

## Single instance using beanstalk (RDS)

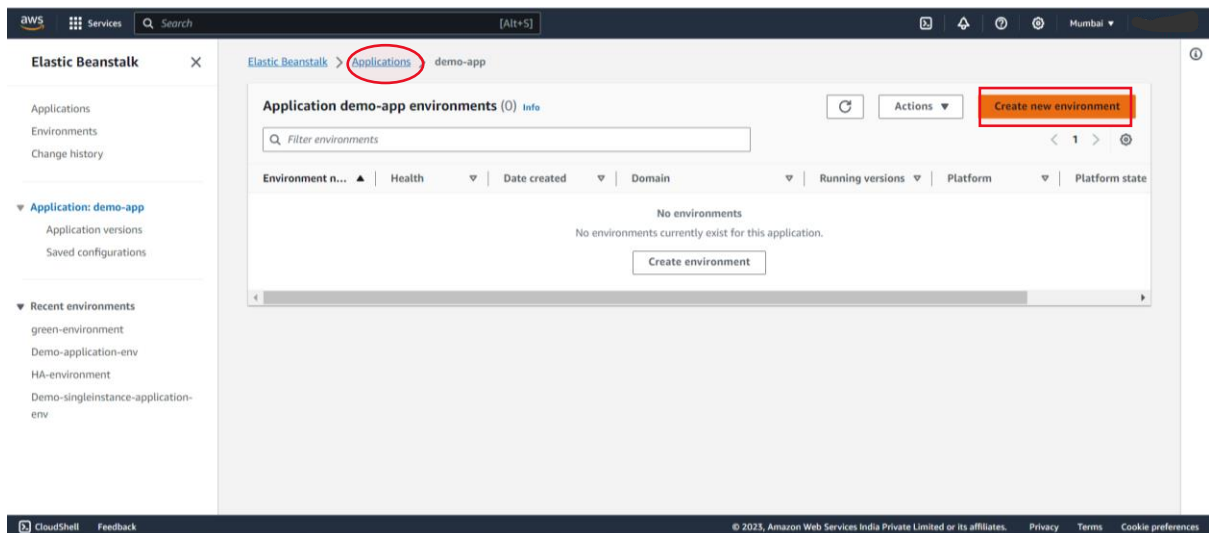
The screenshot shows the Amazon Elastic Beanstalk console landing page. The header includes the AWS logo, 'Services', a search bar, and a location dropdown set to 'Mumbai'. The main content area has a dark blue background with the text 'Amazon Elastic Beanstalk End-to-end web application management.' Below this, a 'Get started' section explains that the service is easy to use for deploying and scaling web applications. To the right, there are two white boxes: 'Get started' with a 'Create application' button, and 'Pricing' which states there is no additional charge for Elastic Beanstalk itself. The footer contains 'CloudShell', 'Feedback', and copyright information for Amazon Web Services India Private Limited.

Click on create application

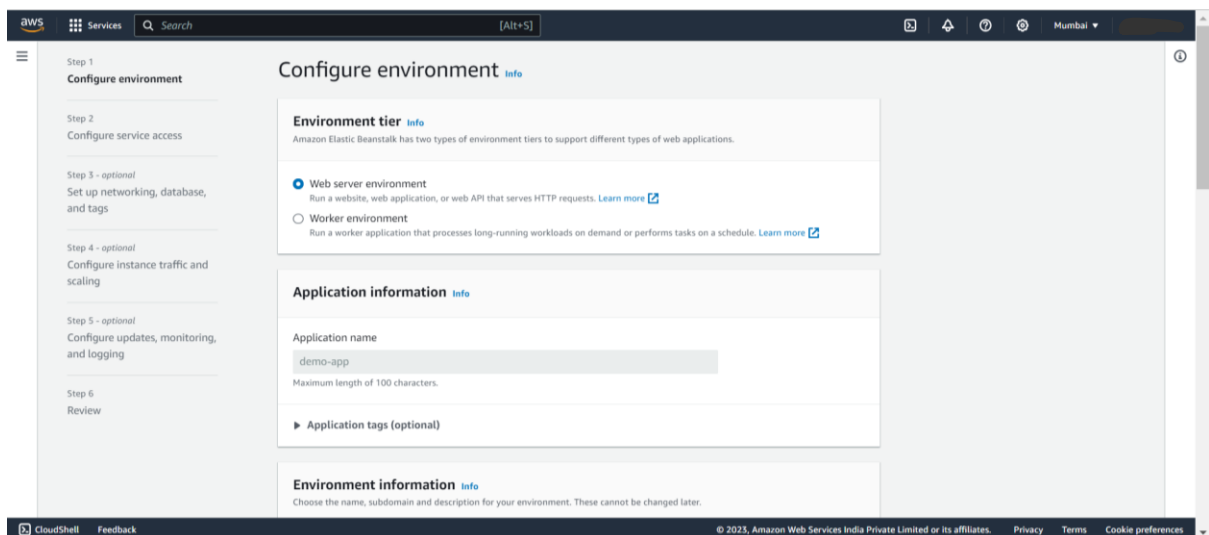
This screenshot shows the 'Applications' page in the Elastic Beanstalk console. The left sidebar has a menu with 'Applications' (selected), 'Environments', and 'Change history'. Below this is a list of 'Recent environments'. The main area is titled 'Applications (0) info' and contains a search bar and a table. The table has columns for 'Application name', 'Environments', 'Date created', 'Last modified', and 'ARN'. It currently shows 'No applications' and 'No applications to display'. A 'Create application' button is in the top right corner. The footer is identical to the previous screenshot.

This screenshot shows the 'Create application' form. It is titled 'Application information'. The 'Application name' field contains 'demo-app' with a note 'Maximum length of 100 characters.' The 'Description' field also contains 'demo-app'. Below this is a 'Tags' section with a note 'Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive.' and a button 'Add new tag'. At the bottom right are 'Cancel' and 'Create' buttons. The footer is identical to the previous screenshots.

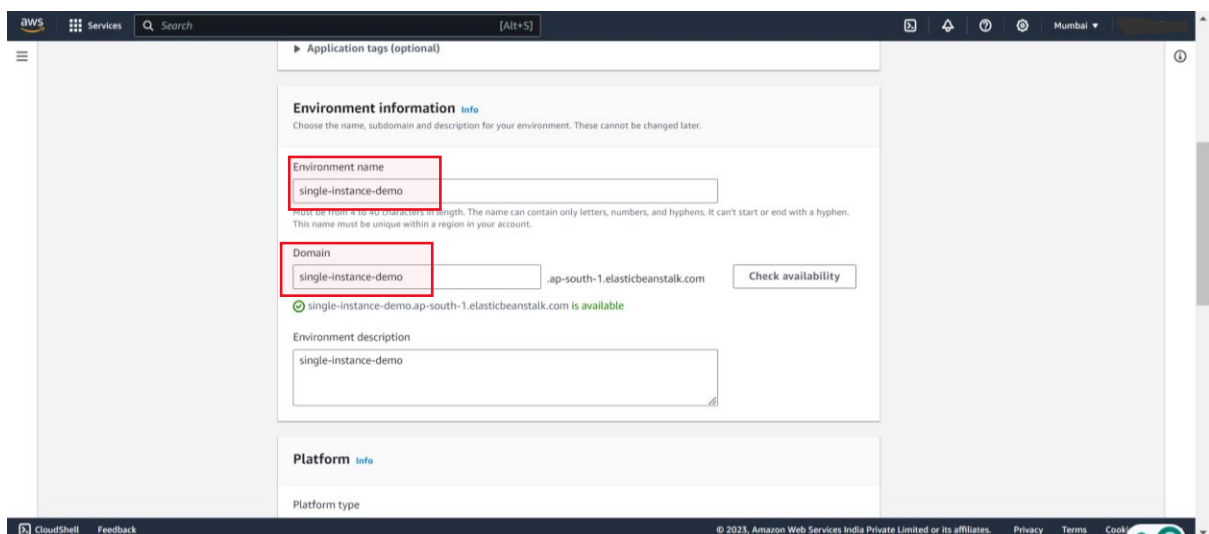
Give application name



Inside applications create a new env



Select webserver env



Give the env name and give the domain name and check for its availability

**Platform Info**

Platform type

- ☒ Managed platform  
Platforms published and maintained by Amazon Elastic Beanstalk. [Learn more](#)
- ☐ Custom platform  
Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform

PHP

Platform branch

PHP 8.2 running on 64bit Amazon Linux 2023

Platform version

4.0.1 (Recommended)

**Application code Info**

☒ Sample application  
Application versions that you have uploaded.

☐ Existing version

Select the platform type as PHP and keep the rest settings as default

**Application code Info**

☒ Sample application  
Application versions that you have uploaded.

☐ Existing version

☐ 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

Choose the application code as a sample application you can also upload the code of your choice and presets as a single instance.

**Service access**

IAM roles, assumed by Elastic Beanstalk as a service role, and EC2 instance profiles allow Elastic Beanstalk to create and manage your environment. Both the IAM role and instance profile must be attached to IAM managed policies that contain the required permissions. [Learn more](#)

Service role

- ☐ Create and use new service role
- ☒ Use an existing service role

Existing service roles

Choose an existing IAM role for Elastic Beanstalk to assume as a service role. The existing IAM role must have the required IAM managed policies.

aws-elasticbeanstalk-service-role

EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. [Learn more](#)

Choose a key pair

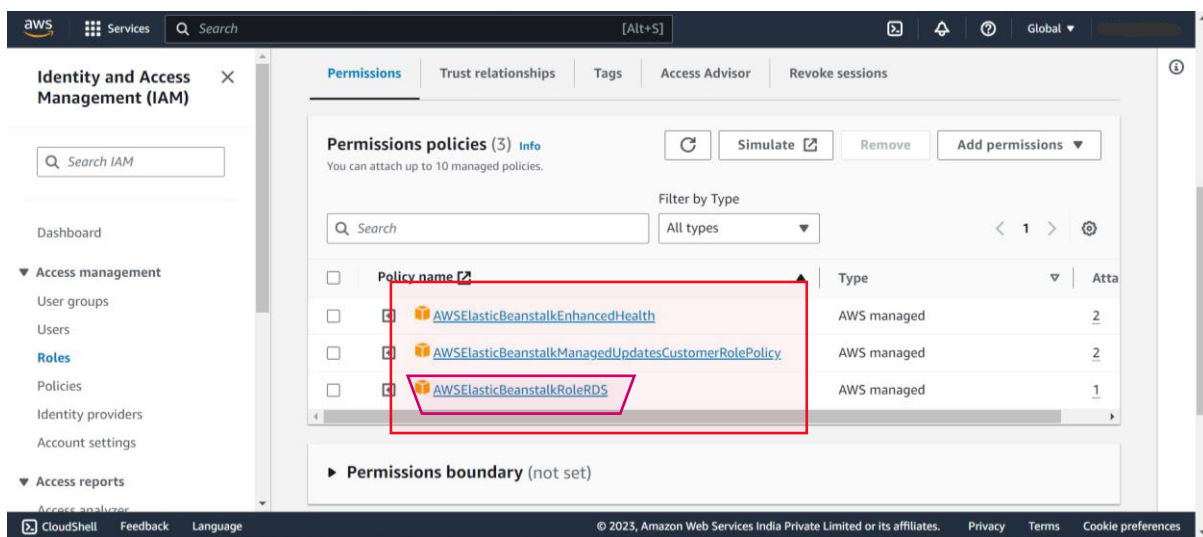
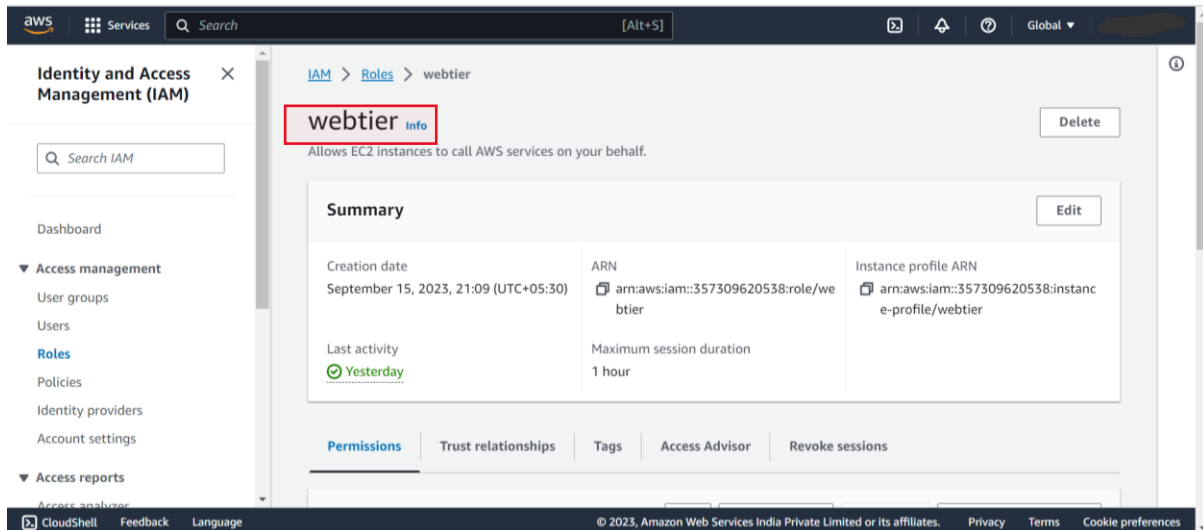
EC2 instance profile

Choose an IAM instance profile with managed policies that allow your EC2 instances to perform required operations.

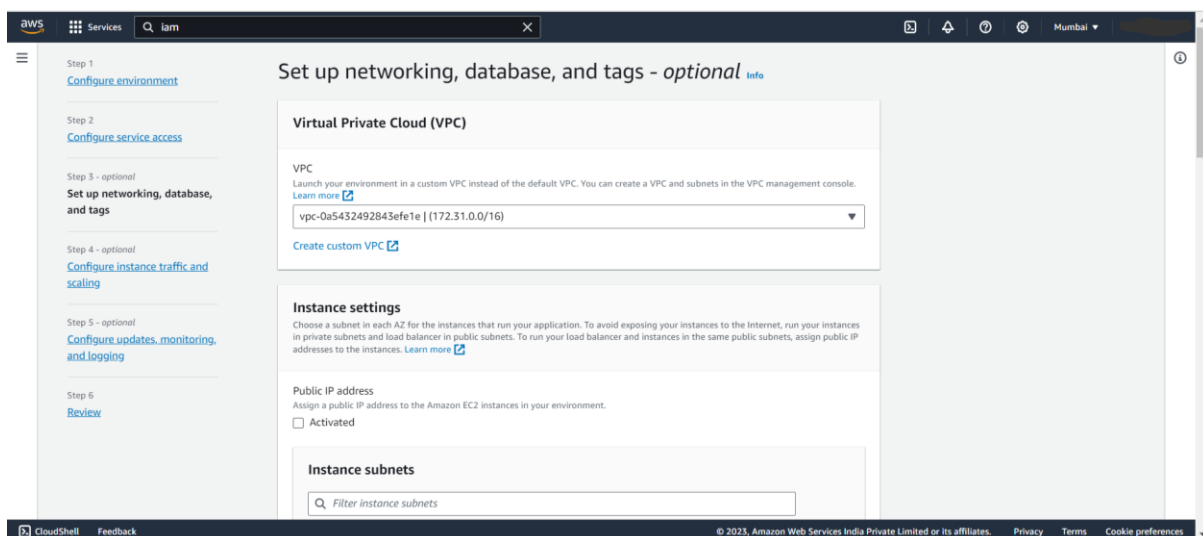
webtier

[View permission details](#)

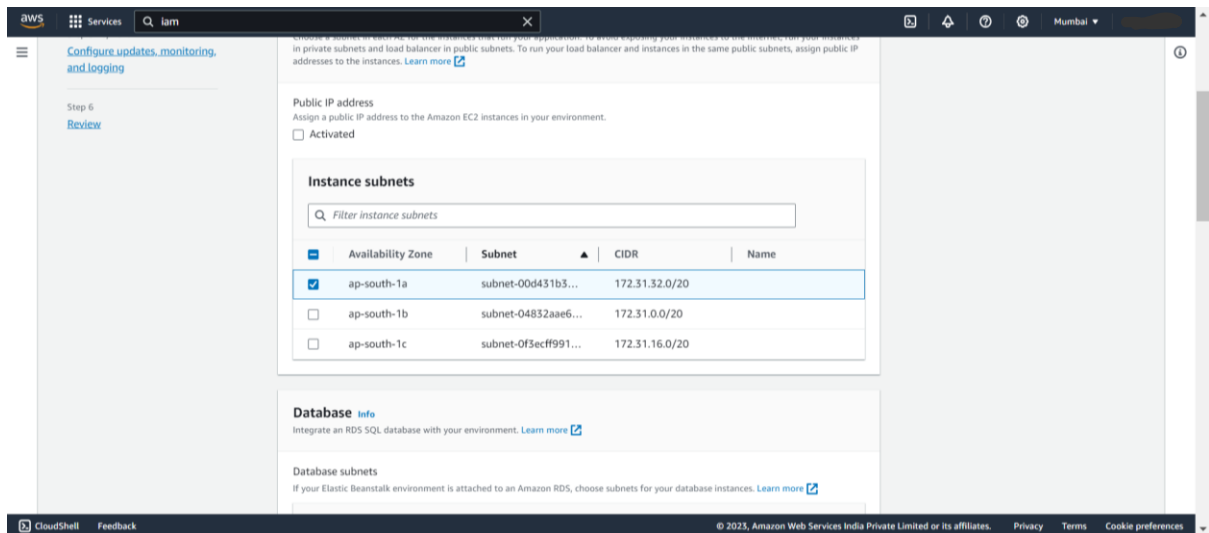
Select the existing role and give the ec2 instance profile an existing role created in IAM.



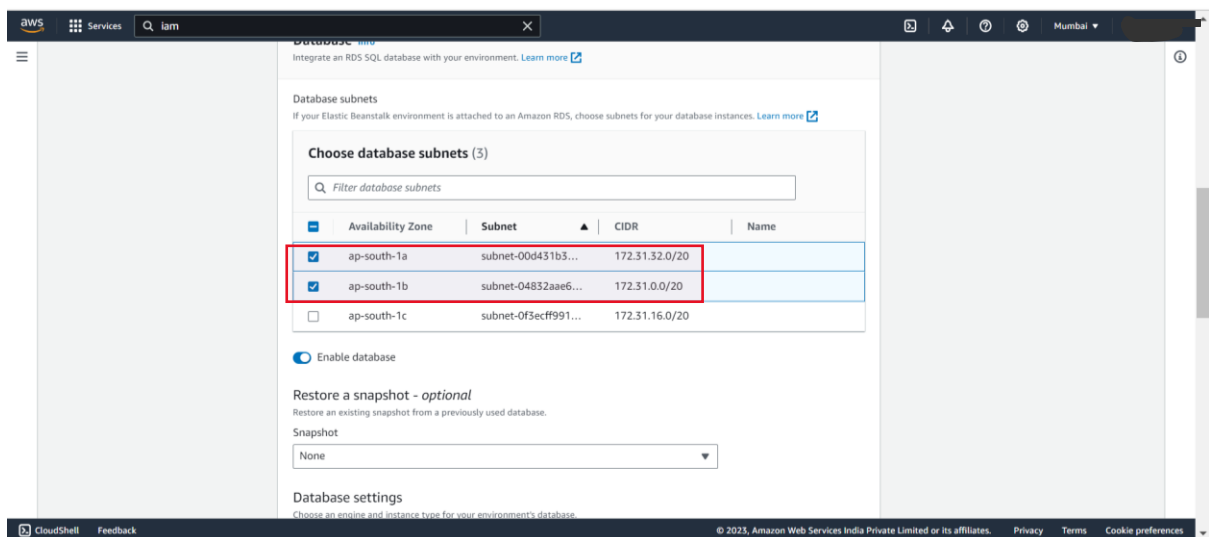
Make sure you have the following permission in your roles and **AWSManagedElasticBeanstalkRoleRDS** if you want to create Database.



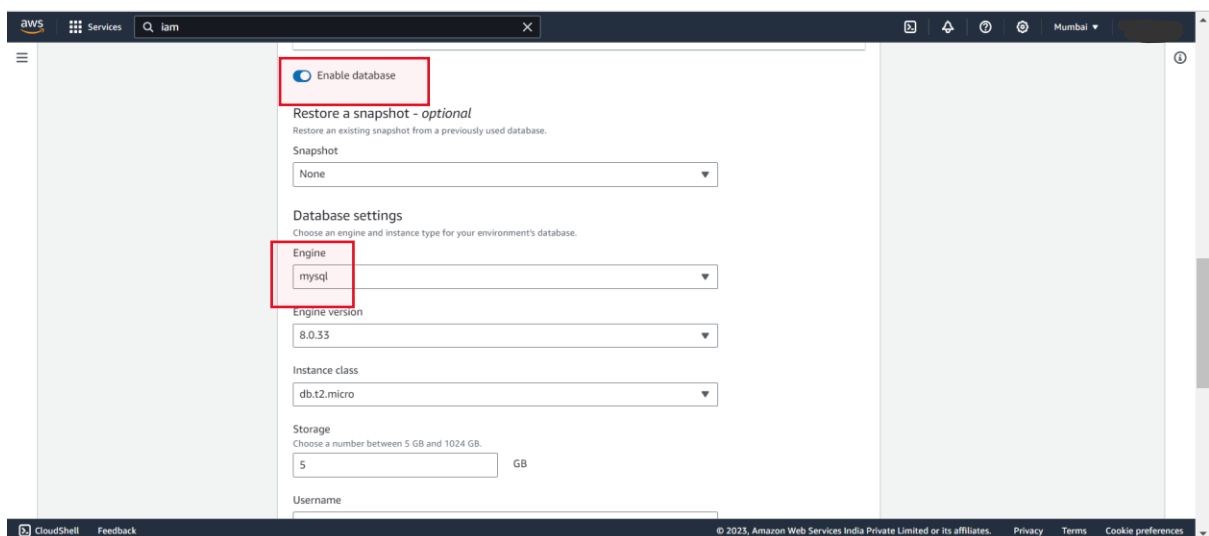
Select VPC as the default VPC



Select the AZ



For RDS database subnet group we need to select at least 2 subnets because in RDS we need to select at least 2 AZ.



## Enable database select engine as MYSQL

8.0.33

Instance class

db.t2.micro

Storage

Choose a number between 5 GB and 1024 GB.

5 GB

Username

admin

Password

\*\*\*\*\*

Availability

Low (one AZ)

Database deletion policy

This policy applies when you decouple a database or terminate the environment coupled to it.

☐ Create snapshot

Elastic Beanstalk saves a snapshot of the database and then deletes it. You can restore a database from a snapshot when you add a DB to an Elastic Beanstalk environment or when you create a standalone database. You might incur charges for storing database snapshots.

☐ Retain

The decoupled database will remain available and operational external to Elastic Beanstalk.

CloudShell Feedback

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Keep the rest settings as default and give the username and password select low AZ.

Availability

Low (one AZ)

Database deletion policy

This policy applies when you decouple a database or terminate the environment coupled to it.

☐ Create snapshot

Elastic Beanstalk saves a snapshot of the database and then deletes it. You can restore a database from a snapshot when you add a DB to an Elastic Beanstalk environment or when you create a standalone database. You might incur charges for storing database snapshots.

☐ Retain

The decoupled database will remain available and operational external to Elastic Beanstalk.

☒ Delete

Elastic Beanstalk terminates the database. The database will no longer be available.

Tags

Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. [Learn more](#)

No tags associated with the resource.

Add new tag

You can add 50 more tags.

Cancel Skip to review Previous Next

CloudShell Feedback

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Keep deletion policy to delete and click next.

Step 1

[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](#)

Configure instance traffic and scaling - optional [info](#)

Instances [info](#)

Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

Root volume type

General Purpose 3(SSD)

Size

The number of gigabytes of the root volume attached to each instance.

10 GB

IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

3000 IOPS

Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125 MiB/s

Amazon CloudWatch monitoring

CloudShell Feedback

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Select root volume as general purpose SSD 3 and rest as default

The image displays three sequential screenshots of the AWS IAM console, showing the configuration of an Amazon EMR environment.

**First Screenshot:** The 'Throughput' section is visible, showing a value of 125 MiB/s. Below it, the 'Amazon CloudWatch monitoring' section shows a monitoring interval of 5 minutes. The 'Instance metadata service (IMDS)' section shows that IMDSv1 is deactivated. The 'EC2 security groups' section shows a search bar and a table with one security group named 'default'.

**Second Screenshot:** The 'EC2 security groups' section is expanded, showing a table with one security group named 'default' with Group ID 'sg-0aee0230bac1564c4'. Below this, the 'Capacity' section is visible, showing the 'Auto scaling group' configuration.

**Third Screenshot:** The 'Auto scaling group' configuration is shown. The 'Environment type' is set to 'Single instance'. The 'Instances' section shows a minimum of 1 instance and a maximum of 1 instance. The 'Fleet composition' section shows 'On-Demand instance' selected. The 'Maximum spot price' section shows 'Default' selected. The 'On-Demand base' section shows a minimum number of 0 On-Demand instances. The 'On-Demand above base' section shows a percentage of 0.

0 %

**Capacity rebalancing**  
Specifies whether to enable the capacity rebalancing feature for Spot Instances in your Auto Scaling Group. This option is only relevant when `EnableSpot` is true in the `aws:ec2:instances` namespace, and there is at least one Spot Instance in your Auto Scaling group.

☐ Turn on capacity rebalancing

**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.

☐ **arm64 - new**  
This architecture uses AWS Graviton2 processors. You might have to recompile some third-party tools and libraries.

**Instance types**  
Add instance types for your fleet. Change the order that the instances are in to set the preferred launch order. This only affects On-Demand instances. We recommend you include at least two instance types. [Learn more](#)

Choose x86 instance types

t3.micro X t3.small X

**AMI ID**  
Elastic Beanstalk selects a default Amazon Machine Image (AMI) for your environment based on the Region, platform version, and processor architecture that you choose. [Learn more](#)

ami-01b825ad455a4c60c

**Availability Zones**  
Number of Availability Zones (AZs) to use.

Keep the settings as default

tools and libraries.

**Instance types**  
Add instance types for your fleet. Change the order that the instances are in to set the preferred launch order. This only affects On-Demand instances. We recommend you include at least two instance types. [Learn more](#)

Choose x86 instance types

t3.micro X t3.small X

**AMI ID**  
Elastic Beanstalk selects a default Amazon Machine Image (AMI) for your environment based on the Region, platform version, and processor architecture that you choose. [Learn more](#)

ami-01b825ad455a4c60c

**Availability Zones**  
Number of Availability Zones (AZs) to use.

Any

**Placement**  
Specify Availability Zones (AZs) to use.

Choose Availability Zones (AZs)

**Scaling cooldown**  
360 seconds

Cancel Skip to review Previous **Next**

Step 1  
[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

**Configure updates, monitoring, and logging - optional** [Info](#)

**Monitoring** [Info](#)

**Health reporting**  
Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The `EnvironmentHealth` custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#).

**System**

☐ Basic

☒ **Enhanced**

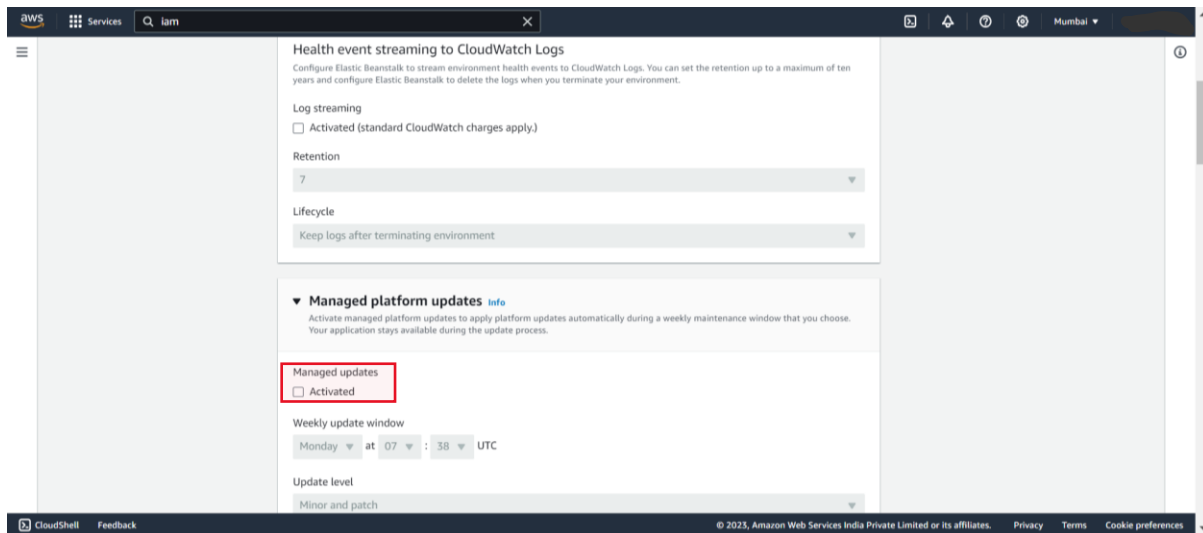
**CloudWatch Custom Metrics - Instance**  
Choose metrics

**CloudWatch Custom Metrics - Environment**  
Choose metrics

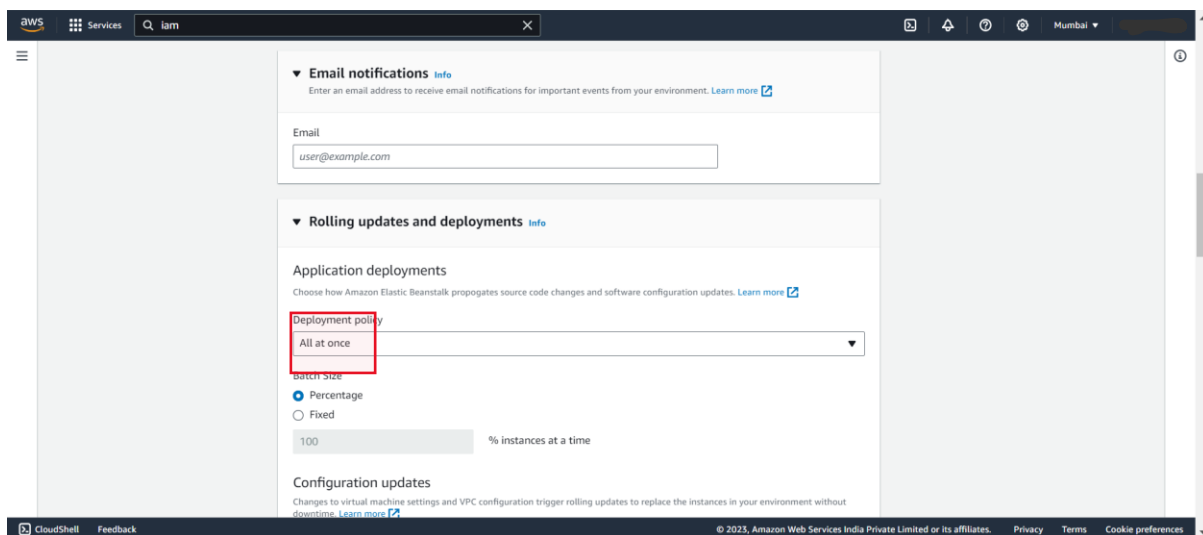
**Health event streaming to CloudWatch Logs**  
Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

**Log streaming**  
☐ Activated (standard CloudWatch charges apply.)

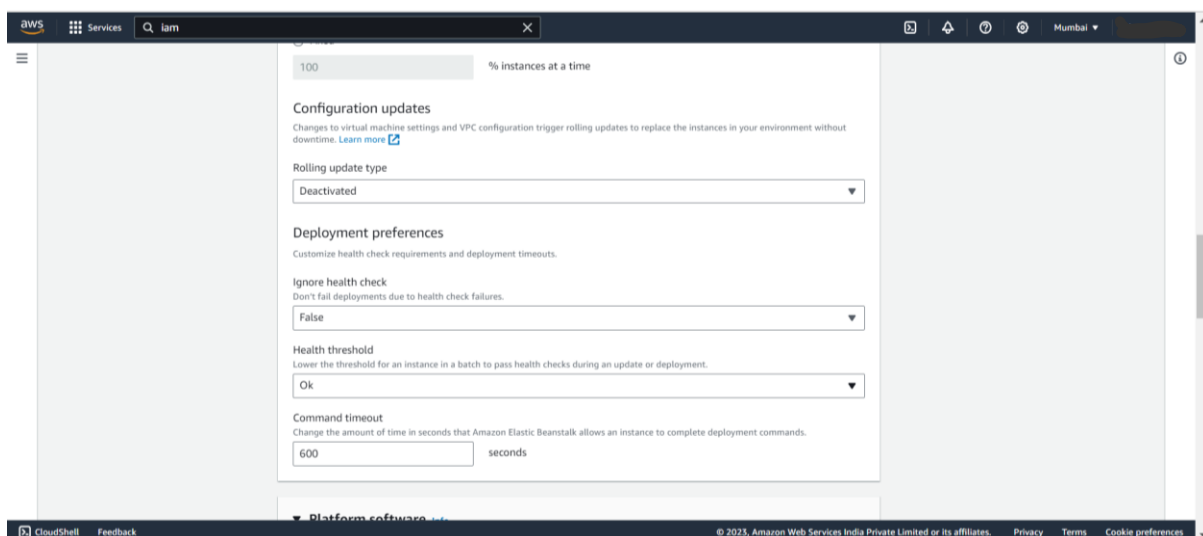




Untick the managed updates checkbox



As this is single instance, we have a deployment policy as all at once.



Platform software [info](#)

Configure the options available to your specific platform. These include the proxy server and OS environment properties. [Learn more](#)

### Container options

Proxy server  
Nginx

Document root  
The child directory of your project that acts as the public facing web root. If your root document is stored in your project directory, leave this set to /. If your root document is in a child directory (e.g., /public), set this value to match the child directory. Values should begin with a / character, and may NOT begin with a . (period).

Memory limit  
The amount of memory allocated to the PHP environment. This value is written to a .ini configuration file located in the/etc/php.d/ directory.  
256M

Zlib output compression  
Whether PHP should use compression for output. This value is written to a .ini configuration file located in the/etc/php.d/ directory.  
Off

Allow URL fopen  
Whether the PHP's file functions are allowed to retrieve data from remote locations, such as websites or FTP servers. This value is written to a .ini configuration file located in the/etc/php.d/ directory.  
On

The maximum time a script is allowed to run before the environment terminates it. This helps prevent poorly written scripts from tying up the server.  
60 seconds

Amazon X-Ray  
Amazon X-Ray is a service that collects data about the requests and responses that your application serves and receives. You can use the tools that X-Ray offers to view and filter the data that it provides to identify potential issues and optimization opportunities.

X-Ray daemon  
(service charges may apply.)  
☐ Activated

S3 log storage  
Configure the instances in your environment to upload rotated logs to Amazon S3. [Learn more](#)

Rotate logs  
(standard S3 charges apply.)  
☐ Activated

Instance log streaming to CloudWatch logs  
Configure the instances in your environment to stream logs to CloudWatch logs. You can set the retention to up to 10 years and configure Elastic Beanstalk to delete the logs when you terminate your environment. [Learn more](#)

Log streaming  
(standard CloudWatch charges apply.)  
☐ Activated

Retention

Step 1: [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

## Review [info](#)

### Step 1: Configure environment [Edit](#)

Environment information

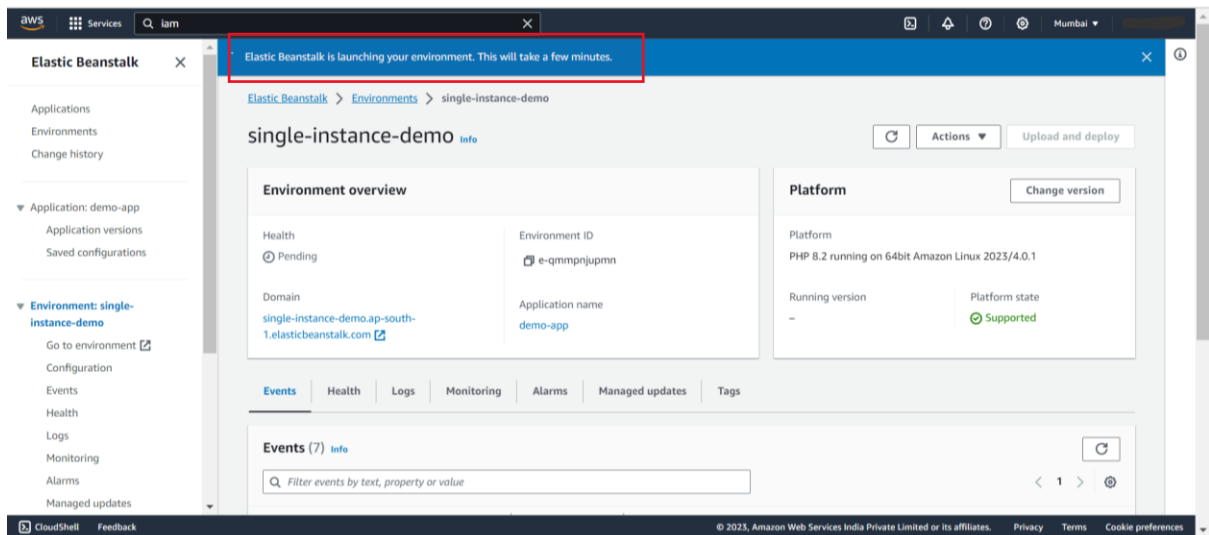
Environment tier Web server environment	Application name demo-app
Environment name single-instance-demo	Application code Sample application
Platform arn:aws:elasticbeanstalk:ap-south-1:platform/PHP 8.2 running on 64bit Amazon Linux 2023/4.0.1	

### Step 2: Configure service access [Edit](#)

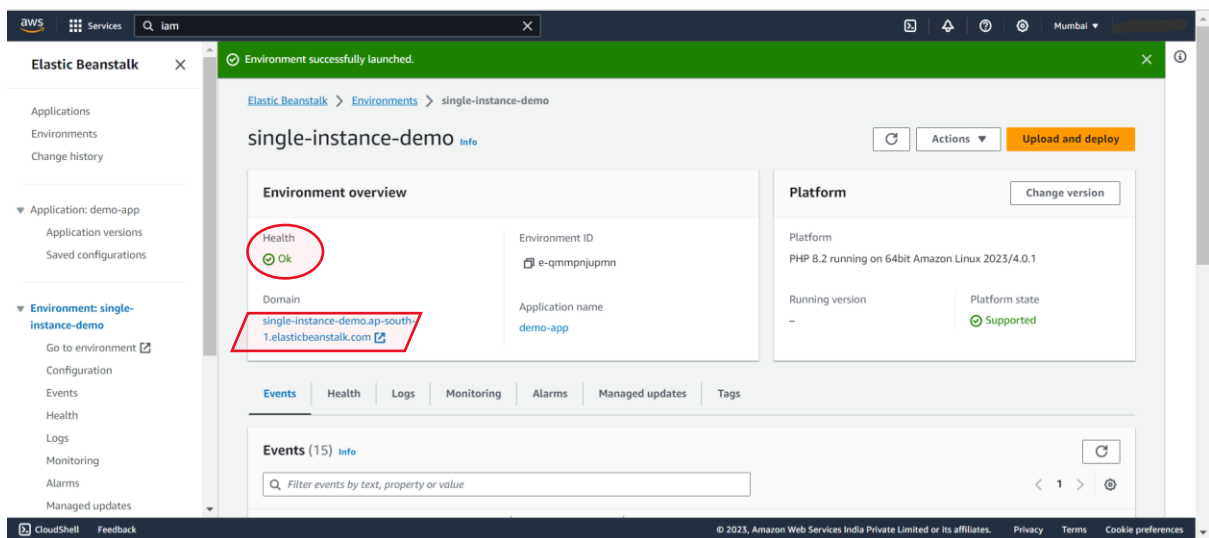
Service access [info](#)

Configure the service role and EC2 instance profile that Elastic Beanstalk uses to manage your environment. Choose an EC2 key pair to securely log in to your EC2 instances.

Keep remaining everything as it as and click on review and launch



Single instance env being launched



Env launched successfully as our health check is OK if we click on the domain name, we can go to our PHP Sample application

# Congratulations!

Your AWS Elastic Beanstalk *PHP* application is now running on your own dedicated environment in the AWS Cloud

You are running PHP version 8.2.7

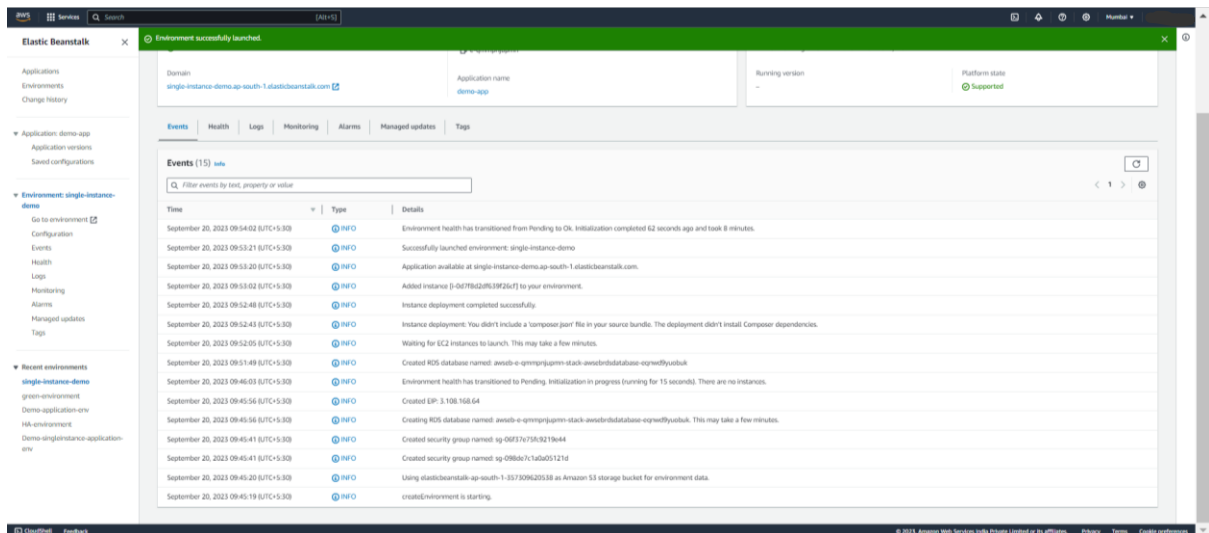
This environment is launched with Elastic Beanstalk PHP Platform

### What's Next?

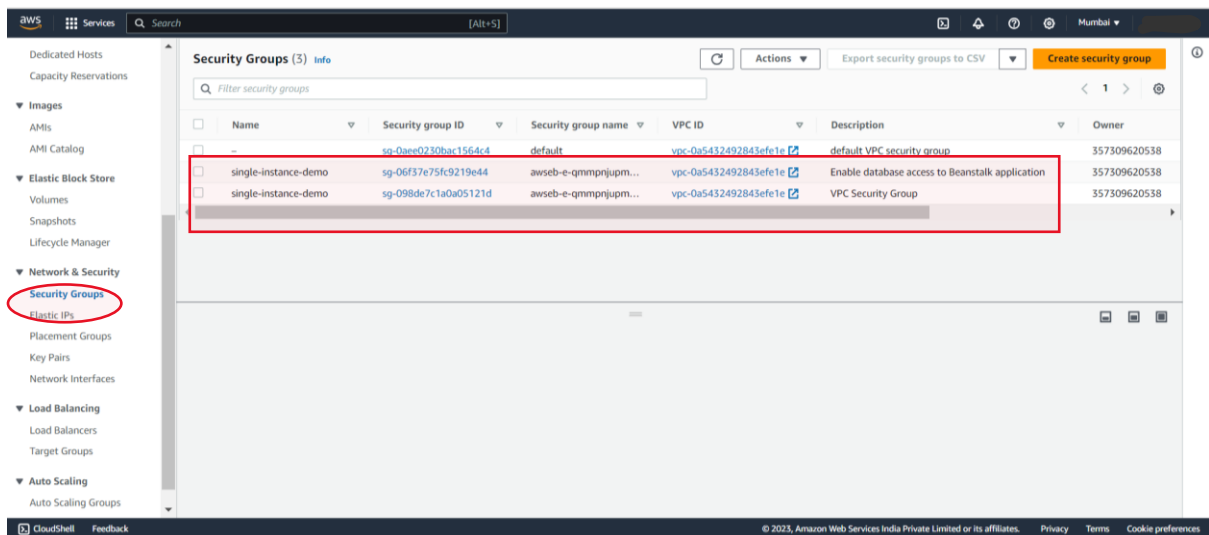
- [AWS Elastic Beanstalk overview](#)
- [Deploying AWS Elastic Beanstalk Applications in PHP Using Eb and Git](#)
- [Using Amazon RDS with PHP](#)
- [Customizing the Software on EC2 Instances](#)
- [Customizing Environment Resources](#)

### AWS SDK for PHP

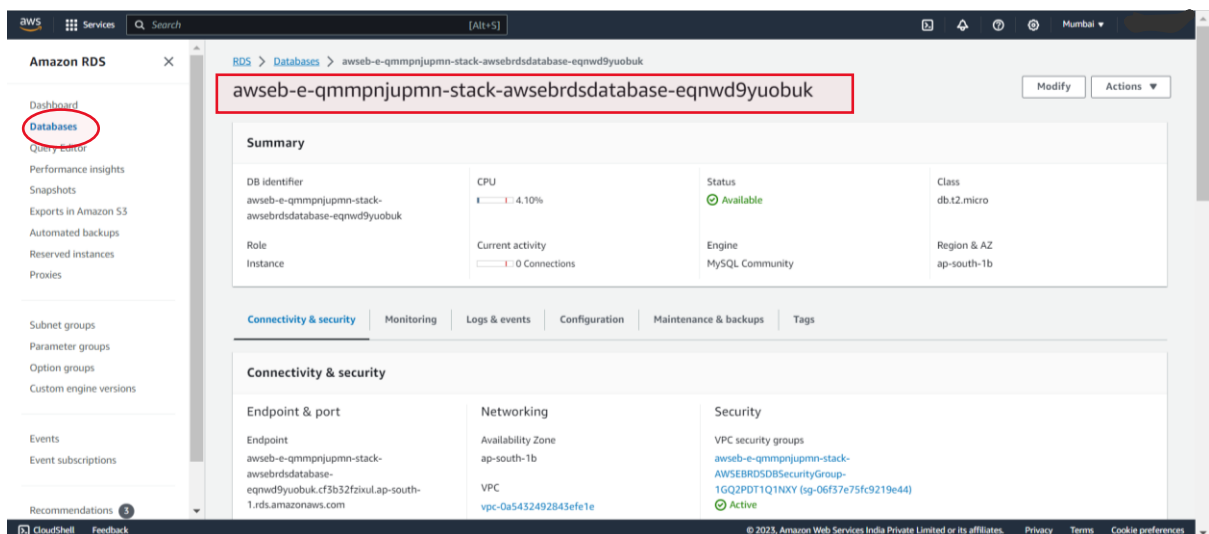
- [AWS SDK for PHP home](#)
- [PHP developer center](#)
- [AWS SDK for PHP on GitHub](#)



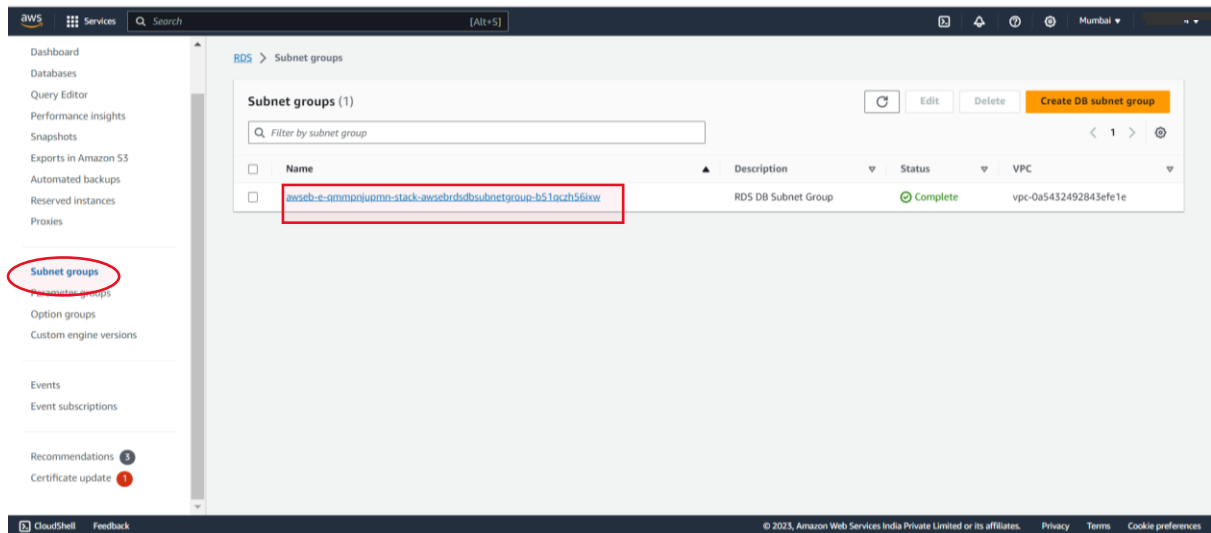
Following things has been automatically created by launching the single instance.



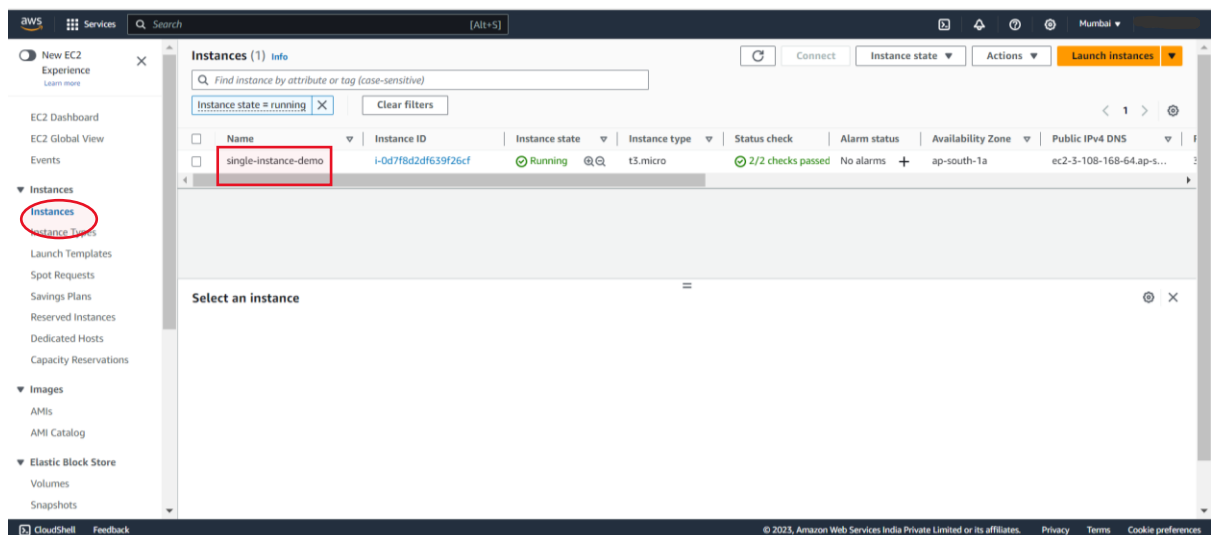
security groups have been created one SG is of database and another is of VPC.



RDS database created



Subnet group has been created for our database



Instance has also been launched by EBS.

Successfully we have launched EBS using single instance deployment.