Module 9: Elastic Beanstalk Assignment

Assignment Submitted By:-Hitesh Chauhan
Course Offered: -Advanced Cloud Computing and Devops
Assignment By: -Intellipaat
Trainer: -Puneet Gavri
Date Of Submission: -07/12/2024

Problem Statement:

You work for XYZ Corporation. Your corporation wants to launch a new web-based application and they do not want their servers to be running all the time. It should also be managed by AWS. Implement suitable solutions.

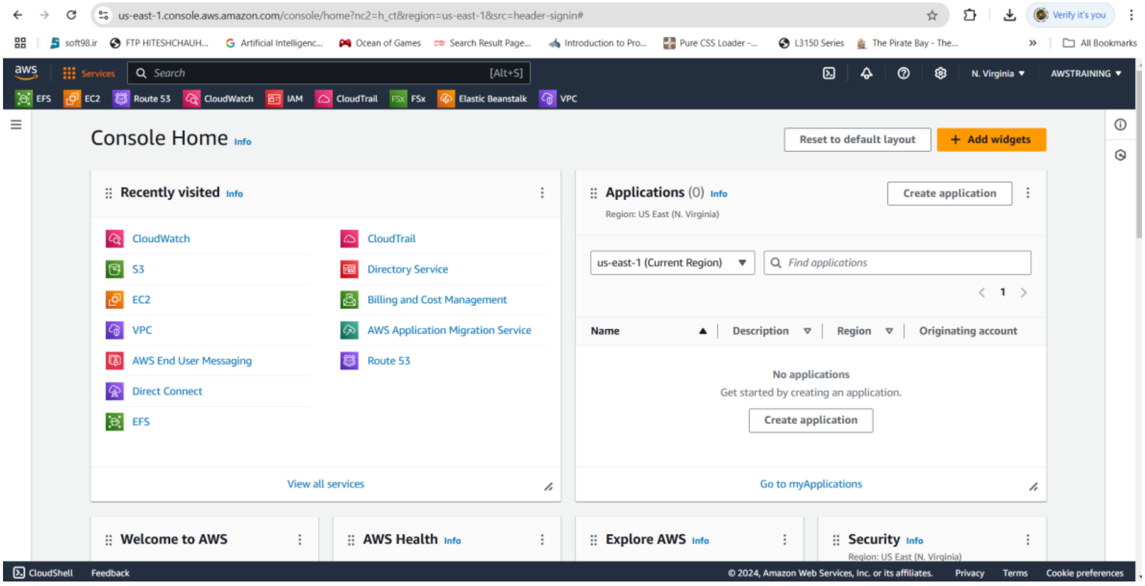
Tasks To Be Performed:

1. Create an Elastic Beanstalk environment with the runtime as PHP.

2. Upload a simple PHP file to the environment once created.

ANSWER:

Login to AWS Management Console and First Of Search IAM



us-east-1.console.aws.amazon.com/console/home?nc2=h_ct&region=us-east-1&src=header-signin#

Console Home Info

Recently visited Info

CloudWatch	CloudTrail
S3	Directory Service
EC2	Billing and Cost Management
VPC	AWS Application Migration Service
AWS End User Messaging	Route 53
Direct Connect	
EFS	

Applications (0) Info

Region: US East (N. Virginia)

us-east-1 (Current Region)

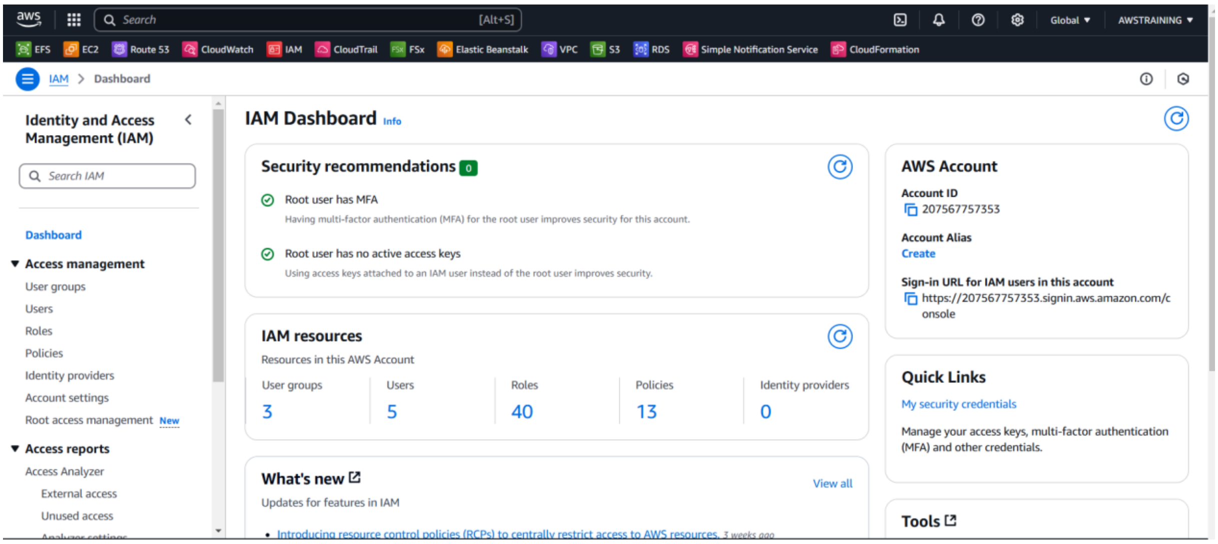
Name	Description	Region	Originating account

No applications

Get started by creating an application.

© 2024, Amazon Web Services, Inc. or its affiliates.

After Search IAM,You will see like this page.



IAM Dashboard Info

Security recommendations 0

Root user has MFA

Having multi-factor authentication (MFA) for the root user improves security for this account.

Root user has no active access keys

Using access keys attached to an IAM user instead of the root user improves security.

IAM resources

Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
3	5	40	13	0

What's new

Updates for features in IAM

- Introducing resource control policies (RCPs) to centrally restrict access to AWS resources. 3 weeks ago

AWS Account

Account ID

207567757353

Account Alias

Create

Sign-in URL for IAM users in this account

https://207567757353.signin.aws.amazon.com/console

Quick Links

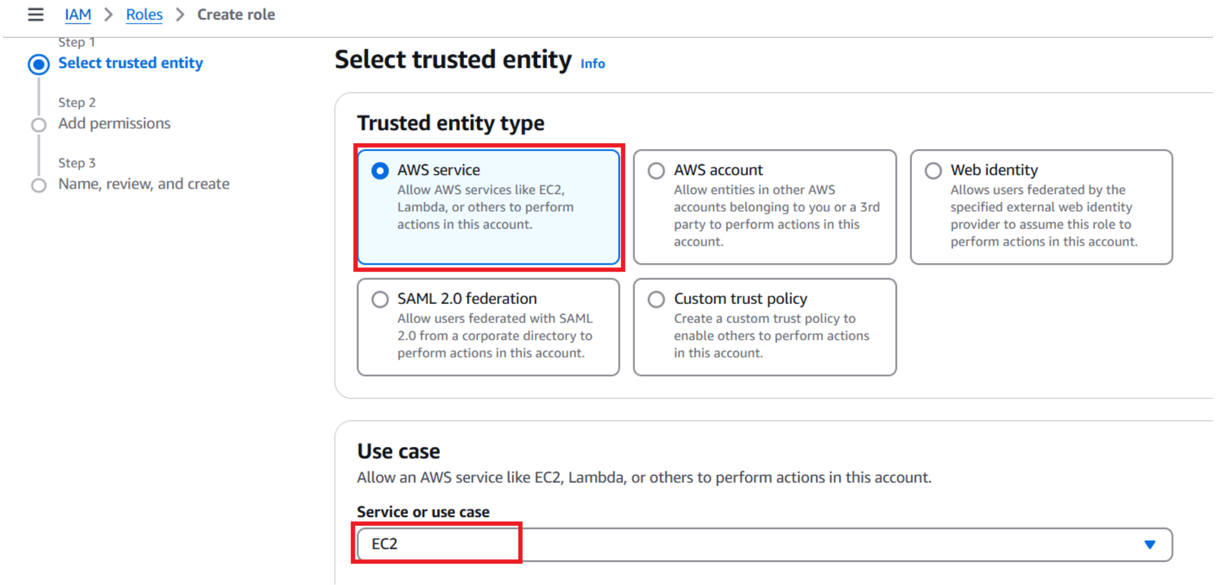
My security credentials

Manage your access keys, multi-factor authentication (MFA) and other credentials.

First Of We need to assigned the to ec2 as a elastic beans talk role.

Go To IAM Then Click Role And you will see like this.

Then go to aws service then choose ec2



IAM > Roles > Create role

Step 1
● Select trusted entity

Step 2
○ Add permissions

Step 3
○ Name, review, and create

Select trusted entity Info

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

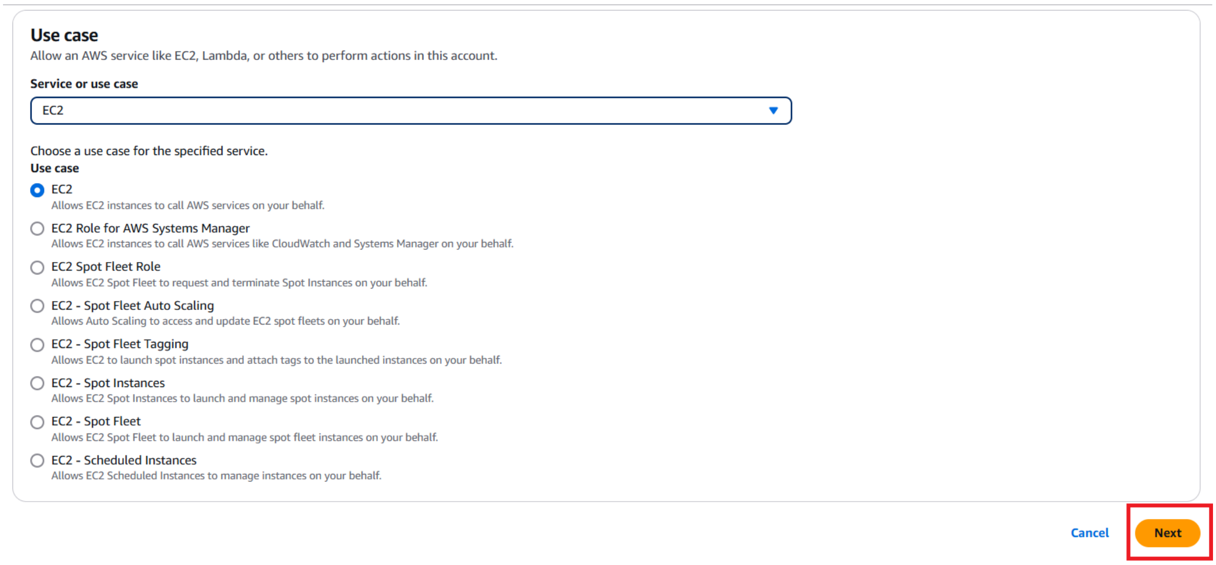
Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

EC2

Then Click Next



Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

EC2

Choose a use case for the specified service.

Use case

● EC2
Allows EC2 instances to call AWS services on your behalf.

○ EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.

○ EC2 Spot Fleet Role
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.

○ EC2 - Spot Fleet Auto Scaling
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.

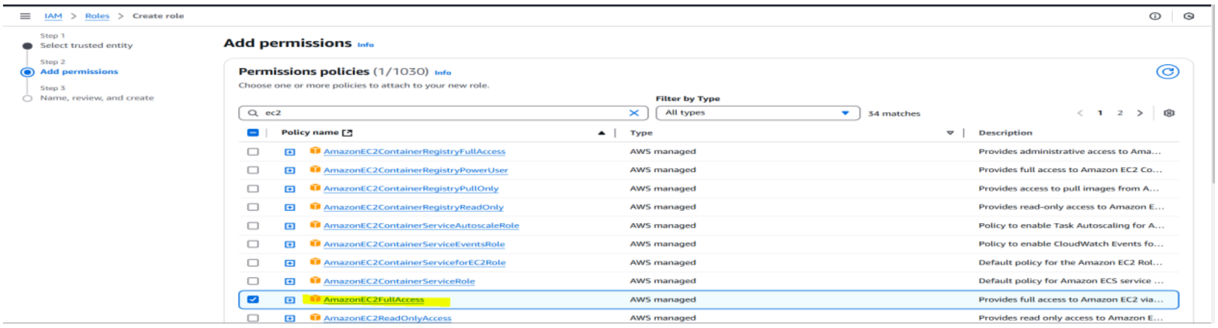
○ EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.

○ EC2 - Spot Instances
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.

○ EC2 - Spot Fleet
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.

○ EC2 - Scheduled Instances
Allows EC2 Scheduled Instances to manage instances on your behalf.

After Click next you choose to AmazonEC2FullAccess



IAM > Roles > Create role

Step 1
● Select trusted entity

Step 2
● Add permissions

Step 3
○ Name, review, and create

Add permissions Info

Permissions policies (1/1030) Info

Choose one or more policies to attach to your new role.

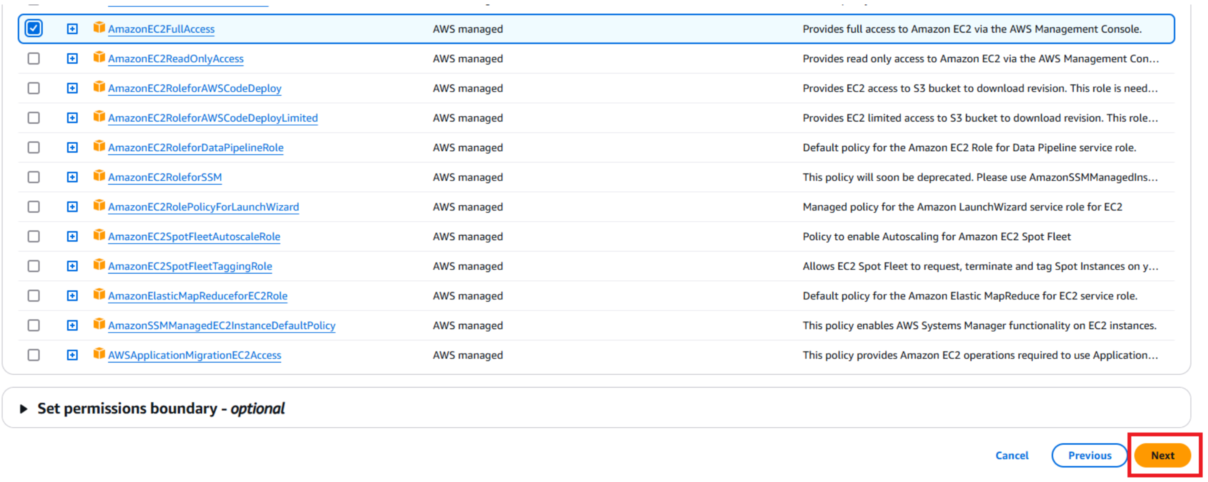
Search: ec2

Filter by Type: All types

34 matches

	Policy name	Type	Description
□	AmazonEC2ContainerRegistryFullAccess	AWS managed	Provides administrative access to Ama...
□	AmazonEC2ContainerRegistryPowerUser	AWS managed	Provides full access to Amazon EC2 Co...
□	AmazonEC2ContainerRegistryPullOnly	AWS managed	Provides access to pull images from A...
□	AmazonEC2ContainerRegistryReadOnly	AWS managed	Provides read-only access to Amazon E...
□	AmazonEC2ContainerServiceAutoscaleRole	AWS managed	Policy to enable Task Autoscaling for A...
□	AmazonEC2ContainerServiceEventsRole	AWS managed	Policy to enable CloudWatch Events fo...
□	AmazonEC2ContainerServiceforEC2Role	AWS managed	Default policy for the Amazon EC2 Rol...
□	AmazonEC2ContainerServiceRole	AWS managed	Default policy for Amazon ECS service ...
☑	AmazonEC2FullAccess	AWS managed	Provides full access to Amazon EC2 via...
□	AmazonEC2ReadOnlyAccess	AWS managed	Provides read only access to Amazon E...

Then Click Next

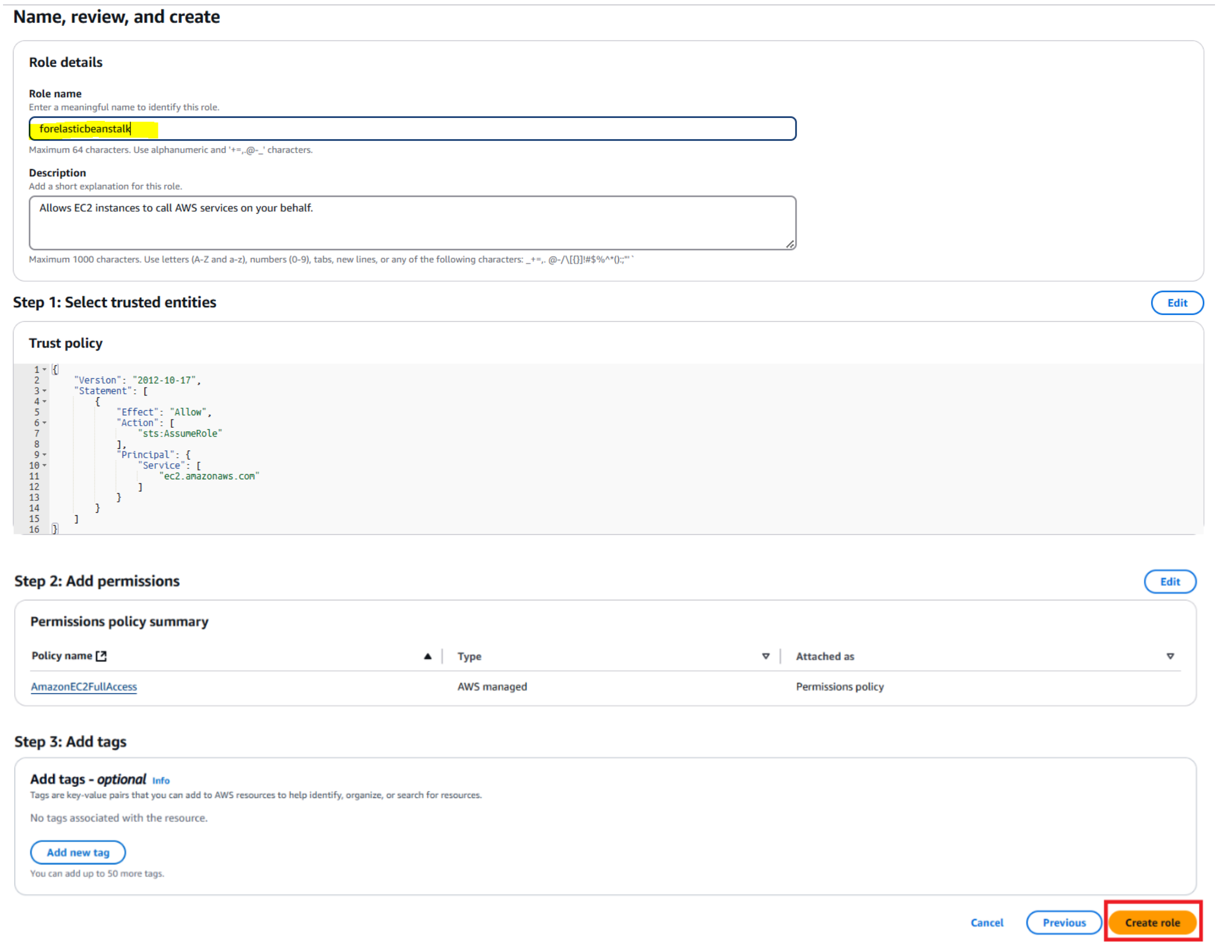


		Policy name	Type	Description
☑	⊞	AmazonEC2FullAccess	AWS managed	Provides full access to Amazon EC2 via the AWS Management Console.
□	⊞	AmazonEC2ReadOnlyAccess	AWS managed	Provides read only access to Amazon EC2 via the AWS Management Con…
□	⊞	AmazonEC2RoleforAWSCodeDeploy	AWS managed	Provides EC2 access to S3 bucket to download revision. This role is need…
□	⊞	AmazonEC2RoleforAWSCodeDeployLimited	AWS managed	Provides EC2 limited access to S3 bucket to download revision. This role…
□	⊞	AmazonEC2RoleforDataPipelineRole	AWS managed	Default policy for the Amazon EC2 Role for Data Pipeline service role.
□	⊞	AmazonEC2RoleforSSM	AWS managed	This policy will soon be deprecated. Please use AmazonSSMManagedIns…
□	⊞	AmazonEC2RolePolicyForLaunchWizard	AWS managed	Managed policy for the Amazon LaunchWizard service role for EC2
□	⊞	AmazonEC2SpotFleetAutoscaleRole	AWS managed	Policy to enable Autoscaling for Amazon EC2 Spot Fleet
□	⊞	AmazonEC2SpotFleetTaggingRole	AWS managed	Allows EC2 Spot Fleet to request, terminate and tag Spot Instances on y…
□	⊞	AmazonElasticMapReduceforEC2Role	AWS managed	Default policy for the Amazon Elastic MapReduce for EC2 service role.
□	⊞	AmazonSSMManagedEC2InstanceDefaultPolicy	AWS managed	This policy enables AWS Systems Manager functionality on EC2 instances.
□	⊞	AWSApplicationMigrationEC2Access	AWS managed	This policy provides Amazon EC2 operations required to use Application…

▸ Set permissions boundary - *optional*

Then Assign the name of Role Name

In My case the role name is **forelasticbeanstalk**



Name, review, and create

Role details

Role name
Enter a meaningful name to identify this role.
forelasticbeanstalk
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/\[{}]!#$%^*();"' `

Step 1: Select trusted entities

Trust policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sts:AssumeRole"
            ],
            "Principal": {
                "Service": [
                    "ec2.amazonaws.com"
                ]
            }
        }
    ]
}
```

Step 2: Add permissions

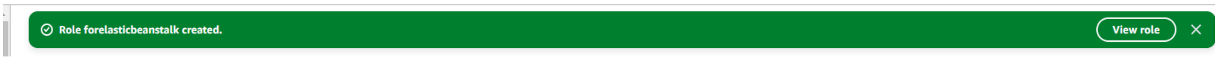
Permissions policy summary

Policy name	Type	Attached as
AmazonEC2FullAccess	AWS managed	Permissions policy

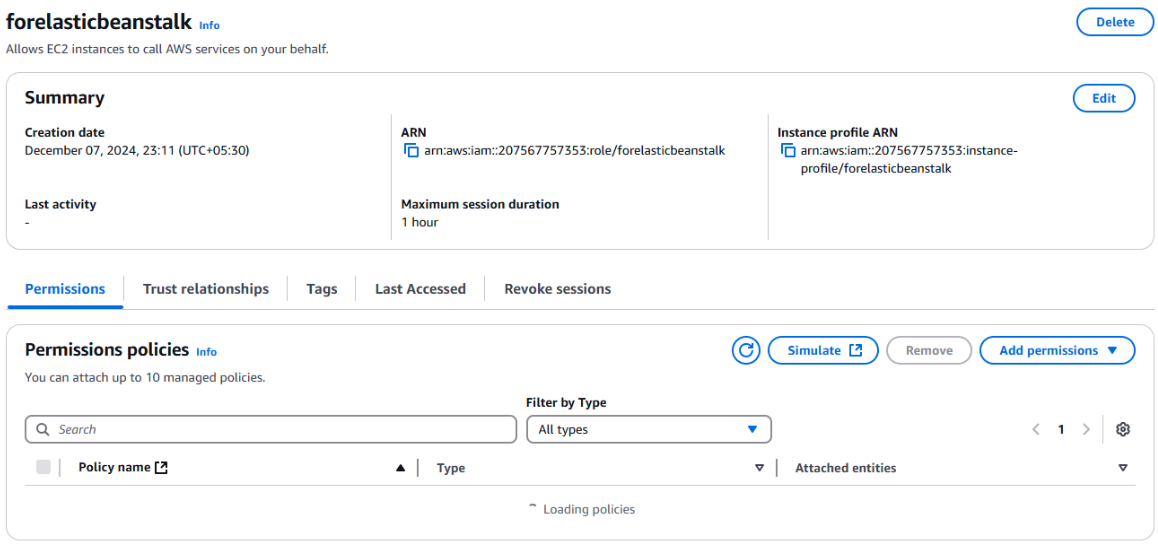
Step 3: Add tags

Add tags - *optional* Info
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
No tags associated with the resource.
You can add up to 50 more tags.

Then Review the role and click next Your role has been created.



Role forelasticbeanstalk created.



forelasticbeanstalk Info

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date
December 07, 2024, 23:11 (UTC+05:30)

ARN
arn:aws:iam::207567757353:role/forelasticbeanstalk

Instance profile ARN
arn:aws:iam::207567757353:instance-profile/forelasticbeanstalk

Last activity
-

Maximum session duration
1 hour

Permissions policies Info

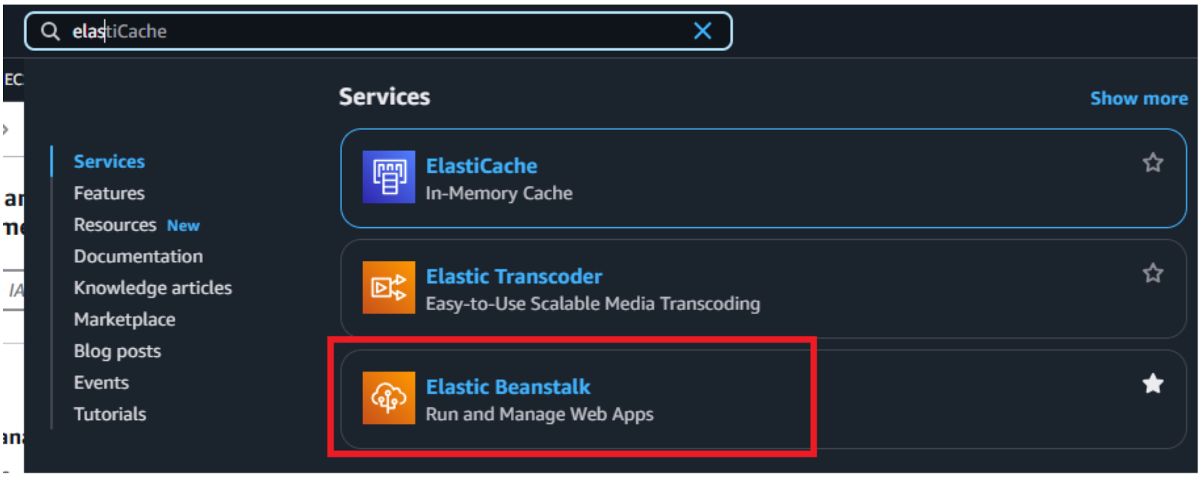
You can attach up to 10 managed policies.

Filter by Type: All types

☐	Policy name	Type	Attached entities

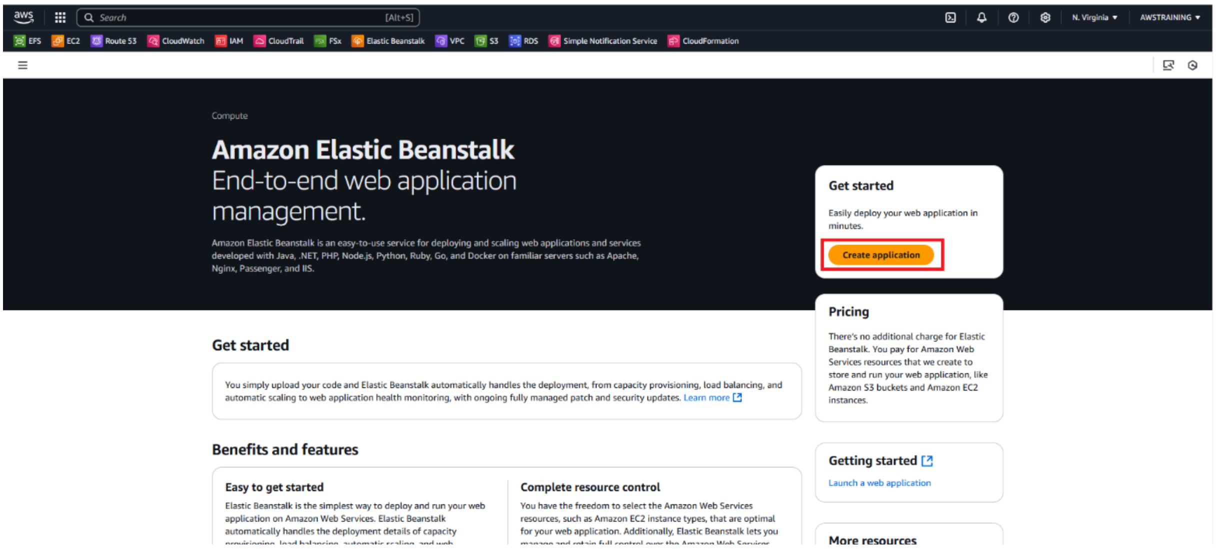
Loading policies

Now You need to search here elastic beanstalk Service.click on it.



elastiCache

After Click you will see like this page.then click Create Application.



Compute

Amazon Elastic Beanstalk
End-to-end web application management.

Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

Get started

Easily deploy your web application in minutes.

Pricing

There's no additional charge for Elastic Beanstalk. You pay for Amazon Web Services resources that we create to store and run your web application, like Amazon S3 buckets and Amazon EC2 instances.

Get started

You simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, and automatic scaling to web application health monitoring, with ongoing fully managed patch and security updates. Learn more

Benefits and features

Easy to get started

Elastic Beanstalk is the simplest way to deploy and run your web application on Amazon Web Services. Elastic Beanstalk automatically handles the deployment details of capacity provisioning, load balancing, automatic scaling, and web

Complete resource control

You have the freedom to select the Amazon Web Services resources, such as Amazon EC2 instance types, that are optimal for your web application. Additionally, Elastic Beanstalk lets you manage and retain full control over the Amazon Web Services

Getting started

Launch a web application

More resources

After Click the Create application.

You can see Configure Environement.

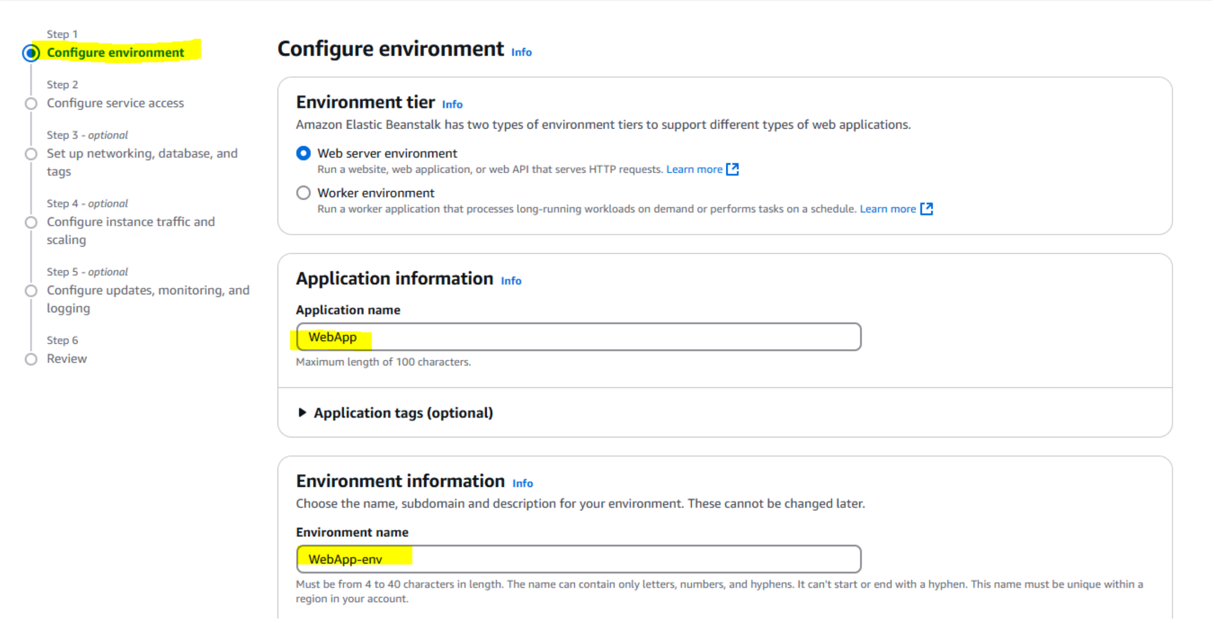
In This Scenario there is two Options.

1.Web Server Environment

2.Worker Environment.

So I choose Web server Environment.

Then Assign the application name.



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 environment Info

Environment tier Info

Amazon Elastic Beanstalk has two types of environment tiers to support different types of web applications.

● Web server environment
Run a website, web application, or web API that serves HTTP requests. Learn more

○ Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more

Application information Info

Application name

WebApp

Maximum length of 100 characters.

▸ **Application tags (optional)**

Environment information Info

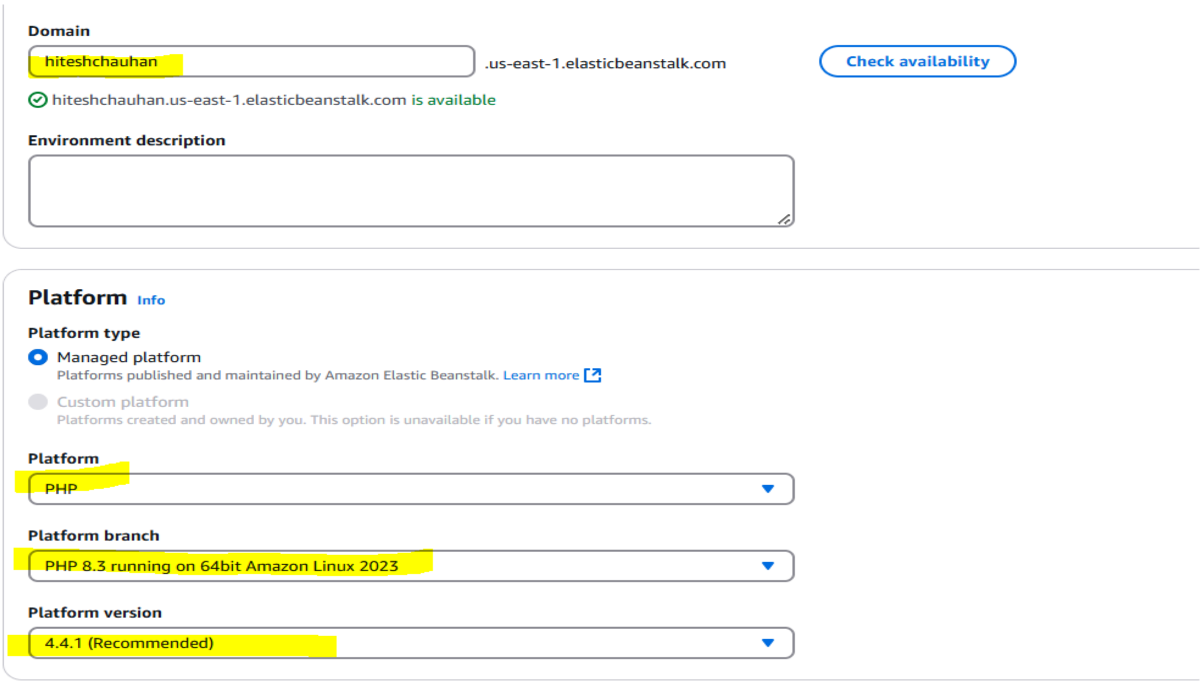
Choose the name, subdomain and description for your environment. These cannot be changed later.

Environment name

WebApp-env

Must be from 4 to 40 characters in length. The name can contain only letters, numbers, and hyphens. It can't start or end with a hyphen. This name must be unique within a region in your account.

Then You need Assign the Domain.in my case I assigned hiteshchauhan



Domain

hiteshchauhan .us-east-1.elasticbeanstalk.com Check availability

✓ hiteshchauhan.us-east-1.elasticbeanstalk.com is available

Environment description

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 8.3 running on 64bit Amazon Linux 2023

Platform version

4.4.1 (Recommended)

Platform Type

By default Managed platform

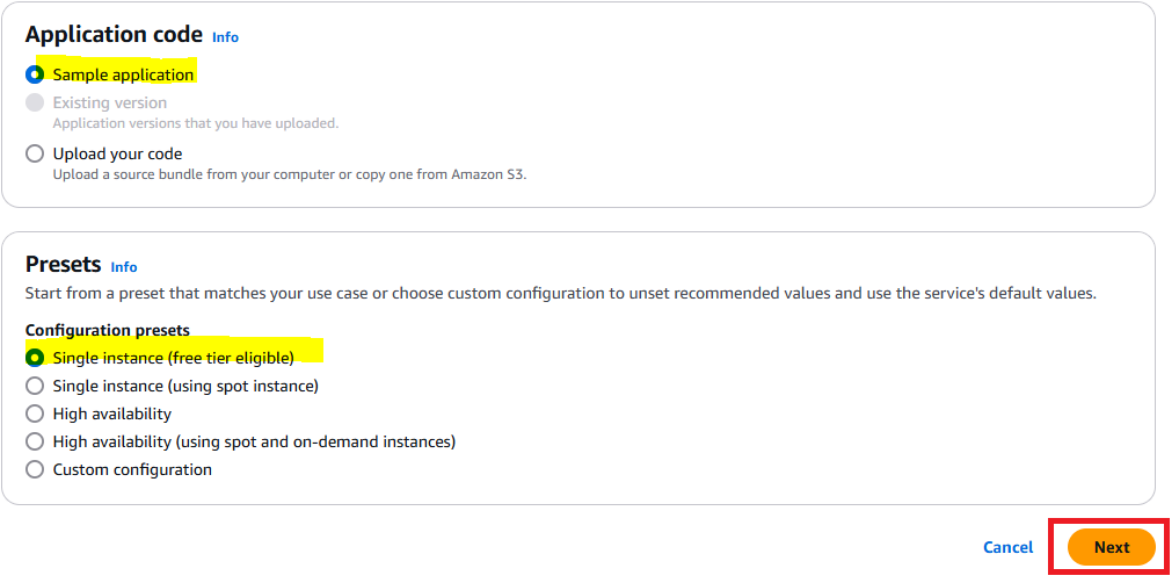
Then you will see **Platform**

So I choosed PHP(As per the task performed)

Then **Platform branch** will be PHP 8.3 running on 64bit Amazon Linux 2023.

Platform version will be 4.4.1.

In Below Screenshot Application Code Will be Sample Application and presets Section, Configuration presents will be single instance(free tier eligible) then click next.



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)

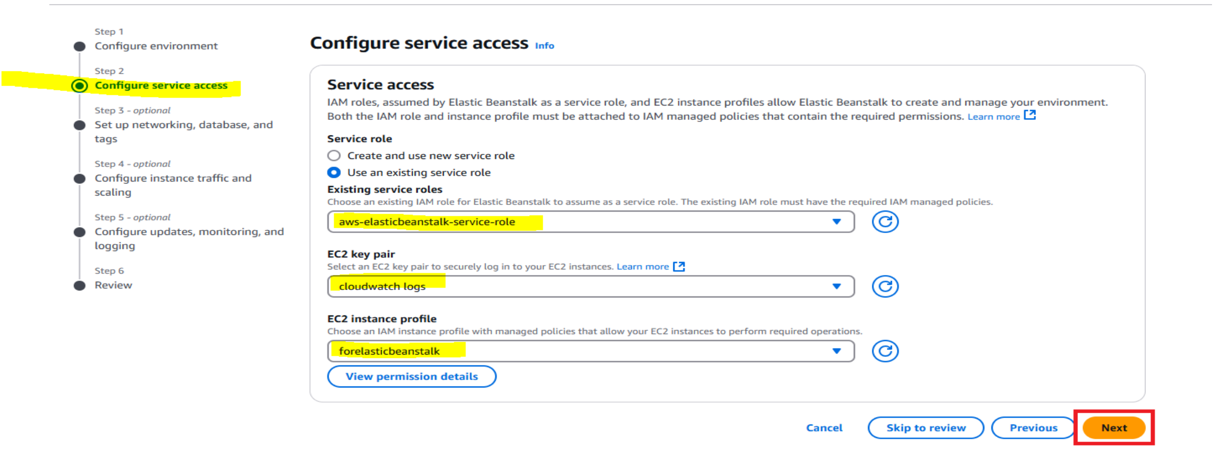
○ Single instance (using spot instance)

○ High availability

○ High availability (using spot and on-demand instances)

○ Custom configuration

In Configure service access Will be use an existing role **aws-elasticbeanstalk-service-role** and ec2 key pair will be **cloudwatch logs** and ec2 instance profile will be **forelasticbeanstalk** then 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 service access Info

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

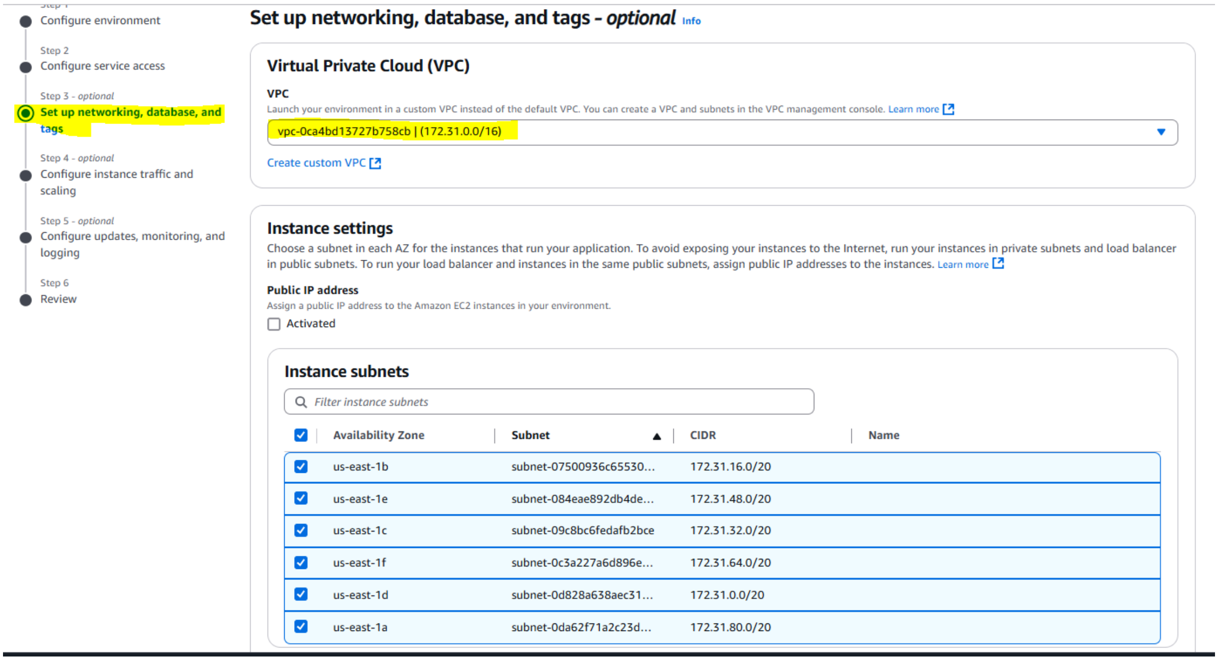
cloudwatch logs

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

forelasticbeanstalk

Now You need to setup networking,database and tags.

In VPC,I choosed by default vpc and instance settings will be choosed all az's.



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

Set up networking, database, and tags - *optional* Info

Virtual Private Cloud (VPC)

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-0ca4bd13727b758cb | (172.31.0.0/16)

Create custom VPC

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

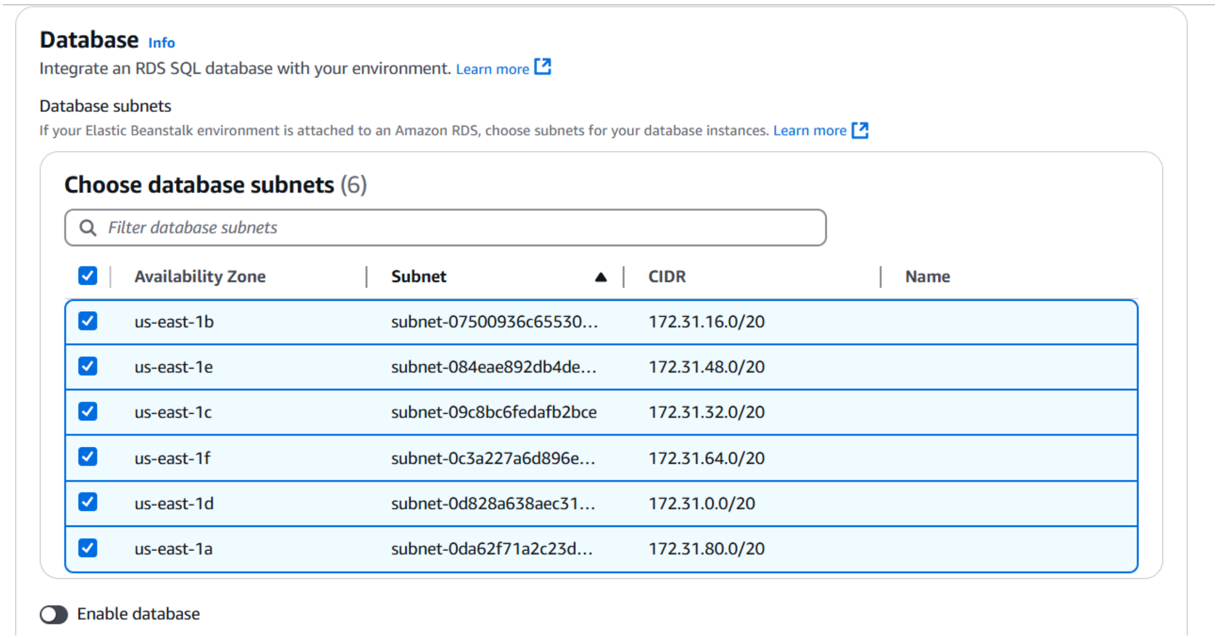
Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
□ Activated

Instance subnets

Filter instance subnets

☑	Availability Zone	Subnet ▲	CIDR	Name
☑	us-east-1b	subnet-07500936c65530...	172.31.16.0/20	
☑	us-east-1e	subnet-084eae892db4de...	172.31.48.0/20	
☑	us-east-1c	subnet-09c8bc6fedafb2bce	172.31.32.0/20	
☑	us-east-1f	subnet-0c3a227a6d896e...	172.31.64.0/20	
☑	us-east-1d	subnet-0d828a638aec31...	172.31.0.0/20	
☑	us-east-1a	subnet-0da62f71a2c23d...	172.31.80.0/20	

In Database Setup choosed all database subnets.



Database Info

Integrate an RDS SQL database with your environment. Learn more

Database subnets
If your Elastic Beanstalk environment is attached to an Amazon RDS, choose subnets for your database instances. Learn more

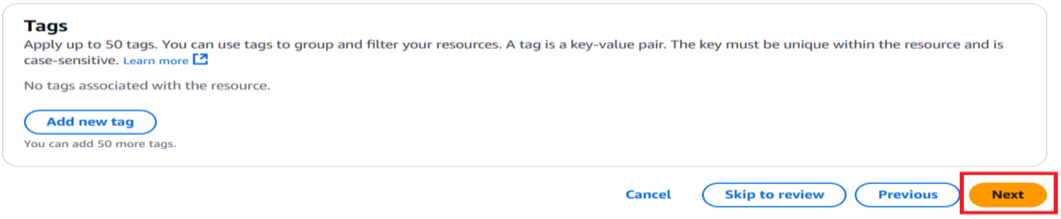
Choose database subnets (6)

Filter database subnets

☑	Availability Zone	Subnet ▲	CIDR	Name
☑	us-east-1b	subnet-07500936c65530...	172.31.16.0/20	
☑	us-east-1e	subnet-084eae892db4de...	172.31.48.0/20	
☑	us-east-1c	subnet-09c8bc6fedafb2bce	172.31.32.0/20	
☑	us-east-1f	subnet-0c3a227a6d896e...	172.31.64.0/20	
☑	us-east-1d	subnet-0d828a638aec31...	172.31.0.0/20	
☑	us-east-1a	subnet-0da62f71a2c23d...	172.31.80.0/20	

Enable database

Then Click Next



Tags

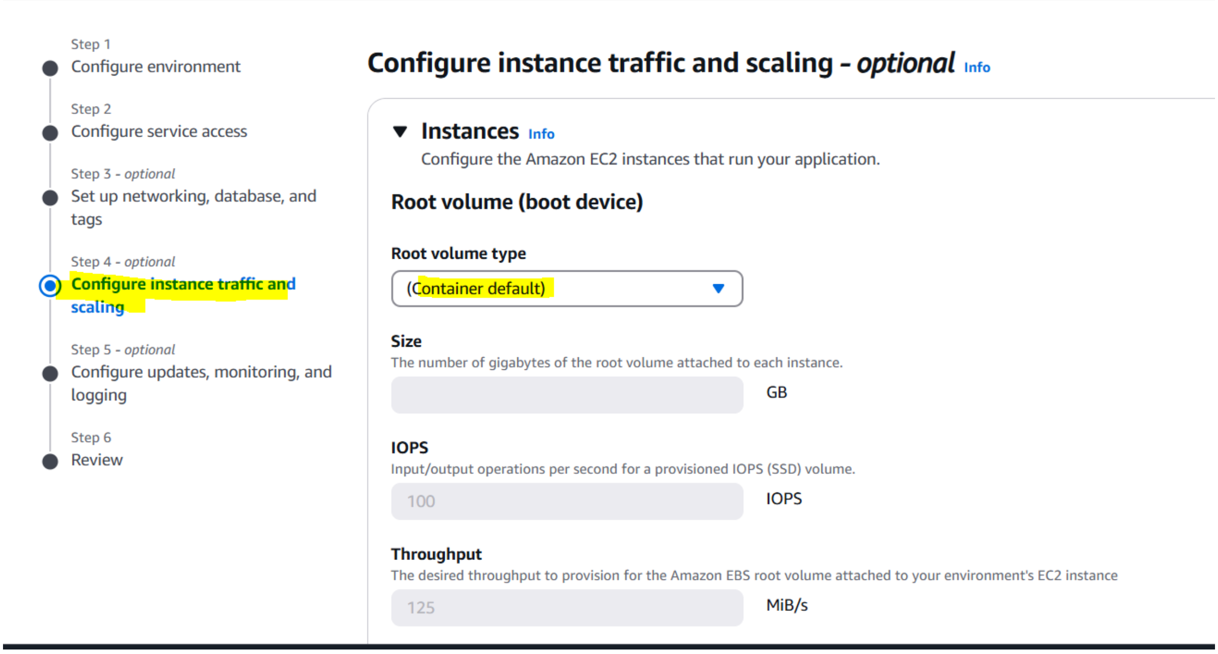
Apply up to 50 tags. You can use tags to group and filter your resources. A tag is a key-value pair. The key must be unique within the resource and is case-sensitive. Learn more

No tags associated with the resource.

Add new tag
You can add 50 more tags.

After Click Next We need to Configure instance traffic and Scaling require.

Root Volume will be use default setting.



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)

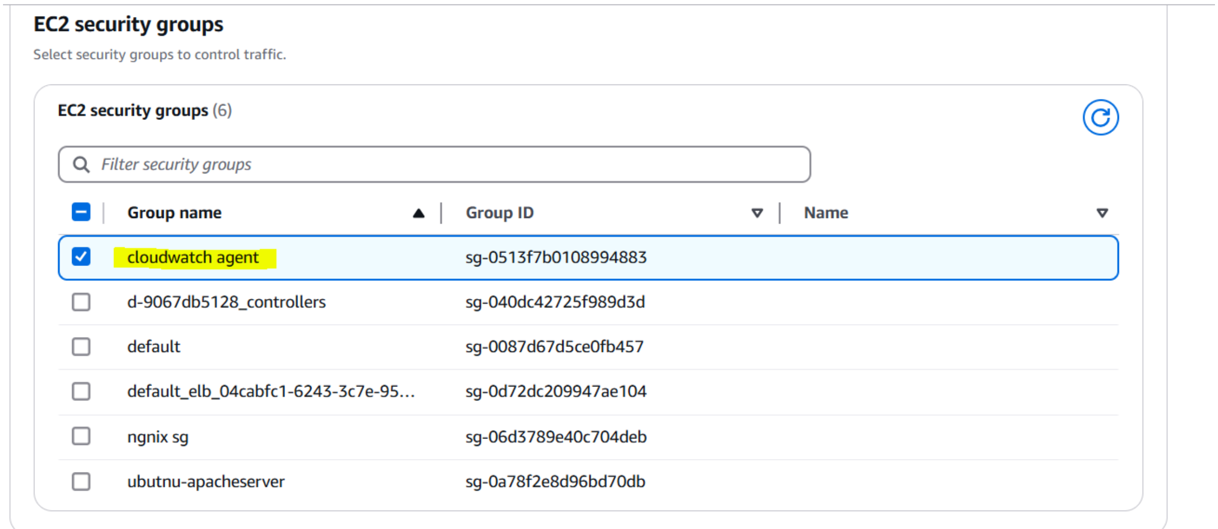
Root volume type
(Container default)

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

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

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

Now EC2 Security Groups Setting I choosed Cloudwatch agent.



EC2 security groups

Select security groups to control traffic.

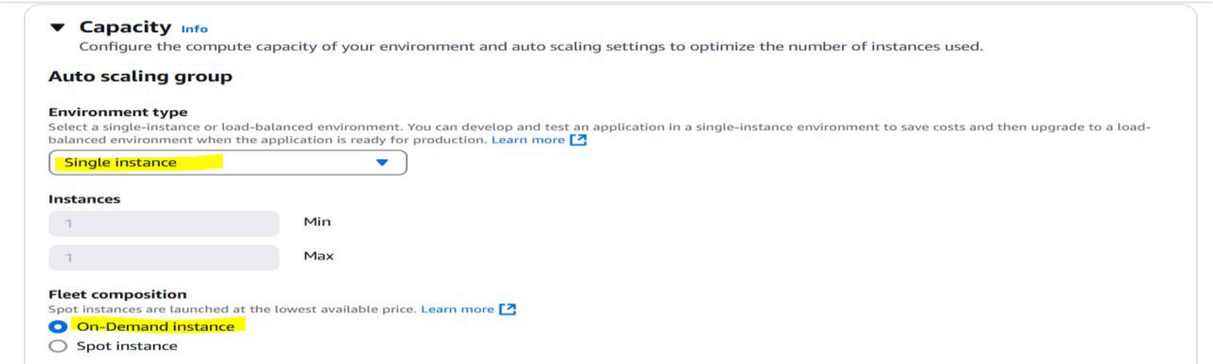
EC2 security groups (6)

Filter security groups

	Group name	Group ID	Name
☑	cloudwatch agent	sg-0513f7b0108994883	
□	d-9067db5128_controllers	sg-040dc42725f989d3d	
□	default	sg-0087d67d5ce0fb457	
□	default_elb_04cabfc1-6243-3c7e-95...	sg-0d72dc209947ae104	
□	ngnix sg	sg-06d3789e40c704deb	
□	ubutnu-apacheserver	sg-0a78f2e8d96bd70db	

Then Capacity Section

In auto scaling group environment type will be single instance.



▼ **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

Single instance ▼

Instances

1 Min

1 Max

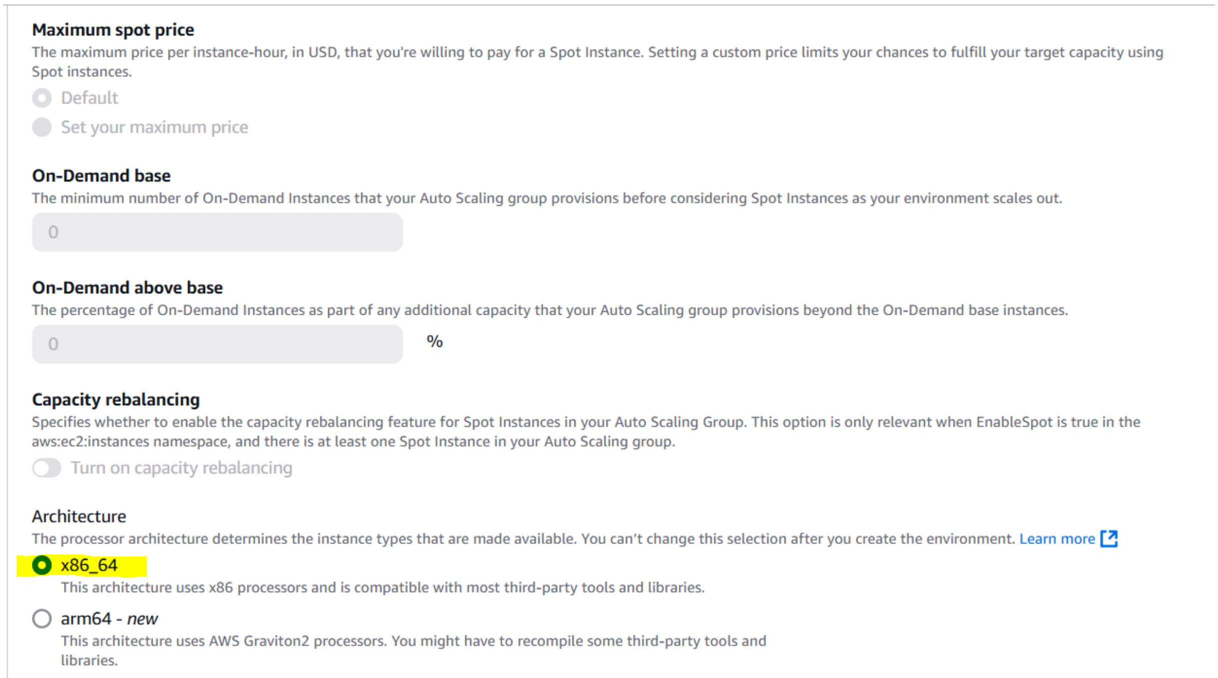
Fleet composition

Spot instances are launched at the lowest available price. Learn more

● On-Demand instance

○ Spot instance

Architecture Will be x86_64.



Maximum spot price

The maximum price per instance-hour, in USD, that you're willing to pay for a Spot Instance. Setting a custom price limits your chances to fulfill your target capacity using Spot instances.

○ Default

○ Set your maximum price

On-Demand base

The minimum number of On-Demand Instances that your Auto Scaling group provisions before considering Spot Instances as your environment scales out.

0

On-Demand above base

The percentage of On-Demand Instances as part of any additional capacity that your Auto Scaling group provisions beyond the On-Demand base instances.

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

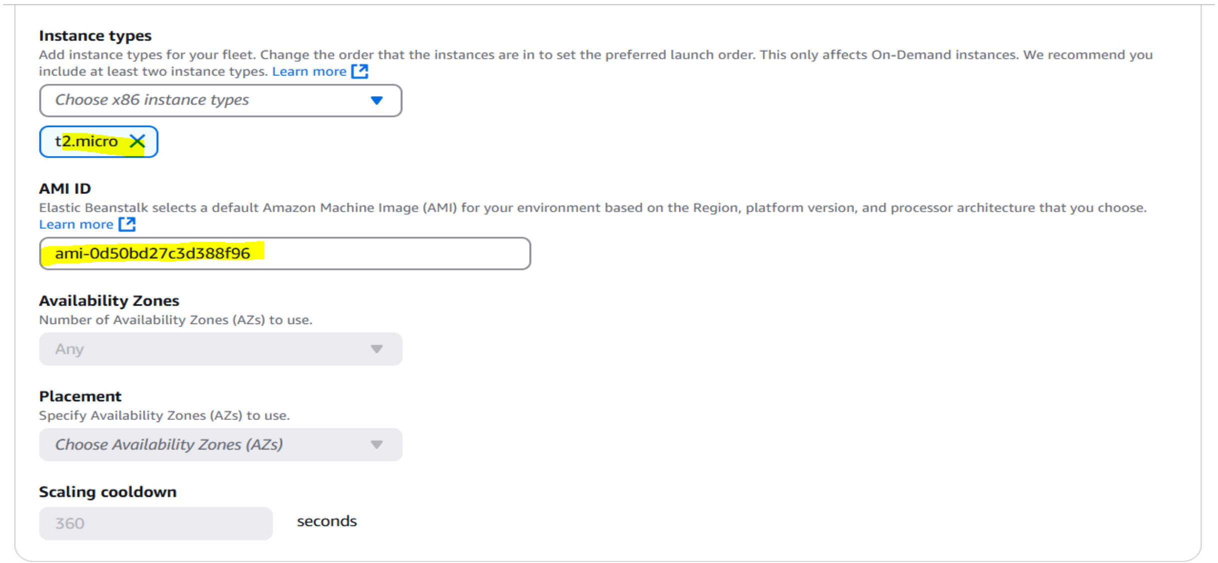
● x86_64

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 Type Will be t2.micro,AMI ID will be used by default.



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 ▼

t2.micro ✕

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-0d50bd27c3d388f96

Availability Zones

Number of Availability Zones (AZs) to use.

Any ▼

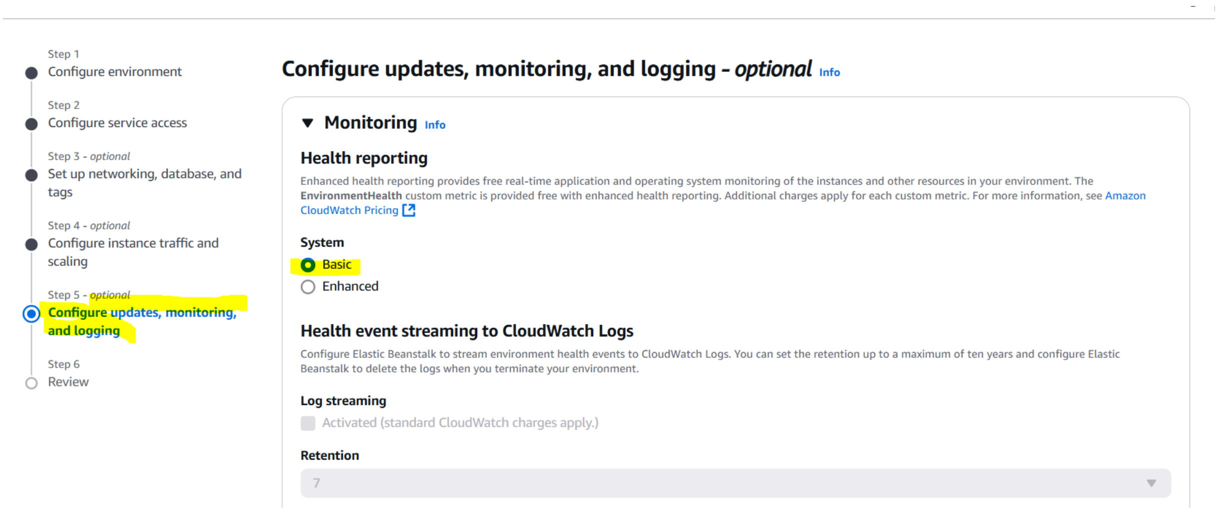
Placement

Specify Availability Zones (AZs) to use.

Choose Availability Zones (AZs) ▼

Scaling cooldown

360 seconds



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

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.

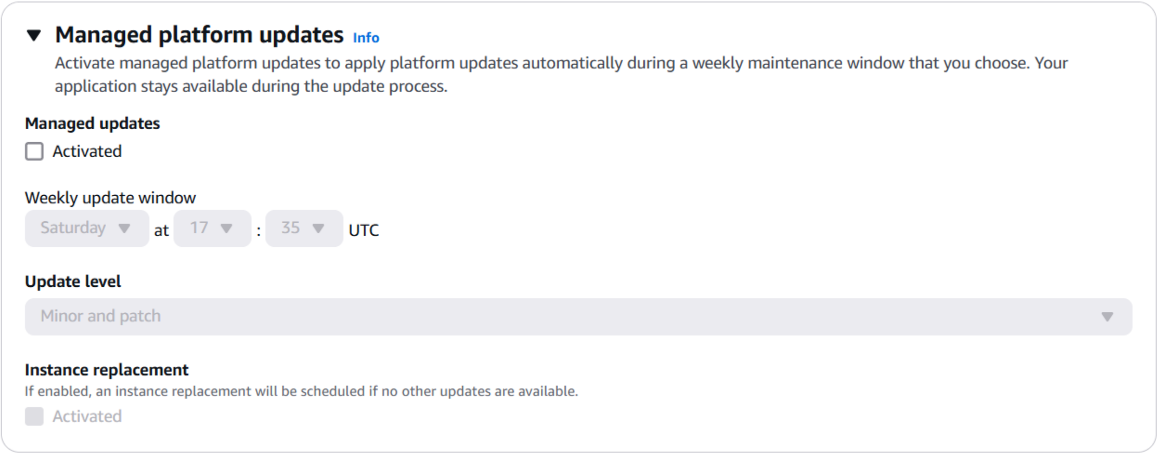
Log streaming

□ Activated (standard CloudWatch charges apply.)

Retention

7

Managed platform updates not required in this case.



▼ **Managed platform updates** Info

Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

Managed updates

□ Activated

Weekly update window

Saturday at 17 : 35 UTC

Update level

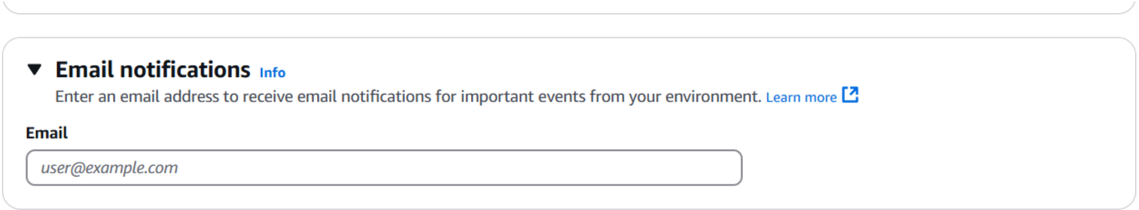
Minor and patch

Instance replacement

If enabled, an instance replacement will be scheduled if no other updates are available.

□ Activated

Not Require To Email Notified to received email.



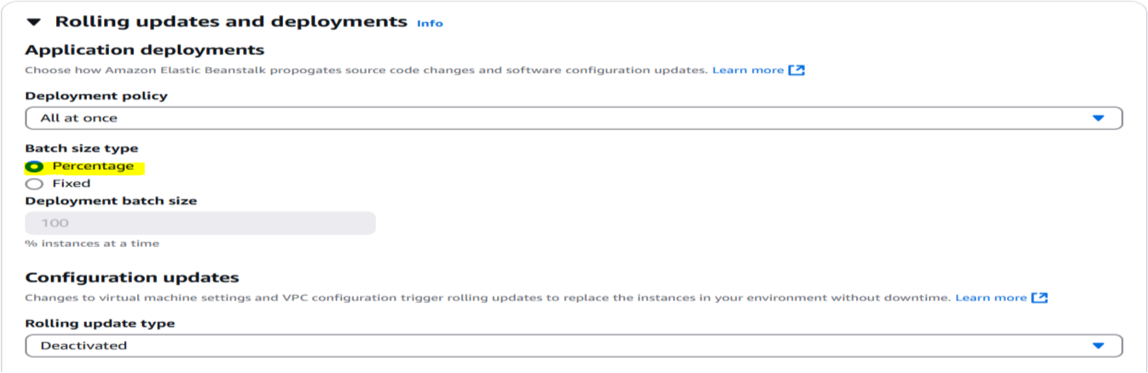
▼ **Email notifications** Info

Enter an email address to receive email notifications for important events from your environment. Learn more

Email

user@example.com

Rolling updates and deployment is by default.



▼ **Rolling updates and deployments** Info

Application deployments

Choose how Amazon Elastic Beanstalk propogates source code changes and software configuration updates. Learn more

Deployment policy

All at once

Batch size type

● Percentage
○ Fixed

Deployment batch size

100

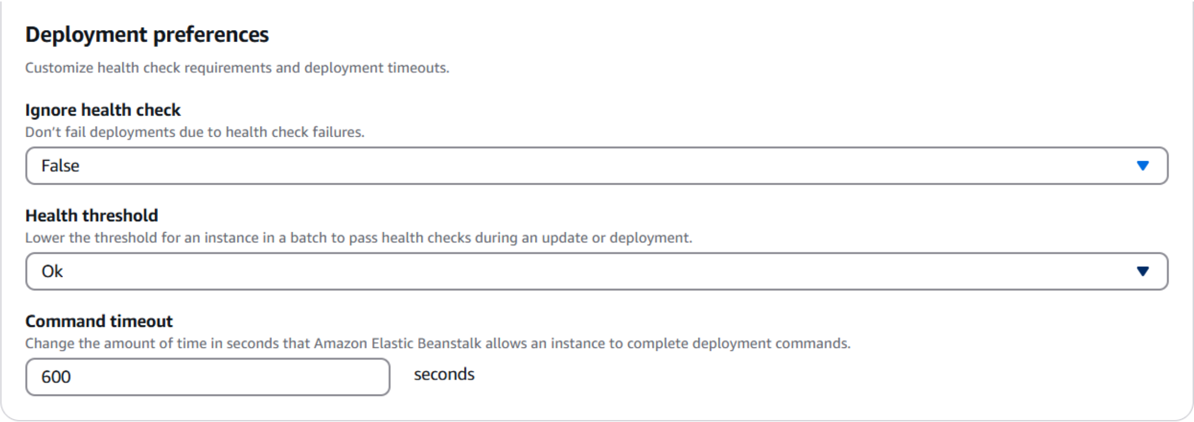
% instances at a time

Configuration updates

Changes to virtual machine settings and VPC configuration trigger rolling updates to replace the instances in your environment without downtime. Learn more

Rolling update type

Deactivated



Deployment preferences

Customize health check requirements and deployment timeouts.

Ignore health check

Don't fail deployments due to health check failures.

False ▼

Health threshold

Lower the threshold for an instance in a batch to pass health checks during an update or deployment.

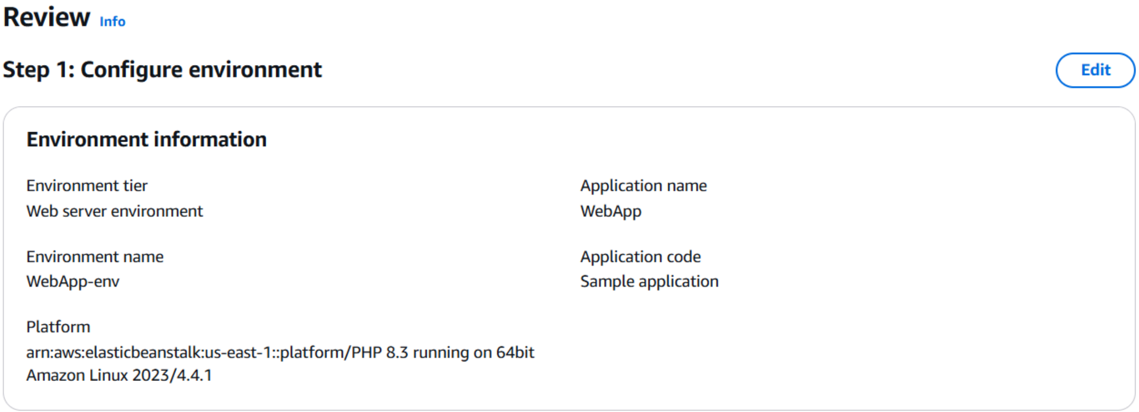
Ok ▼

Command timeout

Change the amount of time in seconds that Amazon Elastic Beanstalk allows an instance to complete deployment commands.

600 seconds

Now In This Stage we need to all this things need to review



Step 1: Configure environment

Environment information

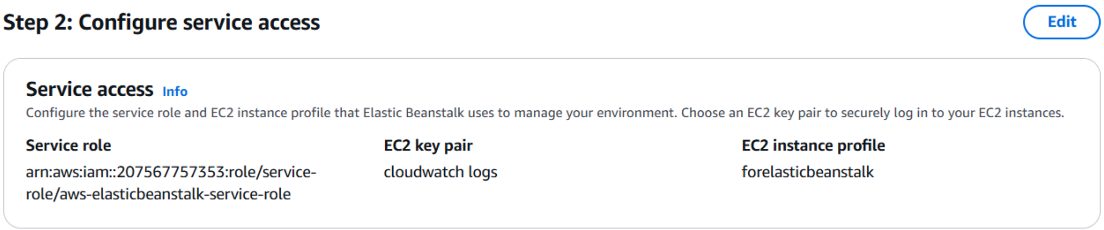
Environment tier
Web server environment

Application name
WebApp

Environment name
WebApp-env

Application code
Sample application

Platform
arn:aws:elasticbeanstalk:us-east-1::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.4.1



Step 2: Configure service access

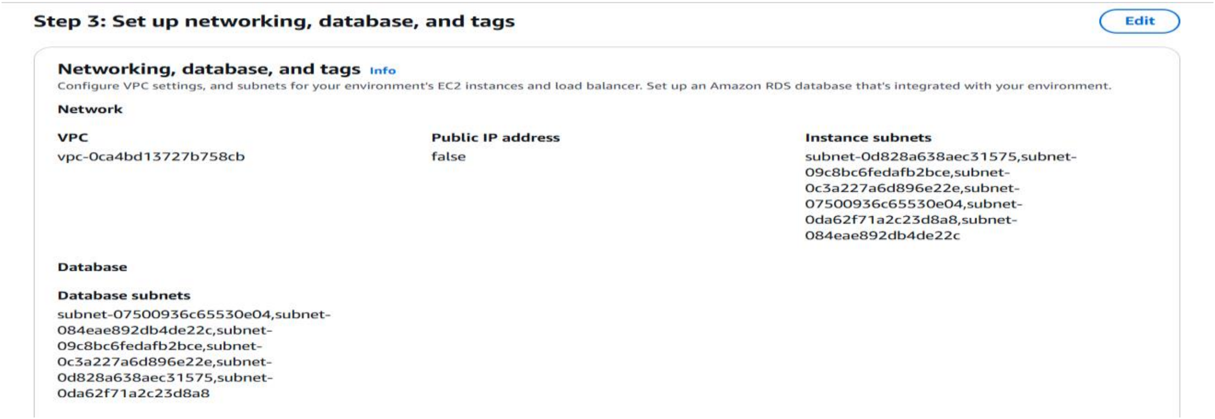
Service access

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.

Service role
arn:aws:iam::207567757353:role/service-role/aws-elasticbeanstalk-service-role

EC2 key pair
cloudwatch logs

EC2 instance profile
forelasticbeanstalk



Step 3: Set up networking, database, and tags

Networking, database, and tags

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

Network

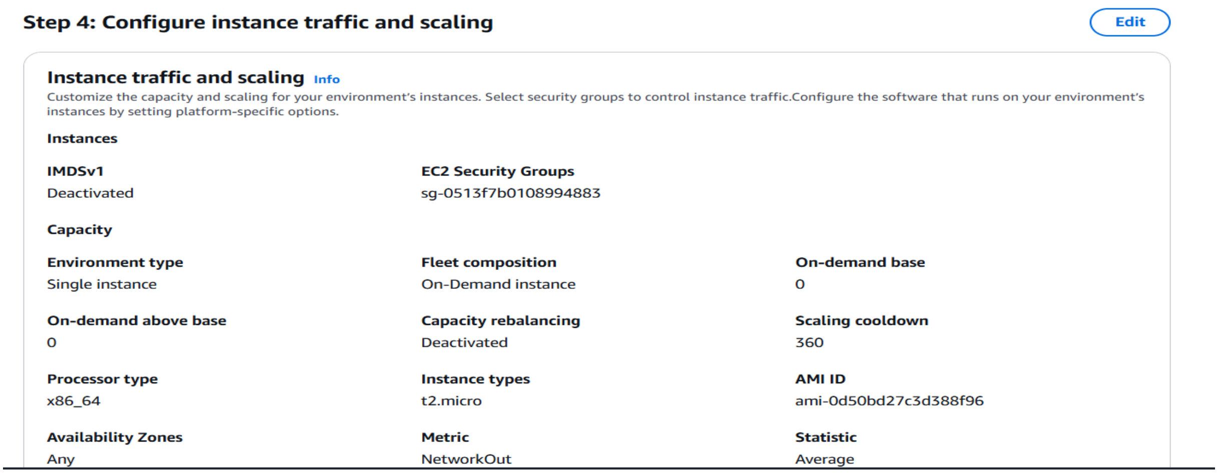
VPC
vpc-0ca4bd13727b758cb

Public IP address
false

Instance subnets
subnet-0d828a638aec31575,subnet-09c8bc6fedafb2bce,subnet-0c3a227a6d896e22e,subnet-07500936c65530e04,subnet-0da62f71a2c23d8a8,subnet-084eae892db4de22c

Database

Database subnets
subnet-07500936c65530e04,subnet-084eae892db4de22c,subnet-09c8bc6fedafb2bce,subnet-0c3a227a6d896e22e,subnet-0d828a638aec31575,subnet-0da62f71a2c23d8a8



Step 4: Configure instance traffic and scaling

Instance traffic and scaling

Customize the capacity and scaling for your environment's instances. Select security groups to control instance traffic.Configure the software that runs on your environment's instances by setting platform-specific options.

Instances

IMDSv1
Deactivated

EC2 Security Groups
sg-0513f7b0108994883

Capacity

Environment type
Single instance

Fleet composition
On-Demand instance

On-demand base
0

On-demand above base
0

Capacity rebalancing
Deactivated

Scaling cooldown
360

Processor type
x86_64

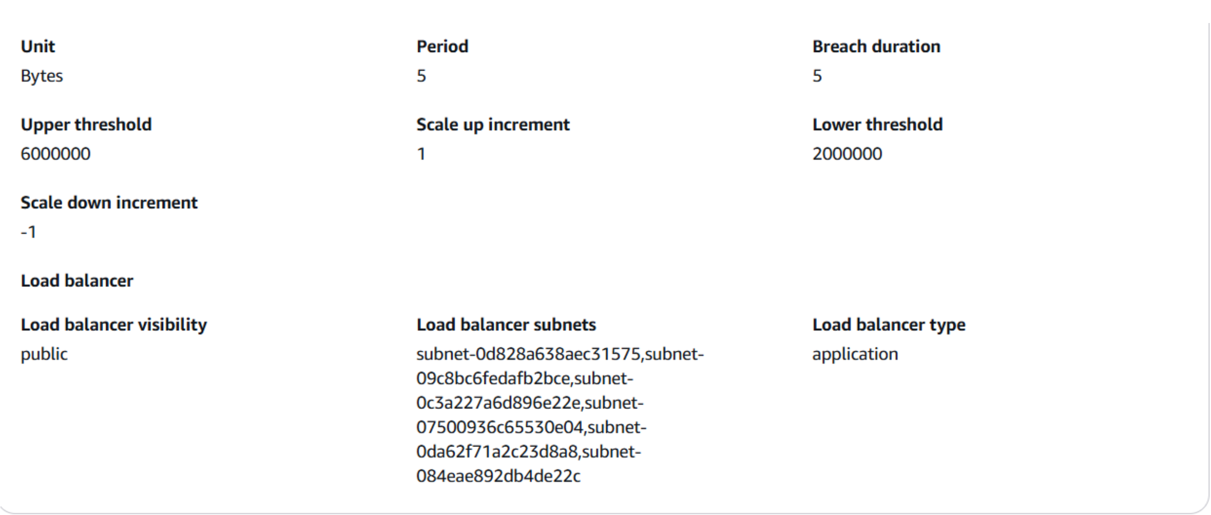
Instance types
t2.micro

AMI ID
ami-0d50bd27c3d388f96

Availability Zones
Any

Metric
NetworkOut

Statistic
Average



Unit
Bytes

Period
5

Breach duration
5

Upper threshold
6000000

Scale up increment
1

Lower threshold
2000000

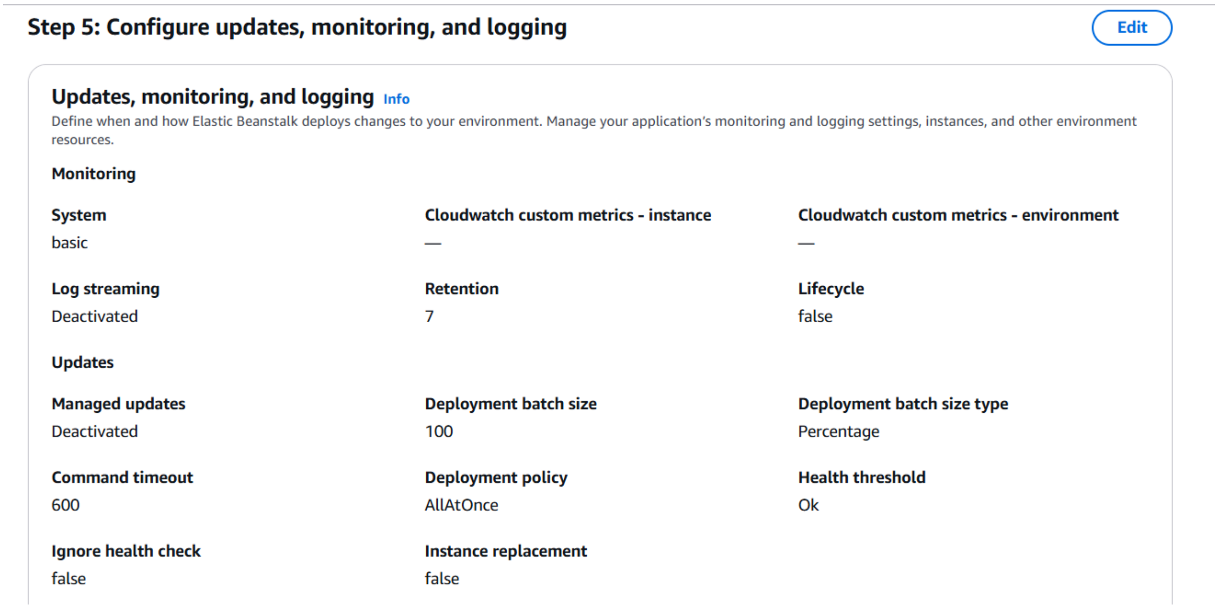
Scale down increment
-1

Load balancer

Load balancer visibility
public

Load balancer subnets
subnet-0d828a638aec31575,subnet-09c8bc6fedafb2bce,subnet-0c3a227a6d896e22e,subnet-07500936c65530e04,subnet-0da62f71a2c23d8a8,subnet-084eae892db4de22c

Load balancer type
application



Step 5: Configure updates, monitoring, and logging

Updates, monitoring, and logging

Define when and how Elastic Beanstalk deploys changes to your environment. Manage your application's monitoring and logging settings, instances, and other environment resources.

Monitoring

System
basic

Cloudwatch custom metrics - instance
—

Cloudwatch custom metrics - environment
—

Log streaming
Deactivated

Retention
7

Lifecycle
false

Updates

Managed updates
Deactivated

Deployment batch size
100

Deployment batch size type
Percentage

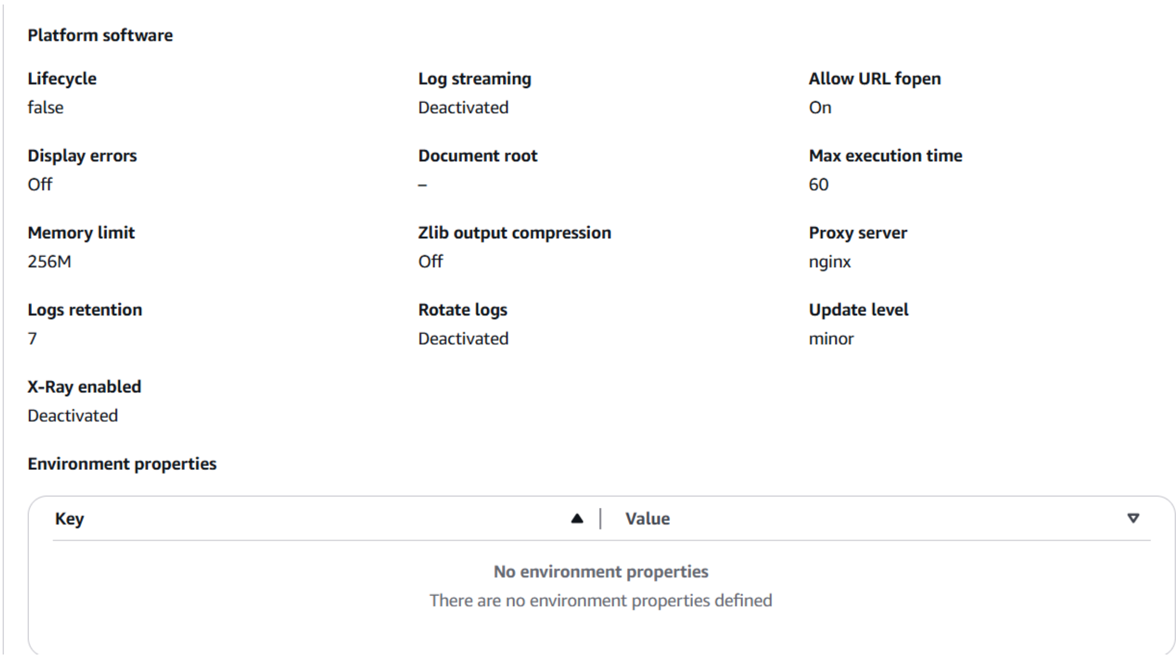
Command timeout
600

Deployment policy
AllAtOnce

Health threshold
Ok

Ignore health check
false

Instance replacement
false



Platform software

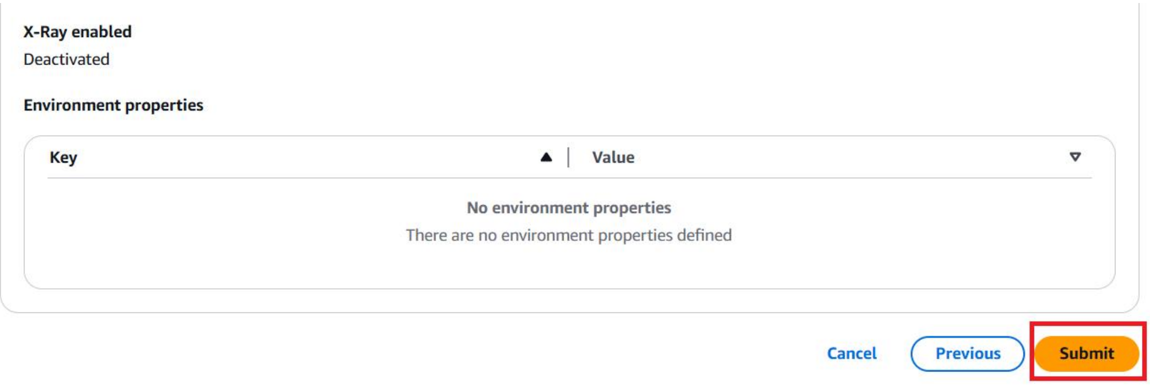
Lifecycle false	**Log streaming** Deactivated	**Allow URL fopen** On
Display errors Off	**Document root** –	**Max execution time** 60
Memory limit 256M	**Zlib output compression** Off	**Proxy server** nginx
Logs retention 7	**Rotate logs** Deactivated	**Update level** minor
X-Ray enabled Deactivated		

Environment properties

Key ▲	Value ▽

No environment properties

There are no environment properties defined



X-Ray enabled
Deactivated

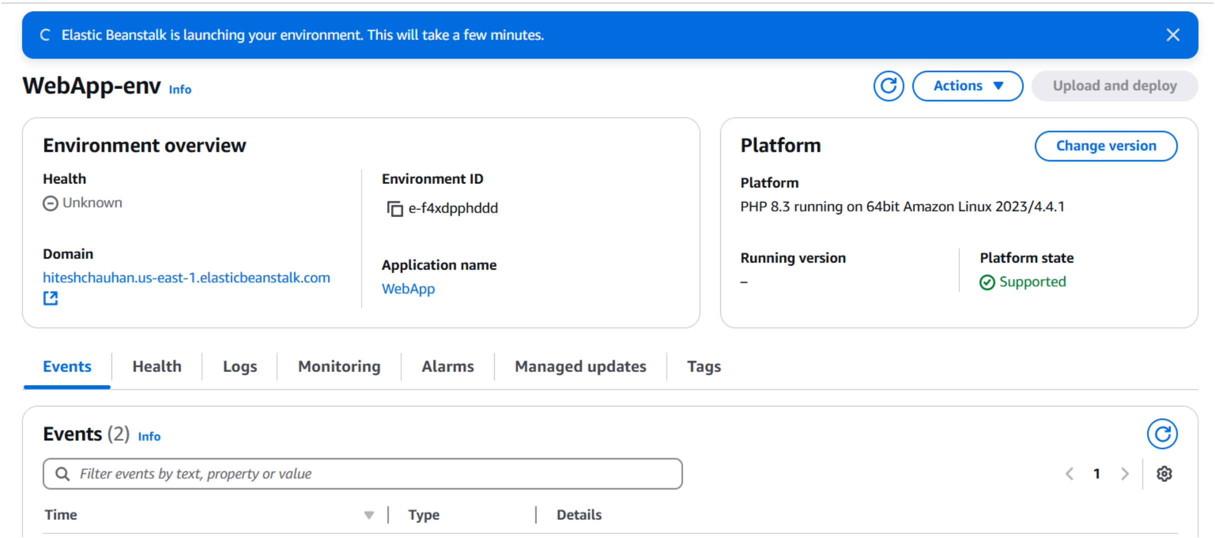
Environment properties

Key ▲	Value ▽

No environment properties

There are no environment properties defined

After Review Click the submit.



Elastic Beanstalk is launching your environment. This will take a few minutes.

WebApp-env Info

Environment overview

Health
Unknown

Environment ID
e-f4xdpphddd

Domain
hiteshchauhan.us-east-1.elasticbeanstalk.com

Application name
WebApp

Platform

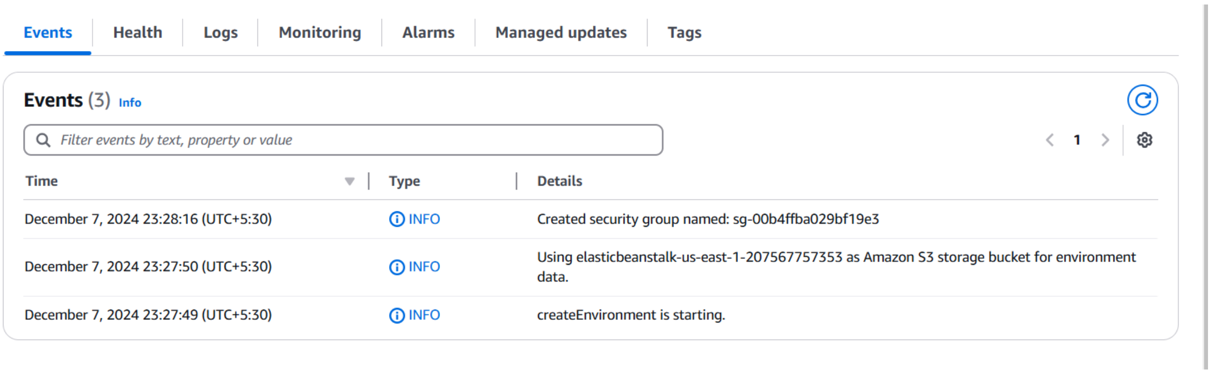
Platform
PHP 8.3 running on 64bit Amazon Linux 2023/4.4.1

Running version
–

Platform state
Supported

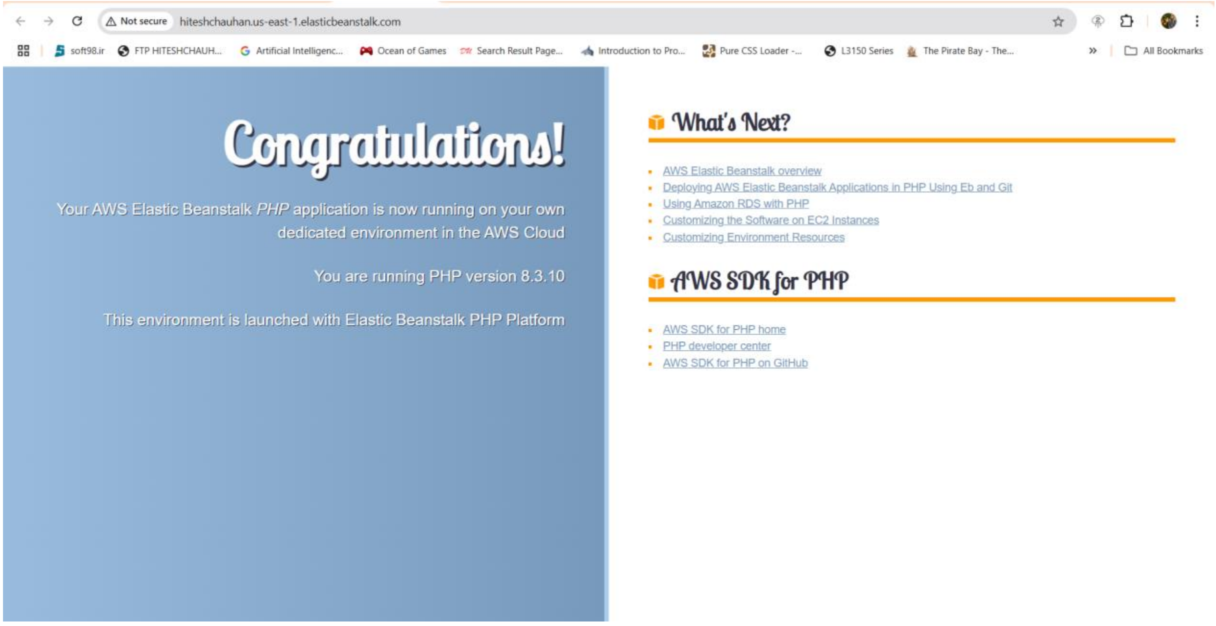
Events (2) Info

Time ▼	Type	Details



Time	Type	Details
December 7, 2024 23:28:16 (UTC+5:30)	INFO	Created security group named: sg-00b4ffba029bf19e3
December 7, 2024 23:27:50 (UTC+5:30)	INFO	Using elasticbeanstalk-us-east-1-207567757353 as Amazon S3 storage bucket for environment data.
December 7, 2024 23:27:49 (UTC+5:30)	INFO	createEnvironment is starting.

After Deployed the elasticbeanstalk.you will see by default PHP Demo Page.



Congratulations!

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

You are running PHP version 8.3.10

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

That's IT for All.