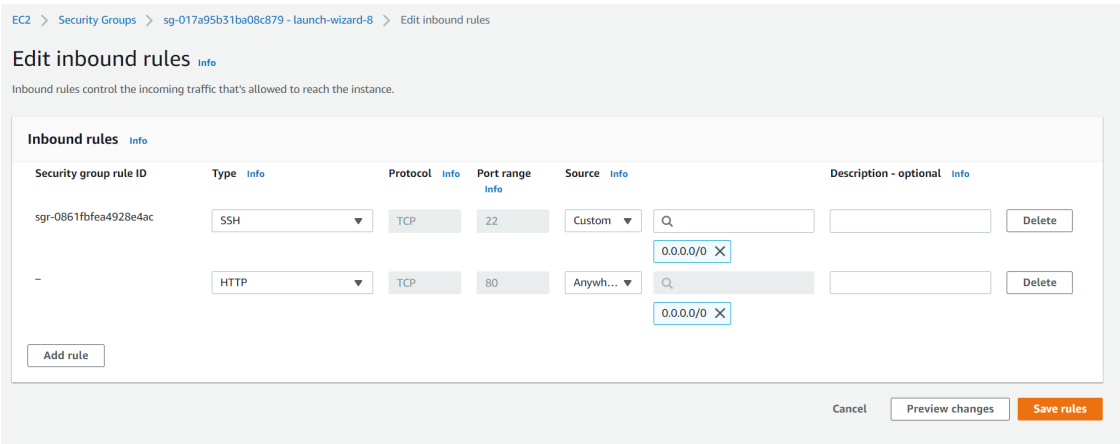
```
[root@ip-192-168-2-174 ec2-user]# sudo amazon-linux-extras install -y  php7.2
Topic php7.2 has end-of-support date of 2020-11-30
Installing php-pdo, php-fpm, php-mysqlnd, php-cli, php-json
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-kernel-5.10 amzn2extra-php7.2
0 metadata files removed
0 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                          | 3.7 kB  00:00:00
amzn2extra-docker                                                                   | 3.0 kB  00:00:00
amzn2extra-kernel-5.10                                                              | 3.0 kB  00:00:00
amzn2extra-php7.2                                                                   | 3.0 kB  00:00:00
(1/9): amzn2-core/2/x86_64/group_gz                                                 | 2.5 kB  00:00:00
(2/9): amzn2-core/2/x86_64/updateinfo                                               | 556 kB  00:00:00
(3/9): amzn2extra-docker/2/x86_64/updateinfo                                        | 8.0 kB  00:00:00
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo                                   |  24 kB  00:00:00
(5/9): amzn2extra-php7.2/2/x86_64/updateinfo                                        |   76 B  00:00:00
(6/9): amzn2extra-docker/2/x86_64/primary_db                                        | 101 kB  00:00:00
(7/9): amzn2extra-php7.2/2/x86_64/primary_db                                        | 580 kB  00:00:00
(8/9): amzn2extra-kernel-5.10/2/x86_64/primary_db                                   |  15 MB  00:00:00
(9/9): amzn2-core/2/x86_64/primary_db                                               |  69 MB  00:00:01
Resolving Dependencies
--> Running transaction check
---> Package php-cli.x86_64 0:7.2.34-1.amzn2 will be installed
--> Processing Dependency: php-common(x86-64) = 7.2.34-1.amzn2 for package: php-cli-7.2.34-1.amzn2.x86_64
```

```
[root@ip-192-168-2-174 ec2-user]# cd /var/www/html/
[root@ip-192-168-2-174 html]# ls
[root@ip-192-168-2-174 html]# vim index.php
[root@ip-192-168-2-174 html]# cat index.php
<pre>
<?php
print `/usr/sbin/ifconfig`;
?>
</pre>
```

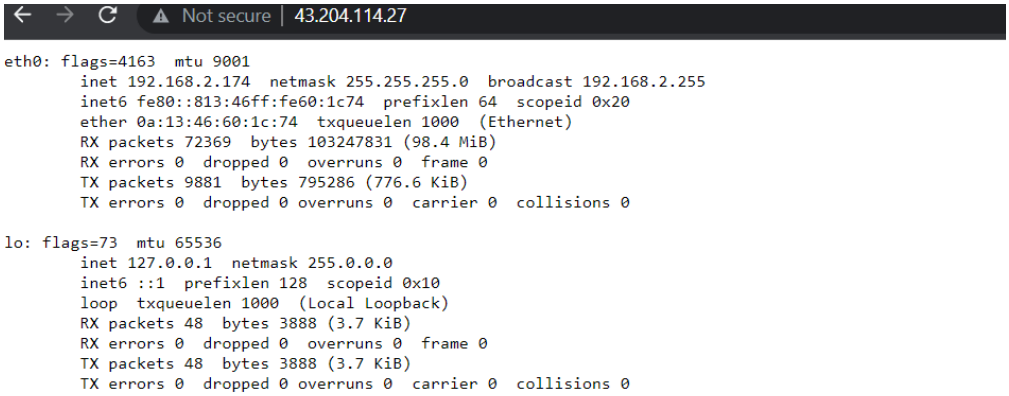
Added sample php file in the webserver root directory /var/www/html

```
[root@ip-192-168-2-174 html]# systemctl start httpd
[root@ip-192-168-2-174 html]# systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[root@ip-192-168-2-174 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
  Drop-In: /usr/lib/systemd/system/httpd.service.d
           └─php-fpm.conf
   Active: active (running) since Tue 2023-02-21 09:43:23 UTC; 11s ago
     Docs: man:httpd.service(8)
 Main PID: 3922 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─3922 /usr/sbin/httpd -DFOREGROUND
           ├─3928 /usr/sbin/httpd -DFOREGROUND
           ├─3929 /usr/sbin/httpd -DFOREGROUND
           ├─3930 /usr/sbin/httpd -DFOREGROUND
           ├─3931 /usr/sbin/httpd -DFOREGROUND
           └─3932 /usr/sbin/httpd -DFOREGROUND

Feb 21 09:43:23 ip-192-168-2-174.ap-south-1.compute.internal systemd[1]: Starting The Apache HTTP Server...
Feb 21 09:43:23 ip-192-168-2-174.ap-south-1.compute.internal systemd[1]: Started The Apache HTTP Server.
[root@ip-192-168-2-174 html]#
```

Started service of the httpd web server.



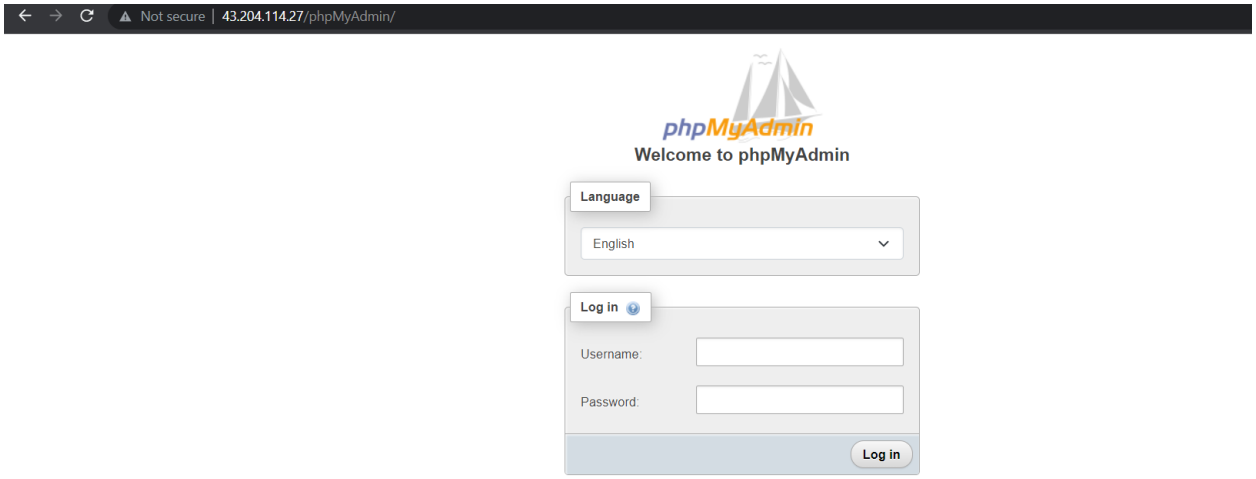
Added a security rule to access the instance also on the port num 80.



Accessing the web server and sample web page accessed successfully

```
[root@ip-192-168-2-174 html]# wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2023-02-21 09:52:33--  https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
Resolving www.phpmyadmin.net (www.phpmyadmin.net)... 143.244.33.159, 143.244.33.173, 143.244.33.174, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|143.244.33.159|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://files.phpmyadmin.net/phpMyAdmin/5.2.1/phpMyAdmin-5.2.1-all-languages.tar.gz [following]
--2023-02-21 09:52:34--  https://files.phpmyadmin.net/phpMyAdmin/5.2.1/phpMyAdmin-5.2.1-all-languages.tar.gz
Resolving files.phpmyadmin.net (files.phpmyadmin.net)... 89.187.162.143, 89.187.163.84, 143.244.33.161, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|89.187.162.143|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13041998 (12M) [application/octet-stream]
Saving to: 'phpMyAdmin-latest-all-languages.tar.gz'

100%[===================================================================================================================>] 13,041,998  14.2MB/s   in 0.9s

2023-02-21 09:52:36 (14.2 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13041998/13041998]

[root@ip-192-168-2-174 html]# ls
index.php  phpMyAdmin-latest-all-languages.tar.gz
[root@ip-192-168-2-174 html]# mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

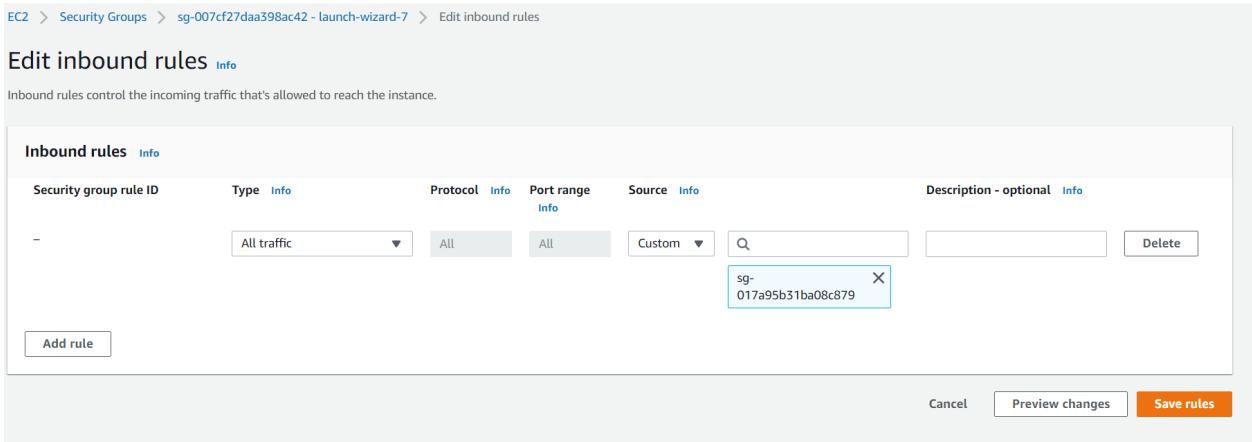
Download the phpmyadmin prebuilt webpage tar file and created a directory for phpmyadmin in the webserver root folder and exact that tar file in it.



Accessed the phpmyadmin web page successfully

Accessing private instance from the public instance (Bastion host)

Added rule in the private instance allow all traffic from the public instance with the help of public instance security group id.



Created a key-pair file of the private instance in the public instance and change the mode of the key-pair file as read only mode

```
[root@ip-192-168-2-174 ~]# ls -lh test.pem
-rw-r--r-- 1 root root 1.7K Feb 21 10:16 test.pem
[root@ip-192-168-2-174 ~]# chmod 400 test.pem
[root@ip-192-168-2-174 ~]# ls -lh test.pem
-r-------- 1 root root 1.7K Feb 21 10:16 test.pem
[root@ip-192-168-2-174 ~]#
```

Login into the private instance from the public instance using the ssh

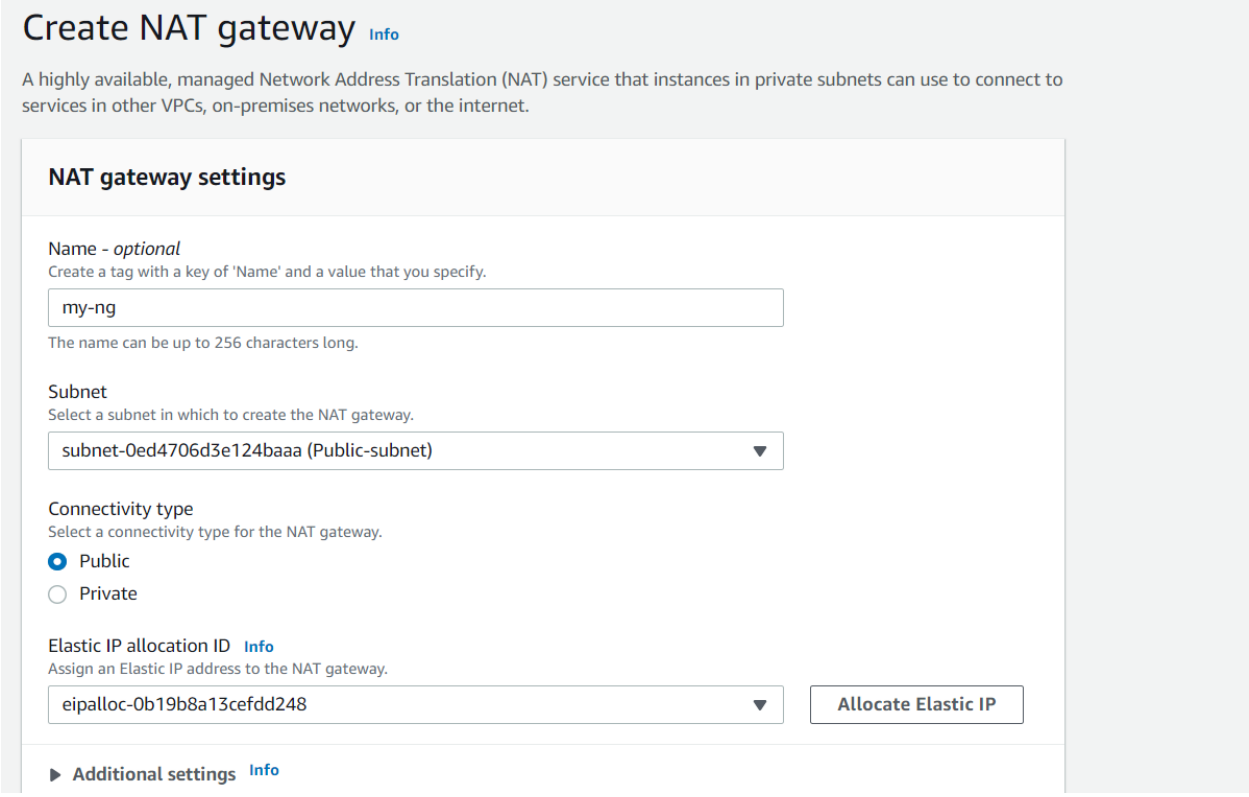
```
[root@ip-192-168-2-174 ~]# ssh ec2-user@192.168.1.49 -i test.pem
The authenticity of host '192.168.1.49 (192.168.1.49)' can't be established.
ECDSA key fingerprint is SHA256:nnntuDmpoEQFgwMCuy7Y7f4PhU2iGhGl7oJYKMYusjA.
ECDSA key fingerprint is MD5:7a:10:c9:0e:72:1a:fa:af:be:33:63:8f:2f:66:d6:84.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.49' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-192-168-1-49 ~]$
[ec2-user@ip-192-168-1-49 ~]$
```

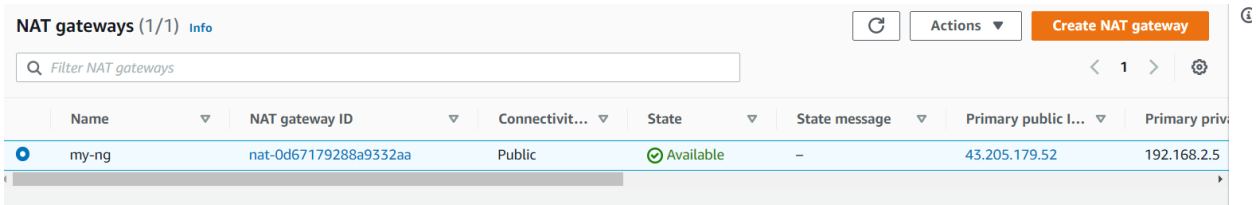
Installing the mysql-server in the private instance because it doesn't have public access for that we have to configure the NAT gateway.

```
[ec2-user@ip-192-168-1-49 ~]$ yum install mysql-server -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
You need to be root to perform this command.
[ec2-user@ip-192-168-1-49 ~]$ sudo su
[root@ip-192-168-1-49 ec2-user]# yum install mysql-server -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Could not retrieve mirrorlist https://amazonlinux-2-repos-ap-south-1.s3.dualstack.ap-south-1.amazonaws.com/2/core/latest/x86_64/mirr
12: Timeout on https://amazonlinux-2-repos-ap-south-1.s3.dualstack.ap-south-1.amazonaws.com/2/core/latest/x86_64/mirror.list: (28, "
-ap-south-1.s3.dualstack.ap-south-1.amazonaws.com port 443 after 2702 ms: Couldn't connect to server")


 One of the configured repositories failed (Unknown),
 and yum doesn't have enough cached data to continue. At this point the only
 safe thing yum can do is fail. There are a few ways to work "fix" this:

     1. Contact the upstream for the repository and get them to fix the problem.

     2. Reconfigure the baseurl/etc. for the repository, to point to a working
        upstream. This is most often useful if you are using a newer
        distribution release than is supported by the repository (and the
        packages for the previous distribution release still work).

     3. Run the command with the repository temporarily disabled
            yum --disablerepo=<repoid> ...
```

Creating NAT gateway

Creating a NAT gateway in the public subnet with the public access connectivity.

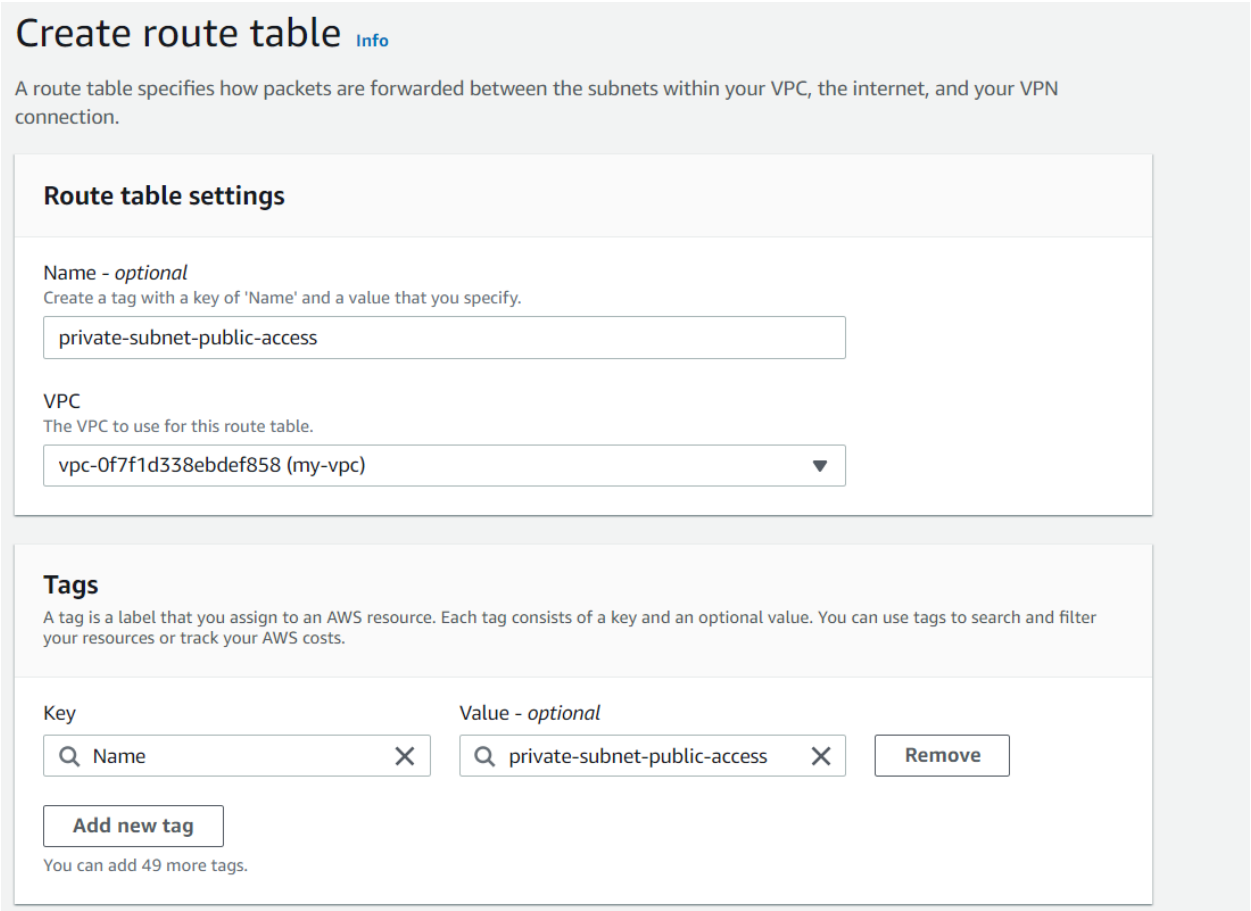




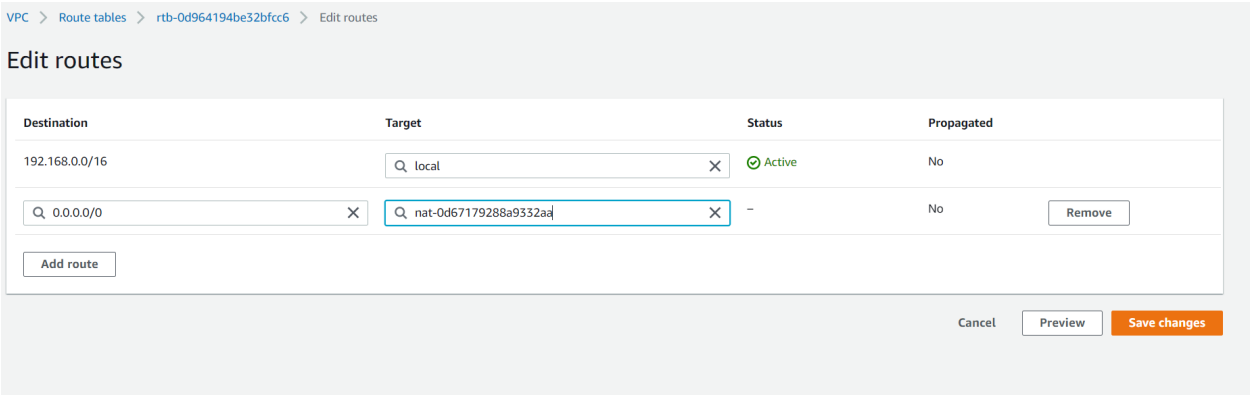
NAT gateway created successfully

Creating the Route table for NAT gateway

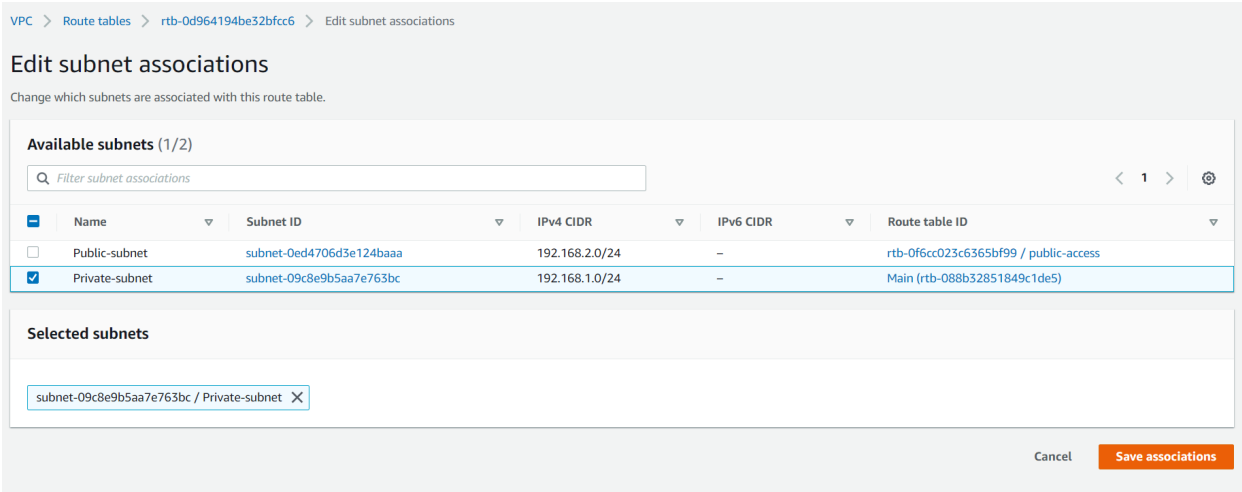
Similarly as Internet gateway it have created a route table to route between the private subnet and public subnet.



Creating a route table



Edited the routes in the route table with the target as nat gateway destination as anywhere.



Route table associated with the private subnet now have end to end connectivity between the private and public subnet.

```
[ec2-user@ip-192-168-1-49 ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=109 time=2.91 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=109 time=2.12 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=109 time=2.09 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=109 time=2.14 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.099/2.319/2.912/0.342 ms
[ec2-user@ip-192-168-1-49 ~]$
```

Now private instance have internet access now packages can be installed in the private instance.

Installing the mysql-server in the private instance

To install the mysql-server we have to install epel repo because that repo contains the mysql-server package.

amazon-linux-extras install epel -y

```
[root@ip-192-168-1-49 ec2-user]# amazon-linux-extras install epel -y
Installing epel-release
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-epel amzn2extra-kernel-5.10
13 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                                | 3.7 kB  00:00:00
amzn2extra-docker                                                                         | 3.0 kB  00:00:00
amzn2extra-epel                                                                           | 3.0 kB  00:00:00
amzn2extra-kernel-5.10                                                                    | 3.0 kB  00:00:00
(1/9): amzn2-core/2/x86_64/group_gz                                                       | 2.5 kB  00:00:00
(2/9): amzn2-core/2/x86_64/updateinfo                                                     | 556 kB  00:00:00
(3/9): amzn2extra-epel/2/x86_64/primary_db                                                | 1.8 kB  00:00:00
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo                                         |  24 kB  00:00:00
(5/9): amzn2extra-docker/2/x86_64/updateinfo                                              | 8.0 kB  00:00:00
(6/9): amzn2extra-epel/2/x86_64/updateinfo                                                |   76 B  00:00:00
(7/9): amzn2extra-kernel-5.10/2/x86_64/primary_db                                         |  15 MB  00:00:00
(8/9): amzn2extra-docker/2/x86_64/primary_db                                              | 101 kB  00:00:00
(9/9): amzn2-core/2/x86_64/primary_db                                                     |  69 MB  00:00:01
Resolving Dependencies
--> Running transaction check
---> Package epel-release.noarch 0:7-11 will be installed
--> Finished Dependency Resolution
```

Installing the mysql server package

yum install https://dev.mysql.com/get/mysql80-community-release-el7-5.noarch.rpm

```
[root@ip-192-168-1-49 ec2-user]# yum install https://dev.mysql.com/get/mysql80-community-release-el7-5.noarch.rpm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
mysql80-community-release-el7-5.noarch.rpm                                                                          |  11 kB  00:00:00
Examining /var/tmp/yum-root-gT_nUr/mysql80-community-release-el7-5.noarch.rpm: mysql80-community-release-el7-5.noarch
Marking /var/tmp/yum-root-gT_nUr/mysql80-community-release-el7-5.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package mysql80-community-release.noarch 0:el7-5 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================================================================================
 Package                                    Arch                     Version                  Repository                                   Size
================================================================================================================================================
Installing:
 mysql80-community-release                  noarch                   el7-5                    /mysql80-community-release-el7-5.noarch      9.1 k

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

Total size: 9.1 k
```

yum install mysql-community-server -y

```
[root@ip-192-168-1-49 ec2-user]# yum install mysql-server -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Existing lock /var/run/yum.pid: another copy is running as pid 32466.
Another app is currently holding the yum lock; waiting for it to exit...
  The other application is: yum
    Memory : 320 M RSS (612 MB VSZ)
    Started: Tue Feb 21 10:29:55 2023 - 00:04 ago
    State  : Running, pid: 32466
Another app is currently holding the yum lock; waiting for it to exit...
  The other application is: yum
    Memory : 335 M RSS (628 MB VSZ)
    Started: Tue Feb 21 10:29:55 2023 - 00:06 ago
    State  : Running, pid: 32466
Another app is currently holding the yum lock; waiting for it to exit...
  The other application is: yum
    Memory : 335 M RSS (628 MB VSZ)
    Started: Tue Feb 21 10:29:55 2023 - 00:08 ago
    State  : Running, pid: 32466
271 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package mysql-community-server.x86_64 0:8.0.32-1.el7 will be installed
--> Processing Dependency: mysql-community-common(x86-64) = 8.0.32-1.el7 for package: mysql-community-server-8.0.32-1.el7.x86_64
--> Processing Dependency: mysql-community-icu-data-files = 8.0.32-1.el7 for package: mysql-community-server-8.0.32-1.el7.x86_64
--> Processing Dependency: mysql-community-client(x86-64) >= 8.0.11 for package: mysql-community-server-8.0.32-1.el7.x86_64
```

Starting the service of the mysql server and enabling it permanently.

```
[root@ip-192-168-1-49 ec2-user]# systemctl status mysqld
● mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
[root@ip-192-168-1-49 ec2-user]# systemctl start mysqld
[root@ip-192-168-1-49 ec2-user]# systemctl enable mysqld
[root@ip-192-168-1-49 ec2-user]# systemctl status mysqld
● mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2023-02-21 10:31:58 UTC; 9s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
 Main PID: 32649 (mysqld)
   Status: "Server is operational"
   CGroup: /system.slice/mysqld.service
           └─32649 /usr/sbin/mysqld

Feb 21 10:31:51 ip-192-168-1-49.ap-south-1.compute.internal systemd[1]: Starting MySQL Server...
Feb 21 10:31:58 ip-192-168-1-49.ap-south-1.compute.internal systemd[1]: Started MySQL Server.
[root@ip-192-168-1-49 ec2-user]#
```

After enabling it you can see it start running on the port number 3306

```
[root@ip-192-168-1-49 ec2-user]# netstat -tnlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      2633/rpcbind
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      3207/sshd
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      3070/master
tcp6       0      0 :::111                  :::*                    LISTEN      2633/rpcbind
tcp6       0      0 :::22                   :::*                    LISTEN      3207/sshd
tcp6       0      0 :::33060                :::*                    LISTEN      32762/mysqld
tcp6       0      0 :::3306                 :::*                    LISTEN      32762/mysqld
[root@ip-192-168-1-49 ec2-user]#
```

While installing the mysql-server it configured with the temporary password we can get it from log file of the mysql-server

cat /var/log/mysqld.log | grep "A temporary password"

```
[root@ip-192-168-1-49 ec2-user]# cat /var/log/mysqld.log | grep "A temporary password"
2023-02-21T10:31:53.806552Z 6 [Note] [MY-010454] [Server] A temporary password is generated for root@localhost: xr570fsbNx-e
[root@ip-192-168-1-49 ec2-user]#
```

Securing the mysql server with the new password

mysql_secure_installation : setting the mysql server with new credentials.

```
[root@ip-192-168-1-49 ec2-user]# mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root:

The existing password for the user account root has expired. Please set a new password.

New password:

Re-enter new password:
The 'validate_password' component is installed on the server.
The subsequent steps will run with the existing configuration
of the component.
Using existing password for root.

Estimated strength of the password: 100
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y

New password:

Re-enter new password:
```

Now login again to the public instance create a same file like config.sample.inc.php with the name config.inc.php in the web server root folder which consists of a phpMyAdmin folder.

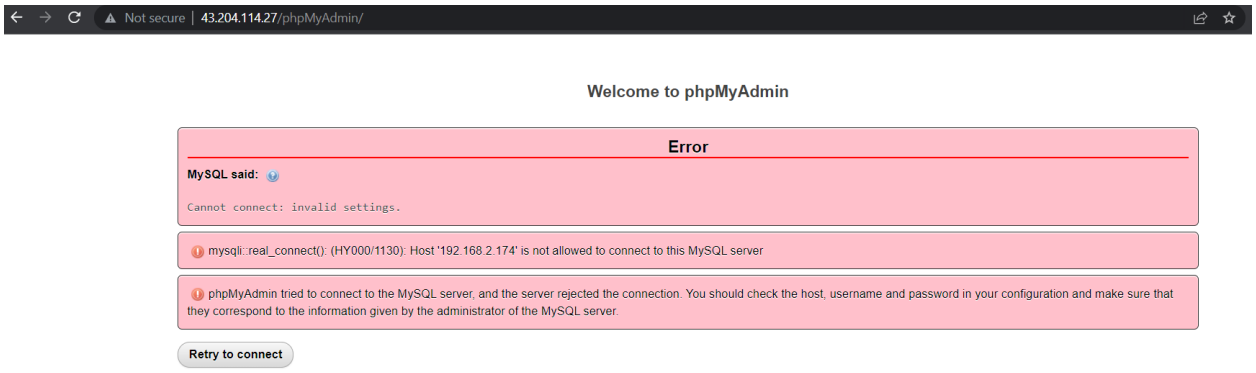
```
[root@ip-192-168-2-174 html]# cd phpMyAdmin/
[root@ip-192-168-2-174 phpMyAdmin]# ls
babel.config.json  composer.lock          doc          index.php  LICENSE       README              setup                  templates  vendor
ChangeLog          config.sample.inc.php  examples     js         locale        RELEASE-DATE-5.2.1  show_config_errors.php  themes     yarn.lock
composer.json      CONTRIBUTING.md        favicon.ico  libraries  package.json  robots.txt          sql                    url.php
[root@ip-192-168-2-174 phpMyAdmin]# cp config.sample.inc.php config.inc.php
```

Open the config.inc.php file edit the content as like below with your content. In my case my mysql server is running on the ip address 192.168.1.49 and port number 3306.

```
$i++;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'config';
/* Server parameters */
$cfg['Servers'][$i]['host'] = '192.168.1.49:3306';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = 'Krup@$123';
$cfg['Servers'][$i]['compress'] = false;
$cfg['Servers'][$i]['AllowNoPassword'] = true;
```

Accessing the phpMyAdmin webpage

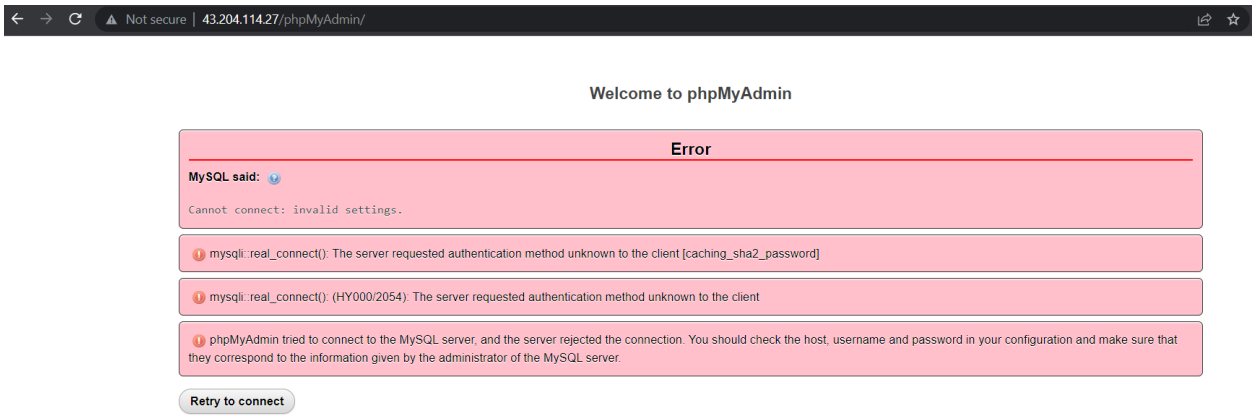
After accessing the phpMyAdmin web page revert back with the error message as 192.168.2.174 ip address of the phpMyAdmin server (public instance) which does not have access to mysql-server.



Created a root user with the private address of the phpMyAdmin instance and granted full access to the instance.

```
[root@ip-192-168-1-49 ec2-user]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> USE mysql;
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> CREATE USER 'root'@'192.168.2.174' IDENTIFIED BY 'Krup@$123';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT ALL ON *.* TO 'root'@'192.168.2.174';
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)
```

Again it revert back with the **caching_sha2_password** as authentication which is unknown we have to add the authentication method.

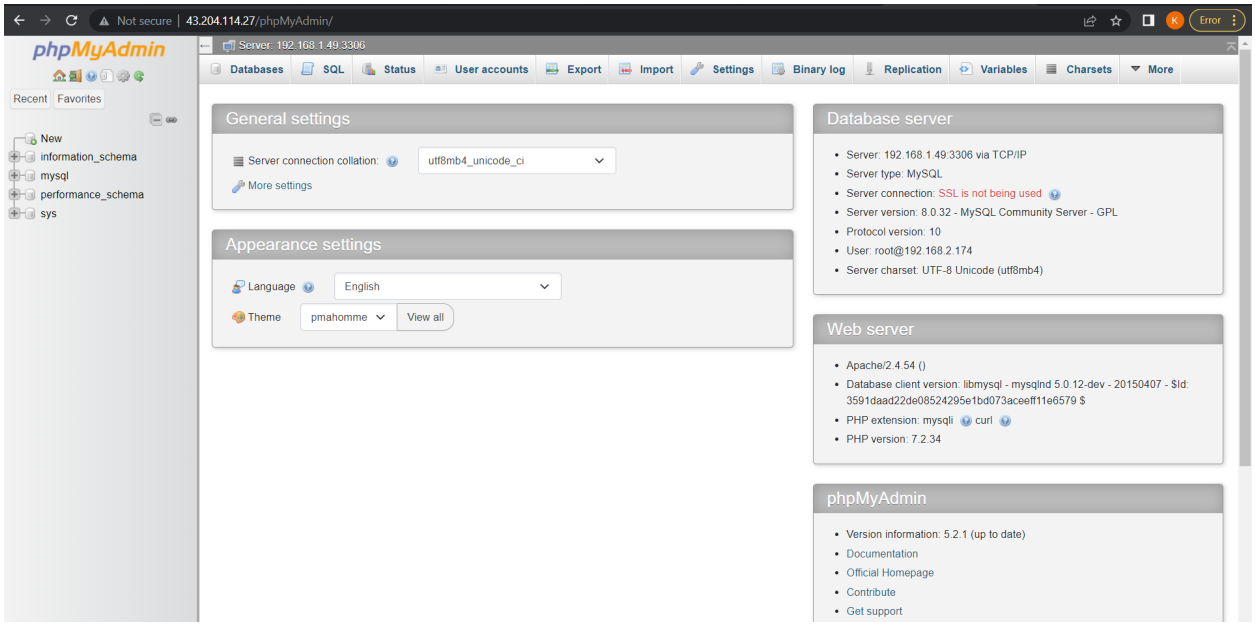


For that have to add the authentication method as **default-authentication-plugin=mysql_native_password** in the /etc/my.cnf configuration file

```
[root@ip-192-168-1-49 ec2-user]# cat /etc/my.cnf
# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/8.0/en/server-configuration-defaults.html

[mysqld]
#
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove the leading "# " to disable binary logging
# Binary logging captures changes between backups and is enabled by
# default. It's default setting is log_bin=binlog
# disable_log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M
#
# Remove leading # to revert to previous value for default_authentication_plugin,
# this will increase compatibility with older clients. For background, see:
# https://dev.mysql.com/doc/refman/8.0/en/server-system-variables.html#sysvar_default_authentication_plugin
default-authentication-plugin=mysql_native_password
```

Alter the previously created root user with the mysql_native_password
alter user 'root'@'192.168.2.174' identified with mysql_native_password by 'Krup@$123';

```
[root@ip-192-168-1-49 ec2-user]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> alter user 'root'@'192.168.2.174' identified with mysql_native_password by 'Krup@$123';
Query OK, 0 rows affected (0.01 sec)

mysql>
```



Finally phpMyadmin webpage was accessed successfully.