

Roll NO. A041

Cloud Computing Practical No: 02 Writeup:

1.Platform as a service

Platform as a Service (PaaS) is a cloud computing service model that provides a platform allowing customers to develop, run, and manage applications without dealing with the complexity of building and maintaining the underlying infrastructure. PaaS sits between Infrastructure as a Service (IaaS) and Software as a Service (SaaS) in the cloud computing service stack.

Characteristics of PaaS:

Development Frameworks: PaaS provides a set of tools, services, and development frameworks that allow developers to build, test, and deploy applications more efficiently.

Middleware Services: PaaS often includes middleware services such as databases, messaging systems, and caching services. These services abstract the underlying infrastructure, making it easier for developers to focus on application development.

Automated Deployment and Scaling: PaaS platforms automate the deployment and scaling of applications. Developers can easily scale their applications up or down based on demand without having to manage the underlying infrastructure.

Integrated Development Tools: PaaS offers integrated development tools, including code repositories, version control, and collaboration tools, facilitating collaboration among development teams.

Multi-Tenancy: PaaS platforms are designed to support multiple users or tenants, allowing developers to share development tools and resources in a secure and efficient manner.

Advantages of PaaS:

Faster Development: PaaS accelerates the development process by providing ready-to-use components and services.

Cost-Efficiency: Users can avoid the costs and complexities associated with maintaining and managing infrastructure.

Scalability: PaaS platforms offer automatic scaling, ensuring applications can handle varying workloads.

Disadvantages of PaaS:

Less Control: Developers have less control over the underlying infrastructure compared to IaaS.

Dependency on Provider: Users are dependent on the PaaS provider for updates, security patches, and overall platform stability.

2.Amazon Elastic Beanstalk

Amazon Elastic Beanstalk is a web infrastructure management service. It handles deployment and scaling for web applications and services. Elastic Beanstalk can automatically manage setup, configuration, scaling and provisioning for other AWS

services. AWS Elastic Beanstalk is an AWS-managed service for web applications. Elastic Beanstalk is a pre-configured EC2 server that can directly take up your application code and environment configurations and use it to automatically provision and deploy the required resources within AWS to run the web application. Unlike EC2 which is Infrastructure as a service, Elastic Beanstalk is a Platform As A Service (PAAS) as it allows users to directly use a pre-configured server for their application. Of course, you can deploy applications without ever having to use elastic beanstalk but that would mean having to choose the appropriate service from the vast array of services offered by AWS, manually provisioning these AWS resources, and stitching them up together to form a complete web application. Elastic Beanstalk abstracts the underlying configuration work and allows you as a user to focus on more pressing matters.

3.)Components of Amazon ElasticBeanStalk

- **Application:** Elastic Beanstalk directly takes in our project code. So Elastic Beanstalk application is named the same as your project home directory.
- **Application Environments:** Users may want their application to run on different environments like DEV, UAT, and PROD. You can create and configure different environments to run applications on different stages.
- **Environment Health:** One of the most lucrative features of running applications on AWS or most of the other cloud platforms is automated health checks. AWS runs automatic health checks on all EC-2 deployments (Elastic Beanstalk is a managed EC-2 service) which can be monitored from the AWS console. For example, in the case of web applications AWS will regularly, as scheduled by the developers, ping the application to check if the response is status code 200 and if the application is running as expected. Health check responses:
 - **Red:** The application failed all health tests.
 - **Yellow:** The application failed some of the health tests.
 - **Grey:** The application is updating.
 - **Green:** The application passed the health check successfully.
- **Isolated:** All environments within a single application are isolated from each other (independent of each others' running states). Needless to say, two different applications are also isolated.
- **Scalability:** Using Auto-Scaling within Elastic Beanstalk makes the application dynamically scalable.
- **Elastic Load Balancing:** All the web requests to the application are not directly relayed to application instances. They first hit the Elastic Load Balancer (ELB), which, as the name suggests, balances the load across all the application instances.
- **Language support:** Elastic Beanstalk supports the applications developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
- **Pricing:** There is no extra charge for using Elastic Beanstalk. Users are only required to pay for the services and resources provisioned by Elastic Beanstalk Service.
- **Automatic Provisioning:** Elastic Beanstalk takes away the burden of choosing the right services and configuring their security groups to work together.
- **Impossible to Outgrow:** AWS claims that since Elastic Beanstalk uses the Auto Scaling feature it can, in theory, handle any amount of internet traffic.

4.)IAM:

IAM stands for Identity and Access Management. In the context of Amazon Web Services (AWS), IAM refers to the service that allows you to manage access to AWS resources securely. IAM enables you to control who (authentication) can do what (authorization) in your AWS environment.

Here are key aspects of AWS Identity and Access Management (IAM):

Users and Groups:

Users: Represent individuals or entities that interact with AWS services. Each user has a unique set of security credentials.

Groups: Users can be organized into groups, and permissions can be assigned to groups, making it easier to manage access.

Roles: IAM roles define a set of permissions for making AWS service requests. Roles are not associated with a specific user or group but can be assumed by users, applications, or services when needed.

Policies: IAM policies are JSON documents that define permissions. They can be attached to users, groups, or roles, specifying what actions are allowed or denied on what resources.

Access Keys: IAM provides access keys (access key ID and secret access key) for programmatic access to AWS services. These keys are often used by developers and applications.

Multi-Factor Authentication (MFA): IAM supports MFA, an additional layer of security that requires users to provide a second form of authentication (such as a code from a virtual or hardware MFA device) in addition to their password.

Identity Federation: IAM allows you to integrate with external identity providers, such as Active Directory or social identity providers, to grant temporary access to AWS resources.

Resource-Level Permissions: IAM policies can define permissions not only at the service level but also at the resource level. This allows fine-grained control over access to specific AWS resources.

IAM Roles for EC2 Instances: IAM roles can be assigned to EC2 instances, allowing applications running on those instances to securely access AWS resources without embedding credentials in the code.

Policy Conditions: IAM policies can include conditions that must be met for the policy to be in effect. Conditions can be based on factors such as the time of day, the source IP address, or the use of MFA.

Implement paas using elastic beanstalk for the following.

- 1. Server**
- 2. Java**
- 3. Python**
- 4. Node.js**

For Server

1) Sign In to your aws acc

Sign in

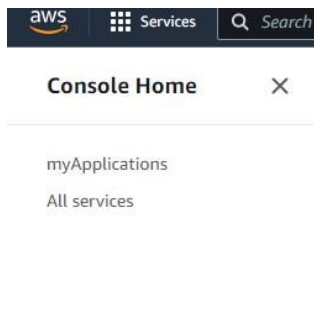
☒ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

Root user email address

Next

2.Select all services



3.Select Elastic Beanstalk



4.Click on Create Application



5.Enter application name and description and then click on create

Create new application [Info](#)

Application information

Application name

Maximum length of 100 characters.

Description

6. Now create environment

[Elastic Beanstalk](#) > [Applications](#) > MyWebApp

Application MyWebApp environments (1) [Info](#)

< 1 > [Settings](#)

Environ... ▲ | Health ▼ | Date cre... ▼ | Domain ▼ | Running ... ▼ | Platform ▼ | Pla

[Create new environment](#)

7. For Configure environment everything will be default except platform since we are doing for server we will choose .Net on Windows server Platform branch and platform version will be default. Click on next.

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

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

Must be from 1 to 42 characters or 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.

Domain

 [Check availability](#)

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

Platform branch

Platform version

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

8. For Service access you have to create a role since it is by default over here you can create your own role .

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.

↻

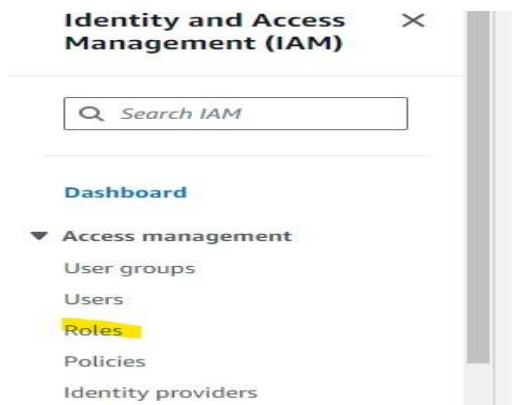
View permission details

Cancel
Skip to review
Previous
Next

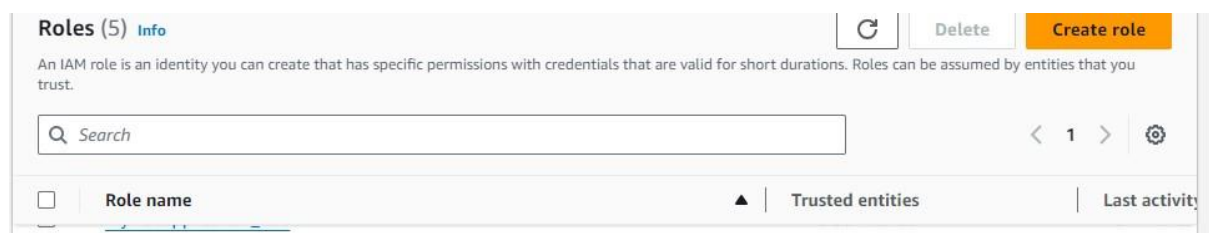
9. To create role go to services IAM



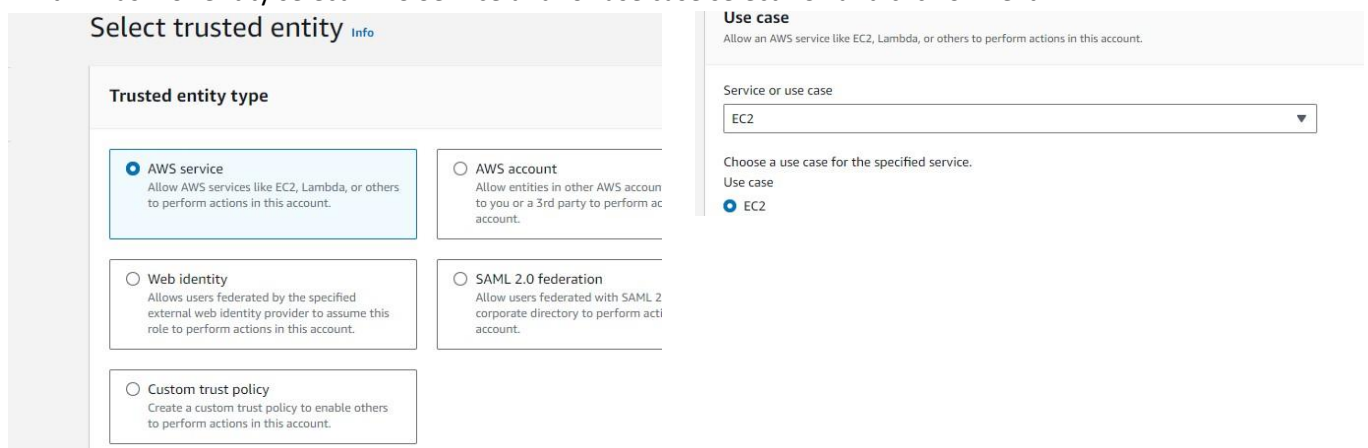
10. Inside IAM dashboard go to roles



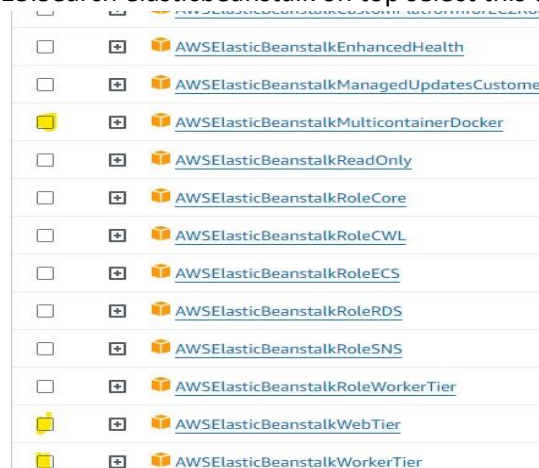
11. Create role



12. It will ask for entity select AWS Service and for use case select EC2 and click on next



13. Search elasticbeanstalk on top select this three opt and click on next



14. Give role name and click on create role

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.

Add new tag

You can add up to 50 more tags.

[Cancel](#) [Previous](#) [Create role](#)

15.Role will get created

Role MyWebAPP created. [View role](#) [X](#)

Roles (6) [Info](#) [Refresh](#) [Delete](#) [Create role](#)

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

[<](#) [1](#) [>](#) [Settings](#)

<input type="checkbox"/>	Role name	Trusted entities	Last activity
--------------------------	-----------	------------------	---------------

16.Now you can again go to your service access that is your 8 step now inside existing service roles refresh it and click on dropdown you can see over there your role name which you have created above automatically comes there click on that

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.

[Refresh](#)

EC2 key pair

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

[Refresh](#)

EC2 instance profile

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

[Refresh](#)

[View permission details](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

17.For instance profile click on dropdown your role name will come select that then enter next

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.

[Refresh](#)

EC2 key pair

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

[Refresh](#)

EC2 instance profile

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

[Refresh](#)

[View permission details](#)

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

18.After clicking on next it will ask for VPC click on dropdown and select which is default not down the ip address of this

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-08f10f70caf202076 | (172.31.0.0/16)

[Create custom VPC](#)

19. The IP address you have selected above same over here select that IP address and click on next

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
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

Database

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 (3)

Filter database subnets

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

20. for configure instance everything will be by default enter next

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

Instances

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.

8

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

MB/s

Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances.

Monitoring interval

5 minute

Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

IMDSv1

21. For configure update also everything will be default click to next

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 ▼

22. Review will display click on submit

Review [Info](#)

Step 1: Configure environment [Edit](#)

Environment information

Environment tier	Application name
Web server environment	MyWebApp
Environment name	Application code
MyWebApp-env	Sample application
Platform	
amazon-elasticbeanstalk-ec2-v1:platform/iis-10.0 running on 64-bit Windows Server 2019/2.13.2	

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.

Service role	EC2 instance profile
arn:aws:iam::836706723086:role/MyWebAPP	MyWebAPP

Step 3: Set up networking, database, and tags [Edit](#)

23. You can see it is getting launch

search [\[Alt+S\]](#)

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

[Elastic Beanstalk](#) > [Environments](#) > Arisha

Arisha [Info](#)

Environment overview

Platform

24. Now you can go to instance from my service you can see The instance is running.

Search [Alt+S]

Instances (2) Info

Find instance by attribute or tag (case-sensitive)

Instance state = running X Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
Mywebapp-env	i-0be4f576697dcb5c4	Running	t3.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-13-234-183-3.ap-s...	13.234.183.3
Arisha	i-0dfbc5d95c1a61fd6	Running	t3.large	2/2 checks passed	View alarms +	ap-south-1b	ec2-65-2-112-232.ap-s...	65.2.112.232

25. Click on domain

Health

Warning

Domain

Arisha.eba-m5pwiltt.ap-south-1.elasticbeanstalk.com

Events (9) Info

Filter events by text, property or value

Time Type

26. It will get open into another browser now Elastic Beanstalk Server is running on your own dedicated environment.

Congratulations!

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

What's Next?

- [AWS Elastic Beanstalk overview](#)
- [AWS Elastic Beanstalk concepts](#)
- [Deploying Applications in .NET Using AWS Toolkit for Visual Studio](#)
- [Managing .NET Environment Settings](#)
- [Working with Logs](#)

AWS SDK for .NET

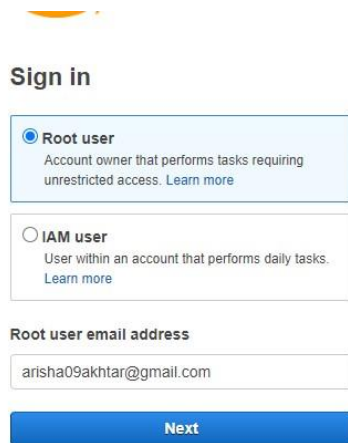
- [AWS SDK for .NET home](#)
- [AWS Toolkit for Visual Studio home](#)
- [Windows and .NET developer center](#)
- [AWS SDK for .NET documentation](#)
- [AWS SDK for .NET on GitHub](#)

AWS .NET Services

- [Generate test events for AWS X-Ray Service](#)

For Java

1.) Sign In to your aws acc



Sign in

☒ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

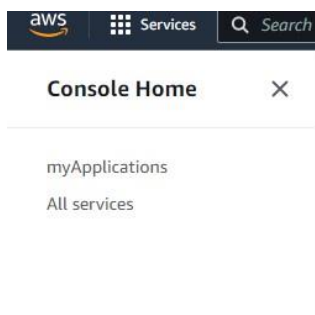
☐ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

Root user email address

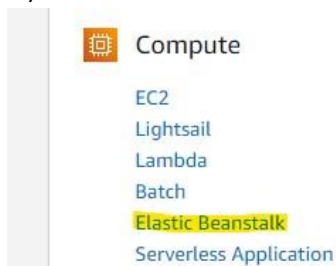
arisha09akhtar@gmail.com

Next

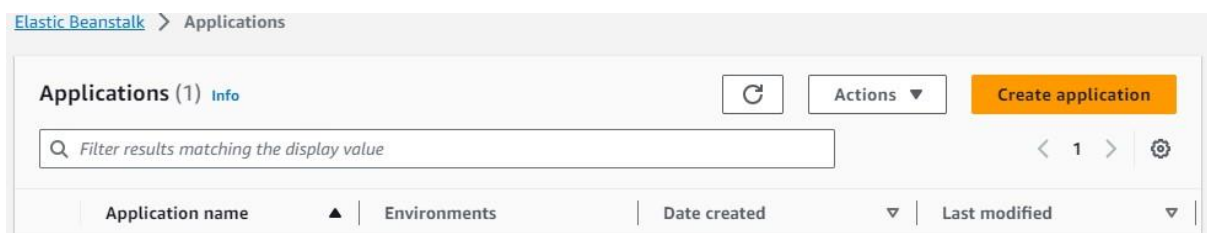
2.)Select all services



3.)Select Elastic Beanstalk



4.) Click on Create Application



5.) Enter application name and description and then click on create

Create new application [Info](#)

Application information

Application name

Maximum length of 100 characters.

Description

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.

You can add 50 more tags.

6. Now create environment

[Elastic Beanstalk](#) > [Applications](#) > MyWebApp

Application MyWebApp environments (1) [Info](#)

< 1 >

Environ...	Health	Date cre...	Domain	Running ...	Platform	Pla
------------	--------	-------------	--------	-------------	----------	-----

7. For Configure environment everything will be default except platform since we are doing for Java we will choose .Net on Windows server Platform branch and platform version will be default. Click on next.

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

Maximum length of 100 characters.

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

Platform branch

Platform version

8.) For Service access you have to create a role since it is by default over here you can create your own role .

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.

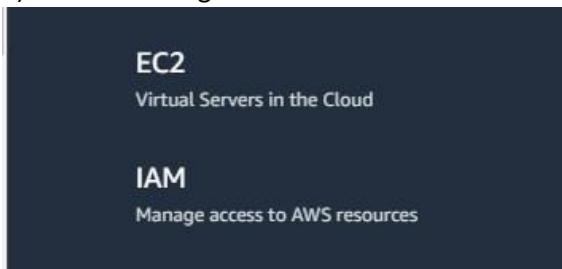
EC2 key pair

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

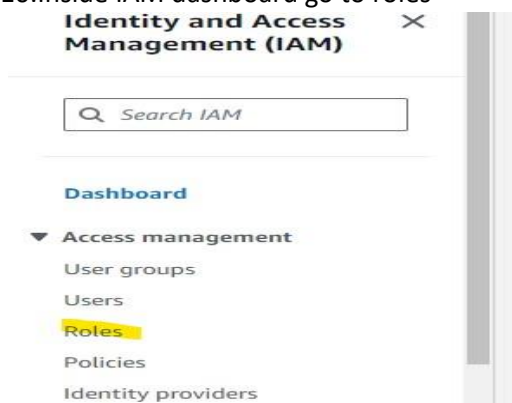
EC2 instance profile

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

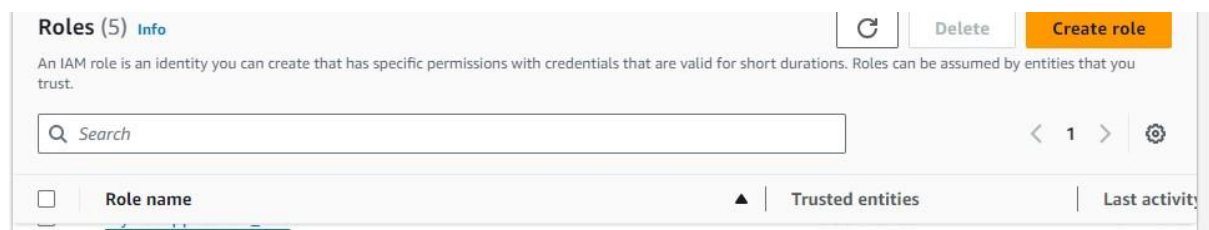
9.)To create role go to services IAM



10. Inside IAM dashboard go to roles



11. Create role



12. It will ask for entity select AWS Service and for use case select EC2 and click on next

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 account 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 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

Choose a use case for the specified service.

Use case

☒ **EC2**

13. Search elasticbeanstalk on top select this three opt and click on next

<input type="checkbox"/>		AWSElasticBeanstalkEnhancedHealth
<input type="checkbox"/>		AWSElasticBeanstalkManagedUpdatesCustom
<input checked="" type="checkbox"/>		AWSElasticBeanstalkMulticontainerDocker
<input type="checkbox"/>		AWSElasticBeanstalkReadOnly
<input type="checkbox"/>		AWSElasticBeanstalkRoleCore
<input type="checkbox"/>		AWSElasticBeanstalkRoleCWL
<input type="checkbox"/>		AWSElasticBeanstalkRoleECS
<input type="checkbox"/>		AWSElasticBeanstalkRoleRDS
<input type="checkbox"/>		AWSElasticBeanstalkRoleSNS
<input type="checkbox"/>		AWSElasticBeanstalkRoleWorkerTier
<input checked="" type="checkbox"/>		AWSElasticBeanstalkWebTier
<input checked="" type="checkbox"/>		AWSElasticBeanstalkWorkerTier

14. Give role name and click on create role

Name, review, and create

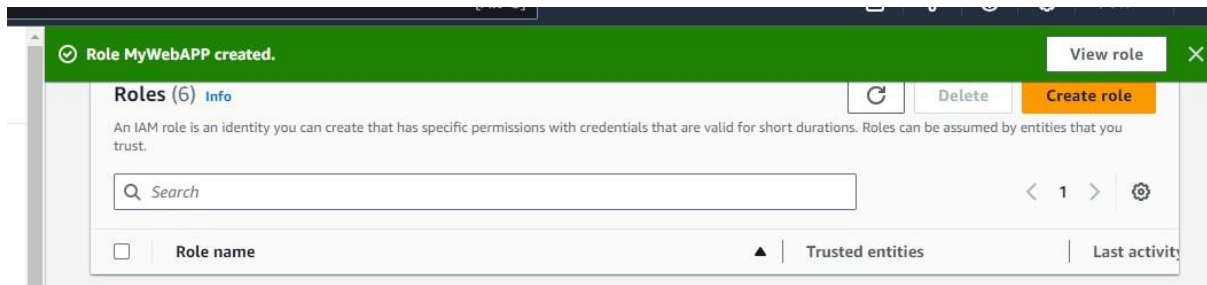
Role details

Role name
Enter a meaningful name to identify this role.

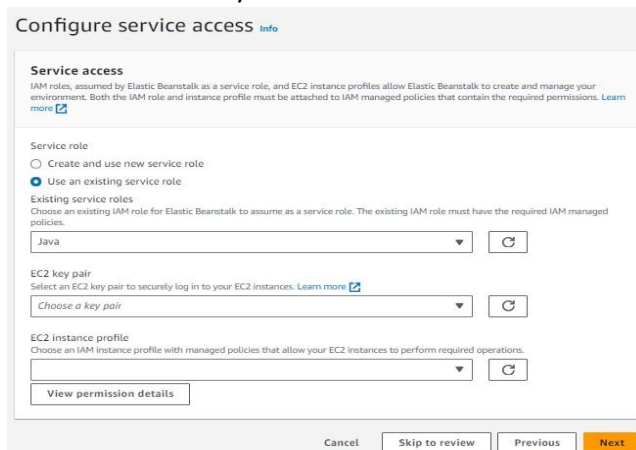
Maximum 64 characters. Use alphanumeric and '+=, @-_' characters.

Description
Add a short explanation for this role.

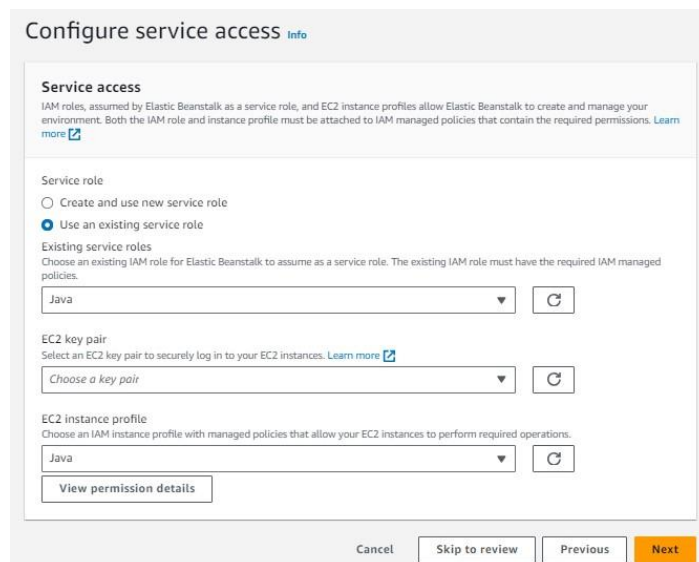
15. Role will get created



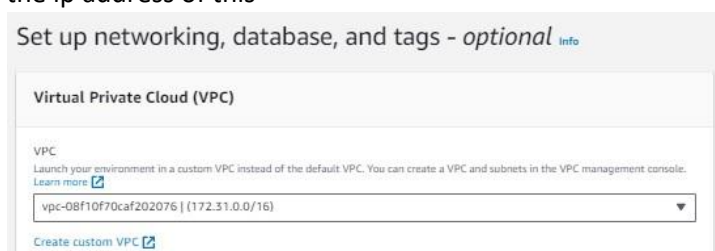
16. Now you can again go to your service access that is your 8 step now inside existing service roles refresh it and click on dropdown you can see over there your role name which you have created above automatically comes there click on that



17. For instance profile click on dropdown your role name will come select that then enter next



18. After clicking on next it will ask for VPC click on dropdown and select which is default not down the ip address of this



19. The ip address you have selected above same over here select that ip address and click on next

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

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

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 (3)

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

20.for configure instance everything will be by default enter next

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

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.
 IOPS

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

Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances.

Monitoring interval

Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

IMDSv1

21.For configure update also everthing will be default click to next

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 ▼

22. Review will display click on submit

Review [Info](#)

Step 1: Configure environment [Edit](#)

Environment information

Environment tier	Application name
Web server environment	MyWebApp
Environment name	Application code
MyWebApp-env	Sample application
Platform	
amazonelasticbeanstalkap-south-1:platform/15.10.0 running on 64bit Windows Server 2019/2.13.2	

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.

Service role	EC2 instance profile
arn:aws:iam::836706723086:role/MyWebAPP	MyWebAPP

Step 3: Set up networking, database, and tags [Edit](#)

23. You can see it is getting launch

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

[Elastic Beanstalk](#) > [Environments](#) > JAVA-env

JAVA-env [Info](#)

[Refresh](#)
[Actions](#)
[Upload and deploy](#)

Environment overview

Health	Environment ID
Unknown	e-kj2ix5xkpg
Domain	Application name
-	java

Platform [Change version](#)

Platform	
Corretto 21 running on 64bit Amazon Linux 2023/4.2.0	
Running version	Platform state
-	Supported

24. Now you can go to EC2 from my service you can see The instance is running.

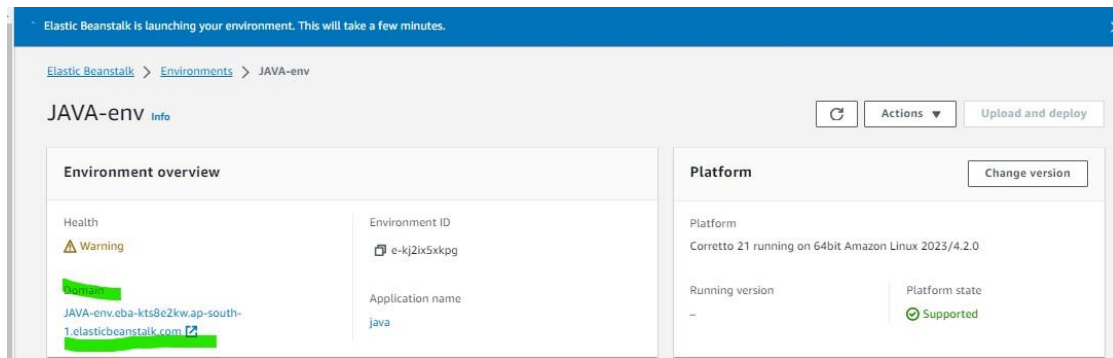
Instances (3) [Info](#)
[Refresh](#)
[Connect](#)
[Instance state](#)
[Actions](#)
[Launch instances](#)

Find Instance by attribute or tag (case-sensitive)
 Any state

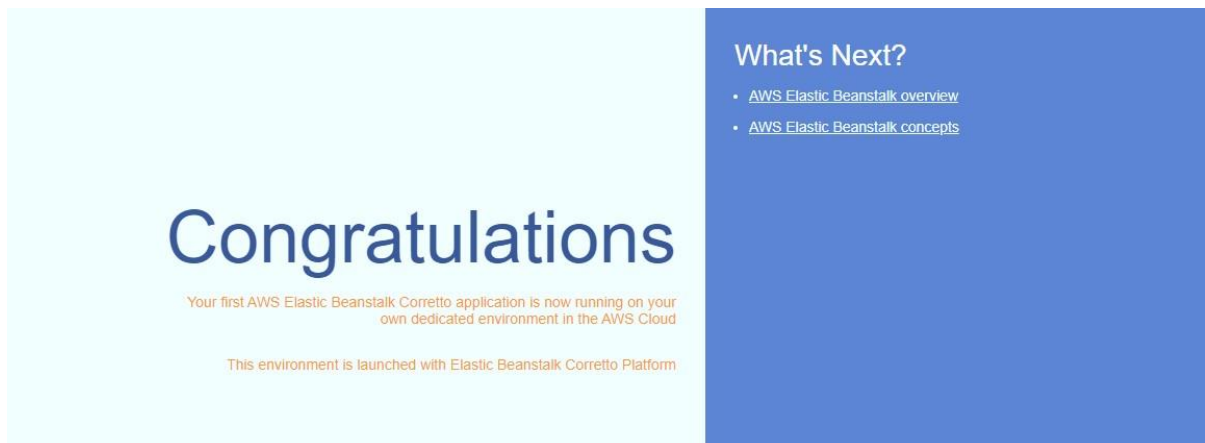
Instance state = running
 Clear filters

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4
<input type="checkbox"/>	Mywebapp-env	i-0be4f376697dcb5c4	Running	t3.micro	2/2 checks passed	View alarms	ap-south-1b	ec2-13-234-183-3.ap-s...	13.234.183.
<input type="checkbox"/>	Alisha	i-0dfbc5d95c1a61fd6	Running	t3.large	2/2 checks passed	View alarms	ap-south-1b	ec2-65-2-112-232.ap-s...	65.2.112.23
<input type="checkbox"/>	JAVA-env	i-02381801472b48cea	Running	t3.micro	Initializing	View alarms	ap-south-1b	ec2-43-204-156-218.ap...	43.204.156.

25. Click on domain

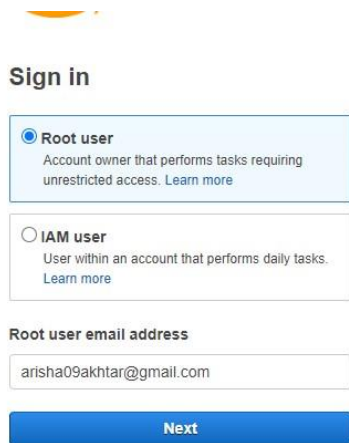


26. It will get open into another browser now Elastic Beanstalk Server is running on your own dedicated environment.



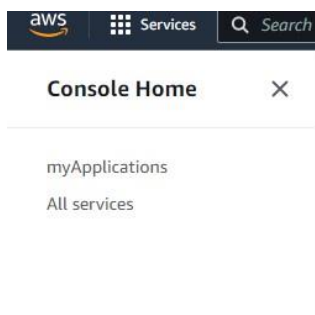
For Python

1.) Sign In to your aws acc

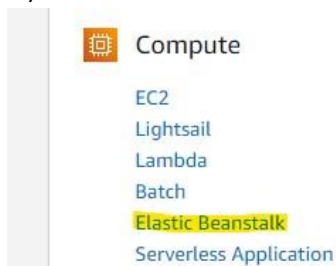


The image shows the AWS Sign in page. At the top, there is a "Sign in" heading. Below it, there are two radio button options: "Root user" (selected) and "IAM user". The "Root user" option has a description: "Account owner that performs tasks requiring unrestricted access. [Learn more](#)". The "IAM user" option has a description: "User within an account that performs daily tasks. [Learn more](#)". Below these options, there is a text input field labeled "Root user email address" containing the email "arisha09akhtar@gmail.com". At the bottom, there is a blue "Next" button.

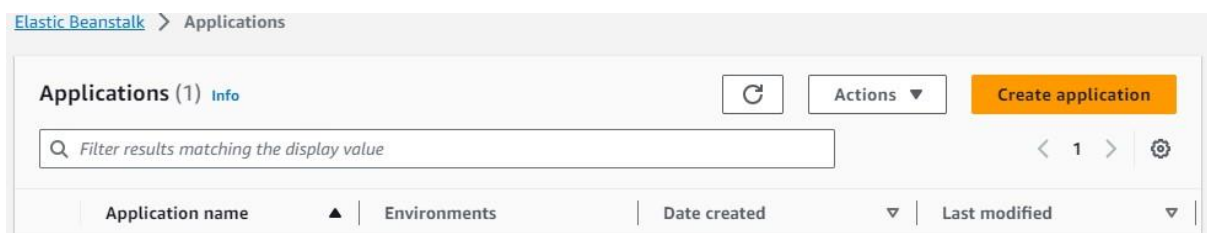
2.)Select all services



3.)Select Elastic Beanstalk



4.) Click on Create Application



5.) Enter application name and description and then click on create

Create new application [Info](#)

Application information

Application name

Maximum length of 100 characters.

Description

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.

You can add 50 more tags.

6. Now create environment

[Elastic Beanstalk](#) > [Applications](#) > MyWebApp

Application MyWebApp environments (1) [Info](#)

< 1 >

Environ...	Health	Date cre...	Domain	Running ...	Platform	Pla
------------	--------	-------------	--------	-------------	----------	-----

7. For Configure environment everything will be default except platform since we are doing for python we will choose python Platform branch and platform version will be default. Click on next.

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

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

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.

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

Platform branch

Platform version

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.

8.) For Service access you have to create a role since it is by default over here you can create your own role .

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.

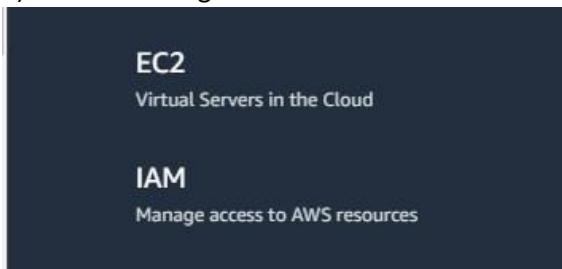
EC2 key pair

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

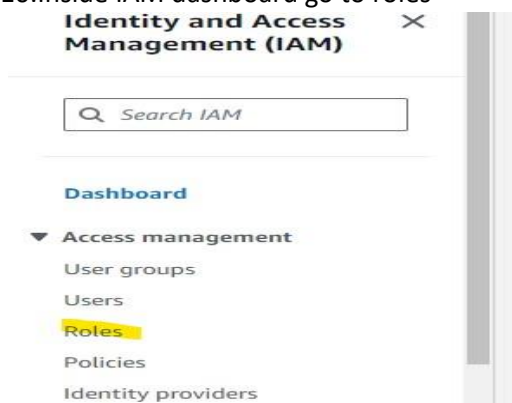
EC2 instance profile

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

9.)To create role go to services IAM



10. Inside IAM dashboard go to roles



11. Create role

Roles (5) [Info](#)

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>			

12. It will ask for entity select AWS Service and for use case select EC2 and click on next

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 account 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 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 ▼

Choose a use case for the specified service.

☒ **EC2**

<input type="checkbox"/>		 <u>AWSElasticBeanstalkEnhancedHealth</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkManagedUpdatesCustom</u>
		 <u>AWSElasticBeanstalkMulticontainerDocker</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkReadOnly</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleCore</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleCWL</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleECS</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleRDS</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleSNS</u>
<input type="checkbox"/>		 <u>AWSElasticBeanstalkRoleWorkerTier</u>
		 <u>AWSElasticBeanstalkWebTier</u>
		 <u>AWSElasticBeanstalkWorkerTier</u>

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

PYTHON

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 alphanumeric and '+=,._@-.' characters.

Q search [Alt+S]

ess (M)

Role PYTHON created. View role

[IAM](#) > Roles

Roles (9) [Info](#)

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Q Search

Name	Type	Actions
RolePYTHON	AssumeRole	iam:CreateRole, iam:DeleteRole, iam:ListRoles, iam:PutRolePermissionsBoundary, iam:UpdateRolePermissionsBoundary
...		

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.

PYTHON

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.

17. For instance profile click on dropdown your role name will come select that then enter next

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.

PYTHON

18. After clicking on next it will ask for VPC click on dropdown and select which is default not down the ip address of this

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-08f10f70caf202076 | (172.31.0.0/16)

[Create custom VPC](#)

19. The ip address you have selected above same over here select that ip address and click on next

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

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

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 (3)

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

20.for configure instance everything will be by default enter next

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

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.
 IOPS

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

Amazon CloudWatch monitoring
The time interval between when metrics are reported from the EC2 instances.

Monitoring interval

Instance metadata service (IMDS)
Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

IMDSv1

21.For configure update also everthing will be default click to next

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 [↕](#)

22. Review will display click on submit

Review [Info](#)

Step 1: Configure environment [Edit](#)

Environment information

Environment tier	Application name
Web server environment	PYTHON
Environment name	Application code
PYTHON-env	Sample application
Platform	
amazonaws:elasticbeanstalk:ap-south-1::platform/Python 3.11 running on 64bit Amazon Linux 2023/4.0.8	

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.

Service role	EC2 instance profile
arn:aws:iam::836706723086:role/PYTHON	PYTHON

23. You can see it is getting launch

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

Elastic Beanstalk > Environments > PYTHON-env

PYTHON-env [Info](#)

[Refresh](#)
[Actions](#)
[Upload and deploy](#)

Environment overview

Health	Environment ID
Unknown	e-2mu2g6icky
Domain	Application name
-	PYTHON

Platform [Change version](#)

Platform	Platform state
Python 3.11 running on 64bit Amazon Linux 2023/4.0.8	Supported
Running version	
-	

24. Now you can go to EC2 from my service you can see The instance is running.

Instances (4) Info

Q Find Instance by attribute or tag (case-sensitive)

Any state

Instance state = running

Clear filters

Refresh

Connect

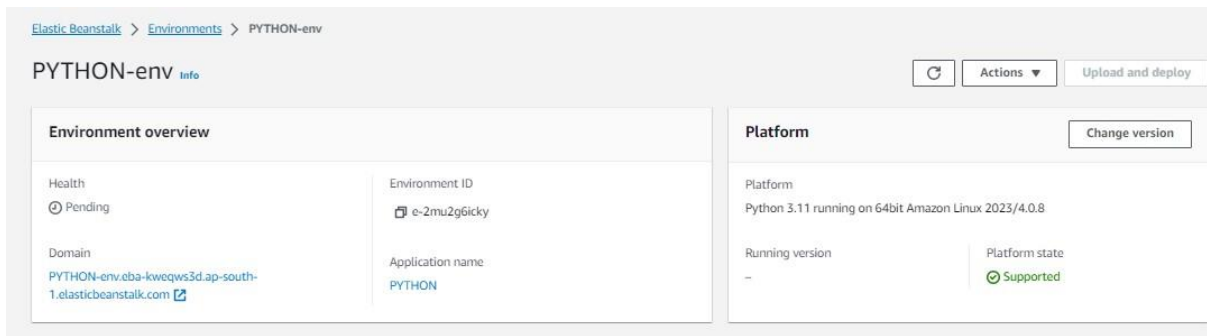
Instance state

Actions

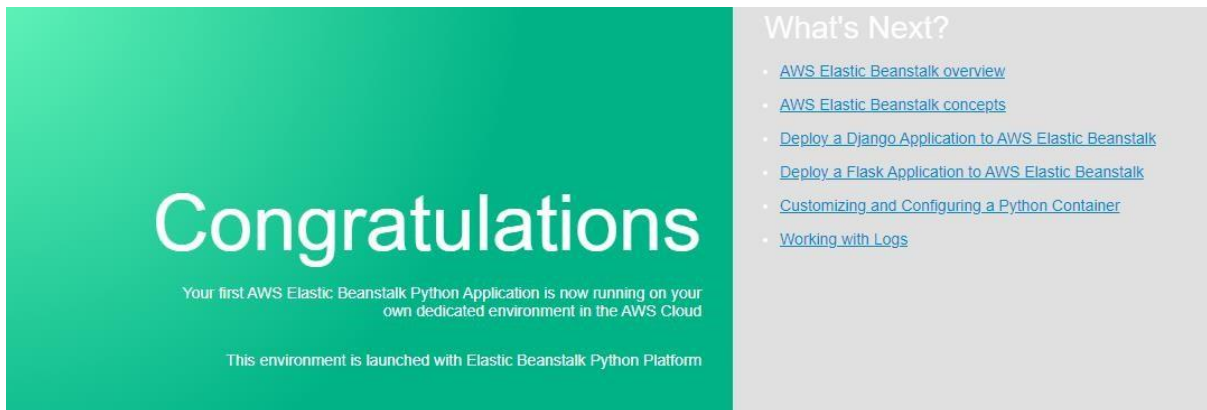
Launch instances

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
<input type="checkbox"/>	Mywebapp-env	i-0be4f376697dcb5c4	Running	t3.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-13-234-183-3.ap-s...	13.234.183.3	13.234.183.3
<input type="checkbox"/>	Arisha	i-0dfbc5d95c1a61fd6	Running	t3.large	2/2 checks passed	View alarms +	ap-south-1b	ec2-65-2-112-232.ap-s...	65.2.112.232	65.2.112.232
<input type="checkbox"/>	JAVA-env	i-02381801472b48cea	Running	t3.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-43-204-156-218.ap...	43.204.156.218	43.204.156.218
<input type="checkbox"/>	PYTHON-env	i-0458f92852efb5a91	Running	t3.micro	Initializing	View alarms +	ap-south-1b	-	-	-

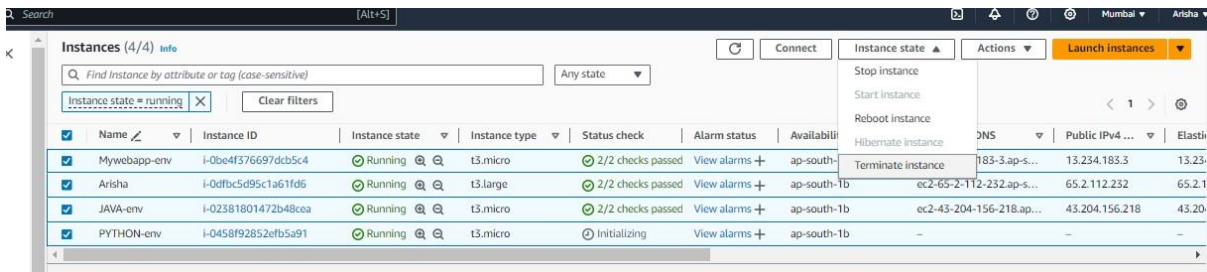
25. Click on domain



26. It will get open into another browser now Elastic Beanstalk Server is running on your own dedicated environment.

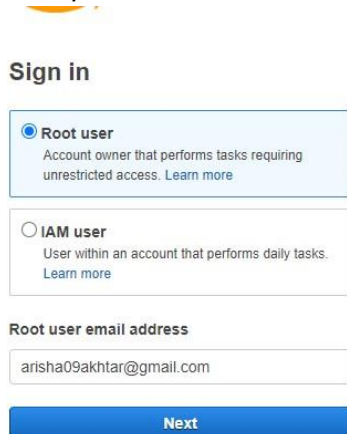


27. Last step don't forget to terminate the instances that have been created

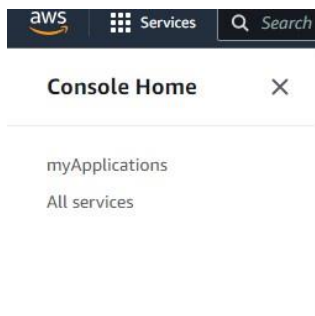


For Tomcat:

1.) Sign In to your aws acc



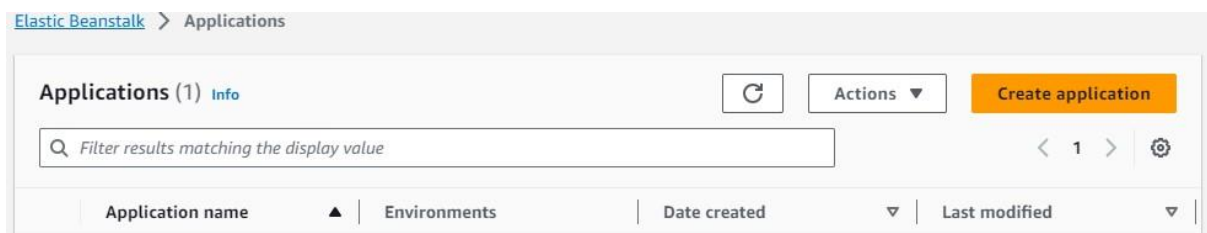
2.)Select all services



3.)Select Elastic Beanstalk



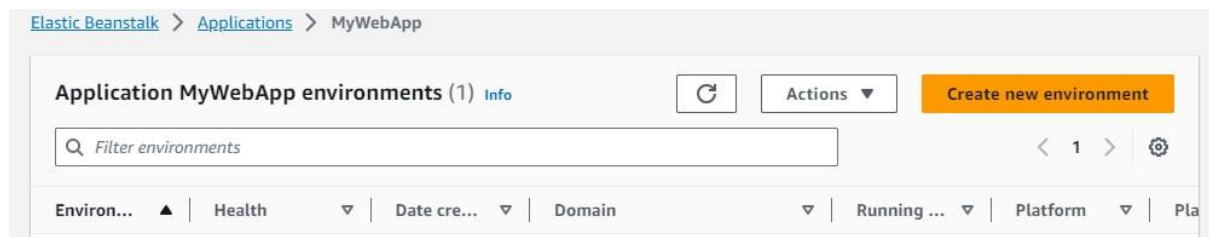
6.) Click on Create Application



7.) Enter application name and description and then click on create

The screenshot shows the 'Create new application' form in the AWS Elastic Beanstalk console. The breadcrumb navigation at the top reads 'Elastic Beanstalk > Create application'. The main heading is 'Create new application Info'. The form is divided into two sections: 'Application information' and 'Tags'. In the 'Application information' section, there is a text input field for 'Application name' with the value 'Tomcat' and a note 'Maximum length of 100 characters.' Below it is a text area for 'Description'. In the 'Tags' section, there is 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. Learn more' and a button 'Add new tag'.

8. Now create environment



7. For Configure environment everything will be default except platform since we are doing for tomcat we will choose tomcat Platform branch and platform version will be default. Click on next.

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
Tomcat
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
Tomcat-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.

Domain
Leave blank for autogenerated value .ap-south-1.elasticbeanstalk.com [Check 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
Tomcat

Platform branch
Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023

Platform version
5.1.3 (Recommended)

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.

Version label
Unique name for this version of your application code.
code

Source code origin. Maximum size 500 MB

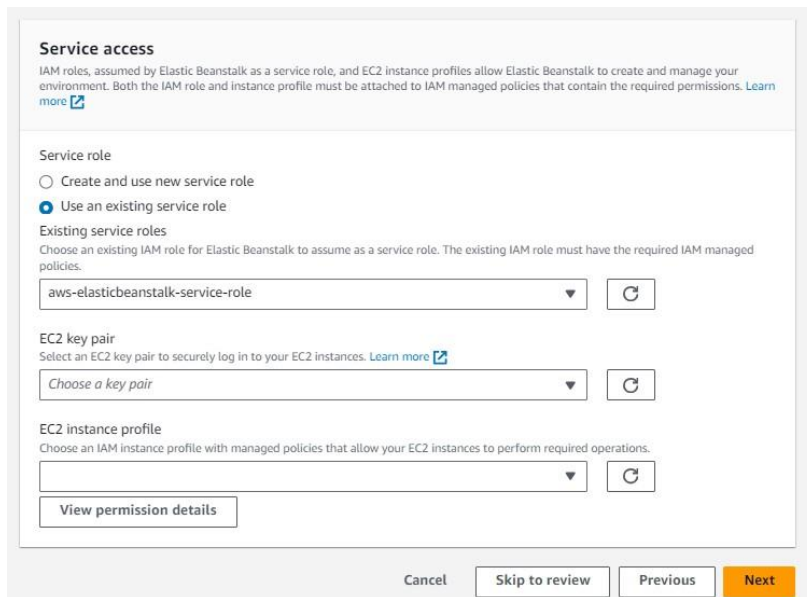
☒ **Local file**

Upload application
[Choose file](#)

File name: **Calendar.war**
File must be less than 500MB max file size

☐ **Public S3 URL**

8.) For Service access you have to create a role since it is by default over here you can create your own role .



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.

EC2 key pair

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

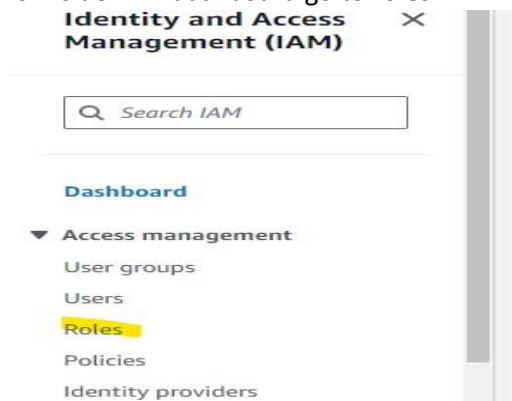
EC2 instance profile

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

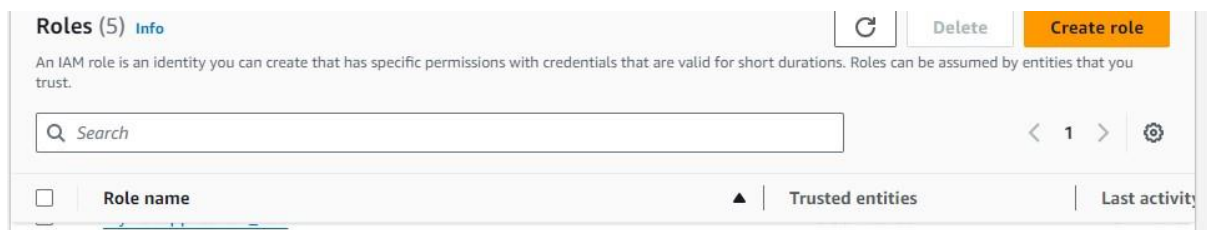
9.)To create role go to services IAM



10. Inside IAM dashboard go to roles



11. Create role



Roles (5) [Info](#)

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

<input type="checkbox"/>	Role name	Trusted entities	Last activity
--------------------------	-----------	------------------	---------------

12. It will ask for entity select AWS Service and for use case select EC2 and click on next

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 account 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.

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

☐ **SAML 2.0 federation**
Allow users federated with SAML 2 corporate directory 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 ▼

Choose a use case for the specified service.

☒ **EC2**

<input type="checkbox"/>			<u>AWSElasticBeanstalkEnhancedHealth</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkManagedUpdatesCustomer</u>
			<u>AWSElasticBeanstalkMulticontainerDocker</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkReadOnly</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleCore</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleCWL</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleECS</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleRDS</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleSNS</u>
<input type="checkbox"/>			<u>AWSElasticBeanstalkRoleWorkerTier</u>
			<u>AWSElasticBeanstalkWebTier</u>
			<u>AWSElasticBeanstalkWorkerTier</u>

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

PYTHON

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 alphanumeric and '+=, @_-' characters.

17. For instance profile click on dropdown your role name will come select that then enter next

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.

TomCat

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.

TomCat

18. After clicking on next it will ask for VPC click on dropdown and select which is default not down the ip address of this

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-08f10f70caf202076 | (172.31.0.0/16)

[Create custom VPC](#)

19. The ip address you have selected above same over here select that ip address and click on next

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

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

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 (3)

	Availability Zone	Subnet	CIDR	Name
<input type="checkbox"/>	ap-south-1a	subnet-0099214b...	172.31.32.0/20	
<input checked="" type="checkbox"/>	ap-south-1b	subnet-058970f16...	172.31.0.0/20	
<input type="checkbox"/>	ap-south-1c	subnet-07fe5d047...	172.31.16.0/20	

20.for configure instance everything will be by default enter next

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

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.
 IOPS

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

Amazon CloudWatch monitoring
The time interval between when metrics are reported from the EC2 instances.

Monitoring interval

Instance metadata service (IMDS)
Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#)

IMDSv1

21.For configure update also everthing will be default click to next

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

22.Review will display click on submit

Review [Info](#)

Step 1: Configure environment [Edit](#)

Environment information

Environment tier

Web server environment

Application name

PYTHON

Environment name

PYTHON-env

Application code

Sample application

Platform

arn:aws:elasticbeanstalk:ap-south-1::platform/Python 3.11 running on 64bit Amazon Linux 2023/4.0.8

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.

Service role

arn:aws:iam::836706723086:role/PYTHON

EC2 instance profile

PYTHON

23. It will get open into another browser now Elastic Beanstalk Server is running on your own dedicated environment.

.

GWT Calendar

Click on day to get date popup. Example Datepicker. Built with the tomcat war builder.
<http://code.google.com/p/gwt-examples/>

< January >

< 2024 >

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

.