AWS Architecture for Cloud-Based PHP Application

1. Introduction

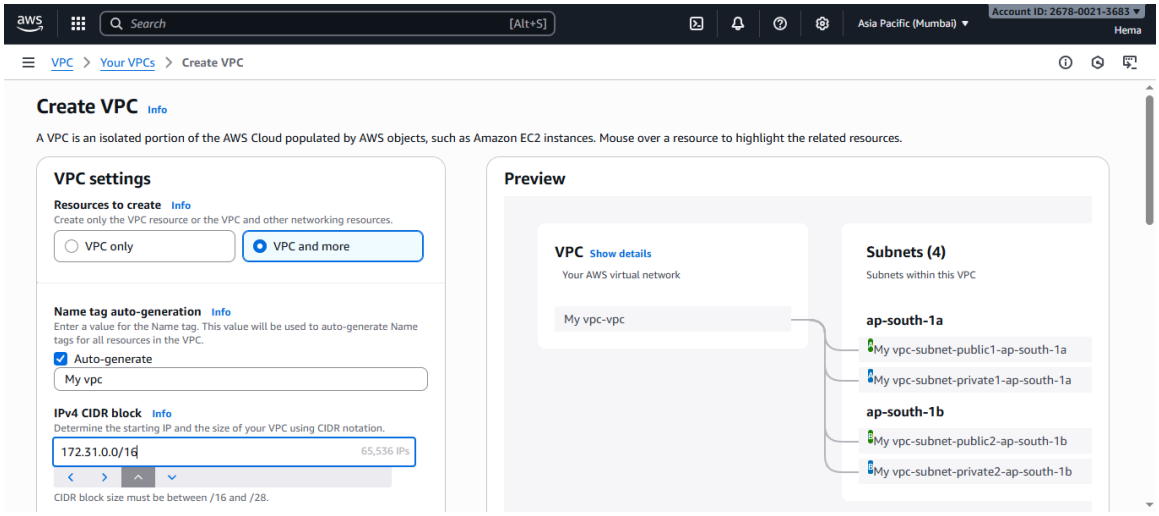
This document describes the AWS solution architecture for hosting a **PHP-based dynamic web application** using **AWS managed services**.
The goal is to design a **secure, scalable, highly available, and cost-effective** cloud environment using a **Load Balancer for high availability**.

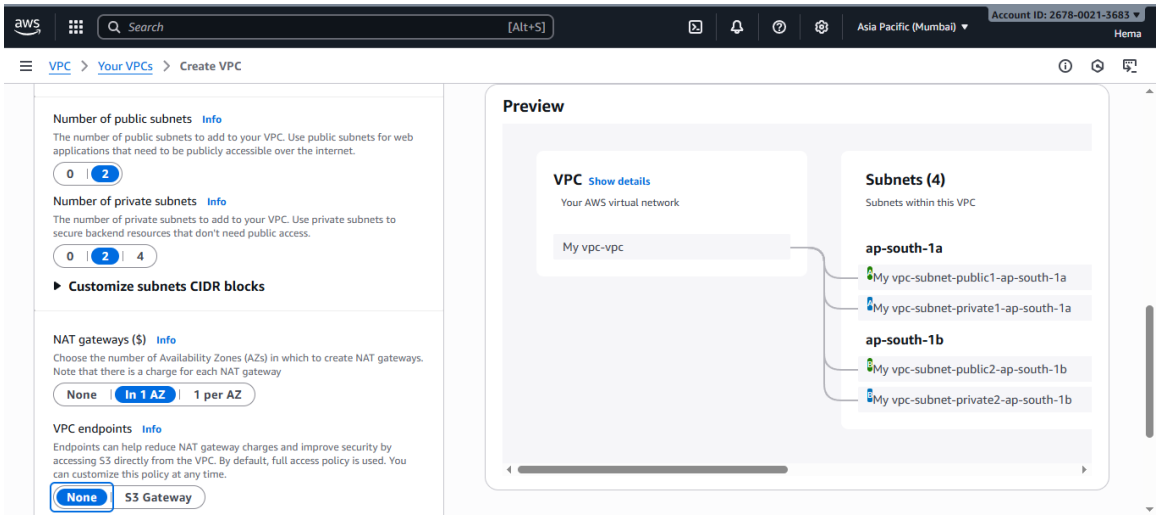
Create a Secure VPC

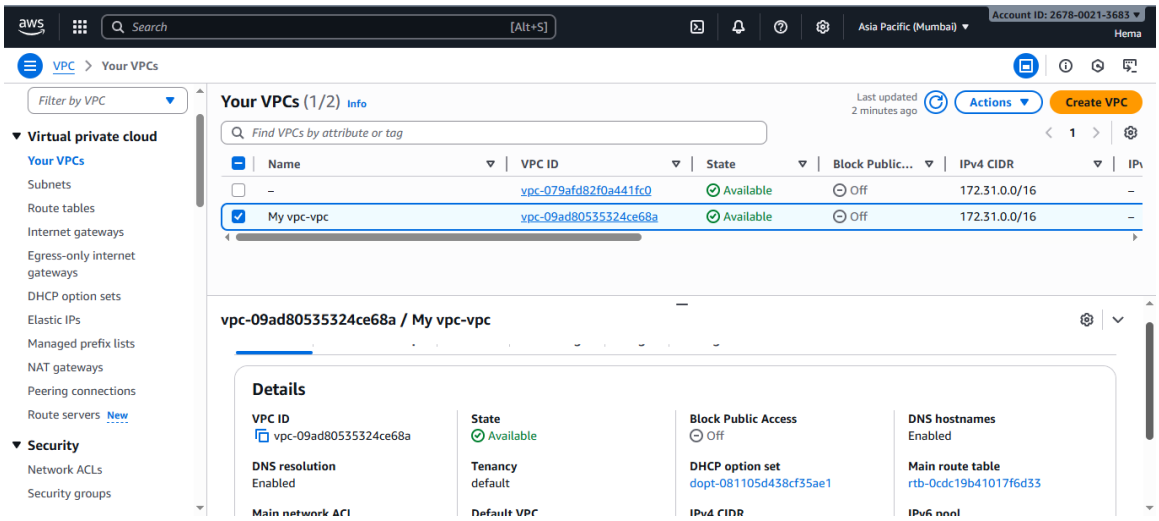
Action:

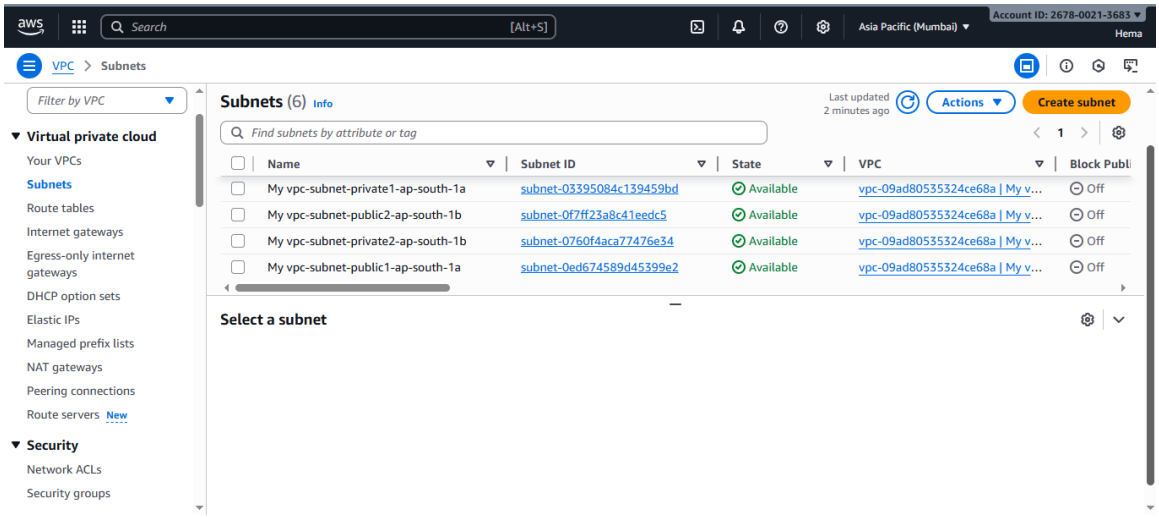
1. Go to **VPC** in AWS.
2. Click **Create VPC → VPC and more**.
3. Configure:
    - **VPC CIDR:** `172.31.0.0/16`
    - **2 Public Subnets:** For load balancer
    - **2 Private Subnets:** For EC2 and RDS
4. Create an **Internet Gateway (IGW)** and attach it to the VPC.
5. Create a **NAT Gateway** in a public subnet to allow private subnets internet access.
6. Set up **Route Tables**:
    - Public route table → IGW
    - Private route table → NAT Gateway
7. Enable **DNS hostnames** and **DNS resolution** in the VPC settings.

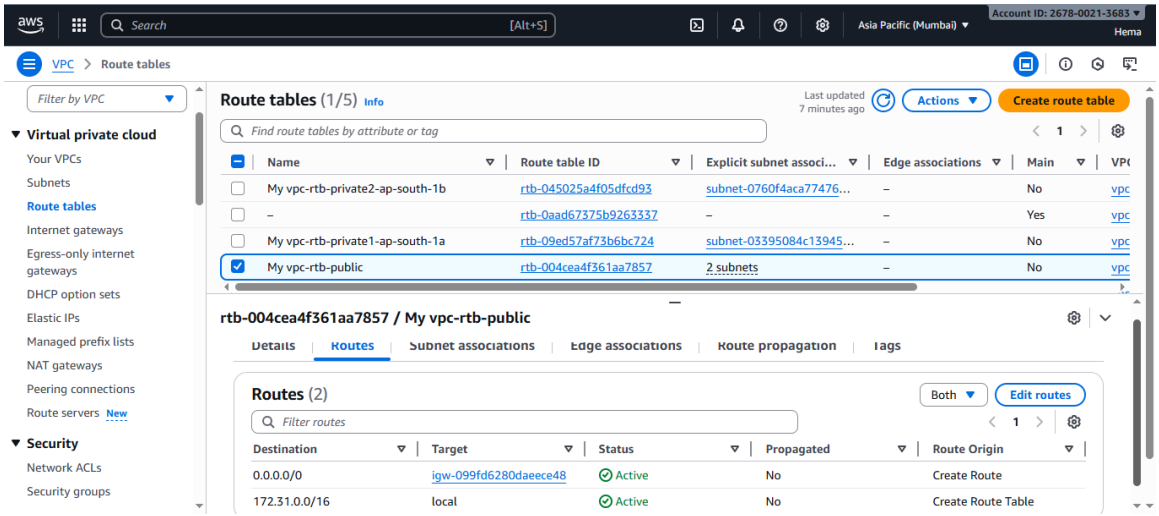
Screenshot:

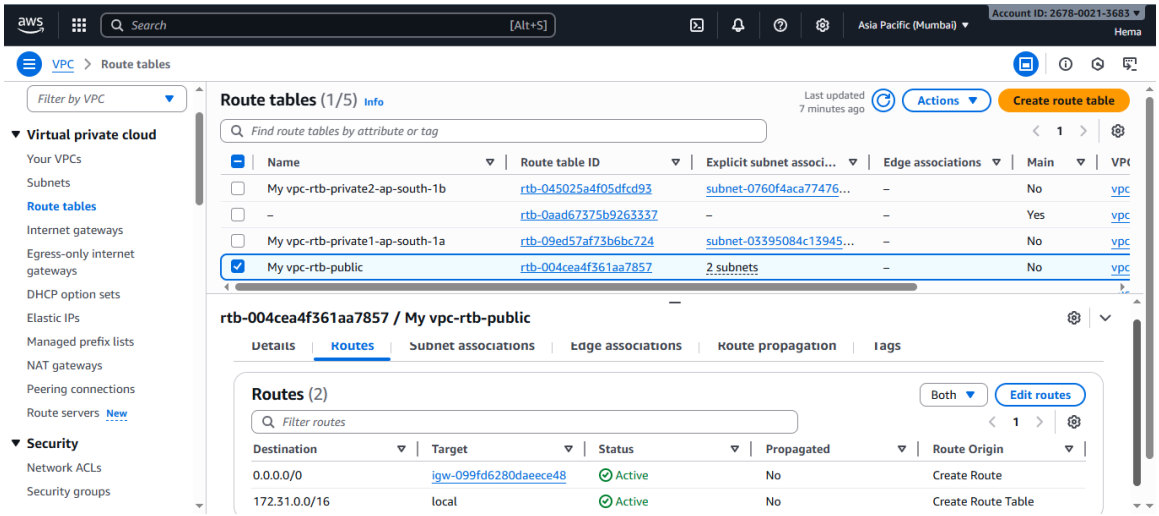


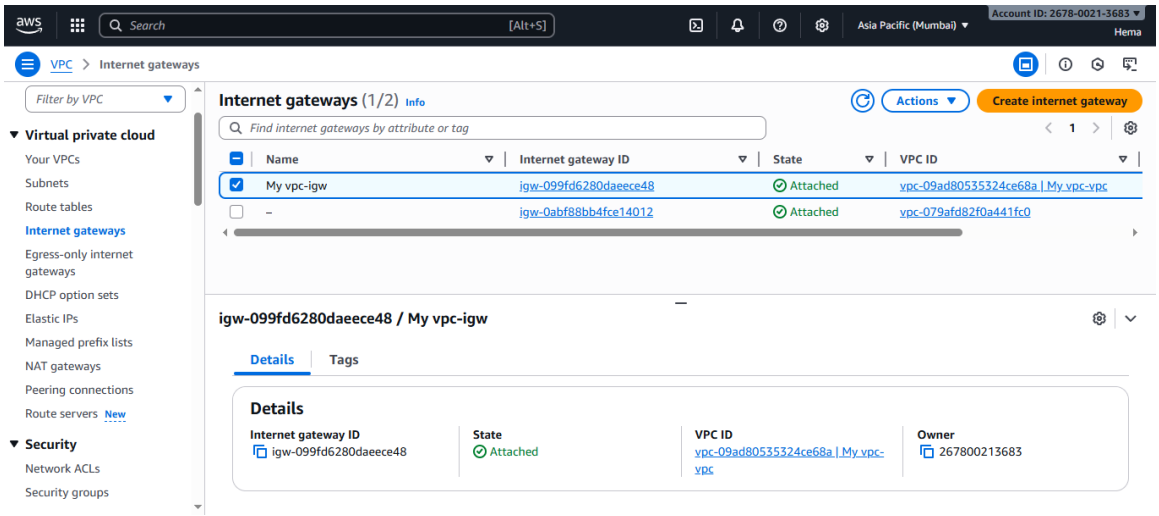

Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Your VPCs > Create VPC
Number of public subnets Info
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.
0 2
Number of private subnets Info
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.
0 2 4
Customize subnets CIDR blocks
NAT gateways ($) Info
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway
None In 1 AZ 1 per AZ
VPC endpoints Info
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.
None S3 Gateway
Preview
VPC Show details
Your AWS virtual network
My vpc-vpc
Subnets (4)
Subnets within this VPC
ap-south-1a
My vpc-subnet-public1-ap-south-1a
My vpc-subnet-private1-ap-south-1a
ap-south-1b
My vpc-subnet-public2-ap-south-1b
My vpc-subnet-private2-ap-south-1b

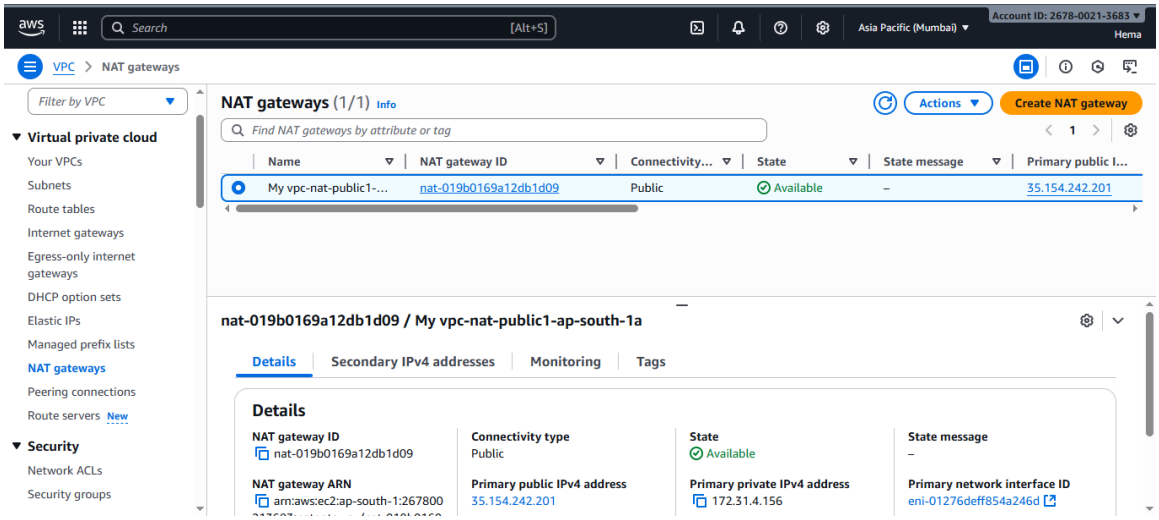

Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Your VPCs
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Route servers New
Security
Network ACLs
Security groups
Your VPCs (1/2) Info
Last updated 2 minutes ago
Actions
Create VPC
Find VPCs by attribute or tag
Name | VPC ID | State | Block Public... | IPv4 CIDR
– | vpc-079afd82f0a441fc0 | Available | Off | 172.31.0.0/16
My vpc-vpc | vpc-09ad80535324ce68a | Available | Off | 172.31.0.0/16
vpc-09ad80535324ce68a / My vpc-vpc
Details
VPC ID
vpc-09ad80535324ce68a
State
Available
Block Public Access
Off
DNS hostnames
Enabled
DNS resolution
Enabled
Tenancy
default
DHCP option set
dopt-081105d438cf35ae1
Main route table
rtb-0cdc19b41017f6d33


Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Subnets
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Route servers New
Security
Network ACLs
Security groups
Subnets (6) Info
Last updated 2 minutes ago
Actions
Create subnet
Find subnets by attribute or tag
Name | Subnet ID | State | VPC | Block Publi
My vpc-subnet-private1-ap-south-1a | subnet-03395084c139459bd | Available | vpc-09ad80535324ce68a | My v... | Off
My vpc-subnet-public2-ap-south-1b | subnet-0f7ff23a8c41eedc5 | Available | vpc-09ad80535324ce68a | My v... | Off
My vpc-subnet-private2-ap-south-1b | subnet-0760f4aca77476e34 | Available | vpc-09ad80535324ce68a | My v... | Off
My vpc-subnet-public1-ap-south-1a | subnet-0ed674589d45399e2 | Available | vpc-09ad80535324ce68a | My v... | Off
Select a subnet


Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Route tables
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Route servers New
Security
Network ACLs
Security groups
Route tables (1/5) Info
Last updated 7 minutes ago
Actions
Create route table
Find route tables by attribute or tag
Name | Route table ID | Explicit subnet associ... | Edge associations | Main | VP
My vpc-rtb-private2-ap-south-1b | rtb-045025a4f05dfcd93 | subnet-0760f4aca77476... | – | No | vpc
– | rtb-0aad67375b9263337 | – | – | Yes | vpc
My vpc-rtb-private1-ap-south-1a | rtb-09ed57af73b6bc724 | subnet-03395084c13945... | – | No | vpc
My vpc-rtb-public | rtb-004cea4f361aa7857 | 2 subnets | – | No | vpc
rtb-004cea4f361aa7857 / My vpc-rtb-public
Details | 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-099fd6280daeece48 | Active | No | Create Route
172.31.0.0/16 | local | Active | No | Create Route Table


Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Route tables
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Route servers New
Security
Network ACLs
Security groups
Route tables (1/5) Info
Last updated 7 minutes ago
Actions
Create route table
Find route tables by attribute or tag
Name | Route table ID | Explicit subnet associ... | Edge associations | Main | VP
My vpc-rtb-private2-ap-south-1b | rtb-045025a4f05dfcd93 | subnet-0760f4aca77476... | – | No | vpc
– | rtb-0aad67375b9263337 | – | – | Yes | vpc
My vpc-rtb-private1-ap-south-1a | rtb-09ed57af73b6bc724 | subnet-03395084c13945... | – | No | vpc
My vpc-rtb-public | rtb-004cea4f361aa7857 | 2 subnets | – | No | vpc
rtb-004cea4f361aa7857 / My vpc-rtb-public
Details | 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-099fd6280daeece48 | Active | No | Create Route
172.31.0.0/16 | local | Active | No | Create Route Table


Search [Alt+S]
Asia Pacific (Mumbai)
Account ID: 2678-0021-3683
Hema
VPC > Internet gateways
Filter by VPC
Virtual private cloud
Your VPCs
Subnets
Route tables
Internet gateways
Egress-only internet gateways
DHCP option sets
Elastic IPs
Managed prefix lists
NAT gateways
Peering connections
Route servers New
Security
Network ACLs
Security groups
Internet gateways (1/2) Info
Actions
Create internet gateway
Find internet gateways by attribute or tag
Name | Internet gateway ID | State | VPC ID
My vpc-igw | igw-099fd6280daeece48 | Attached | vpc-09ad80535324ce68a | My vpc-vpc
– | igw-0abf88bb4fce14012 | Attached | vpc-079afd82f0a441fc0
igw-099fd6280daeece48 / My vpc-igw
Details | Tags
Details
Internet gateway ID
igw-099fd6280daeece48
State
Attached
VPC ID
vpc-09ad80535324ce68a | My vpc-vpc
Owner
267800213683

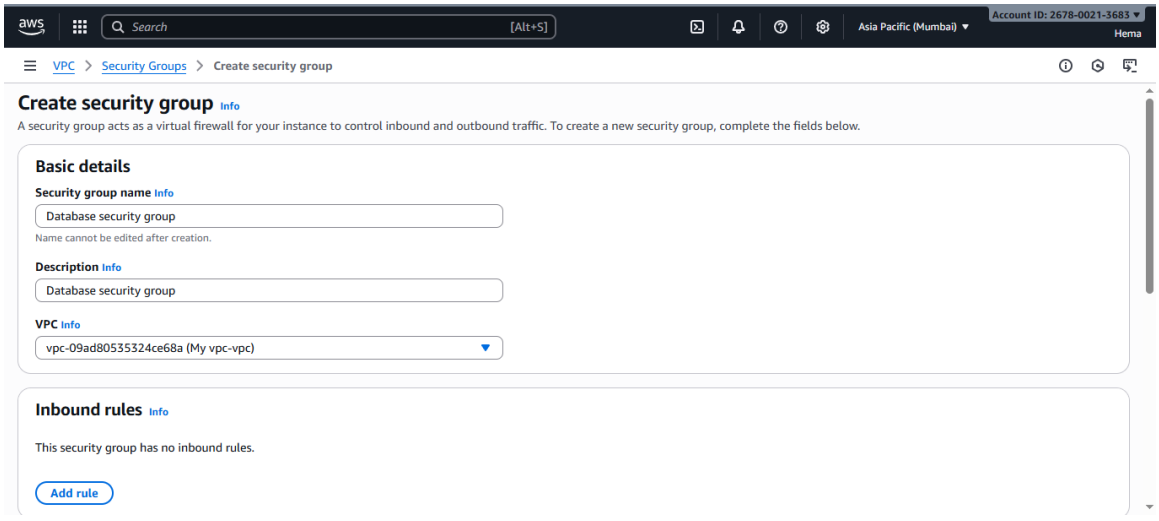


Create Security Groups

Action:

1. Create **the Security Groups**:
   - **Database security group:** Allow inbound MySQL (3306) from EC2 SG only.
2. Apply the security groups to the respective resources.

Screenshot:

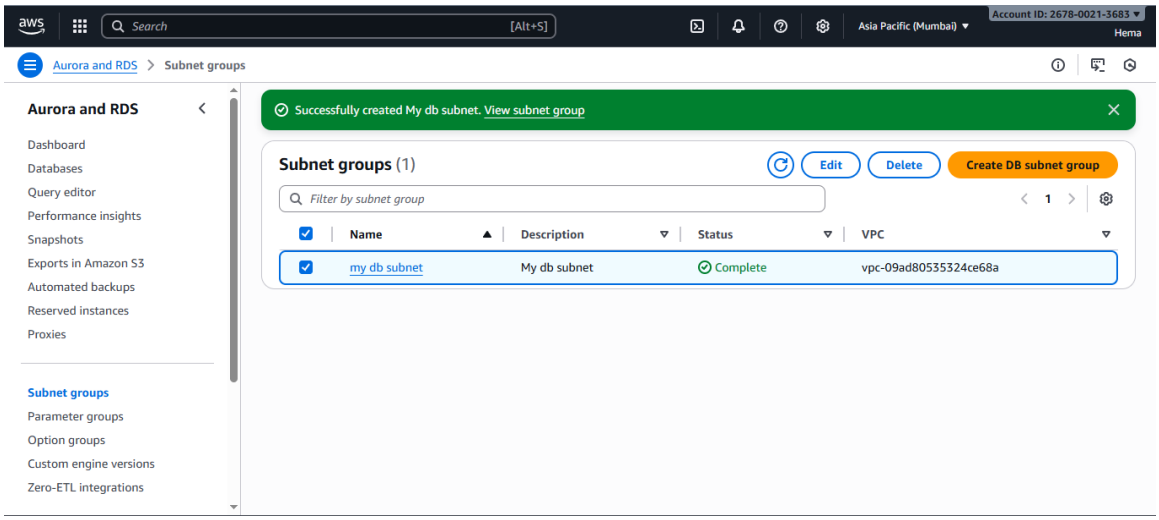


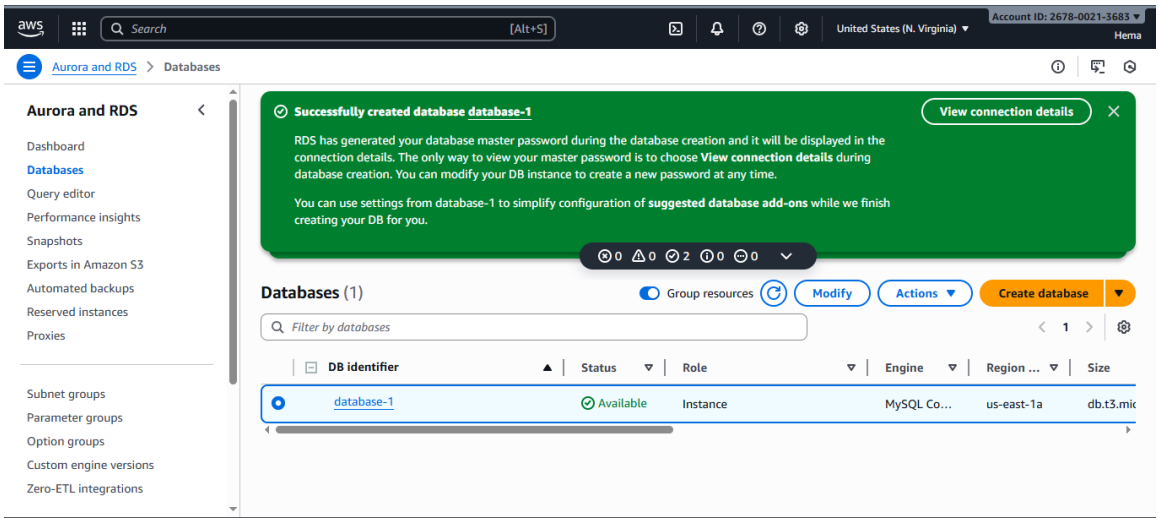
Create Amazon RDS (Database)

Action:

1. Go to **RDS → Create Database**.
2. Choose **Standard Create → MySQL**
3. Enable **Multi-AZ Deployment** for high availability.
4. Under **Connectivity**:
    - Select your custom **VPC**
    - Create a **DB Subnet Group** with private subnets
    - Attach the **RDS Security Group**
5. Note the **DB endpoint** and credentials for your application.

Screenshot:



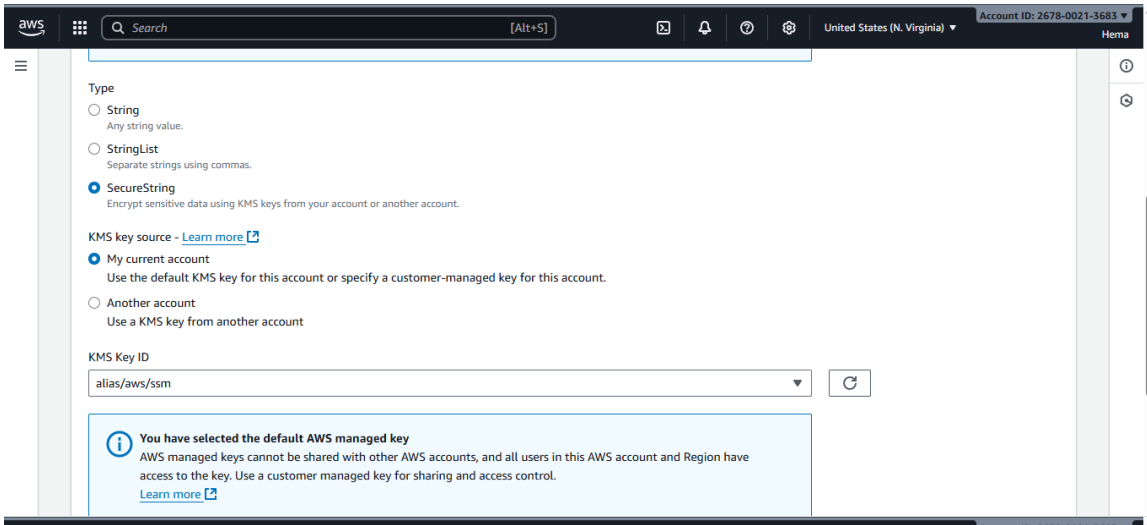


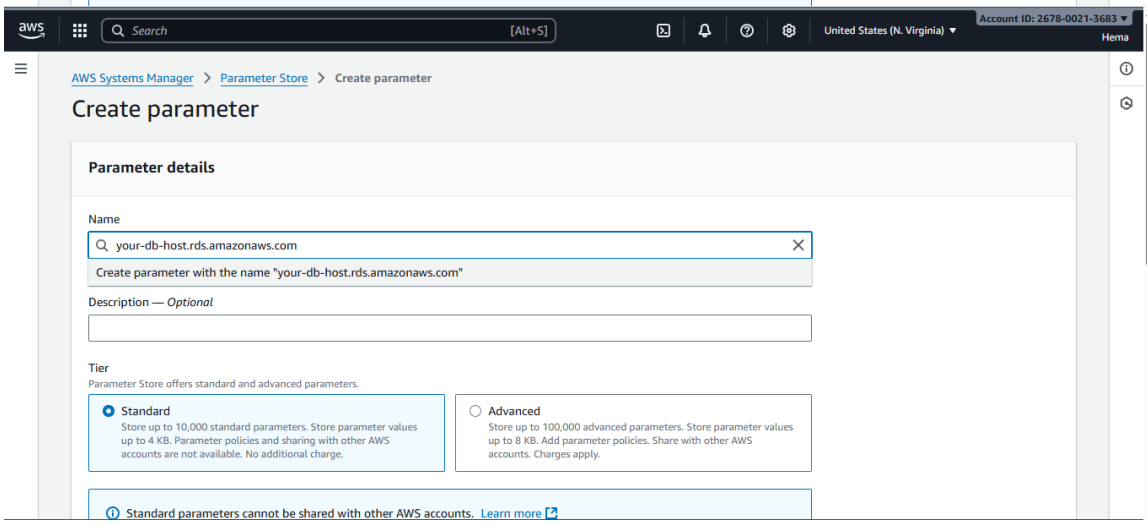
Store Secrets in AWS Systems Manager Parameter Store

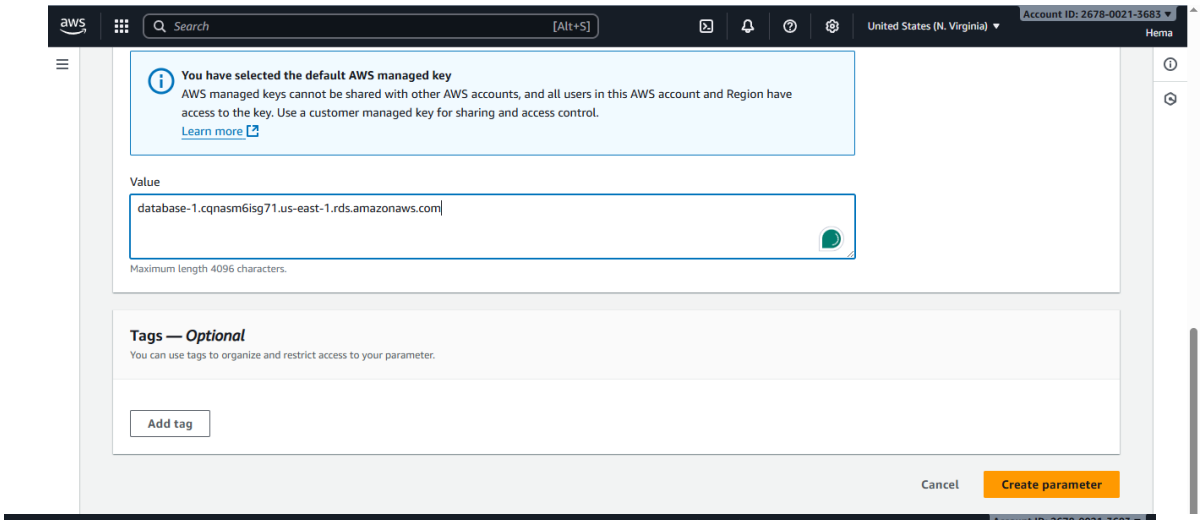
Action:

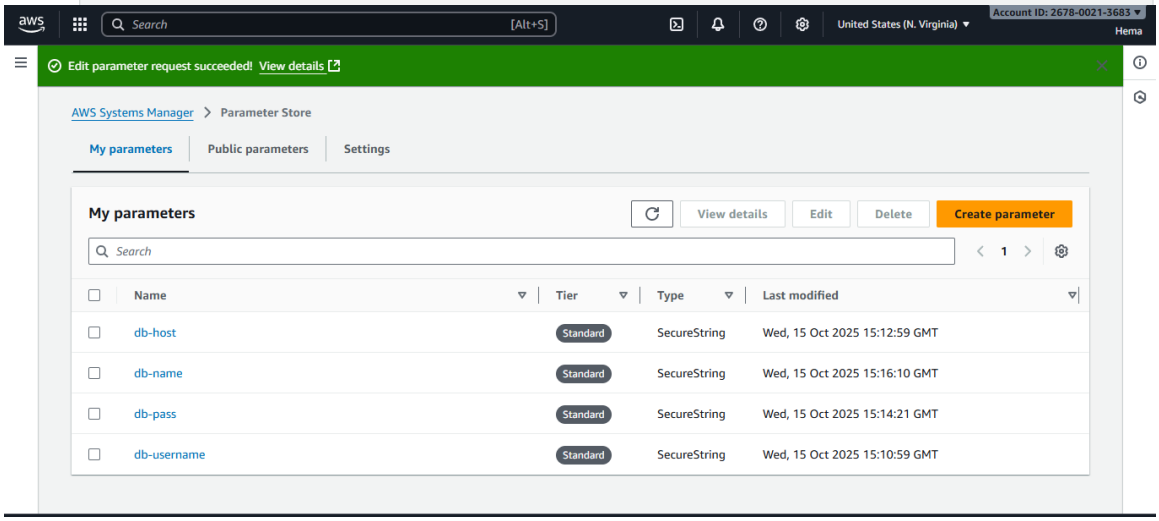
1. Navigate to **Systems Manager → Parameter Store**.
2. Create **SecureString Parameters** for:
    - `/app/db/username`
    - `/app/db/password`
    - `/app/db/host`
3. Use **AWS KMS default key** for encryption.

Screenshot:







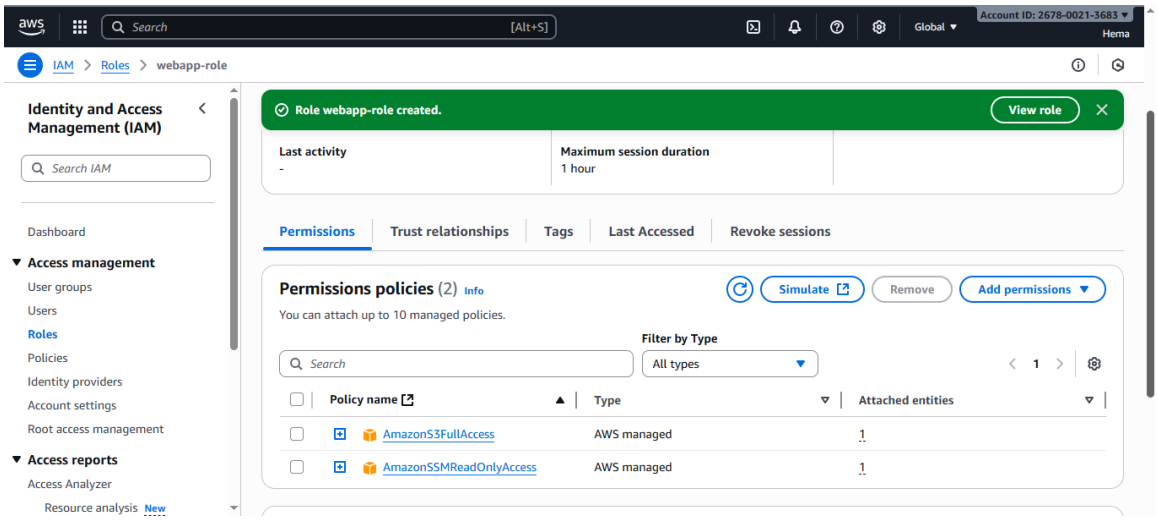


Configure IAM Roles and Policies

Action:

1. Go to **IAM → Roles → Create Role**.
2. Select **EC2** as trusted entity.
3. Attach managed policies:
   - `AmazonSSMReadOnlyAccess` (to read Parameter Store)
   - `AmazonS3FullAccess` (optional, for storing assets)
4. Name the role (Web`appRole`)

Screenshot:

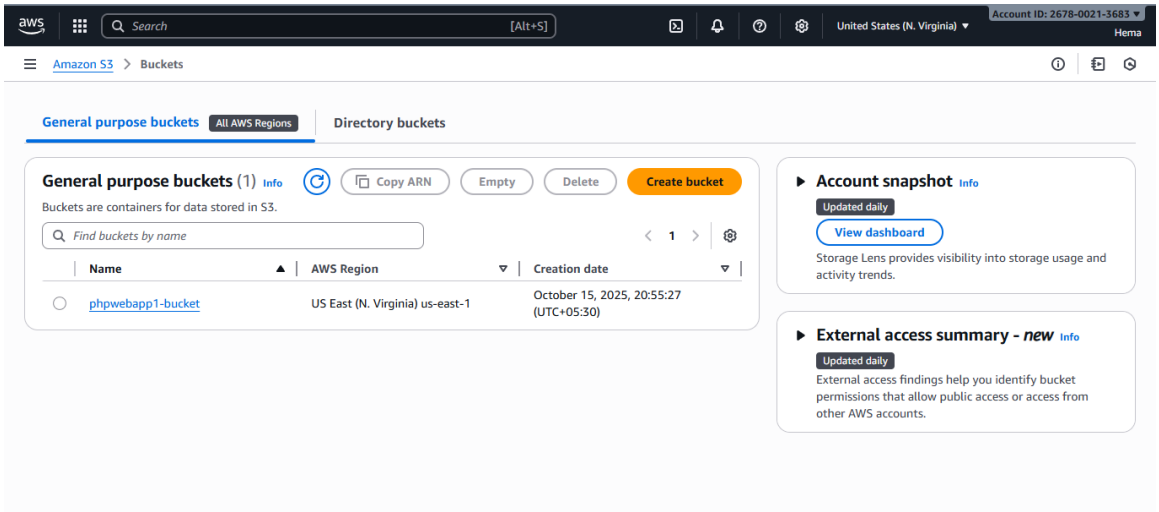


Create S3 Bucket for Application Assets

Action:

1. Navigate to **S3 → Create Bucket**.
2. Name: `phpwebapp1-bucket`
3. Enable:
    - **Server-Side Encryption (SSE-S3)**

Screenshot:

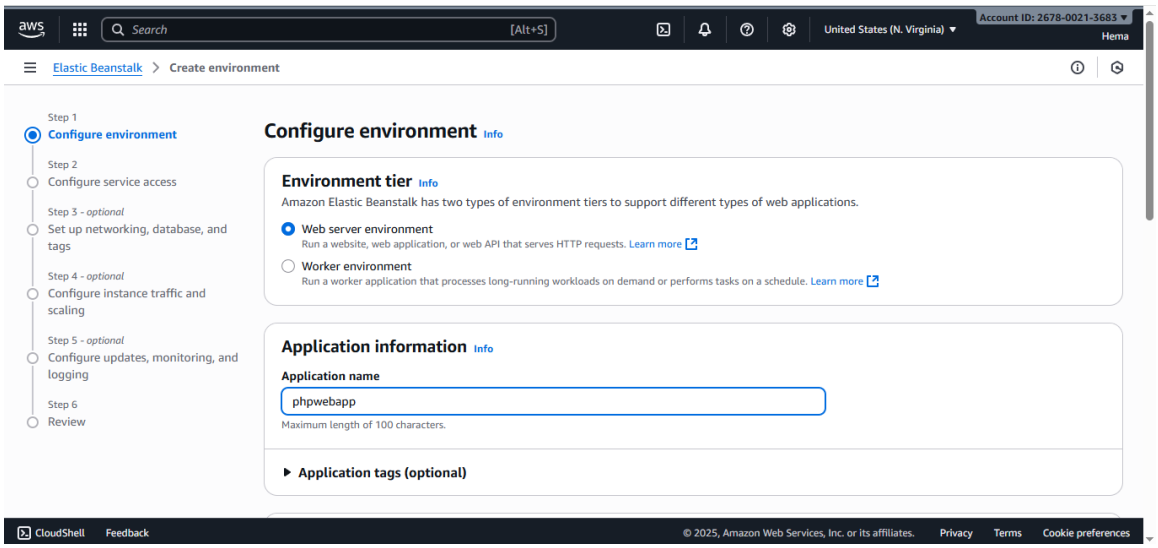


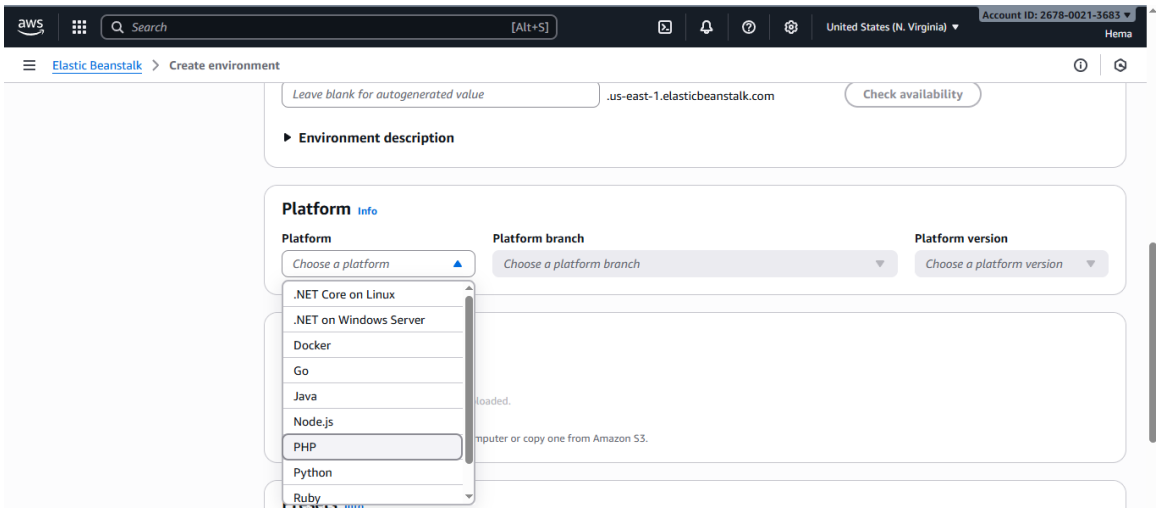
Deploy PHP Application using Elastic Beanstalk

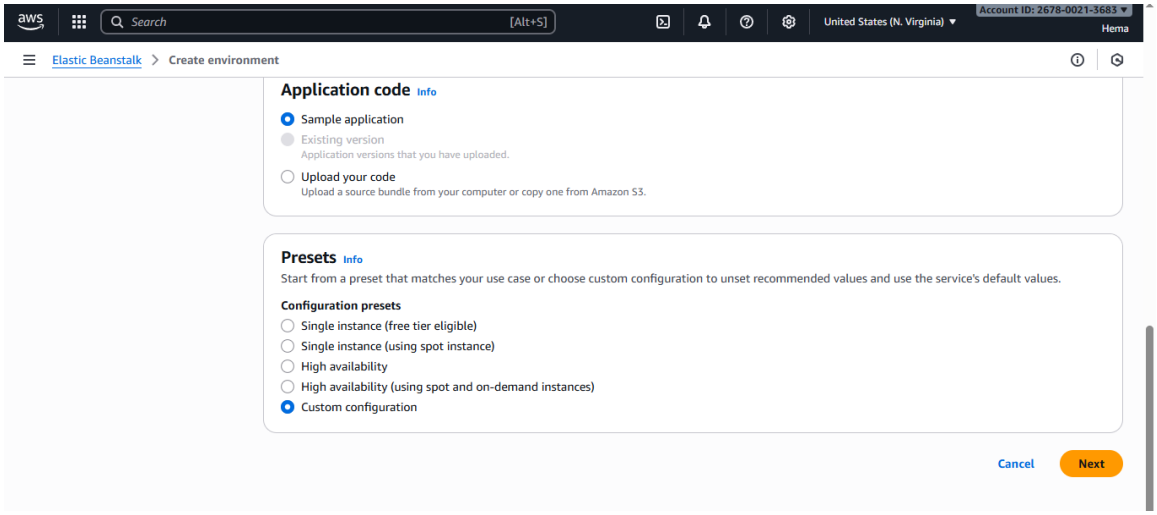
Action:

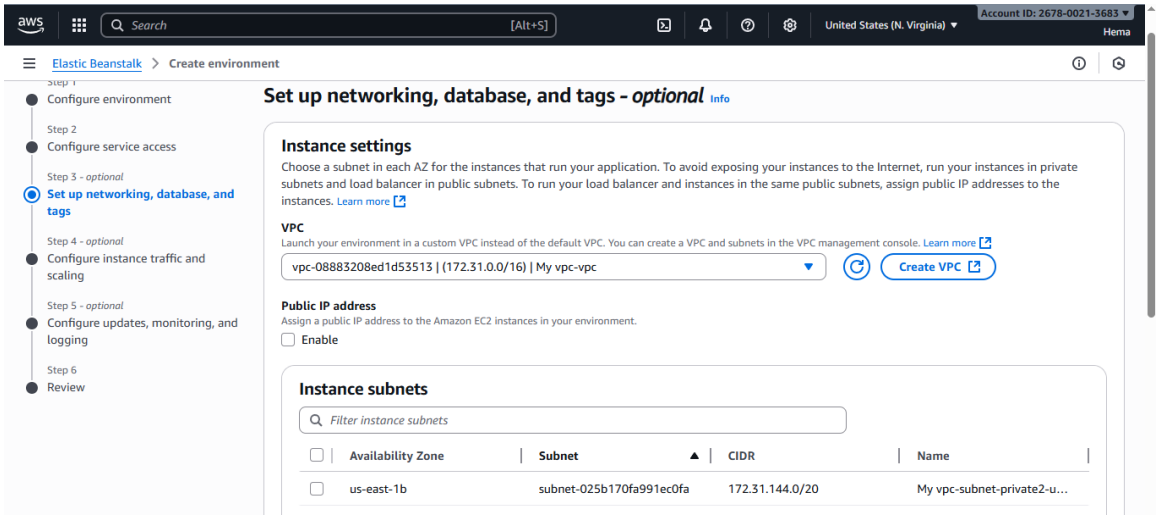
1. Go to **Elastic Beanstalk → Create Application**.
2. Application Name: `php-webapp`
3. Platform: **PHP**
4. Upload your application ZIP file.
5. Under **Configuration**:
    - Environment Type: **Load balanced, auto-scaling**
    - Select your custom **VPC** and **subnets**
    - Attach **EC2AppRole** IAM role
6. Deploy the application and verify environment health.

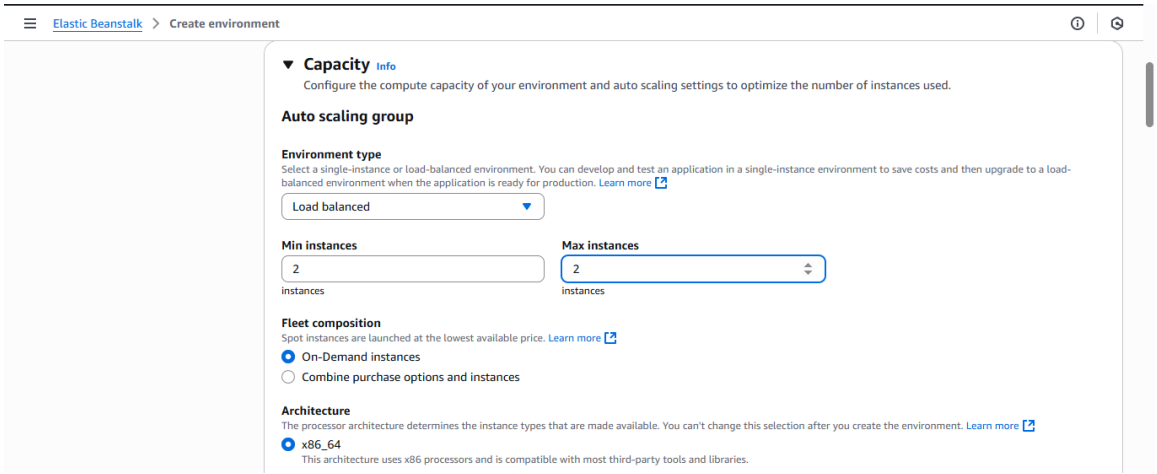
Screenshot:

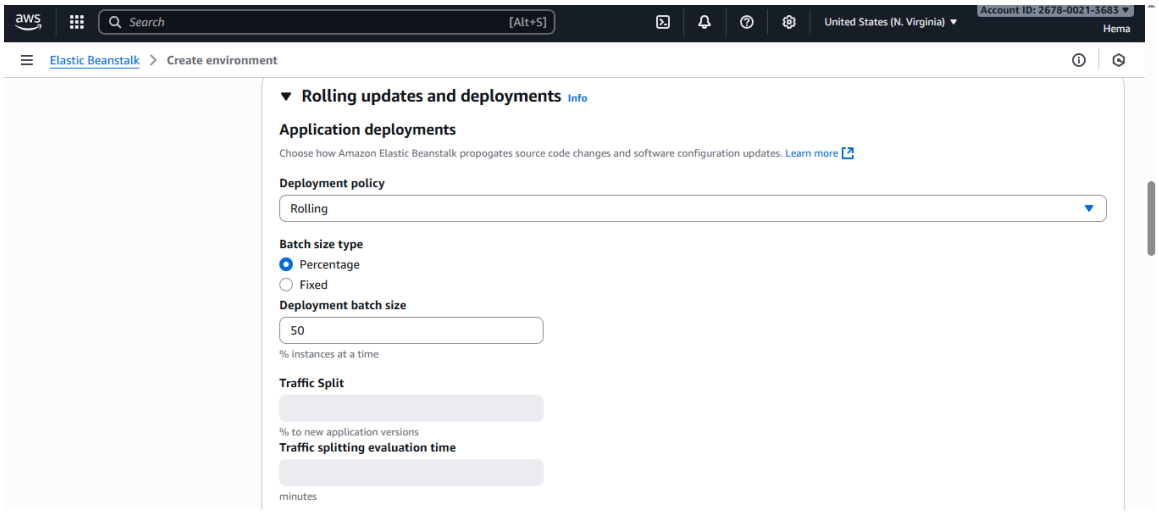





Search
[Alt+S]
United States (N. Virginia)
Account ID: 2678-0021-3683
Hema
Elastic Beanstalk
Create environment
Application code Info
Sample application
Existing version
Application versions that you have uploaded.
Upload your code
Upload a source bundle from your computer or copy one from Amazon S3.
Presets Info
Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.
Configuration presets
Single instance (free tier eligible)
Single instance (using spot instance)
High availability
High availability (using spot and on-demand instances)
Custom configuration
Cancel
Next


Search
[Alt+S]
United States (N. Virginia)
Account ID: 2678-0021-3683
Hema
Elastic Beanstalk
Create environment
Configure environment
Step 2
Configure service access
Step 3 - optional
Set up networking, database, and tags
Step 4 - optional
Configure instance traffic and scaling
Step 5 - optional
Configure updates, monitoring, and logging
Step 6
Review
Set up networking, database, and tags - optional Info
Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more
VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console. Learn more
vpc-08883208ed1d53513 | (172.31.0.0/16) | My vpc-vpc
Create VPC
Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
Enable
Instance subnets
Filter instance subnets
Availability Zone
Subnet
CIDR
Name
us-east-1b
subnet-025b170fa991ec0fa
172.31.144.0/20
My vpc-subnet-private2-u...


Elastic Beanstalk
Create environment
Capacity Info
Configure the compute capacity of your environment and auto scaling settings to optimize the number of instances used.
Auto scaling group
Environment type
Select a single-instance or load-balanced environment. You can develop and test an application in a single-instance environment to save costs and then upgrade to a load-balanced environment when the application is ready for production. Learn more
Load balanced
Min instances
2
instances
Max instances
2
instances
Fleet composition
Spot instances are launched at the lowest available price. Learn more
On-Demand instances
Combine purchase options and instances
Architecture
The processor architecture determines the instance types that are made available. You can't change this selection after you create the environment. Learn more
x86_64
This architecture uses x86 processors and is compatible with most third-party tools and libraries.



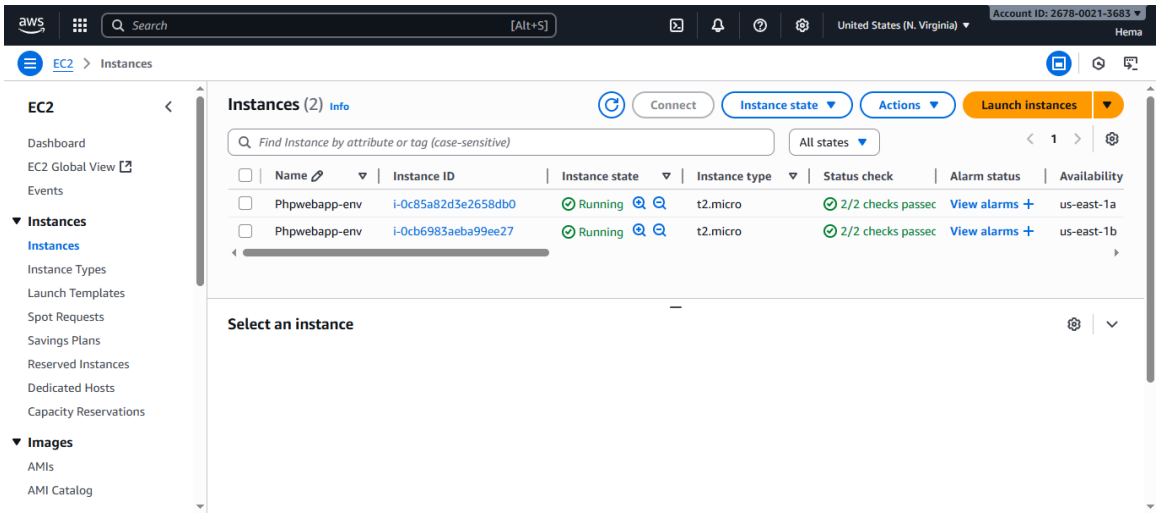
--→Create environment

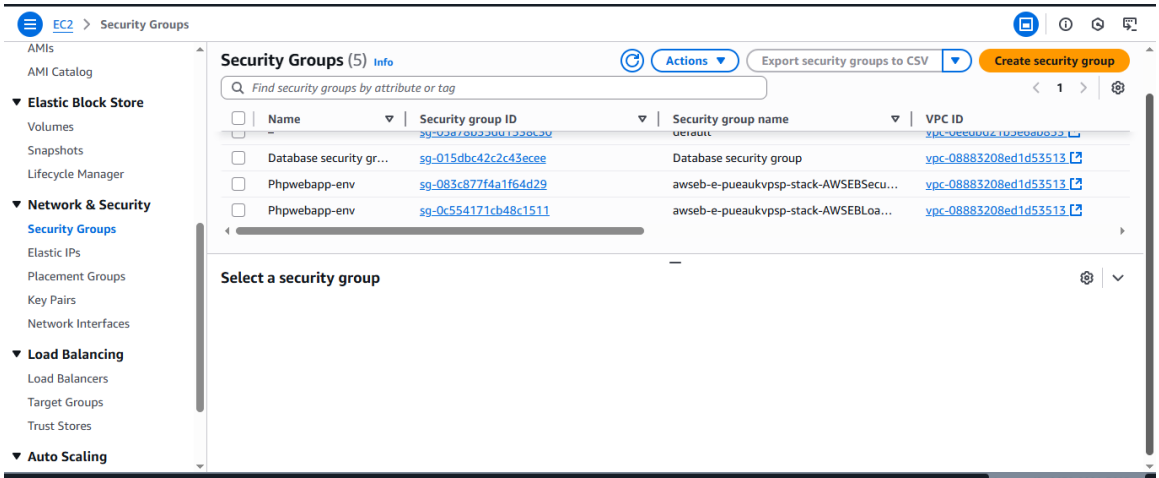
Set Up Load Balancer and Auto Scaling

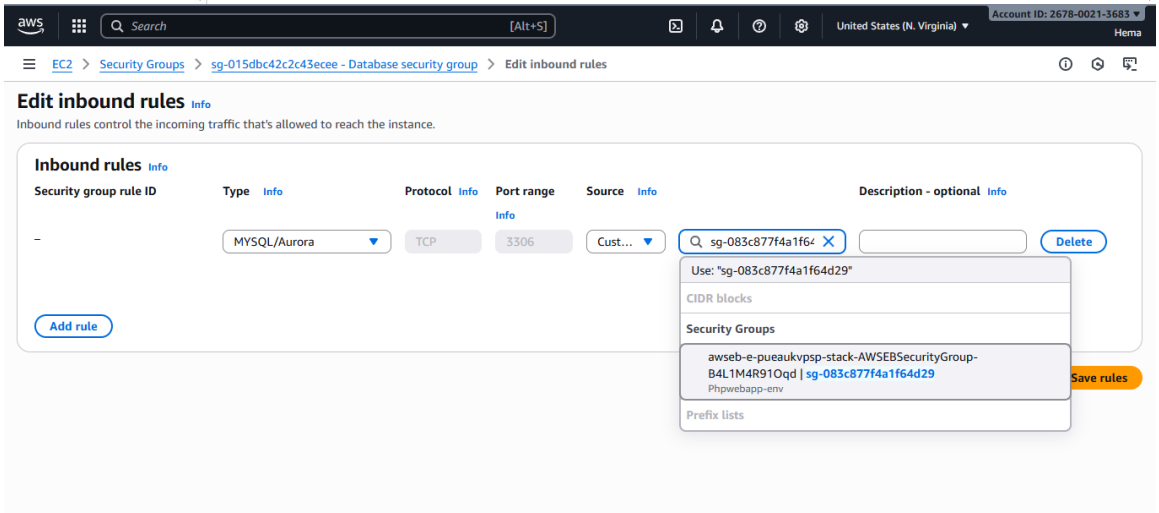
Action:

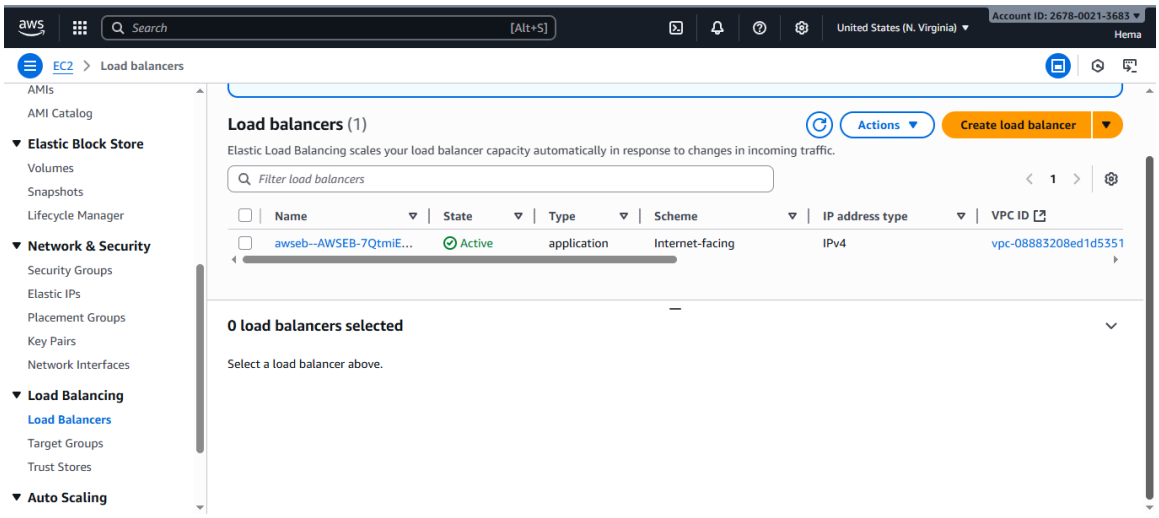
1. Open **Elastic Beanstalk → Configuration → Capacity**.
2. Configure:
   - **Minimum Instances:** 2
   - **Maximum Instances:** 2
   - **Scaling Trigger:** CPU Utilization
   - Scale Out: CPU > 70%
   - Scale In: CPU < 30%
3. Elastic Beanstalk automatically creates a **Load Balancer** to distribute traffic among EC2 instances.

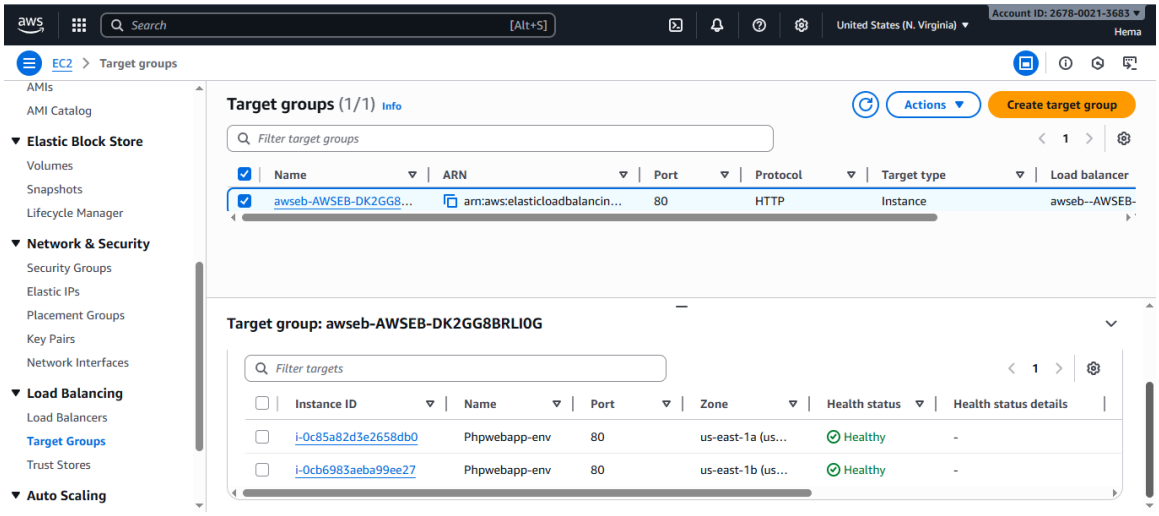
Screenshot:

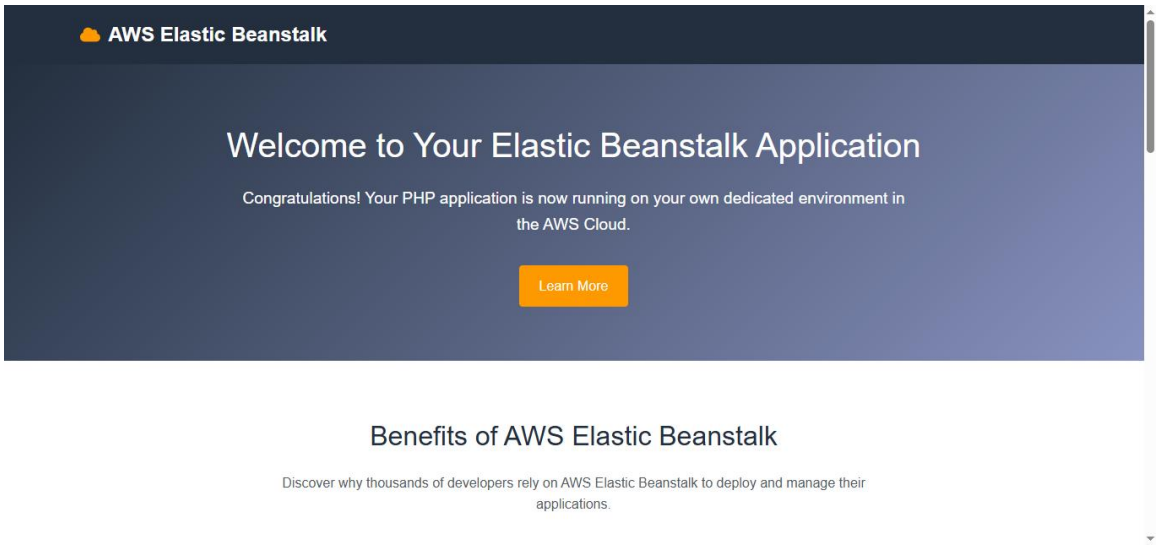

aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2678-0021-3683
Hema
EC2 > Instances
EC2
Dashboard
EC2 Global View
Events
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Instances (2) Info
Connect
Instance state
Actions
Launch instances
Find Instance by attribute or tag (case-sensitive)
All states
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability
Phpwebapp-env
i-0c85a82d3e2658db0
Running
t2.micro
2/2 checks passec
View alarms
us-east-1a
Phpwebapp-env
i-0cb6983aeba99ee27
Running
t2.micro
2/2 checks passec
View alarms
us-east-1b
Select an instance


EC2 > Security Groups
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Network Interfaces
Load Balancing
Load Balancers
Target Groups
Trust Stores
Auto Scaling
Security Groups (5) Info
Actions
Export security groups to CSV
Create security group
Find security groups by attribute or tag
Name
Security group ID
Security group name
VPC ID
Database security gr...
sg-015dbc42c2c43ecee
Database security group
vpc-08883208ed1d53513
Phpwebapp-env
sg-083c877f4a1f64d29
awseb-e-pueaukvpsp-stack-AWSEBSecu...
vpc-08883208ed1d53513
Phpwebapp-env
sg-0c554171cb48c1511
awseb-e-pueaukvpsp-stack-AWSEBLoa...
vpc-08883208ed1d53513
Select a security group


aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 2678-0021-3683
Hema
EC2 > Security Groups > sg-015dbc42c2c43ecee - Database security group > Edit inbound rules
Edit inbound rules Info
Inbound rules control the incoming traffic that's allowed to reach the instance.
Inbound rules Info
Security group rule ID
Type Info
Protocol Info
Port range Info
Source Info
Description - optional Info
MYSQL/Aurora
TCP
3306
Cust...
sg-083c877f4a1f64
Delete
Use: "sg-083c877f4a1f64d29"
CIDR blocks
Security Groups
awseb-e-pueaukvpsp-stack-AWSEBSecurityGroup-B4L1M4R91Oqd | sg-083c877f4a1f64d29
Phpwebapp-env
Prefix lists
Add rule
Save rules

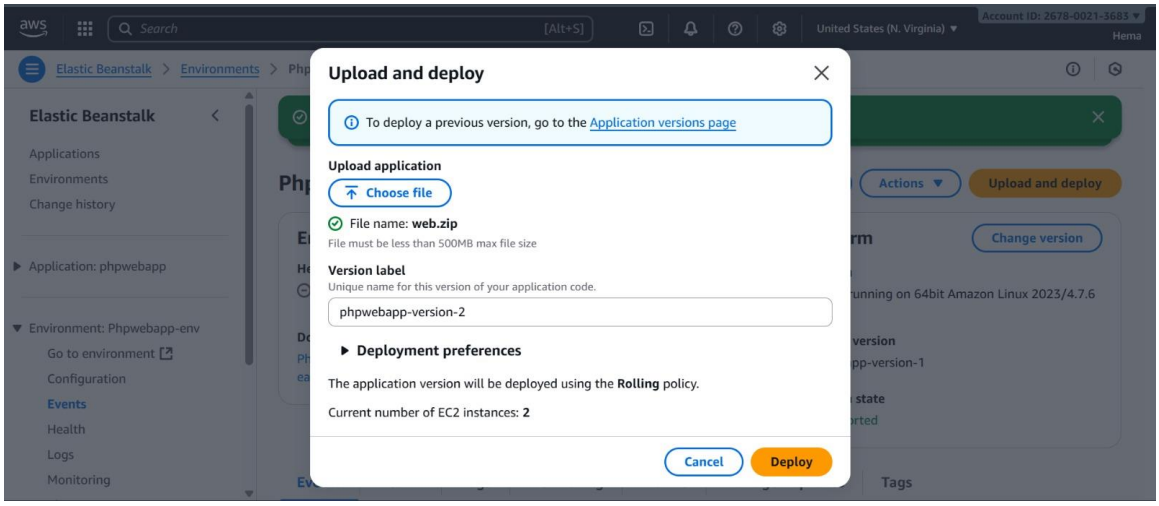


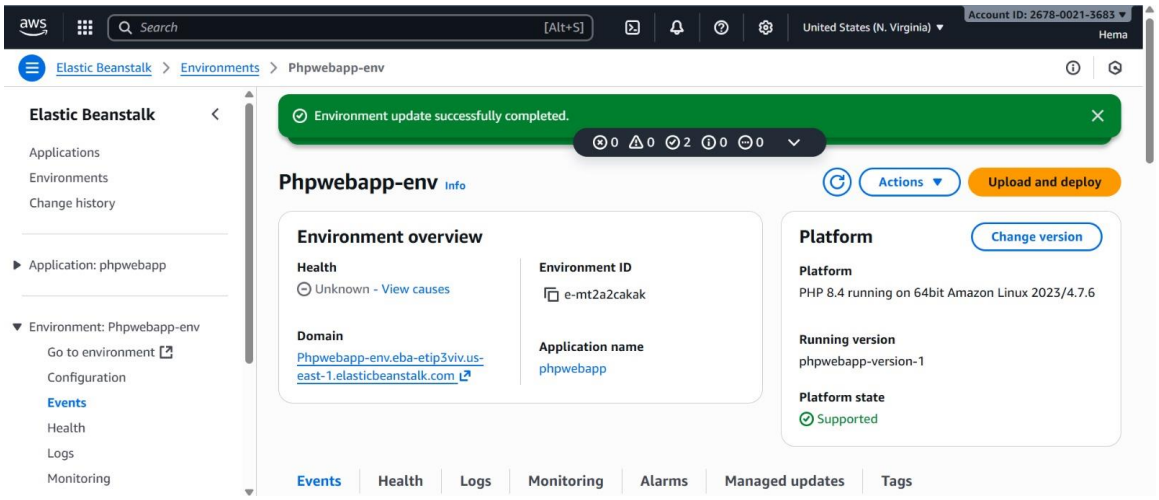




Benefits of AWS Elastic Beanstalk

Discover why thousands of developers rely on AWS Elastic Beanstalk to deploy and manage their applications.


Upload and deploy
To deploy a previous version, go to the Application versions page
Upload application
Choose file
File name: web.zip
File must be less than 500MB max file size
Version label
Unique name for this version of your application code.
phpwebapp-version-2
Deployment preferences
The application version will be deployed using the Rolling policy.
Current number of EC2 instances: 2
Cancel
Deploy


Environment update successfully completed.
Phpwebapp-env
Info
Actions
Upload and deploy
Environment overview
Health
Unknown - View causes
Environment ID
e-mt2a2cakak
Domain
Phpwebapp-env.eba-etip3viv.us-east-1.elasticbeanstalk.com
Application name
phpwebapp
Platform
Change version
Platform
PHP 8.4 running on 64bit Amazon Linux 2023/4.7.6
Running version
phpwebapp-version-1
Platform state
Supported
Events
Health
Logs
Monitoring
Alarms
Managed updates
Tags

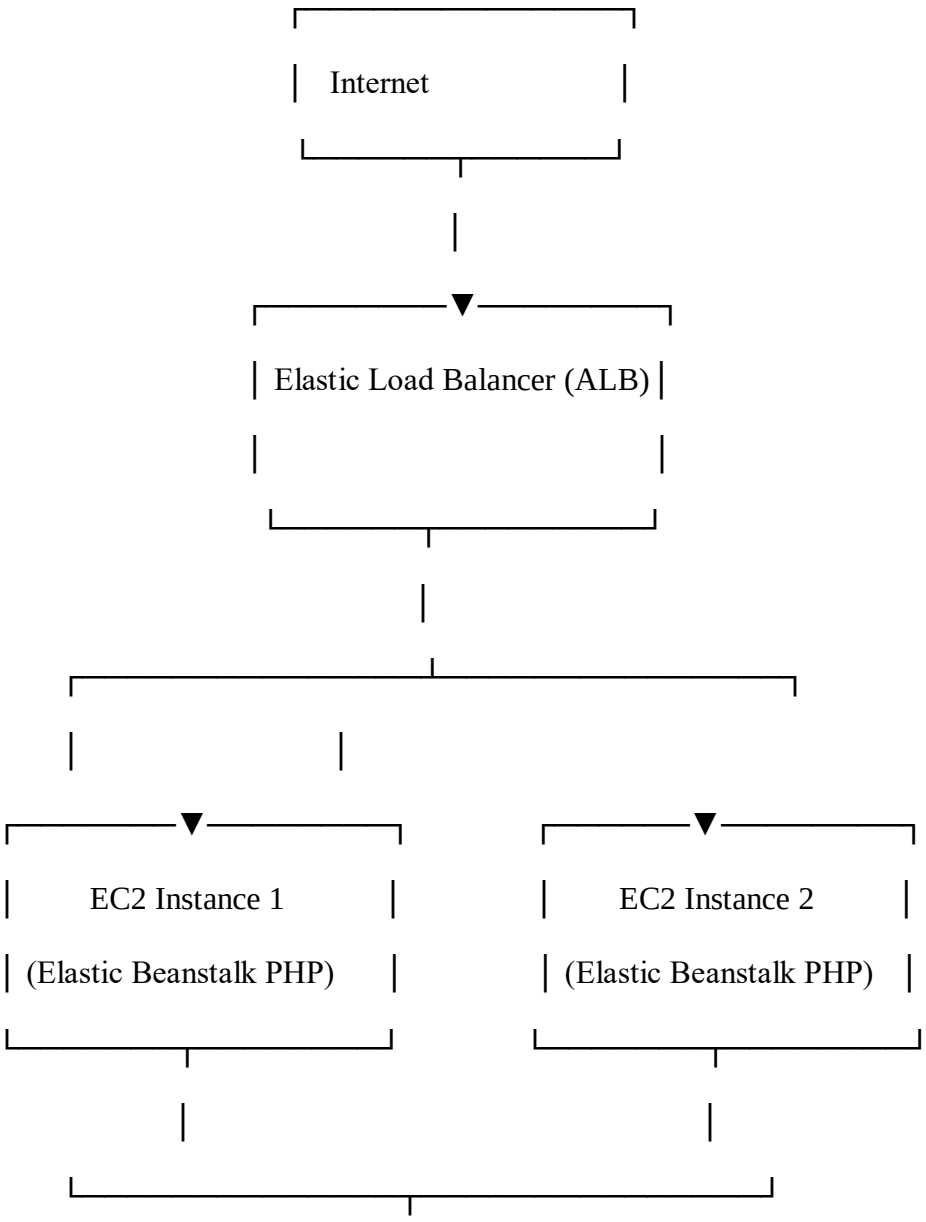

Not secure
phpwebapp-env.eba-etip3viv.us-east-1.elasticbeanstalk.com/index.php
Contact Us
Name:
Hema
Email:
hems25112002@gmail.com
Message:
For Test Check
Send
Thanks for contacting us, Hema! We'll get back to you at hems25112002@gmail.com.

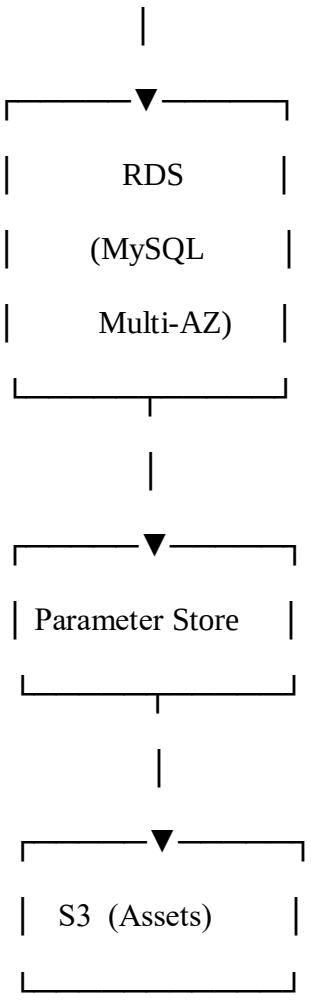
Final Architecture Diagram

Action:
Draw a diagram showing:

- VPC with public/private subnets
- Elastic Beanstalk (PHP app) in private subnet
- Load Balancer in public subnet
- RDS in private subnet
- S3 for static assets
- IAM roles and Parameter Store integration

Screenshot:





✅**Conclusion**
This architecture provides a **secure, scalable, and highly available AWS environment** for a PHP web application using:

- **VPC** for network isolation
- **Security Groups** for access control
- **RDS** for managed database
- **Parameter Store** for secrets
- **Elastic Beanstalk** for app deployment
- **EC2 + Load Balancer** for high availability
- **S3** for static assets