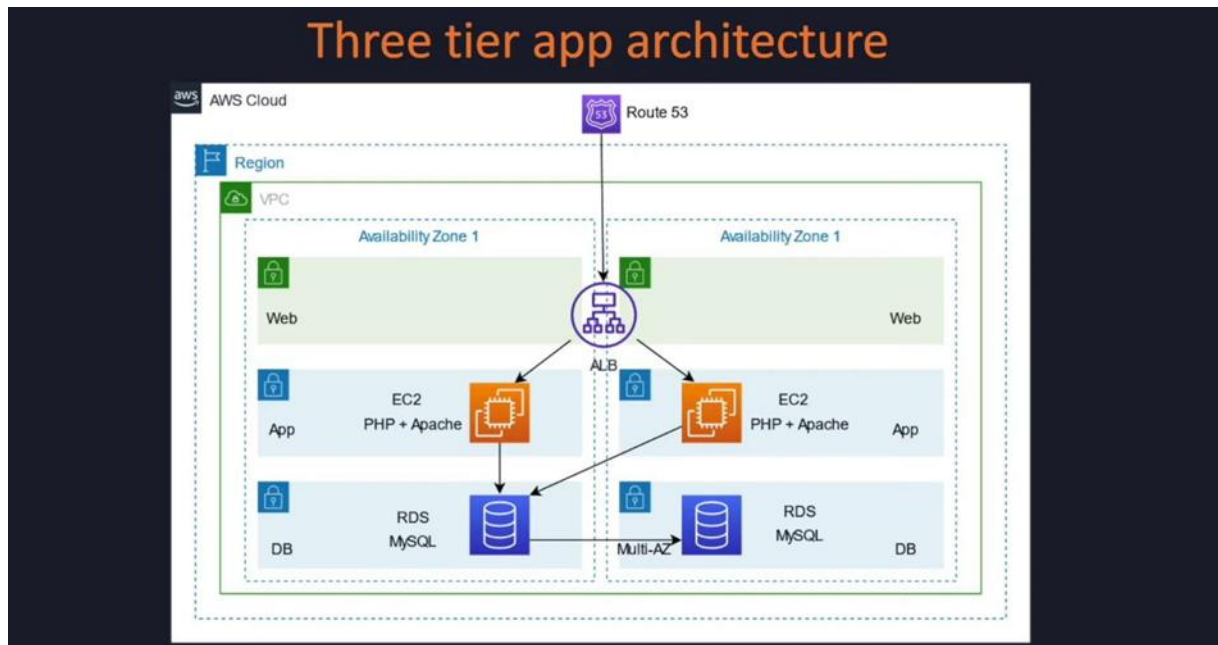


AUTHOR : ISRAR AHMAD KHAN

MOB NO: 8888337050

EMAIL: ik427094@gmail.com

AWS Challenge (Designing Robust Web Application on AWS)



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

Lets Proceed:

Part 1

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

-00f9c2f4d0864299f

You successfully created vpc-00f9c2f4d0864299f / awsProject-vpc

vpc-00f9c2f4d0864299f / awsProject-vpc

Actions

| | | | |
|---|---|---|---|
| Details Info | | | |
| VPC ID vpc-00f9c2f4d0864299f | State Available | Block Public Access Off | DNS hostnames Disabled |
| DNS resolution Enabled | Tenancy default | DHCP option set dopt-0b634855d4173a33e | Main route table rtb-0c1f83ced391c0f79 |
| Main network ACL acl-0823afc39fbc464db | Default VPC No | IPv4 CIDR 20.0.0.0/20 | IPv6 pool - |
| IPv6 CIDR (Network border group) - | Network Address Usage metrics Disabled | Route 53 Resolver DNS Firewall rule groups - | Owner ID 889692314784 |

Resource map CIDRs Flow logs Tags Integrations

Resource map Info Show all details

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

subnet-0ff331027d79be81a / web-pub-sub1

Actions

| | | | |
|---|--|---|--|
| Details | | | |
| Subnet ID subnet-0ff331027d79be81a | Subnet ARN arn:aws:ec2:us-east-1:889692314784:subnet/subnet-0ff331027d79be81a | State Available | Block Public Access Off |
| IPv4 CIDR 20.0.1.0/24 | Available IPv4 addresses 251 | IPv6 CIDR - | IPv6 CIDR association ID - |
| Availability Zone use1-az6 (us-east-1a) | Network border group us-east-1 | VPC vpc-00f9c2f4d0864299f awsProject-vpc | Route table - |
| Network ACL - | Default subnet No | Auto-assign public IPv4 address No | Auto-assign IPv6 address No |
| Auto-assign customer-owned IPv4 address No | Customer-owned IPv4 pool - | Outpost ID - | IPv4 CIDR reservations - |
| IPv6 CIDR reservations - | IPv6-only No | Hostname type IP name | Resource name DNS A record Disabled |
| Resource name DNS AAAA record Disabled | DNS64 Disabled | Owner 889692314784 | |

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

subnet-01a219e3513c4f37d / web-pub-sub2

Actions

| | | | |
|---|--|---|--|
| Details | | | |
| Subnet ID subnet-01a219e3513c4f37d | Subnet ARN arn:aws:ec2:us-east-1:889692314784:subnet/subnet-01a219e3513c4f37d | State Available | Block Public Access Off |
| IPv4 CIDR 20.0.2.0/24 | Available IPv4 addresses 251 | IPv6 CIDR - | IPv6 CIDR association ID - |
| Availability Zone use1-az1 (us-east-1b) | Network border group us-east-1 | VPC vpc-00f9c2f4d0864299f awsProject-vpc | Route table rtb-0c1f83ced391c0f79 |
| Network ACL acl-0823afc39fbc464db | Default subnet No | Auto-assign public IPv4 address No | Auto-assign IPv6 address No |
| Auto-assign customer-owned IPv4 address No | Customer-owned IPv4 pool - | Outpost ID - | IPv4 CIDR reservations - |
| IPv6 CIDR reservations - | IPv6-only No | Hostname type IP name | Resource name DNS A record Disabled |
| Resource name DNS AAAA record Disabled | DNS64 Disabled | Owner 889692314784 | |

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

t-0ecc685c835c6527b

subnet-0ecc685c835c6527b / app-pvt-sub1

Actions

Details

Subnet ID

subnet-0ecc685c835c6527b

IPv4 CIDR

20.0.3.0/24

Availability Zone

use1-az6 (us-east-1a)

Network ACL

acl-0823afc39fbc464db

Auto-assign customer-owned IPv4 address

No

IPv6 CIDR reservations

-

Resource name DNS AAAA record

Disabled

Subnet ARN

arn:aws:ec2:us-east-1:889692314784:subnet/subnet-0ecc685c835c6527b

Available IPv4 addresses

251

Network border group

us-east-1

Default subnet

No

Customer-owned IPv4 pool

-

IPv6-only

No

DNS64

Disabled

State

Available

IPv6 CIDR

-

VPC

vpc-00f9c2f4d0864299f | awsProject-vpc

Auto-assign public IPv4 address

No

Outpost ID

-

Hostname type

IP name

Owner

889692314784

Block Public Access

Off

IPv6 CIDR association ID

-

Route table

rtb-0c1f83ced391c0f79

Auto-assign IPv6 address

No

IPv4 CIDR reservations

-

Resource name DNS A record

Disabled

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

subnet-0cf5d60bdc1c0e2f2 / app-pvt-sub2

Actions

Details

Subnet ID

subnet-0cf5d60bdc1c0e2f2

IPv4 CIDR

20.0.4.0/24

Availability Zone

use1-az1 (us-east-1b)

Network ACL

acl-0823afc39fbc464db

Auto-assign customer-owned IPv4 address

No

IPv6 CIDR reservations

-

Resource name DNS AAAA record

Disabled

Subnet ARN

arn:aws:ec2:us-east-1:889692314784:subnet/subnet-0cf5d60bdc1c0e2f2

Available IPv4 addresses

251

Network border group

us-east-1

Default subnet

No

Customer-owned IPv4 pool

-

IPv6-only

No

DNS64

Disabled

State

Available

IPv6 CIDR

-

VPC

vpc-00f9c2f4d0864299f | awsProject-vpc

Auto-assign public IPv4 address

No

Outpost ID

-

Hostname type

IP name

Owner

889692314784

Block Public Access

Off

IPv6 CIDR association ID

-

Route table

rtb-0c1f83ced391c0f79

Auto-assign IPv6 address

No

IPv4 CIDR reservations

-

Resource name DNS A record

Disabled

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

subnet-0b0ae03134aad9838 / db-pvt-sub1

Actions

Details

Subnet ID

subnet-0b0ae03134aad9838

IPv4 CIDR

20.0.5.0/24

Availability Zone

use1-az6 (us-east-1a)

Network ACL

acl-0823afc39fbc464db

Auto-assign customer-owned IPv4 address

No

IPv6 CIDR reservations

-

Resource name DNS AAAA record

Disabled

Subnet ARN

arn:aws:ec2:us-east-1:889692314784:subnet/subnet-0b0ae03134aad9838

Available IPv4 addresses

251

Network border group

us-east-1

Default subnet

No

Customer-owned IPv4 pool

-

IPv6-only

No

DNS64

Disabled

State

Available

IPv6 CIDR

-

VPC

vpc-00f9c2f4d0864299f | awsProject-vpc

Auto-assign public IPv4 address

No

Outpost ID

-

Hostname type

IP name

Owner

889692314784

Block Public Access

Off

IPv6 CIDR association ID

-

Route table

rtb-0c1f83ced391c0f79

Auto-assign IPv6 address

No

IPv4 CIDR reservations

-

Resource name DNS A record

Disabled

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

subnet-0520306b6908eba1d / db-pvt-sub2 Actions

Details

Subnet ID

subnet-0520306b6908eba1d

IPv4 CIDR

20.0.6.0/24

Availability Zone

use1-az1 (us-east-1b)

Network ACL

acl-0823afc39fbc464db

Auto-assign customer-owned IPv4 address

No

IPv6 CIDR reservations

-

Resource name DNS AAAA record

Disabled

Subnet ARN

arn:aws:ec2:us-east-1:889692314784:subnet/subnet-0520306b6908eba1d

Available IPv4 addresses

251

Network border group

us-east-1

Default subnet

No

Customer-owned IPv4 pool

-

IPv6-only

No

DNS64

Disabled

State

Available

IPv6 CIDR

-

VPC

vpc-00f9c2f4d0864299f | awsProject-vpc

Auto-assign public IPv4 address

No

Outpost ID

-

Hostname type

IP name

Owner

889692314784

Block Public Access

Off

IPv6 CIDR association ID

-

Route table

rtb-0c1f83ced391c0f79

Auto-assign IPv6 address

No

IPv4 CIDR reservations

-

Resource name DNS A record

Disabled

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

NAT gateway nat-0da42da4adf5e865e | my-nat was created successfully. ×

nat-0da42da4adf5e865e / my-nat Actions

Details

NAT gateway ID

nat-0da42da4adf5e865e

NAT gateway ARN

arn:aws:ec2:us-east-1:889692314784:natgateway/nat-0da42da4adf5e865e

VPC

vpc-00f9c2f4d0864299f | awsProject-vpc

Connectivity type

Public

Primary public IPv4 address

-

Subnet

subnet-0ff331027d79be81a / web-pub-sub1

State

Pending

Primary private IPv4 address

-

Created

Sunday, September 28, 2025 at 14:01:57 GMT+5:30

State message

Info

-

Primary network interface ID

-

Deleted

-

Secondary IPv4 addresses

Monitoring

Tags

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

Internet gateway igw-08f05725ce847fc1b successfully attached to vpc-00f9c2f4d0864299f ×

igw-08f05725ce847fc1b / my-igw Actions

Details Info

Internet gateway ID

igw-08f05725ce847fc1b

State

Attached

VPC ID

vpc-00f9c2f4d0864299f | awsProject-vpc

Owner

889692314784

Tags

Search tags

| Key | Value |
|------|--------|
| Name | my-igw |

Manage tags

< 1 > ⚙

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

Updated routes for rtb-09ac9660ae8b969a5 / route-web successfully

rtb-09ac9660ae8b969a5 / route-web

Details

| | | | |
|---|--|---|------------------------|
| Route table ID rtb-09ac9660ae8b969a5 | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (2)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|--------------|--------------------------|-------------|-----------|
| web-pub-sub1 | subnet-0ff331027d79be81a | 20.0.1.0/24 | - |
| web-pub-sub2 | subnet-01a219e3513c4f37d | 20.0.2.0/24 | - |

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

VPC | us-east-1

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#RouteTableDetails:routeTableId=rtb-04a8ffaf823d2d58

VPC dashboard

EC2 Global View

Filter by VPC

Virtual private cloud

Your VPCs

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

NAT gateways

Peering connections

Route servers

rtb-04a8ffaf823d2d58 / route-app

Details

| | | | |
|---|--|---|------------------------|
| Route table ID rtb-04a8ffaf823d2d58 | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (2)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|--------------|--------------------------|-------------|-----------|
| app-pvt-sub1 | subnet-0ecc685c835c6527b | 20.0.3.0/24 | - |
| app-pvt-sub2 | subnet-0cf5d60bdc1c0e2f2 | 20.0.4.0/24 | - |

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

Updated routes for rtb-06da478354cf35bfd / route-db.

rtb-06da478354cf35bfd / route-db

Details

| | | | |
|---|--|---|------------------------|
| Route table ID rtb-06da478354cf35bfd | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (2)

| Name | Subnet ID | IPv4 CIDR | IPv6 CIDR |
|-------------|--------------------------|-------------|-----------|
| db-pvt-sub1 | subnet-0b0ae03134aad9838 | 20.0.5.0/24 | - |
| db-pvt-sub2 | subnet-0520306b6908eba1d | 20.0.6.0/24 | - |

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

rtb-09ac9660ae8b969a5 / route-web Actions

Details [Info](#)

| | | | |
|--|---|--|-------------------------------|
| Route table ID rtb-09ac9660ae8b969a5 | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2) Both Edit routes

Filter routes

| Destination | Target | Status | Propagated | Route Origin |
|-------------|-----------------------|--------|------------|--------------------|
| 0.0.0.0/0 | igw-08f05725ce847fc1b | Active | No | Create Route |
| 20.0.0.0/20 | local | Active | No | Create Route Table |

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

Updated routes for rtb-04a8ffafd823d2d58 / route-app successfully
[Details](#)

rtb-04a8ffafd823d2d58 / route-app Actions

Details [Info](#)

| | | | |
|--|---|--|-------------------------------|
| Route table ID rtb-04a8ffafd823d2d58 | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2) Both Edit routes

Filter routes

| Destination | Target | Status | Propagated | Route Origin |
|-------------|-----------------------|--------|------------|--------------------|
| 0.0.0.0/0 | nat-05903e026a1f95f43 | Active | No | Create Route |
| 20.0.0.0/20 | local | Active | No | Create Route Table |

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

Updated routes for rtb-06da478354cf35bfd / route-db successfully
[Details](#)

rtb-06da478354cf35bfd / route-db Actions

Details [Info](#)

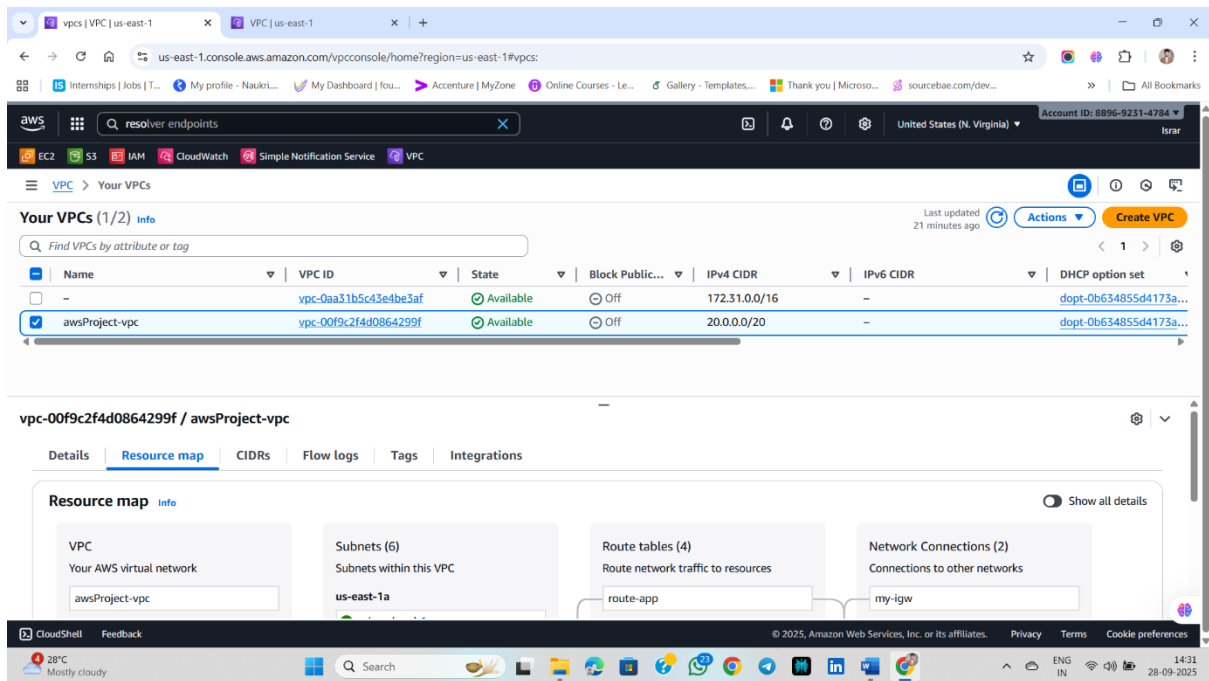
| | | | |
|--|---|--|-------------------------------|
| Route table ID rtb-06da478354cf35bfd | Main <input checked="" type="checkbox"/> No | Explicit subnet associations 2 subnets | Edge associations - |
| VPC vpc-00f9c2f4d0864299f awsProject-vpc | Owner ID 889692314784 | | |

Routes | Subnet associations | Edge associations | Route propagation | Tags

Routes (2) Both Edit routes

Filter routes

| Destination | Target | Status | Propagated | Route Origin |
|-------------|-----------------------|--------|------------|--------------------|
| 0.0.0.0/0 | nat-05903e026a1f95f43 | Active | No | Create Route |
| 20.0.0.0/20 | local | Active | No | Create Route Table |



Part 2

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

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

Instance summary for i-034493cbb78059342 (Jump-server) [Info](#)

Updated less than a minute ago

| | | |
|---|---|--|
| Instance ID i-034493cbb78059342 | Public IPv4 address 3.81.47.214 open address | Private IPv4 addresses 20.0.1.175 |
| IPv6 address - | Instance state Running | Public DNS - |
| Hostname type IP name: ip-20-0-1-175.ec2.internal | Private IP DNS name (IPv4 only) ip-20-0-1-175.ec2.internal | Elastic IP addresses - |
| Answer private resource DNS name - | Instance type t3.micro | AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendation s. Learn more |
| Auto-assigned IP address 3.81.47.214 [Public IP] | VPC ID vpc-00f9c2f4d0864299f (awsProject-vpc) | Auto Scaling Group name - |
| IAM Role - | Subnet ID subnet-0ff331027d79be81a (web-pub-sub1) | Managed false |
| IMDSv2 Required | Instance ARN arn:aws:ec2:us-east-1:889692314784:instance/i-0344 | |

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

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

Instance summary for i-05208bf42570e97cd (app-server1) [Info](#)

Updated less than a minute ago

Instance ID

i-05208bf42570e97cd

IPv6 address

–

Hostname type

IP name: ip-20-0-3-239.ec2.internal

Answer private resource DNS name

–

Auto-assigned IP address

–

IAM Role

–

IMDSv2

Required

Public IPv4 address

–

Instance state

Running

Private IP DNS name (IPv4 only)

ip-20-0-3-239.ec2.internal

Instance type

t3.micro

VPC ID

vpc-00f9c2f4d0864299f (awsProject-vpc)

Subnet ID

subnet-0ecc685c835c6527b (app-pvt-sub1)

Instance ARN

arn:aws:ec2:us-east-1:889692314784:instance/i-05208bf42570e97cd

Private IPv4 addresses

20.0.3.239

Public DNS

–

Elastic IP addresses

–

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendation s.
[Learn more](#)

Auto Scaling Group name

–

Managed

false

Connect

Instance state

Actions

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

| Find Instance by attribute or tag (case-sensitive) | | | | | | | All states | |
|--|-------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|-------------|
| Instance state = running | | | | | | | Clear filters | |
| | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 |
| <input type="checkbox"/> | jump-server | i-0dab1a78544fd316c | Running | t2.micro | 2/2 checks passed | View alarms | us-east-1a | – |
| <input type="checkbox"/> | app-server2 | i-08e6df054281d361c | Running | t2.micro | 2/2 checks passed | View alarms | us-east-1b | – |
| <input type="checkbox"/> | app-server1 | i-0805002007bedSebc | Running | t2.micro | 2/2 checks passed | View alarms | us-east-1a | – |

Copy the private key in your jump-server instance

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

```
>> chmod 400 <your key>
```

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

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


```

Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-filesystem-2.4.65-1.amzn2023.0.1.noarch
libsodium-1.0.19-4.amzn2023.x86_64
mod_http2-2.0.27-1.amzn2023.0.3.x86_64
php8.2-8.2.29-1.amzn2023.0.1.x86_64
php8.2-fpm-8.2.29-1.amzn2023.0.1.x86_64
php8.2-pdo-8.2.29-1.amzn2023.0.1.x86_64
php8.2-xml-8.2.29-1.amzn2023.0.1.x86_64

apr-util-1.6.3-1.amzn2023.0.1.x86_64
httpd-2.4.65-1.amzn2023.0.1.x86_64
httpd-tools-2.4.65-1.amzn2023.0.1.x86_64
libxml2-1.1.43-1.amzn2023.0.2.x86_64
mod_lua-2.4.65-1.amzn2023.0.1.x86_64
php8.2-cgi-8.2.29-1.amzn2023.0.1.x86_64
php8.2-mbstring-8.2.29-1.amzn2023.0.1.x86_64
php8.2-process-8.2.29-1.amzn2023.0.1.x86_64

apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-core-2.4.65-1.amzn2023.0.1.x86_64
libtool-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
nginx-filesystem-1.1.28-0.1.amzn2023.0.2.noarch
php8.2-common-8.2.29-1.amzn2023.0.1.x86_64
php8.2-opcache-8.2.29-1.amzn2023.0.1.x86_64
php8.2-sodium-8.2.29-1.amzn2023.0.1.x86_64

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

```

Install Apache Web Server

>> sudo yum install -y httpd

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

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

Complete!

[ec2-user@ip-20-0-3-239 ~]\$ sudo yum install httpd -y

Last metadata expiration check: 0:05:44 ago on Sun Sep 28 09:42:35 2025.

Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.

Dependencies resolved.

Nothing to do.

Complete!

[ec2-user@ip-20-0-3-239 ~]\$ sudo yum install -y httpd

Last metadata expiration check: 0:06:23 ago on Sun Sep 28 09:42:35 2025.

Package httpd-2.4.65-1.amzn2023.0.1.x86_64 is already installed.

Dependencies resolved.

Nothing to do.

Complete!

[ec2-user@ip-20-0-3-239 ~]\$

Start and Enable Apache

>> sudo systemctl start httpd

>> sudo systemctl enable httpd

>> sudo systemctl is-enabled httpd

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

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

Add User to Apache Group

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

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

Change Ownership and Permissions for Web Directory

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

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

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

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

Install Additional PHP Modules

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

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

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

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

Restart Apache and PHP-FPM

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

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

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

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

2025-09-28 10:03:13 (292 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]
[ec2-user@ip-20-0-3-239 ~]$
```

Download and Set Up phpMyAdmin

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

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

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

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

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

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

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

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

The screenshot shows a web browser window displaying the AWS Management Console. The top navigation bar includes the AWS logo, a search bar, and the account ID: 8896-9231-4784. The main content area shows a list of services: EC2, S3, IAM, CloudWatch, Simple Notification Service, and VPC. Below the services list, there is a terminal window showing the installation of phpMyAdmin 5.2.2. The terminal output includes the following commands and their results:

```
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/CHANGELOG.md
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/LICENSE
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/README.md
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/composer.json
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/src/
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/src/Assert.php
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/src/InvalidArgumentException.php
phpMyAdmin-5.2.2-all-languages/vendor/webmozart/assert/src/Mixin.php
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/CHANGELOG.md
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/LICENSE
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/README.md
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/cliff.toml
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/composer.json
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/dist/
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/dist/merged-ultraslim.json
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/RBDocumentation.php
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/RBEntry.php
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/RBException.php
phpMyAdmin-5.2.2-all-languages/vendor/williamdes/mariadb-mysql-kbs/src/Search.php
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-20-0-3-239 html]$ rm phpMyAdmin-latest-all-languages.tar.gz
[ec2-user@ip-20-0-3-239 html]$
```

Below the terminal window, the browser shows the details of the EC2 instance, including the public IP: 3.81.47.214 and private IP: 20.0.1.175. The bottom of the browser window shows the CloudShell interface with a search bar and various icons.

Create a Test Page and Test the Server

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

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

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

Now ssh into app-server2 and run the below commands

```
[ec2-user@ip-20-0-3-239 ~]$ sudo nano projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 777 projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.4.125

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

```
# Update the system
```


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

Install PHP 8.2

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

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

```
[ec2-user@ip-20-0-3-239 ~]$ sudo chmod 777 projectkey.pem
[ec2-user@ip-20-0-3-239 ~]$ ssh -i "projectkey.pem" ec2-user@20.0.4.125
```



```
#####
~\      \      Amazon Linux 2023
~~\    /~~~~~\
~~\  /   \____\
~~\ /     \___/
   V       ^--> https://aws.amazon.com/linux/amazon-linux-2023
  /m/
 /m/
```

```
[ec2-user@ip-20-0-4-125 ~]$ sudo yum update -y
Amazon Linux 2023 Kernel Livepatch repository                                189 kB/s | 23 kB          00:00
Dependencies resolved.
Nothing to do.
Complete!
```

```
[ec2-user@ip-20-0-4-125 ~]$ sudo dnf install php8.2
Last metadata expiration check: 0:01:08 ago on Sun Sep 28 10:11:59 2025.
Dependencies resolved.
```

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

Install Apache Web Server

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

```
# Start and Enable Apache
```

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

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

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

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

```
# Add User to Apache Group
```



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

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

Install Additional PHP Modules

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

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

Restart Apache and PHP-FPM

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

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

Download and Set Up phpMyAdmin

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

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

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

2025-09-28 10:24:06 (284 MB/s) - 'phpMyAdmin-latest-all-languages.tar.gz' saved [13764534/13764534]
[ec2-user@ip-20-0-4-125 html]$
```

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

Create a Test Page and Test the Server

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

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

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

As you have downloaded Apache and phpMyAdmin to access them, you now need to:

Create the Target groups:

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

app-tg Actions

Details

arn:aws:elasticloadbalancing:us-east-1:889692314784:targetgroup/app-tg/2f7f2865ab1142a5

| | | | |
|--------------------------------|---|----------------------------------|---|
| Target type Instance | Protocol : Port HTTP: 80 | Protocol version HTTP1 | VPC vpc-00f9c2f4d0864299f |
| IP address type IPv4 | Load balancer None associated | | |

| | | | | | |
|--------------------|--|-----------------------------|--------------------------|---------------------------|----------------------------|
| 2 Total targets | 0 Healthy 0 Anomalous | 0 Unhealthy | 2 Unused | 0 Initial | 0 Draining |
|--------------------|--|-----------------------------|--------------------------|---------------------------|----------------------------|

► **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](#)

Now go to Load Balancer:

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

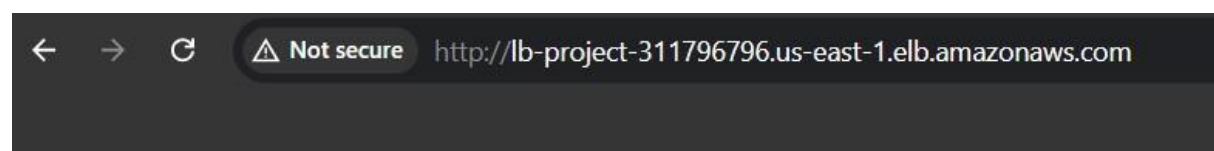
☑ **Successfully created load balancer: project-alb**
It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks.

project-alb Actions

Details

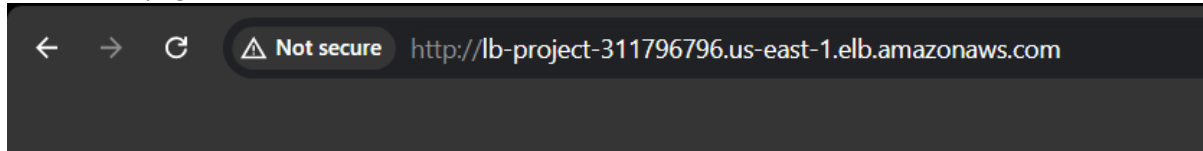
| | | | |
|---|---|---|--|
| Load balancer type Application | Status Active | VPC vpc-00f9c2f4d0864299f | Load balancer IP address type IPv4 |
| Scheme Internet-facing | Hosted zone Z35XDOTRQ7X7K | Availability Zones subnet-0ecc685c835c6527b us-east-1a (use1-az6) subnet-0520306b6908eba1d us-east-1b (use1-az1) | Date created September 28, 2025, 16:05 (UTC+05:30) |
| Load balancer ARN arn:aws:elasticloadbalancing:us-east-1:889692314784:loadbalancer/app/project-alb/b0658bdd93b427db | DNS name info project-alb-168262.us-east-1.elb.amazonaws.com (A Record) | | |

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



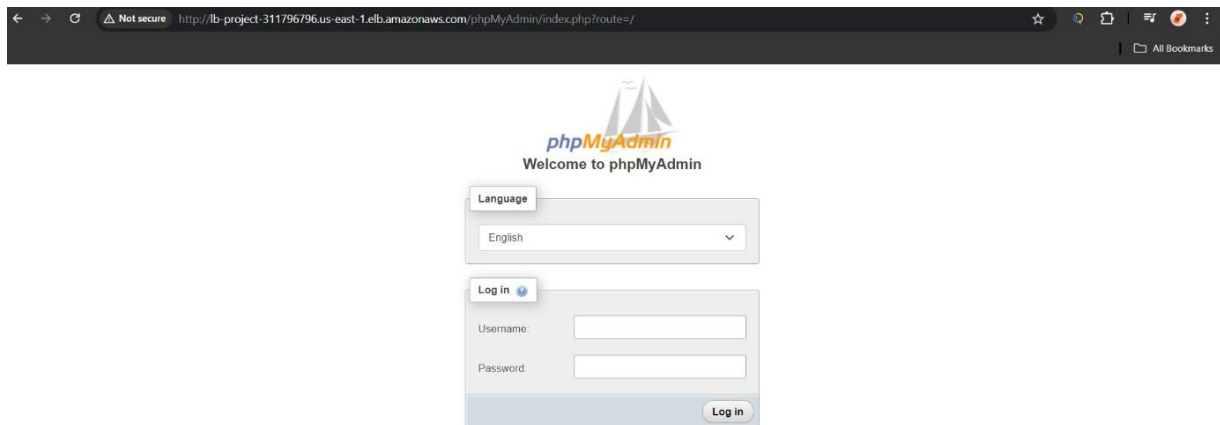
PHP server 1

Refract the page:



PHP server 2

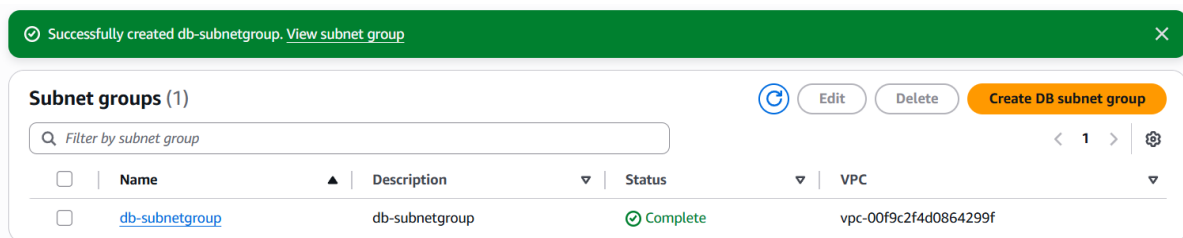
And when you write phpMyAdmin after your domain name it should navigate to



Part 3

Go to RDS

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



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

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

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

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

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

```
>> nano config.inc.php
```

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

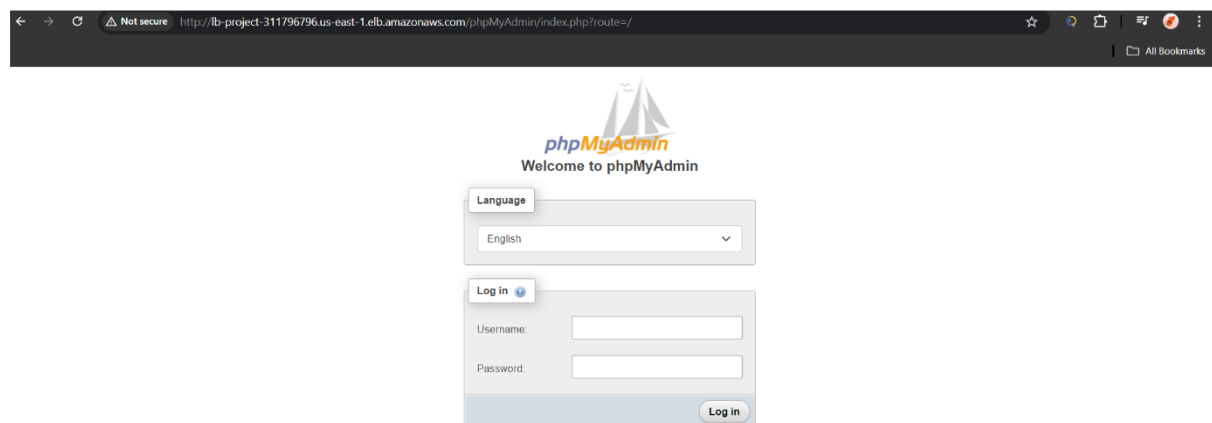
```
GNU nano 2.8 config.inc.php
/**
 * This is needed for cookie based authentication to encrypt the cookie.
 * Needs to be a 32-bytes long string of random bytes. See FAQ 2.10.
 */
$config['blowfish_secret'] = ''; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

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

/**
 * First server
 */
$i++;
/* Authentication type */
$config['Servers'][$i]['auth_type'] = 'cookie';
/* Server parameters */
$config['Servers'][$i]['host'] = 'db-project-cj0e66cakk5j.us-east-1.rds.amazonaws.com';
Broadcast message from root@localhost (Sun 2024-07-14 17:50:59 UTC):
$config['Servers'][$i]['AllowNoPassword'] = false;
The system will power off now!
/**
```

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

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



Once you enter your username and password:

