

# Scalable 3-Tier Web Application on AWS

(Designing Robust Web Application on AWS)

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

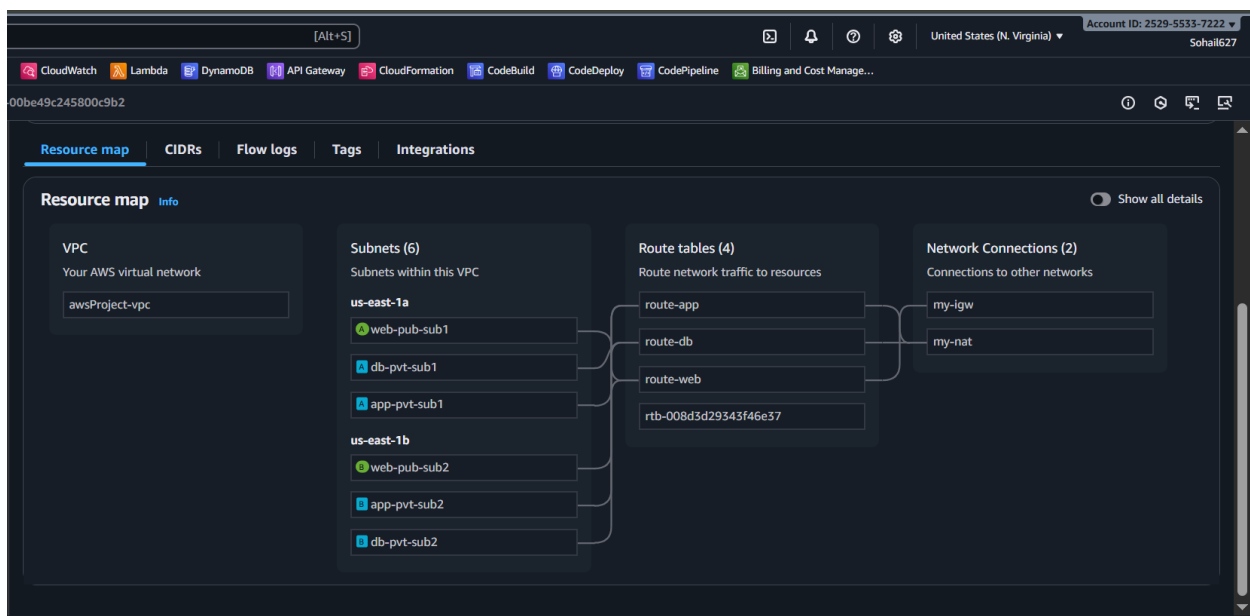
## Part 1 :- VPC Construct

- Create a VPC named `awsProject-vpc` with a CIDR block of `20.0.0.0/20`.
  - `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 public subnets:
  - `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 applications:
  - `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 2 private subnets for databases:
- 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 changes



## Part 2 :- Ec2 Instances

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
- 2) Create a security group as alb-sg and allow port HTTP (80)
- 3) 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
- 4) Launch an Instance with name app-server2, AMI (Amazon Linux), Instance type (t2.micro), Select the keypair as (projectkey) , click on edit button on right side of Network settings select vpc (awsProject-vpc), Subnet (app-pvt-sub2 ), Auto-assign IP (Disable), Select existing security group as (app-sg), now Launch instance

Name	Instance ID	Instance type	Status check	Alarm status	Availability	Public IP	Public IP	Elastic IP	IPv6 IPs	Monitoring
app-server2	i-0973b03e...	t2.micro	2/2 checks passed	View alarms	us-east-1b	-	-	-	-	disabled
app-server1	i-0123299...	t2.micro	2/2 checks passed	View alarms	us-east-1a	-	-	-	-	disabled
jump-server	i-06a69a22...	t2.micro	2/2 checks passed	View alarms	us-east-1a	54.175.171...	-	-	-	disabled

Public IP	IPv6 IPs	Monitoring	Security group name	Key name	Launch time	Platform	Managed	Operator
-	-	disabled	app-sg	projectkey	2025/08/31 14:20 GMT+5:30	Linux/UNIX	false	-
-	-	disabled	app-sg	projectkey	2025/08/31 14:19 GMT+5:30	Linux/UNIX	false	-
-	-	disabled	jump-sg	projectkey	2025/08/31 14:13 GMT+5:30	Linux/UNIX	false	-

## Part 3 :- Server configuration

1. Copy the private key in your jump-server instance

```
SOHAIL@SOHAIL MINGW64 /e
$ scp -i projectkey.pem projectkey.pem ec2-user@54.175.171.131:/home/ec2-user/
projectkey.pem                                100% 1674          0.9KB/s   00:01

[ec2-user@ip-20-0-1-40 ~]$ ls
[ec2-user@ip-20-0-1-40 ~]$ pwd
/home/ec2-user
[ec2-user@ip-20-0-1-40 ~]$ ls
projectkey.pem
[ec2-user@ip-20-0-1-40 ~]$ chmod 400 projectkey.pem
[ec2-user@ip-20-0-1-40 ~]$
```

2. **Now ssh into app-server1 and run the below commands**

```
[ec2-user@ip-20-0-1-40 ~]$ ssh -i projectkey.pem ec2-user@20.0.3.30  
The authenticity of host '20.0.3.30 (20.0.3.30)' can't be established.  
ED25519 key fingerprint is SHA256:1++68a+ZODSJTO4PUlvzbzbcIJjgIIkL+a9It+rElSI.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '20.0.3.30' (ED25519) to the list of known hosts.
```

```
#_
~\##### Amazon Linux 2023
~~\##### 
~~\###|
~~\#/ https://aws.amazon.com/linux/amazon-linux-2023
   V~'->
~~~~
~~.-.-
  _/_/_/_/_
```

```
[ec2-user@ip-20-0-3-30 ~]$
```

```
# Update the system
```

```
>> sudo yum update -y
[ec2-user@ip-20-0-3-30 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository 123 kB/s | 19 kB      00:00
Dependencies resolved.
Nothing to do.
Complete!
```

```
# Install PHP 8.2
```

```
>> sudo dnf install php8.2
[ec2-user@ip-20-0-3-30 ~]$ sudo dnf install php8.2
Last metadata expiration check: 0:00:20 ago on Sun Aug 31 09:14:42 2025.
Dependencies resolved.
=====
Package                                Arch      Version                                Repository
=====
Installing:
  php8.2                                x86_64    8.2.29-1.amzn2023.0.1                amazonlinux 9
Installing dependencies:
>> sudo yum install php8.2-mysqlnd
```

```
[ec2-user@ip-20-0-3-30 ~]$ sudo yum install php8.2-mysqlnd
Last metadata expiration check: 0:00:57 ago on Sun Aug 31 09:14:42 2025.
Dependencies resolved.
```

```
=====
Package                Arch      Version                               Repository      Size
=====
Installing:
php8.2-mysqlnd         x86_64    8.2.29-1.amzn2023.0.1               amazonlinux     148 k
Transaction Summary
=====
Install 1 Package

Total download size: 148 k
Installed size: 442 k
Is this ok [y/N]: y
Downloading Packages:
```

# Install Apache Web Server

```
>> sudo yum install -y httpd
# Start and Enable Apache
>> sudo systemctl start httpd
>> sudo systemctl enable httpd
>> sudo systemctl is-enabled httpd
```

```
[ec2-user@ip-20-0-3-30 ~]$ sudo yum install -y httpd
Last metadata expiration check: 0:01:20 ago on Sun Aug 31 09:14:42 2025.
Package httpd-2.4.64-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-30 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-3-30 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-3-30 ~]$ sudo systemctl is-enabled httpd
enabled
```

# Add User to Apache Group

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

```
[ec2-user@ip-20-0-3-30 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-3-30 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-3-30 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-3-30 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-3-30 ~]$
```

# Install Additional PHP Modules

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

```
[ec2-user@ip-20-0-3-30 ~]$ sudo yum install php-mbstring php-xml -y
Last metadata expiration check: 1:00:08 ago on Sun Aug 31 09:14:42 2025.
Package php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-30 ~]$ sudo yum install php-fpm
Last metadata expiration check: 1:00:20 ago on Sun Aug 31 09:14:42 2025.
Package php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-3-30 ~]$
```

# Restart Apache and PHP-FPM

```
>> sudo systemctl restart httpd
>> sudo systemctl restart php-fpm
```

```
[ec2-user@ip-20-0-3-30 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-3-30 ~]$ sudo systemctl restart php-fpm
```

### # Download and Set Up phpMyAdmin

```
>> cd /var/www/html
```

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

```
[ec2-user@ip-20-0-3-30 ~]$ cd /var/www/html
[ec2-user@ip-20-0-3-30 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-08-31 10:16:44-- 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::53, ...
Connecting to www.phpmyadmin.net (www.phpmyadmin.net)|79.127.206.207|:443... connected.
HTTP request sent, awaiting response... 302 Found
```

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

```
[ec2-user@ip-20-0-3-30 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
phpMyAdmin-5.2.2-all-languages/.rtlcssrc.json
phpMyAdmin-5.2.2-all-languages/CONTRIBUTING.md
phpMyAdmin-5.2.2-all-languages/ChangeLog
phpMyAdmin-5.2.2-all-languages/LICENSE
phpMyAdmin-5.2.2-all-languages/README
phpMyAdmin-5.2.2-all-languages/RELEASE-DATE-5.2.2
phpMyAdmin-5.2.2-all-languages/babel.config.json
phpMyAdmin-5.2.2-all-languages/composer.json
phpMyAdmin-5.2.2-all-languages/composer.lock
phpMyAdmin-5.2.2-all-languages/config.sample.inc.php
phpMyAdmin-5.2.2-all-languages/doc/
phpMyAdmin-5.2.2-all-languages/doc/html/
```

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

```
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/SlimData.php
phpMyAdmin-5.2.2-all-languages/yarn.lock
[ec2-user@ip-20-0-3-30 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-3-30 html]$ |
```

### # Create a Test Page and Test the Server

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

```
>> curl http://localhost
```

```
[ec2-user@ip-20-0-3-30 html]$ echo "PHP server 1" > /var/www/html/index.html
[ec2-user@ip-20-0-3-30 html]$ curl http://localhost
PHP server 1
[ec2-user@ip-20-0-3-30 html]$ exit
logout
Connection to 20.0.3.30 closed.
[ec2-user@ip-20-0-1-40 ~]$ |
```

### 3. Now ssh into app-server2 and run the below commands

```
[ec2-user@ip-20-0-1-40 ~]$ ls
projectkey.pem
[ec2-user@ip-20-0-1-40 ~]$ ssh -i projectkey.pem ec2-user@20.0.4.6
The authenticity of host '20.0.4.6 (20.0.4.6)' can't be established.
ED25519 key fingerprint is SHA256:PH9glsJzYBDwud9Scx/XR3lPIv1qh0/GtgdMP8d7dYM.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '20.0.4.6' (ED25519) to the list of known hosts.

#_
~\_##### Amazon Linux 2023
~~~\_#####\
~~~\_###|
~~~\_#/
~~~\_V~'-'>
~~~~
~~~.-.
~~~\_/m/'-/_/
[ec2-user@ip-20-0-4-6 ~]$ |
```

### # Update the system

```
>> sudo yum update -y
```



```
[ec2-user@ip-20-0-4-6 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository 130 kB/s | 19 kB    00:00
Dependencies resolved.
Nothing to do.
Complete!
```

#### # Install PHP 8.2

```
>> sudo dnf install php8.2
```

```
[ec2-user@ip-20-0-4-6 ~]$ sudo dnf install php8.2
Last metadata expiration check: 0:00:12 ago on Sun Aug 31 10:24:35 2025.
Dependencies resolved.
```

Package	Arch	Version	Repository	Size
Installing:				
php8.2	x86_64	8.2.29-1.amzn2023.0.1	amazonlinux	9.7 k
Installing dependencies:				
apr	x86_64	1.7.5-1.amzn2023.0.4	amazonlinux	129 k

```
>> sudo yum install php8.2-mysqlnd
```

```
[ec2-user@ip-20-0-4-6 ~]$ sudo yum install php8.2-mysqlnd
Last metadata expiration check: 0:00:31 ago on Sun Aug 31 10:24:35 2025.
Dependencies resolved.
```

Package	Arch	Version	Repository	Size
Installing:				
php8.2-mysqlnd	x86_64	8.2.29-1.amzn2023.0.1	amazonlinux	148 k

Transaction Summary

=====  
Install 1 Package  
=====  
Total download size: 148 k  
Installed size: 442 k  
Is this ok [y/N]: y  
Downloading Packages:

#### # Install Apache Web Server

```
>> sudo yum install -y httpd
```

##### # Start and Enable Apache

```
>> sudo systemctl start httpd
```

```
>> sudo systemctl enable httpd
```

```
>> sudo systemctl is-enabled httpd
```

```
[ec2-user@ip-20-0-4-6 ~]$ sudo yum install -y httpd
Last metadata expiration check: 0:02:01 ago on Sun Aug 31 10:24:35 2025.
Package httpd-2.4.64-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
```

```
Nothing to do.
Complete!
```

```
[ec2-user@ip-20-0-4-6 ~]$ sudo systemctl start httpd
[ec2-user@ip-20-0-4-6 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-20-0-4-6 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-20-0-4-6 ~]$
```

#### # Add User to Apache Group

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

##### # Change Ownership and Permissions for Web Directory

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

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

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

```
[ec2-user@ip-20-0-4-6 ~]$ sudo usermod -a -G apache ec2-user
[ec2-user@ip-20-0-4-6 ~]$ sudo chown -R ec2-user:apache /var/www
[ec2-user@ip-20-0-4-6 ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
[ec2-user@ip-20-0-4-6 ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
[ec2-user@ip-20-0-4-6 ~]$
```

## # Install Additional PHP Modules

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

```
>> sudo yum install php-fpm
```

## # Restart Apache and PHP-FPM

```
>> sudo systemctl restart httpd
```

```
>> sudo systemctl restart php-fpm
```

```
[ec2-user@ip-20-0-4-6 ~]$ sudo yum install php-mbstring php-xml -y
Last metadata expiration check: 0:04:19 ago on Sun Aug 31 10:24:35 2025.
Package php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Package php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-6 ~]$ sudo yum install php-fpm
Last metadata expiration check: 0:04:30 ago on Sun Aug 31 10:24:35 2025.
Package php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-20-0-4-6 ~]$ sudo systemctl restart httpd
[ec2-user@ip-20-0-4-6 ~]$ sudo systemctl restart php-fpm
[ec2-user@ip-20-0-4-6 ~]$
```

## # Download and Set Up phpMyAdmin

```
>> cd /var/www/html
```

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

```
[ec2-user@ip-20-0-4-6 ~]$ cd /var/www/html
[ec2-user@ip-20-0-4-6 html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
--2025-08-31 10:30:55-- 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]
--2025-08-31 10:30:55-- 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, 109.61.91.195, 109.61.91.197, ...
Connecting to files.phpmyadmin.net (files.phpmyadmin.net)|109.61.91.230|:443..
```

```
>> mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C
```

```
phpMyAdmin --strip-components 1
```

```
[ec2-user@ip-20-0-4-6 html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
phpMyAdmin-5.2.2-all-languages/.rtlcssrc.json
phpMyAdmin-5.2.2-all-languages/CONTRIBUTING.md
phpMyAdmin-5.2.2-all-languages/ChangeLog
phpMyAdmin-5.2.2-all-languages/LICENSE
phpMyAdmin-5.2.2-all-languages/README
phpMyAdmin-5.2.2-all-languages/RELEASE-DATE-5.2.2
phpMyAdmin-5.2.2-all-languages/babel.config.json
phpMyAdmin-5.2.2-all-languages/composer.json
phpMyAdmin-5.2.2-all-languages/composer.lock
phpMyAdmin-5.2.2-all-languages/config.sample.inc.php
phpMyAdmin-5.2.2-all-languages/doc/
phpMyAdmin-5.2.2-all-languages/doc/html/
```

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

## # Create a Test Page and Test the Server

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

```
>> curl http://localhost
```

```
[ec2-user@ip-20-0-4-6 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-4-6 html]$ echo "PHP server 2" > /var/www/html/index.html
[ec2-user@ip-20-0-4-6 html]$ curl http://localhost
PHP server 2
[ec2-user@ip-20-0-4-6 html]$
```

4. Create the Target groups:

app-tg

Details

arn:aws:elasticloadbalancing:us-east-1:252955337222:targetgroup/app-tg/3c7d3e1af2b2c2f3

Target type

Instance

Protocol : Port

HTTP: 80

Protocol version

HTTP1

VPC

vpc-00be49c245800c9b2

IP address type

IPv4

Load balancer

project-alb

2

Total targets

2

Healthy

0

Unhealthy

0

Unused

0

Initial

0

Draining

0

Anomalous

Distribution of targets by Availability Zone (AZ)

Select values in this table to see corresponding filters applied to the Registered targets table below.

Targets

Monitoring

Health checks

Attributes

Tags

Registered targets (2)

Info

Anomaly mitigation: Not applicable

Deregister

Register targets

Target groups route requests to individual registered targets using the protocol and port number specified. Health checks are performed on all registered targets according to the target group's health check settings. Anomaly detection is automatically applied to HTTP/HTTPS target groups with at least 3 healthy targets.

Filter targets

Instance ID

Name

Port

Zone

Health status

Health status details

Admini...

Overrid...

Launch...

Anomaly detectio...

I-0973b03e6448974de

app-server2

80

us-east-1b (us...

Healthy

-

No override.

No overrid...

August 31...

Normal

I-01232991e2a6be2b8

app-server1

80

us-east-1a (use...

Healthy

-

No override.

No overrid...

August 31...

Normal

5. Create Application Load Balancer

project-alb

Details

Load balancer type

Application

Status

Active

VPC

vpc-00be49c245800c9b2

Load balancer IP address type

IPv4

Scheme

Internet-facing

Hosted zone

Z35SXDOTRQ7X7K

Availability Zones

subnet-05e9916122e328f00 us-east-1a (use1-az2)

Date created

August 31, 2025, 16:21 (UTC+05:30)

Load balancer ARN

arn:aws:elasticloadbalancing:us-east-1:252955337222:loadbalancer/app/project-alb/af73e970789da802

DNS name info

project-alb-1759809495.us-east-1.elb.amazonaws.com (A Record)

Listeners and rules

Network mapping

Resource map

Security

Monitoring

Integrations

Attributes

Capacity

Tags

Listeners and rules (1)

Info

Manage rules

Manage listener

Add listener

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

Filter listeners

Protocol:Port

Default action

Rules

ARN

Security policy

Default SSL/TLS certificate

mTLS

Trust sto

HTTP:80

Forward to target group

app-tg 1 (100%)

1 rule

ARN

Not applicable

Not applicable

Not applicable

Not appl

Target group stickiness: Off

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

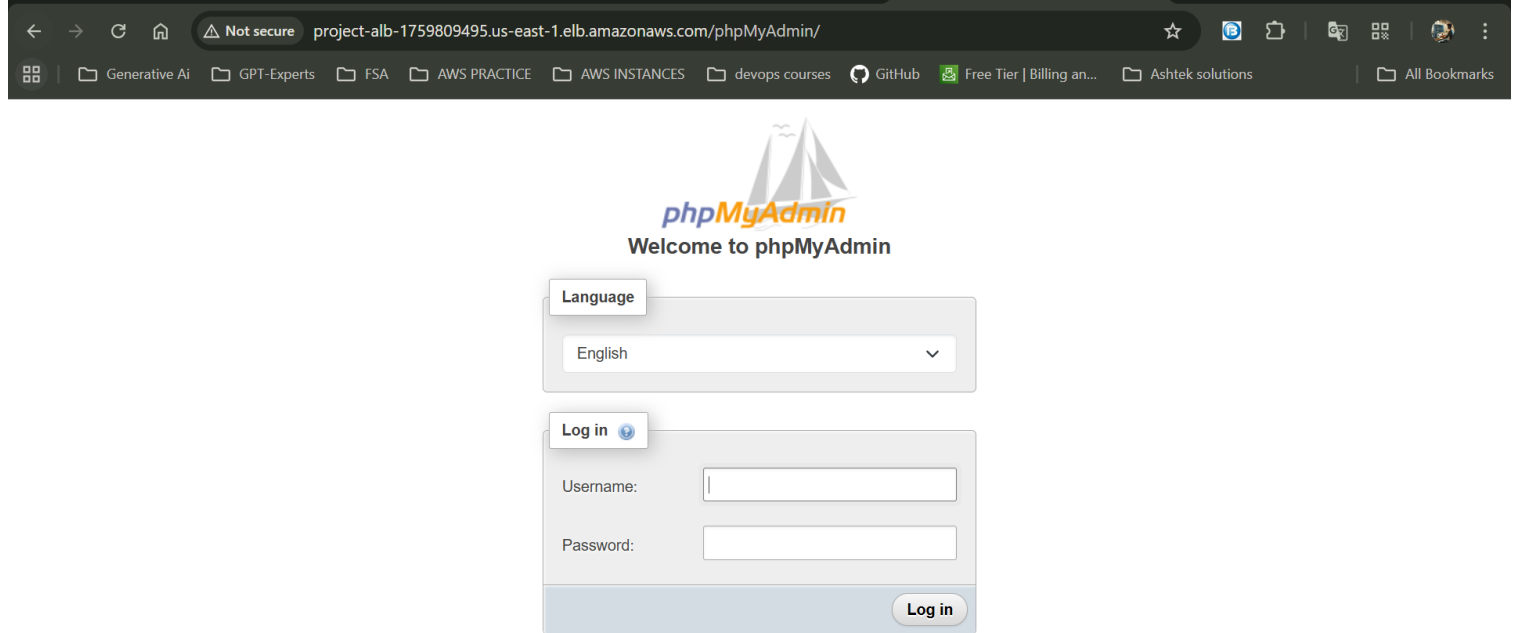
A screenshot of a web browser window. The address bar shows the URL 'project-alb-1759809495.us-east-1.elb.amazonaws.com'. The page content is 'PHP server 1'. The browser's bookmark bar is visible at the bottom, showing various folders like 'Generative AI', 'GPT-Experts', 'FSA', 'AWS PRACTICE', 'AWS INSTANCES', 'devops courses', 'GitHub', 'Free Tier | Billing an...', 'Ashtek solutions', and 'All Bookmarks'.

Refresh the page:

A screenshot of the same web browser window after a page refresh. The address bar still shows the same URL. The page content has changed to 'PHP server 2'. The browser's bookmark bar remains visible at the bottom.



7. Check PhpMyAdmin :- And when you write phpMyAdmin after your domain name it should navigate to



## Part 4 :- Database configurations

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
```

```
[ec2-user@ip-20-0-4-6 html]$ cd /var/www/html/phpMyAdmin
[ec2-user@ip-20-0-4-6 phpMyAdmin]$ ls
CONTRIBUTING.md      composer.lock          libraries              templates
ChangeLog             config.sample.inc.php  locale                themes
LICENSE               doc                    package.json          url.php
README                examples              robots.txt            vendor
RELEASE-DATE-5.2.2    favicon.ico            setup                 yarn.lock
babel.config.json     index.php              show_config_errors.php
composer.json          js                      sql
[ec2-user@ip-20-0-4-6 phpMyAdmin]$ mv config.sample.inc.php config.inc.php
[ec2-user@ip-20-0-4-6 phpMyAdmin]$ nano config.inc.php
```

```
GNU nano 8.3 config.inc.php Modified
*/
$i = 0;

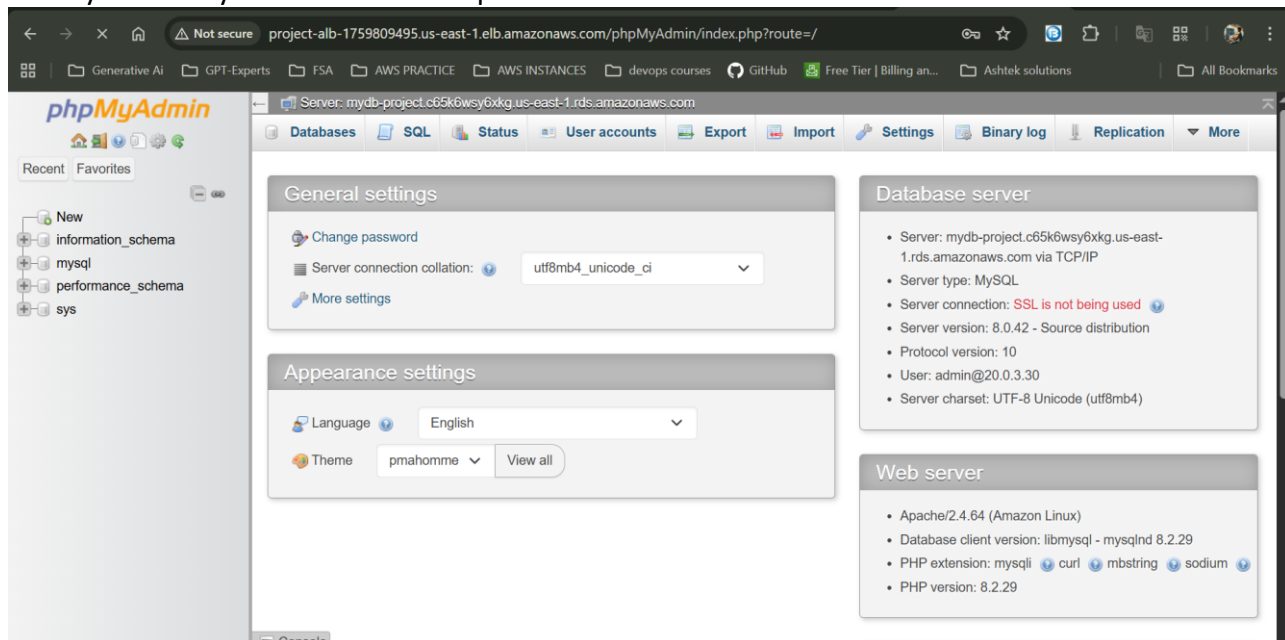
/**
 * First server
 */
$i++;
/* Authentication type */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$cfg['Servers'][$i]['host'] = 'mydb-project.c65k6wsy6xkg.us-east-1.rds.amazonaws.com';
$cfg['Servers'][$i]['compress'] = false;
$cfg['Servers'][$i]['AllowNoPassword'] = false;

/**
 * phpMyAdmin configuration storage settings.
 */

/* User used to manipulate with storage */
// $cfg['Servers'][$i]['controlhost'] = '';
// $cfg['Servers'][$i]['controlport'] = '';

AG Help      AO Write Out  AF Where Is  AK Cut       AT Execute   AC Location
AX Exit      AR Read File  AN Replace  AU Paste    AJ Justify   AL Go To Line
```

Once you enter your username and password:



The end