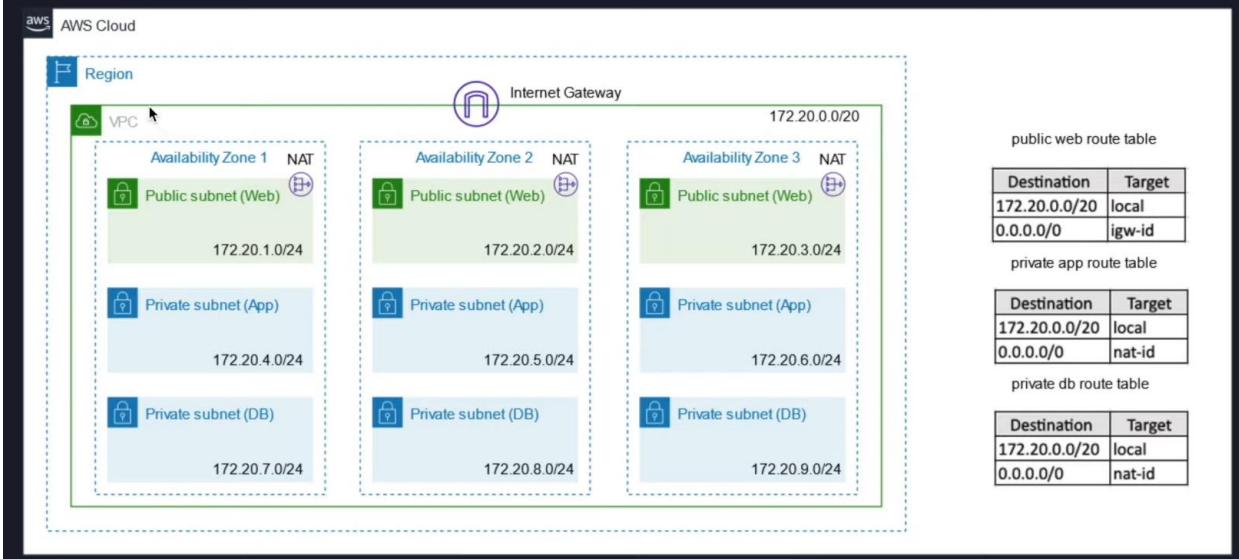
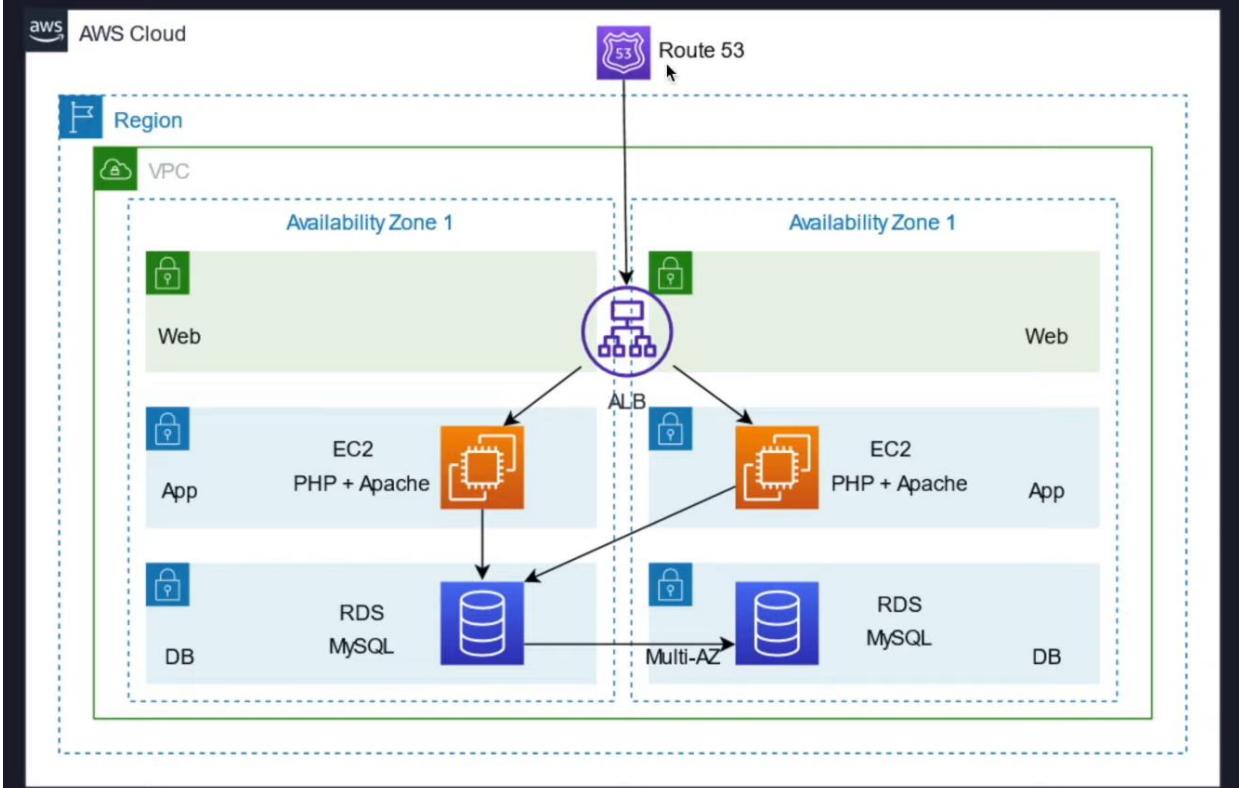


AWS network architecture



Services used:- VPC, ALB, EC2, RDS, Route53, Amazon Certificate Mnagaer

Three tier app architecture



Step 1:- Select Any Region as per your choice, I choosed Ohio

Step 2:- Create VPC in Ohio region with IPv4 CIDR 172.20.0.0/20

| vpc-05aa6573cb454103c / 3-tier-vpc | | | |
|---|--|---|---|
| Details | | Info | |
| VPC ID vpc-05aa6573cb454103c | State Available | DNS hostnames Disabled | DNS resolution Enabled |
| Tenancy Default | DHCP options set dopt-023069ec5f4ff623e | Main route table rtb-0e79957743db6d975 | Main network ACL acl-0eb1115278ec4ede0 |
| Default VPC No | IPv4 CIDR 172.20.0.0/20 | IPv6 pool - | IPv6 CIDR - |
| Route 53 Resolver DNS Firewall rule groups - | Owner ID 829912339674 | | |

Step 3:- Create 9 subnets for 3 servers

Public-web-subnet-1 → us-east-2a → 172.20.1.0/24

Public-web-subnet-2 → us-east-2b → 172.20.2.0/24 For Jump Server instance-1

Public-web-subnet-3 → us-east-2c → 172.20.3.0/24

.....

Private-App-subnet-1 → us-east-2a → 172.20.4.0/24

Private -App-subnet-2 → us-east-2b → 172.20.5.0/24 For App Server instance-2

Private -App-subnet-3 → us-east-2c → 172.20.6.0/24

.....

Private-db-subnet-1 → us-east-2a → 172.20.7.0/24

Private -db-subnet-2 → us-east-2b → 172.20.8.0/24 For db Server instance-3

Private -db-subnet-3 → us-east-2c → 172.20.9.0/24

| Name | Subnet ID | State | VPC | IPv4 CIDR | IPv6 CIDR | Available IPv4 |
|-----------------------|---------------------------|-----------|---------------------------------|---------------|-----------|----------------|
| Public-web-subnet-1 | subnet-0f90937e86c52c758 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.1.0/24 | - | 251 |
| Public-web-subnet-2 | subnet-0b557e53fc4ca0273 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.2.0/24 | - | 251 |
| Private -App-subne... | subnet-0f573d7089f5c8c47 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.5.0/24 | - | 251 |
| Private -db-subnet-2 | subnet-0be612db8e172328fe | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.8.0/24 | - | 251 |
| Private -db-subnet-3 | subnet-0748139fc09ea4227 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.9.0/24 | - | 251 |
| Private -db-subnet-1 | subnet-0791f13da4a663b17 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.7.0/24 | - | 251 |
| Private -App-subne... | subnet-00f381f0a87bf7327 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.6.0/24 | - | 251 |
| Public-web-subnet-3 | subnet-0c52a2a71b2a851df | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.3.0/24 | - | 251 |
| Private-App-subnet-1 | subnet-0c5af36e2f2fe290 | Available | vpc-05aa6573cb454103c 3-ti... | 172.20.4.0/24 | - | 251 |

Step 4: - Create 3 route table for 3 instances don't use by default route table of created VPC

Web-server-rt, App-server-rt, db-server-rt

| Name | Route table ID | Explicit subnet associat... | Edge associations | Main | VPC | Owner ID |
|---------------|------------------------|-----------------------------|-------------------|------|---------------------------------|--------------|
| - | rtb-0e79957743db6d975 | - | - | Yes | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| web-server-rt | rtb-00b9692d31202afb9 | - | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| App-server-rt | rtb-07e3081c7c6e8ae1a | - | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| db-server-rt | rtb-05295c52cae57603 | - | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| - | rtb-07a809fac0c95beeaa | - | - | Yes | vpc-08f53a58058ba642 | 829912339674 |

Step 5: - Associate all subnet to related specific route tables.

Web-server-rt

Route tables > rtb-00b9692d31202afb9 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (3/9)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR | Route table ID |
|---------------------|--------------------------|---------------|-----------|------------------------------|
| Public-web-subnet-2 | subnet-0b557e53fc4ca0273 | 172.20.2.0/24 | - | Main (rtb-0e79957743db6d975) |
| Public-web-subnet-1 | subnet-0f90937e86c32c758 | 172.20.1.0/24 | - | Main (rtb-0e79957743db6d975) |
| Public-web-subnet-3 | subnet-0c52a2a71b2a851df | 172.20.3.0/24 | - | Main (rtb-0e79957743db6d975) |

Selected subnets

subnet-0b557e53fc4ca0273 / Public-web-subnet-2 X subnet-0f90937e86c32c758 / Public-web-subnet-1 X subnet-0c52a2a71b2a851df / Public-web-subnet-3 X

Cancel Save associations

App-server-rt

VPC > Route tables > rtb-07e3081c7c6e8ae1a > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

| Available subnets (3/9) | | | | | | |
|---|-----------------------|--------------------------|---------------|-----------|------------------------------|--|
| <input type="text"/> Filter subnet associations | | | | | | |
| <input checked="" type="checkbox"/> | Name | Subnet ID | IPv4 CIDR | IPv6 CIDR | Route table ID | |
| <input checked="" type="checkbox"/> | Private -App-subnet-2 | subnet-0f373d7089f5c8c47 | 172.20.5.0/24 | - | Main (rtb-0e79957743db6d975) | |
| <input checked="" type="checkbox"/> | Private -App-subnet-3 | subnet-00f581f0a87bf7327 | 172.20.6.0/24 | - | Main (rtb-0e79957743db6d975) | |
| <input checked="" type="checkbox"/> | Private-App-subnet-1 | subnet-0c5af36e2f2fe296 | 172.20.4.0/24 | - | Main (rtb-0e79957743db6d975) | |

Selected subnets

subnet-0f373d7089f5c8c47 / Private -App-subnet-2 subnet-00f581f0a87bf7327 / Private -App-subnet-3 subnet-0c5af36e2f2fe296 / Private-App-subnet-1

Cancel

Db-server-rt

VPC > Route tables > rtb-052a95c52cae37603 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

| Available subnets (3/9) | | | | | | |
|---|----------------------|--------------------------|---------------|-----------|------------------------------|--|
| <input type="text"/> Filter subnet associations | | | | | | |
| <input checked="" type="checkbox"/> | Name | Subnet ID | IPv4 CIDR | IPv6 CIDR | Route table ID | |
| <input checked="" type="checkbox"/> | Private -db-subnet-3 | subnet-0748139fc09e1a427 | 172.20.9.0/24 | - | Main (rtb-0e79957743db6d975) | |
| <input checked="" type="checkbox"/> | Private -db-subnet-2 | subnet-0be612d8e172328fe | 172.20.8.0/24 | - | Main (rtb-0e79957743db6d975) | |
| <input checked="" type="checkbox"/> | Private-db-subnet-1 | subnet-0751f13da4a663b17 | 172.20.7.0/24 | - | Main (rtb-0e79957743db6d975) | |

Selected subnets

subnet-0748139fc09e1a427 / Private -db-subnet-3 subnet-0be612d8e172328fe / Private -db-subnet-2 subnet-0751f13da4a663b17 / Private-db-subnet-1

Cancel

Succcessfully all subnet are explicite associated with their route tables.

aws Services Search for services, features, blogs, docs, and more [Alt+S]

New VPC Experience Tell us what you think

VPC Dashboard EC2 Global View New Filter by VPC: Select a VPC

VIRTUAL PRIVATE CLOUD Your VPCs

Route tables (3) Info

Filter route tables search: server-rt

| <input type="checkbox"/> | Name | Route table ID | Explicit subnet associat... | Edge associations | Main | VPC | Owner ID |
|--------------------------|---------------|-----------------------|-----------------------------|-------------------|------|---------------------------------|--------------|
| <input type="checkbox"/> | web-server-rt | rtb-00b9692d31202afb9 | 3 subnets | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| <input type="checkbox"/> | App-server-rt | rtb-07e3081c7c6e8ae1a | 3 subnets | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |
| <input type="checkbox"/> | db-server-rt | rtb-052a95c52cae37603 | 3 subnets | - | No | vpc-05aa6573cb454103c 3-ti... | 829912339674 |

Step 6: - Create internet gateway → Attach to VPC → do entry in **web-server-rt**

The screenshot shows the 'Edit routes' section of a route table. A new route is being added with a destination of '0.0.0.0/0'. The target is set to 'igw-0c391122185730047 (3-tier-igway)'. The status is 'Active' and propagation is set to 'No'. There is a 'Remove' button next to the target field. At the bottom right are 'Cancel', 'Preview', and 'Save changes' buttons.

Step 7: - Create NAT Gateway → Attach to **public-web-subnet-1** → Allocate Elastic Ip

The screenshot shows the 'NAT gateways' page. A new NAT gateway named '3-tier-NAT-Gateway' is listed. It has a NAT gateway ID of 'nat-0620e5ca23d1c9243', is in a 'Public' state, and is pending. It has an elastic IP address of '172.20.1.168' and a private IP address of 'eni-01aa0f4e'. The 'Create NAT gateway' button is visible at the top right.

Attach NAT gateway to the **App-server-rt & db-server-rt**

The screenshot shows the 'rtb-04955e7870ca987e7 / App-server-rt' route table. In the 'Routes' section, there are two entries: one to 'local' and another to 'nat-0620e5ca23d1c9243'. Both routes are active and not propagated. The 'Edit routes' button is visible at the top right of the routes table.

VPC > Route tables > rtb-083641e2043f56bbe

rtb-083641e2043f56bbe / db-server-rt

Details [Info](#)

| | | | |
|---|--------------------------|---|------------------------|
| Route table ID rtb-083641e2043f56bbe | Main No | Explicit subnet associations 3 subnets | Edge associations - |
| VPC vpc-03938b5bda53c285d 3-tier-vpc | Owner ID 829912339674 | | |

Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

| Destination | Target | Status | Propagated |
|---------------|-----------------------|--------|------------|
| 172.20.0.0/20 | local | Active | No |
| 0.0.0.0/0 | nat-0620e5ca23d1c9243 | Active | No |

[Edit routes](#)

Step 7:- Create 3 EC2 instances with

1. Public-Jump-server (Public+Private)
2. Private-App-server (Private IP only)
3. Private-db-server (Private IP only)

App-server and db-server is only for routing traffic from both server, if any server failovers.

Public-Jump-Server:-

- Create new security group with ssh & http [jump-server-sg]
- Public-web-subnet-1
- Auto assign public IP enable

aws Services Search for services, features, blogs, docs, and more [Alt+S]

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View Events Tags Limits

Instances (1/1) [Info](#)

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|--------------------|---------------------|----------------|---------------|--------------|--------------|-------------------|
| Public-Jump-Server | i-07ebf0aab55d16d4a | Pending | t2.micro | - | No alarms | us-east-2a |

Instance: i-07ebf0aab55d16d4a (Public-Jump-Server)

| Details | Security | Networking | Storage | Status checks | Monitoring | Tags |
|---|---|--|---------|---------------|------------|------|
| Instance ID i-07ebf0aab55d16d4a (Public-Jump-Server) | Public IPv4 address 13.59.207.169 open address | Private IPv4 addresses 172.20.1.225 | | | | |
| IPv6 address - | Instance state Pending | Public IPv4 DNS - | | | | |
| Hostname type | Private IP DNS name (IPv4 only) | Answer private resource DNS name | | | | |

Private-App-Server:-

- Private-App-subnet-1
- Auto assign public IP disable
- Create Security group [App-server-sg]

SSH → custom → jump-server-sg

Http → Anywhere

The screenshot shows the AWS Management Console with the EC2 service selected. The left sidebar shows navigation options like New EC2 Experience, EC2 Dashboard, and Instances. The main pane displays the 'Instances (1/2)' list. It shows two instances:

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone |
|--------------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|
| Private-App-Server | i-081fdc0f814ebe03c | Running | t2.micro | Initializing | No alarms | us-east-2a |
| Public-Jump-Server | i-07ebf0aab55d16d4a | Running | t2.micro | 2/2 checks passed | No alarms | us-east-2a |

Below the list, a detailed view for the 'Private-App-Server' instance is shown. The 'Details' tab is selected, displaying information such as Instance ID (i-081fdc0f814ebe03c), Public IPv4 address (172.20.4.185), and Instance state (Running).

Private-db-Server:-

- Private-db-subnet-1
- Auto assign public IP disable
- Choose existing security group [App-server-sg] of Private-App-server

The screenshot shows the 'Launch instance' wizard. The first step, 'Network settings', is completed. The second step, 'Summary', is shown. The summary includes:

- Number of instances: 1
- Software Image (AMI): Amazon Linux 2 Kernel 5.10 AMI... (ami-0fa9ccfdcb062c84)
- Virtual server type (instance type): t2.micro
- Firewall (security group): App-server-sg (selected)
- Storage (volumes): 1 volume(s) - 8 GiB

A note about the free tier is displayed: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 1000 API calls per month.'

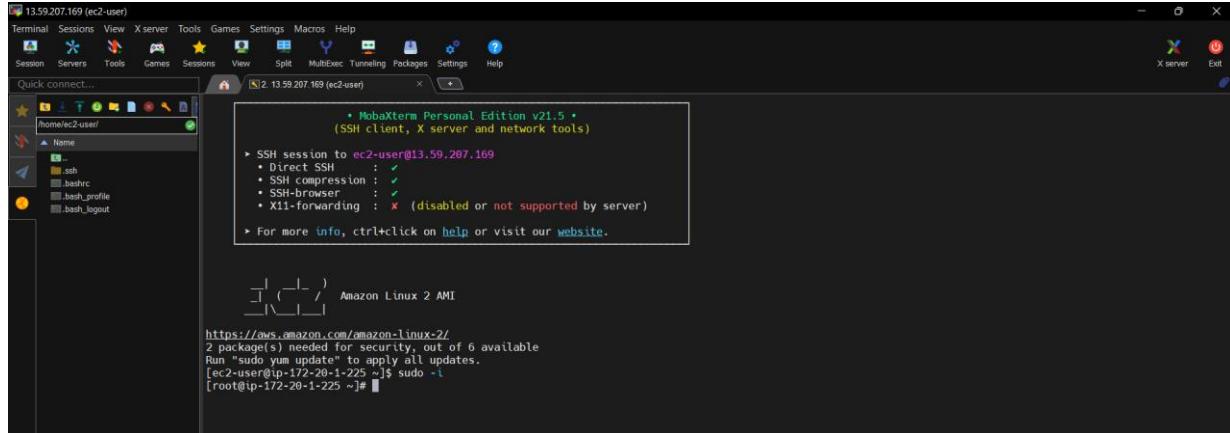
The screenshot shows the AWS Management Console with the EC2 service selected. The main pane displays a table of instances. The first instance, 'Private-db-server', is selected. Below the table, a detailed view for 'Private-db-server' is shown, including its instance ID, state (Running), type (t2.micro), and various network and monitoring details.

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS |
|--------------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|-----------------|
| Private-db-server | i-010ce108831528652 | Running | t2.micro | Initializing | No alarms | us-east-2a | - |
| Private-App-Server | i-081fd0f814eb03c | Running | t2.micro | 2/2 checks passed | No alarms | us-east-2a | - |
| Public-Jump-Server | i-07ebf0aab55d16d4a | Running | t2.micro | 2/2 checks passed | No alarms | us-east-2a | - |

Step 8:- Connect public-jump-server and get SSH into both private servers (App-server & db-server)

- Install php and apache
- Install LAMP web server on the Amazon Linux AMI

Public-Jump-server connect with moba successfully



Now, we have to connect Private-App-Serve and Private-db-server for installing php and lamp server

- Open 2 tabs of jump-server
- Upload 3-tier-key pair in moba both tabs
- In one tab of jump-server get ssh of Private-App-server and same another tab get of jump-server get ssh of Private-db-server

```
# ssh -i /home/ec2-user/3-tier-key.pem ec2-user@<Private-ip of app-server>
```

```

13.59.207.169 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
      • MobaXterm Personal Edition v21.5 •
      (SSH client, X server and network tools)

> SSH session to ec2-user@13.59.207.169
  • Direct SSH : ✓
  • SSH compression : ✓
  • SSH-browser : ✓
  • X11-forwarding : ✘ (disabled or not supported by server)

> For more info, ctrl+click on help or visit our website.

[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 6 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-20-1-225 ~]$ sudo -i
[root@ip-172-20-1-225 ~]# ssh -l /home/ec2-user/3-tier-key.pem ec2-user@172.20.4.185
The authenticity of host '172.20.4.185' (172.20.4.185) can't be established.
EDSA key fingerprint is SHA256:729EF31Y3uBo0B8AM/KqFC7wJks/4AdmMehT9G3JA.
EDSA key fingerprint is MD5:5fcb:ef:08:ce:07:b7:8b:a5:57:fc:1c:75:03:bd:01:d4.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.20.4.185' (EDSA) to the list of known hosts.

[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 6 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-20-4-185 ~]$ 

```

ssh -i /home/ec2-user/3-tier-key.pem ec2-user@<Private-ip of db-server>

```

13.59.207.169 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
      • MobaXterm Personal Edition v21.5 •
      (SSH client, X server and network tools)

> SSH session to ec2-user@13.59.207.169
  • Direct SSH : ✓
  • SSH compression : ✓
  • SSH-browser : ✓
  • X11-forwarding : ✘ (disabled or not supported by server)

> For more info, ctrl+click on help or visit our website.

[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
Last login: Mon May 30 04:32:04 2022 from 202.142.121.39
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 6 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-20-1-225 ~]$ sudo -i
[root@ip-172-20-1-225 ~]# ssh -l /home/ec2-user/3-tier-key.pem ec2-user@172.20.7.19
The authenticity of host '172.20.7.19' (172.20.7.19) can't be established.
EDSA key fingerprint is SHA256:05ZvhmYK0ImLXE1WM41V1tK94IpHyMF+FKnw.
EDSA key fingerprint is MD5:65:6e:e4:96:bb:3a:d8:eb:33:91:5a:fd:69:44:80:ce:7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.20.7.19' (EDSA) to the list of known hosts.

[13.59.207.169 (ec2-user)] x [4. 13.59.207.169 (ec2-user)] x
https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 6 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-20-7-19 ~]$ 

```

Step 9:- Install Lamp server into the both Private instance via getting SSH from moba

- Install Lamp server
- Set file permission
- Install phpMyAdmin

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/install-LAMP.html> refer the Doc

Prepare the Lamp server

```
[ec2-user ~]$ sudo yum update -y  
[ec2-user ~]$ sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2  
cat /etc/system-release  
[ec2-user ~]$ sudo yum install -y httpd mariadb-server  
[ec2-user ~]$ sudo systemctl start httpd  
[ec2-user ~]$ sudo systemctl enable httpd  
[ec2-user ~]$ sudo systemctl is-enabled httpd  
[ec2-user ~]$ curl localhost
```

To set file permissions

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

Install my phpMyAdmin

```
[ec2-user ~]$ sudo yum install php-mbstring php-xml -y  
[ec2-user ~]$ sudo systemctl restart httpd  
[ec2-user ~]$ sudo systemctl restart php-fpm  
[ec2-user ~]$ cd /var/www/html  
[ec2-user html]$ wget  
https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz  
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C  
phpMyAdmin --strip-components 1  
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz  
[ec2-user ~]$ sudo systemctl start mariadb
```

Step 10: - Create Target group [*TgforALB*]

- Target type: - Instances
- Register target for both private instances

Step 1
Specify group details

Step 2
Register targets

Available instances (2/3)

| Instance ID | Name | State | Security groups | Zone | Subnet ID |
|---------------------|--------------------|---------|-----------------|------------|--------------------------|
| i-010ce108831528652 | Private-db-server | running | App-server-sg | us-east-2a | subnet-00cc6ccc2d495db5e |
| i-081fdc0f814eb03c | Private-App-Server | running | App-server-sg | us-east-2a | subnet-02a3650c21f4361f7 |
| i-07ebf0aab55d16d4a | Public-Jump-Server | running | jump-server-sg | us-east-2a | subnet-00676a115181ee6c1 |

2 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with commas)

Include as pending below

Step 11:- Create Application Load balancer

VPC:- 3- tier-vpc

Mapping:- us-east-2a (public-web-subnet-1)

us-east-2b (public-web-subnet-2)

Attach saperately created security group for load balancer

Attach created **TgforALB**

Create New Security group for Application Load Balancer

EC2 > Security Groups > Create security group

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info

Name cannot be edited after creation.

Description Info

VPC Info

Inbound rules Info

| Type <small>Info</small> | Protocol <small>Info</small> | Port range <small>Info</small> | Source <small>Info</small> | Description - optional <small>Info</small> | Delete |
|--------------------------|------------------------------|--------------------------------|---|--|---------------------------------------|
| Custom TCP | TCP | 80 | <input type="text" value="0.0.0.0/0"/> <input type="button" value="X"/> | <input type="text" value="0.0.0.0/0"/> | <input type="button" value="Delete"/> |

Add rule

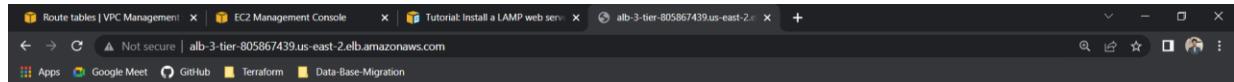
Application Load balancer created successfully

Step 12:- check whether it's working or not ?

```
echo "Hello chetan" > /var/www/html/index.html → Private-App-server
```

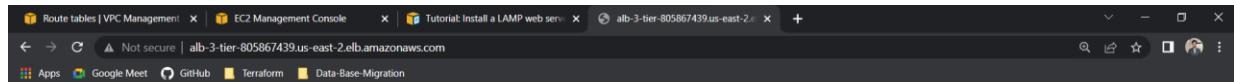
```
echo "Hello Jarvis" > /var/www/html/index.html → Private-db-server
```

Hit DNS and check



Hello Jarvis

After HARD Refreshing



Hello chetan

Both pages are shown successfully

Step 13:- Create Subnet group for 3-tier rds database [[subnetgroup-3-tier](#)]

VPC:- 3-tier-vpc

Subnets selected (2)

| Availability zone | Subnet ID | CIDR block |
|-------------------|--------------------------|---------------|
| us-east-2a | subnet-00cc6ccc2d495db5e | 172.20.7.0/24 |
| us-east-2b | subnet-0fe58e506bfde6cc9 | 172.20.8.0/24 |

Step 14:- Create RDS Database

Standard Create

Engine:- MySQL

Template:- Free tier

DB instance identifier:- jarvis-db

Credentials:- admin/admin123456

VPC:- 3-tier VPC

Subnet group:- automatically select created subnet group subnetgroup-3-tier

Public Access: - NO

Security group: - create New

Availability zone: - No preferences

| | | | | | | | |
|---------------|-----------|------------------|---|--------|-----------------|-------------|-------------|
| DB identifier | jarvis-db | CPU | - | Status | Creating | Class | db.t3.micro |
| Role | Instance | Current activity | | Engine | MySQL Community | Region & AZ | us-east-2a |

| | | |
|-----------------|---|--|
| Endpoint & port | Networking | Security |
| Endpoint | Availability Zone us-east-2a | VPC security groups rds-3tier-security-group (sg-0319f3012d5e7e878) Active |
| Port | VPC 3-tier-vpc (vpc-03938b5bda53c285d) | Publicly accessible No |
| | Subnet group subnetgroup-3-tier | Certificate authority rds-ca-2019 |
| | Subnets subnet-00cc6ccc2d495db5e subnet-0fe58e506bfde6cc9 | Certificate authority date August 23, 2024, 10:20:00 UTC +10:00 |

Succesfully created Database

Step 15:- Modify security group of jarvis-db

Here add Private-App-server security group [App-server-sg] into the jarvis db security group

Inbound rules

| Security group rule ID | Type | Protocol | Port range | Source | Description - optional |
|------------------------|------------|----------|------------|--------|------------------------|
| - | Custom TCP | TCP | 3306 | Custom | |
| Add rule | | | | | |

Source: App-server-sg | sg-07ebdc12370110a70

Description: app

Buttons: Cancel, Preview changes, Save rules

Also add HTTP

Inbound rules

| Security group rule ID | Type | Protocol | Port range | Source | Description - optional |
|------------------------|------------|----------|------------|----------|------------------------|
| - | Custom TCP | TCP | 3306 | Custom | |
| - | HTTP | TCP | 80 | Anywhere | sg-07ebdc12370110a70 |
| Add rule | | | | | |

Source: Anywhere-IP

Description: app

Buttons: Cancel, Preview changes, Save rules

Sucsessfully modified rds escurity group

Inbound security group rules successfully modified on security group (sg-0319f3012d5e7e878 | rds-3tier-security-group)

Details

Security Groups (1/6)

| Name | Security group ID | Security group name | VPC ID | Description | Owner | Inbound rules count | Outbound rule |
|----------------------|--------------------------|----------------------|---------------------------|--------------|----------------------|---------------------|---------------|
| sg-0319f3012d5e7e878 | rds-3tier-security-group | vpc-0393b5bda53c285d | Created by RDS manag... | 829912339674 | 2 Permission entries | 1 Permission en | |
| sg-04078affcf52a251 | jump-server-sg | vpc-0393b5bda53c285d | launch-wizard created ... | 829912339674 | 2 Permission entries | 1 Permission en | |

sg-0319f3012d5e7e878 - rds-3tier-security-group

Inbound rules (2)

| Name | Security group rule... | IP version | Type | Protocol | Port range | Source |
|-----------------------|------------------------|------------|--------------|----------|------------|--------------------------------------|
| sgr-037d96fb886cf6a8 | - | - | MySQL/Aurora | TCP | 3306 | sg-07ebdc12370110a70 / App-server-sg |
| sgr-0cadcc541ee2b9514 | IPv4 | - | HTTP | TCP | 80 | 0.0.0.0/0 |

Buttons: Actions, Export security groups to CSV, Create security group, Manage tags, Edit inbound rules

Step 16:- Rename the file for identification in both servers

```
# cd /var/www/html/phpMyAdmin/
```

```
# ls
# mv config.sample.inc.php config.inc.php
# ls
# vim config.inc.php
```

Here line no 30 paste the endpoint of [Jarvis-db] instead of localhost

| DB identifier | CPU | Status | Class |
|---------------|------------------|-----------------|-------------|
| jarvis-db | 3.54% | Available | db.t3.micro |
| Role | Current activity | Engine | Region & AZ |
| Instance | 0 Connections | MySQL Community | us-east-2a |

Paste in Private-App-server

```
</php>
/*
 * phpMyAdmin sample configuration, you can use it as base for
 * manual configuration. For easier setup you can use setup/
 *
 * All directives are explained in documentation in the doc/ folder
 * or at <https://docs.phpmyadmin.net/>.
 */
declare(strict_types=1);

/**
 * This is needed for cookie based authentication to encrypt password in
 * cookie. Needs to be 32 chars long.
 */
$cfg['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

/**
 * Servers configuration
 */
$i = 0;

/**
 * First server
 */
$i++;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$cfg['Servers'][$i]['host'] = 'jarvis-db.cdsxitiv1prd.us-east-2.rds.amazonaws.com';
$cfg['Servers'][$i]['compress'] = false;
$cfg['Servers'][$i]['AllowNoPassword'] = false;

/**
 * phpMyAdmin configuration storage settings.
 */
/* User used to manipulate with storage */
// $cfg['Servers'][$i]['controlhost'] = '';
// $cfg['Servers'][$i]['controlport'] = '';
// $cfg['Servers'][$i]['controluser'] = 'pma';
-- INSERT --
```

Also paste in Private-db-server

```

<?php
/*
 * phpMyAdmin sample configuration, you can use it as base for
 * manual configuration. For easier setup you can use setup/
 *
 * All directives are explained in documentation in the doc/ folder
 * or at <https://docs.phpmyadmin.net/>.
 */
declare(strict_types=1);

/**
 * This is needed for cookie based authentication to encrypt password in
 * cookie. Needs to be 32 chars long.
 */
$cfg['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

/**
 * Servers configuration
 */
$i = 0;

/**
 * First server
 */
++$i;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$cfg['Servers'][$i]['host'] = 'jarvis-db.cdxltiv1prd.us-east-2.rds.amazonaws.com';
$cfg['Servers'][$i]['compress'] = false;
$cfg['Servers'][$i]['AllowNoPassword'] = false;

/**
 * phpMyAdmin configuration storage settings.
 */
/* User used to manipulate with storage */
// $cfg['Servers'][$i]['controlhost'] = '';
// $cfg['Servers'][$i]['controlport'] = '';
// $cfg['Servers'][$i]['controluser'] = 'pma';
-- INSERT --

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Step 17:- Go to created target group **TgforALB** of database and enable stickiness.

Attributes --> Edit --> Enable Stickiness

The screenshot shows the AWS Lambda console with the URL <https://console.aws.amazon.com/lambda/home#/functions/TgforALB>. The 'Edit attributes' dialog is open, showing the following configuration:

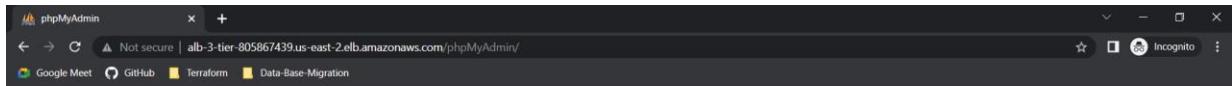
- Deregistration delay:** 300 seconds (0-3600)
- Slow start duration:** 0 seconds (0-900)
- Load balancing algorithm:**
 - Round robin
 - Least outstanding requestsCannot be combined with the Slow start duration attribute.
- Stickiness:**

The type of stickiness associated with this target group. If enabled, the load balancer binds a client's session to a specific instance within the target group.

 - Load balancer generated cookie
 - Application-based cookie
- Stickiness duration:** 1 days

<http://alb-3-tier-805867439.us-east-2.elb.amazonaws.com/phpMyAdmin/index.php>

Hit DNS of Application load balancer/phpMyAdmin and login via masteruser of RDS Database



Welcome to phpMyAdmin

Language

English

Log In

Username: admin

Password: *****

Log in



We can successfully able to login and we can also able to create database and their entries in private application web server.

Step 18:- Create Hosted Zone in **Route53** with specified Domain Name referd in Freenom.com

- As I get **somkuwar.ml**, so I created **somkuwar.ml** named hosted zone
- Do **NS records entry in your Somkuwa.ml Domain in Freenom.com**
- Create A record www/A-record/Alias/Application load balancer/Region/SimpleRouting

The screenshot shows the AWS Route 53 console with the domain 'somkuwar.ml' selected. The 'Records' section lists three entries:

- NS record for 'somkuwar.ml': Value/Route traffic to ns-1889.awsdns-44.co.uk, ns-875.awsdns-45.net, ns-1042.awsdns-02.org, ns-17.awsdns-02.com
- SOA record for 'somkuwar.ml': Value/Route traffic to ns-1889.awsdns-44.co.uk. awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400
- A record for 'www.somkuwar.ml': Value/Route traffic to dualstack.alb-3-tier-805867459.us-east-2.elb.amazonaws.com.

Do entry into purchased freenom domain

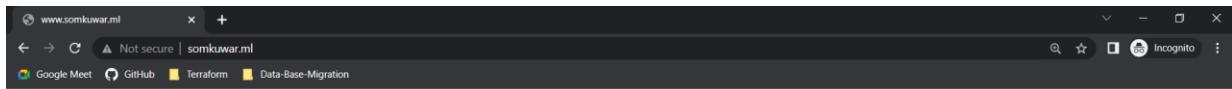
The screenshot shows the Freenom client area for managing the domain 'somkuwar.ml'. Under the 'Nameservers' section, the 'Use custom nameservers (enter below)' option is selected. Five input fields are present for Nameserver 1 through Nameserver 5, each containing a different AWS name server address:

- Nameserver 1: NS-1042.AWSDNS-02.ORG
- Nameserver 2: NS-17.AWSDNS-02.COM
- Nameserver 3: NS-1889.AWSDNS-44.CO.UK
- Nameserver 4: NS-875.AWSDNS-45.NET
- Nameserver 5: (empty)

Create A record in somkuwar.ml hosted zone in Route53 service

Now,

Hit www.somkuwar.ml → it shows my index page

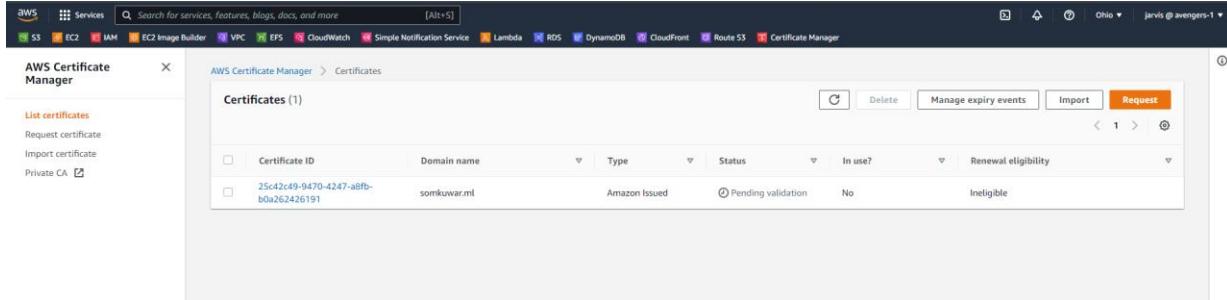


Hello chetan

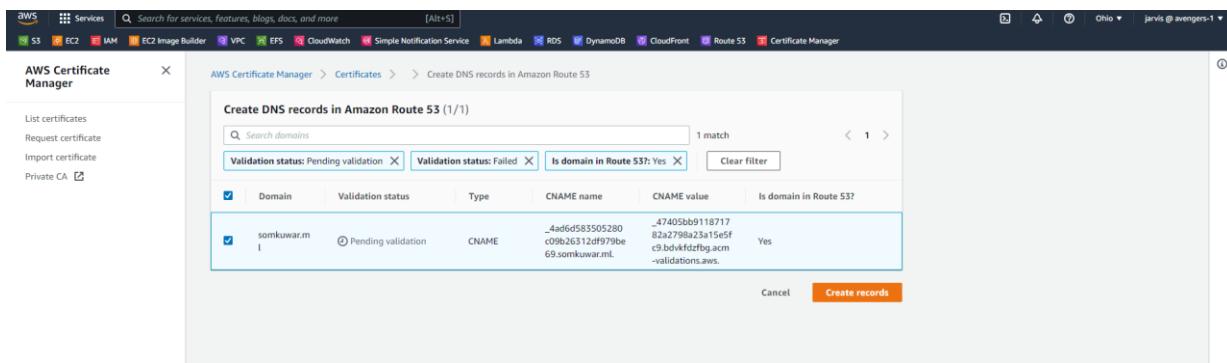
Hit www.somkuwar.ml/phpMyAdmin → it show my phpMyAdmin page and we also can also login their.

CURRENTLY IN TESTING

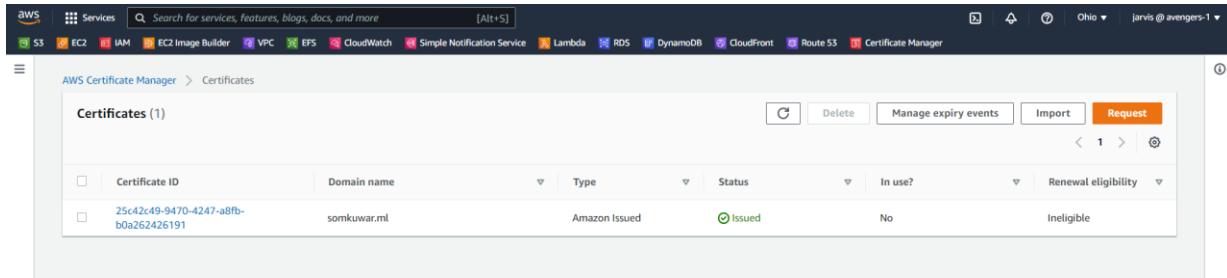
Step 19:- our website is not secured so now we issue SSL certificate from Amazon sertificate Manager and attach to our Domain then it becomes http → https



The screenshot shows the AWS Certificate Manager interface. In the left sidebar, under 'List certificates', there is one entry: '25c42c49-9470-4247-a8fb-b0a262426191' for domain 'somkuwar.ml'. The status is 'Pending validation'.



The screenshot shows the 'Create DNS records in Amazon Route 53' page. It lists a single record for domain 'somkuwar.ml' with type 'CNAME' and value '_4adfd583505280c09b26312df979be69.somkuwar.ml'. The status is 'Pending validation'.



The screenshot shows the AWS Certificate Manager interface again. The same certificate entry is now listed with status 'Issued'.

EC2 > Load balancers > ALB-3-tier : Edit listener

Edit listener

arn:aws:elasticloadbalancing:us-east-2:829912339674:listener/app/ALB-3-tier/622011fd51049dc0/715cc9d24d03b80d

Listener details

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

Protocol Port
HTTP ▲ : 80
HTTP 1-65535
HTTPS **Info**

Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

► 1. Forward to [Info](#) Remove

Add action ▾

Cancel Save changes

EC2 > Load balancers > ALB-3-tier : Edit listener

Edit listener

arn:aws:elasticloadbalancing:us-east-2:829912339674:listener/app/ALB-3-tier/622011fd51049dc0/715cc9d24d03b80d

Listener details

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

Protocol Port
HTTPS ▼ : 443
1-65535

Default actions [Info](#)

Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

▼ 1. Forward to [Info](#) Remove

Target group Weight (0-999)
TgforALB HTTP 1 100%
Target type: Instance, IPv4
Traffic distribution: 100%

Select a target group 0 Create target group [Info](#)

Enable group-level stickiness [Info](#)
If you enable stickiness for your target group, requests routed to it remain in the same group for the duration you specify.

Cancel Save changes

EC2 > Load balancers > ALB-3-tier : Edit listener

Edit listener

arn:aws:elasticloadbalancing:us-east-2:829912339674:listener/app/ALB-3-tier/622011fd51049dc0/715cc9d24d03b80d

Target group Weight (0-999)
TgforALB HTTP 1 100%
Target type: Instance, IPv4
Traffic distribution: 100%

Select a target group 0 Create target group [Info](#)

Enable group-level stickiness [Info](#)
If you enable stickiness for your target group, requests routed to it remain in the same group for the duration you specify.

Add action ▾

Secure listener settings [Info](#)

Security policy
Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections with clients.
ELBSecurityPolicy-2016-08

Compare security policies [Info](#)

Default SSL/TLS certificate
The certificate used if a client connects without SNI protocol, or if there are no matching certificates. This certificate will automatically be added to your listener certificate list.

From ACM somkuwar.ml 25c42c49-9470-4247-a8fb-b0a262425191 [C](#)

Request new ACM certificate [Info](#)

Cancel Save changes

Cancel Save changes

