Practical-2 Platform as a service using AWS.

NAME: Sania Rodrigues
ROLL.NO: A008

Writeup:-

• PLATFORM AS A SERVICE

Platform as a Service (PaaS) is a complete cloud environment that includes everything developers need to build, run, and manage applications—from servers and operating systems to all the networking, storage, middleware, tools, and more.

How does PaaS work?

Unlike IaaS or SaaS service models, PaaS solutions are specific to application and software development and typically include:

Cloud infrastructure: Data centers, storage, network equipment, and servers Middleware software: Operating systems, frameworks, development kits (SDK), libraries, and more

User interface: A graphical user interface (GUI), a command line interface (CLI), an API interface, and in some cases, all three

Benefits of PaaS

- Faster time to market
- Low maintenance
- Easy scalability
- Flexible access
- Cost-effective pricing

• ELASTIC BEANSTALK

Elastic Beanstalk is a service for deploying and scaling web applications and services. Upload your code and Elastic Beanstalk automatically handles the deployment—from capacity provisioning, load balancing, and auto scaling to application health monitoring.

Use cases

Quickly launch web applications

Deploy scalable web applications in minutes without the complexity of provisioning and managing underlying infrastructure.

Create mobile API backends for your applications

Use your favorite programming language to build mobile API backends, and Elastic Beanstalk will manage patches and updates.

Replatform critical business applications

Migrate stateful applications off legacy infrastructure to Elastic Beanstalk and connect securely to your private network.

Platforms for Programming Languages Provided By Elastic Beanstalk are

➢ GO

➢ Java

➢ Node.js

➢ PHP

➢ Python

➢ Ruby

Platforms for Application Servers Provided by Elastic Beanstalk are

➢ Tomcat

➢ Docker

• COMPONENTS OF BEANSTALK

AWS Elastic Beanstalk Components

1. Application Handling:

Elastic Beanstalk adopts the project code directly, naming the application after the project's home directory.

2. Application Environments:

Supports multiple environments (e.g., DEV, UAT, PROD) for running applications at different stages.

3. Automated Health Checks:

AWS conducts automatic health checks on Elastic Beanstalk applications, monitoring EC2 deployments.

4. Health status indicators: Red (failure), Yellow (partial failure), Grey (updating), Green (success), Isolated (environments and applications are isolated).

5. Scalability and Load Balancing:

Utilizes Auto-Scaling for dynamic application scalability.

Elastic Load Balancer (ELB) balances web request loads across application instances. 6. Language Support:

Supports Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker applications on familiar servers.

7. Pricing:

No additional charges for Elastic Beanstalk; users pay for services and resources provisioned by the service.

8. Automatic Provisioning:

Relieves users from selecting services and configuring security groups; handles automatic provisioning.

9. Scalability Assurance:

Leverages Auto Scaling, theoretically capable of handling any amount of internet traffic, as claimed by AWS.

• IAM

Identity and access management (IAM) is a framework of business processes, policies and technologies that facilitates the management of electronic or digital identities. With an IAM framework in place, information technology (IT) managers can control user access to critical information within their organizations. Systems used for IAM include single sign-on systems, two-factor authentication, multifactor authentication and privileged access management.

IAM systems can be deployed on premises, provided by a third-party vendor through a cloud based subscription model or deployed in a hybrid model.

On a fundamental level, IAM encompasses the following components:

how individuals are identified in a system (understand the difference between identity management and authentication);

how roles are identified in a system and how they are assigned to individuals; adding, removing and updating individuals and their roles in a system;

assigning levels of access to individuals or groups of individuals; and

protecting the sensitive data within the system and securing the system itself. IAM Features: Brief Overview

➢ Shared Access:

Facilitates easy resource sharing among project teams.

➢ Cost-Free Access:

IAM feature is free; charges incurred only when accessing other AWS services using IAM users.

➢ Centralized Control:

Provides centralized control over user and group creation, management, and data access within the AWS account.

➢ Permission Granting:

Root account, with administrative rights, grants specific permissions to IAM users for accessing services.

➢ Multifactor Authentication:

Enhances account security with a third-party six-digit code, required along with the password for account logins.

• Implement paas using elastic beanstalk for the following.

1. Server

2. Java

3. Python

4. Node.js

Beanstalk

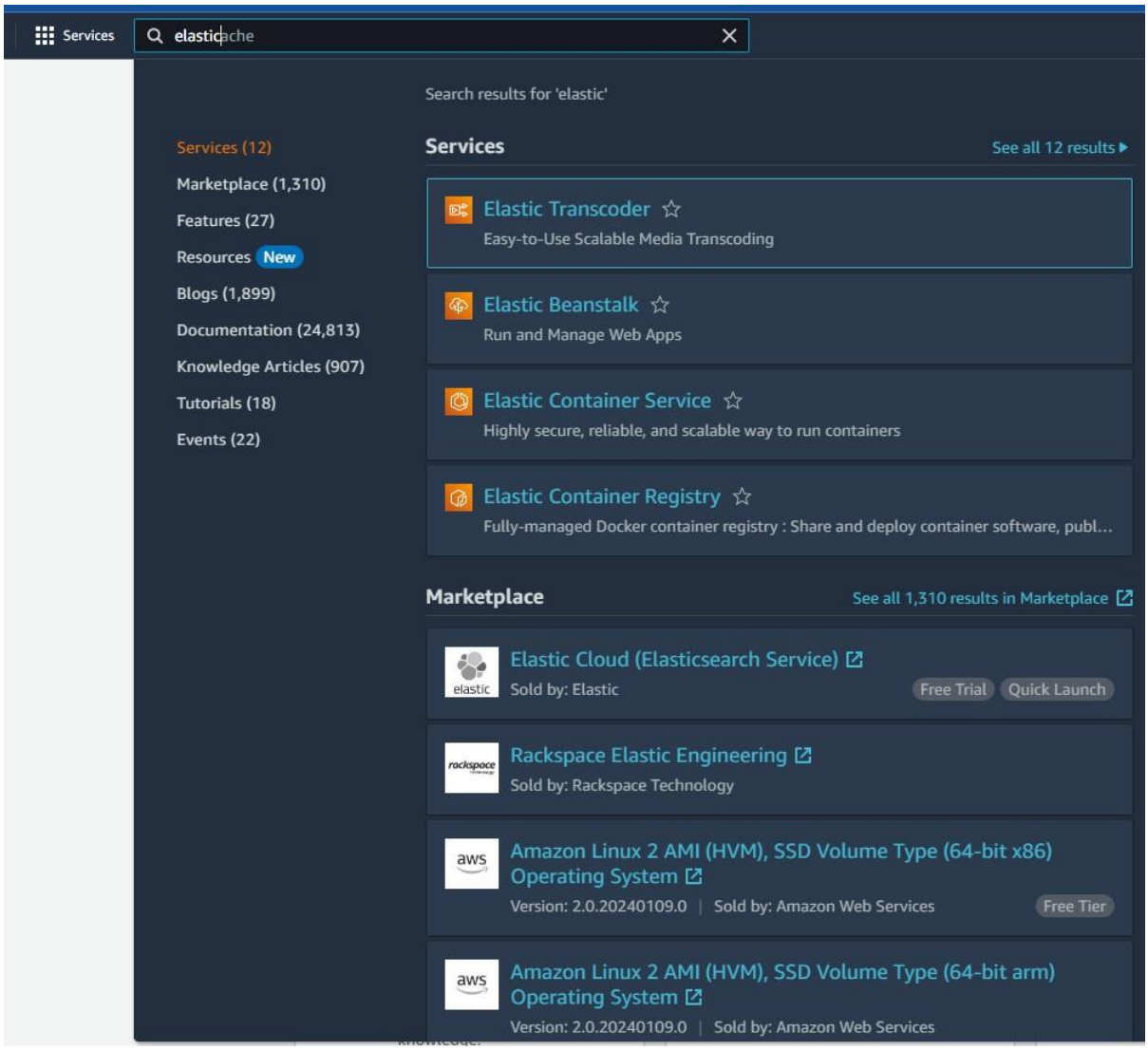
IAM(Identity Access Management)- Roles can be assigned with this

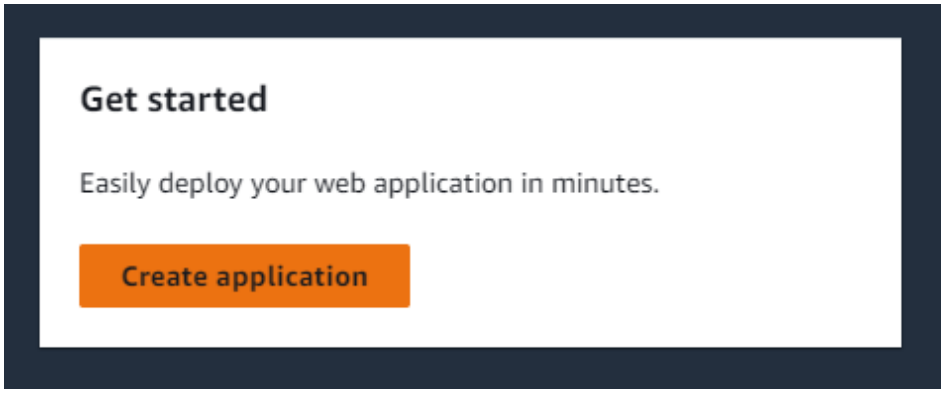
Creating an application

CREATE ENV

In Elastic Beanstalk

• EXECUTING APPLICATIONS
• UPLOADING APPLICATIONS


Services
elasticache
Search results for 'elastic'
Services (12)
Marketplace (1,310)
Features (27)
Resources New
Blogs (1,899)
Documentation (24,813)
Knowledge Articles (907)
Tutorials (18)
Events (22)
Services
See all 12 results
Elastic Transcoder
Easy-to-Use Scalable Media Transcoding
Elastic Beanstalk
Run and Manage Web Apps
Elastic Container Service
Highly secure, reliable, and scalable way to run containers
Elastic Container Registry
Fully-managed Docker container registry : Share and deploy container software, publ...
Marketplace
See all 1,310 results in Marketplace
Elastic Cloud (Elasticsearch Service)
Sold by: Elastic
Free Trial
Quick Launch
Rackspace Elastic Engineering
Sold by: Rackspace Technology
Amazon Linux 2 AMI (HVM), SSD Volume Type (64-bit x86) Operating System
Version: 2.0.20240109.0 | Sold by: Amazon Web Services
Free Tier
Amazon Linux 2 AMI (HVM), SSD Volume Type (64-bit arm) Operating System
Version: 2.0.20240109.0 | Sold by: Amazon Web Services


Get started
Easily deploy your web application in minutes.
Create application

Application information Info

Application name

Purva

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

Purva-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 .eu-north-1.elasticbeanstalk.com Check availability

Environment description

This is a app which will execute a java code

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

Java

Platform branch

Corretto 21 running on 64bit Amazon Linux 2023

Platform version

4.2.0 (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.

Version label

Source code origin. Maximum size 500 MB

- ( ) Local file
- (•) Public S3 URL

https://s3.us-east-1.amazonaws.com

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

Java

Platform branch

Corretto 21 running on 64bit Amazon Linux 2023

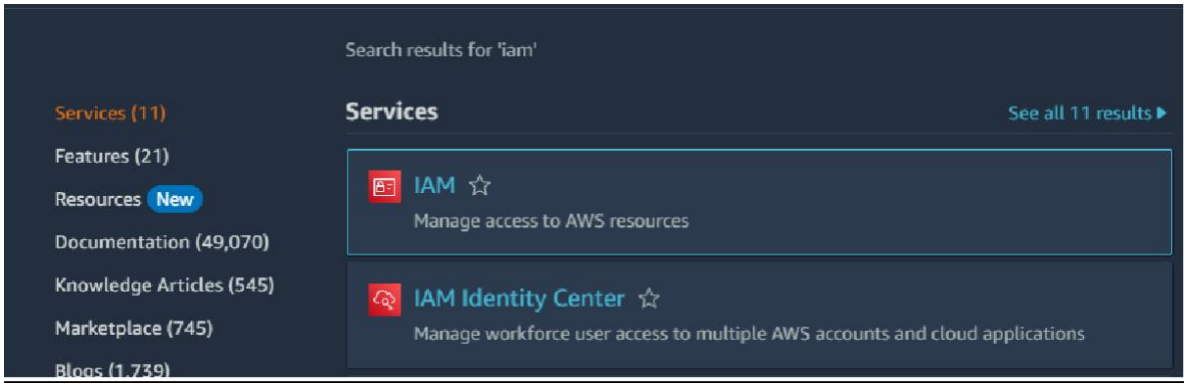
Platform version

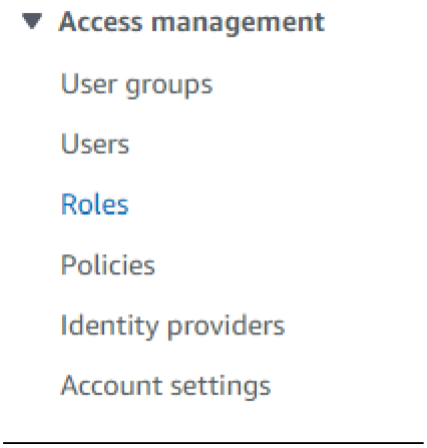
4.2.0 (Recommended)

Application code Info

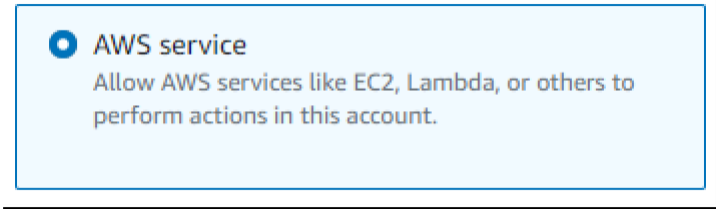
- (•) Sample application
- ( ) Existing version
  Application versions that you have uploaded.
- ( ) Upload your code
  Upload a source bundle from your computer or copy one from Amazon S3.

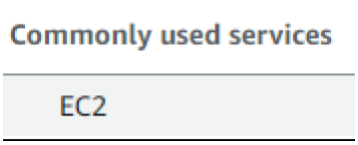
Creating a role for an application.

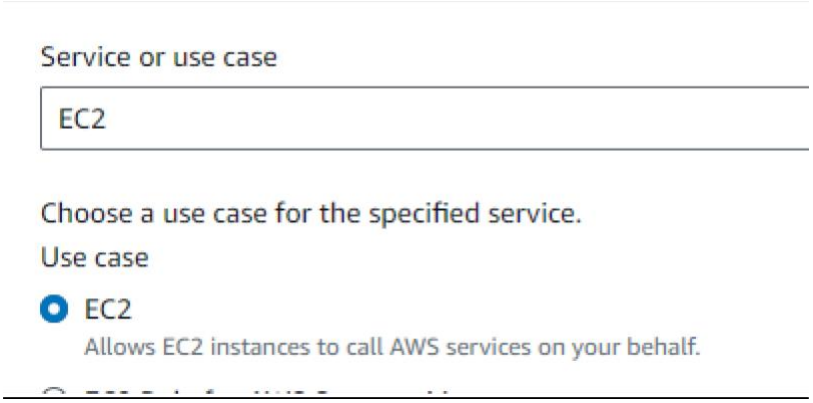



Access management
User groups
Users
Roles
Policies
Identity providers
Account settings

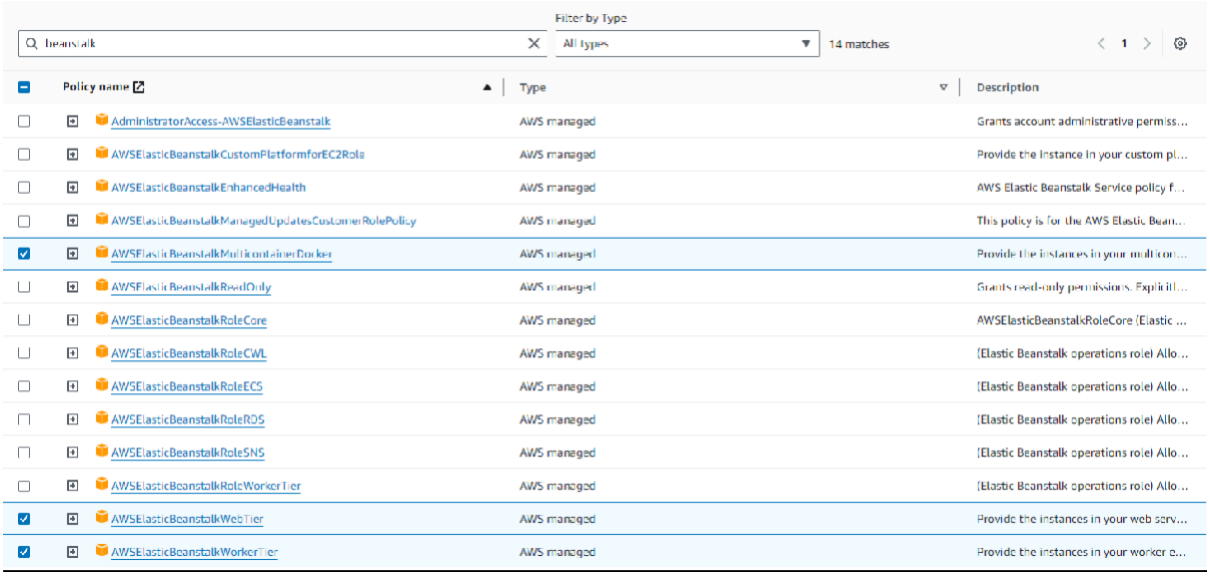

Create role


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


Commonly used services
EC2


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.

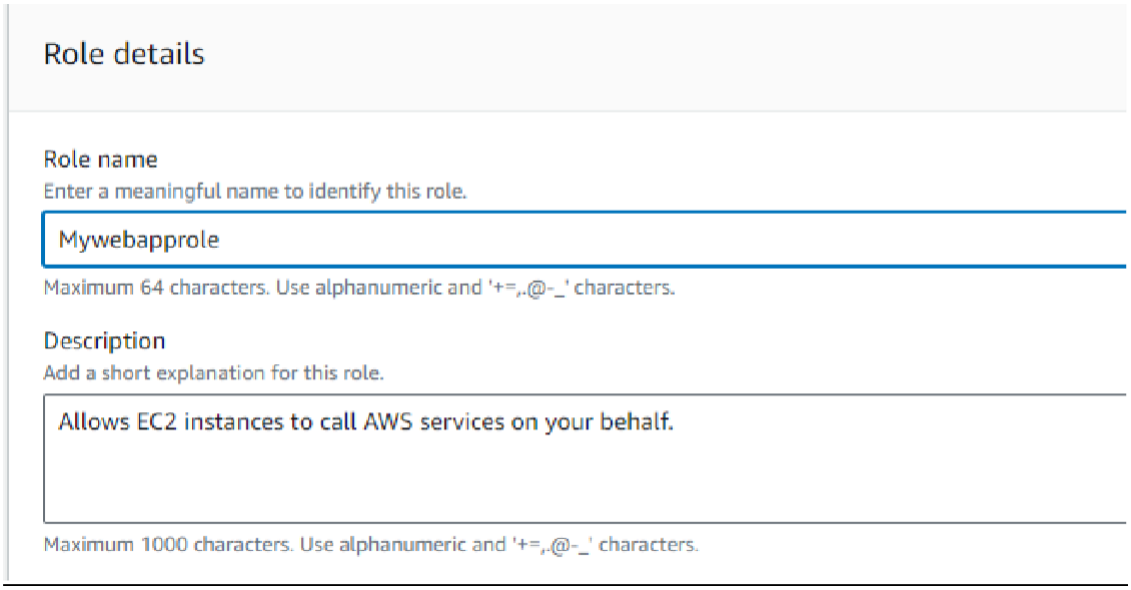
Click next

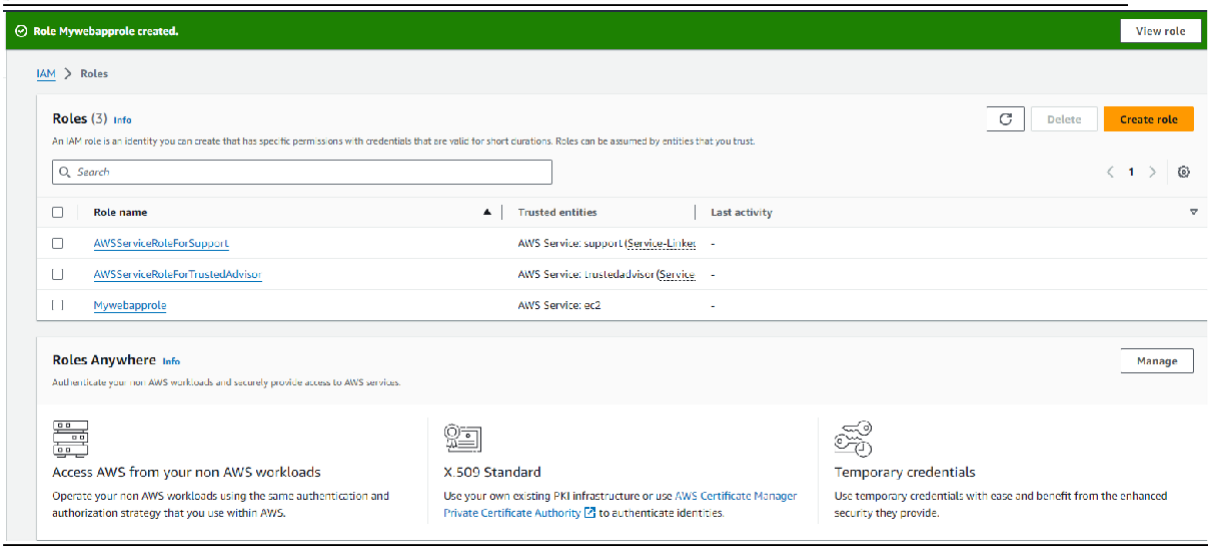


Worker tier-applications that you run

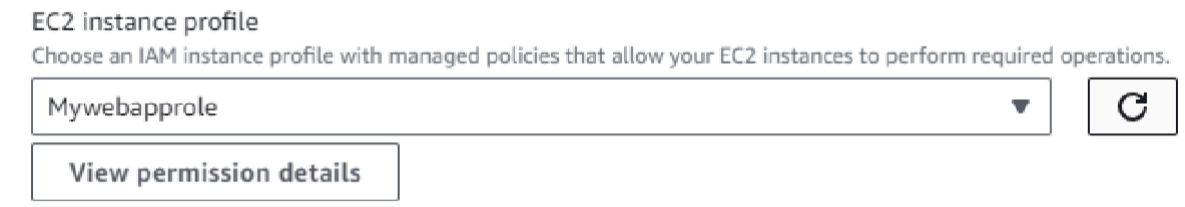
Multicontainer -webserver

Click next

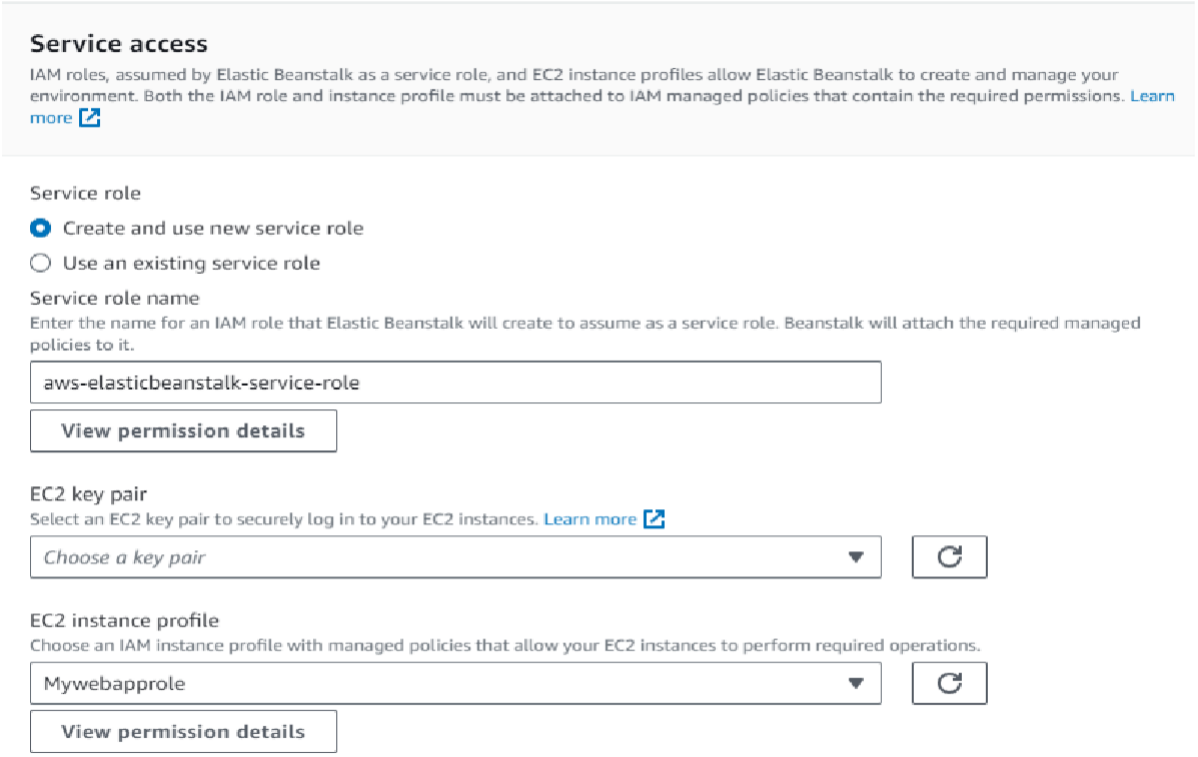




Go to the previous window

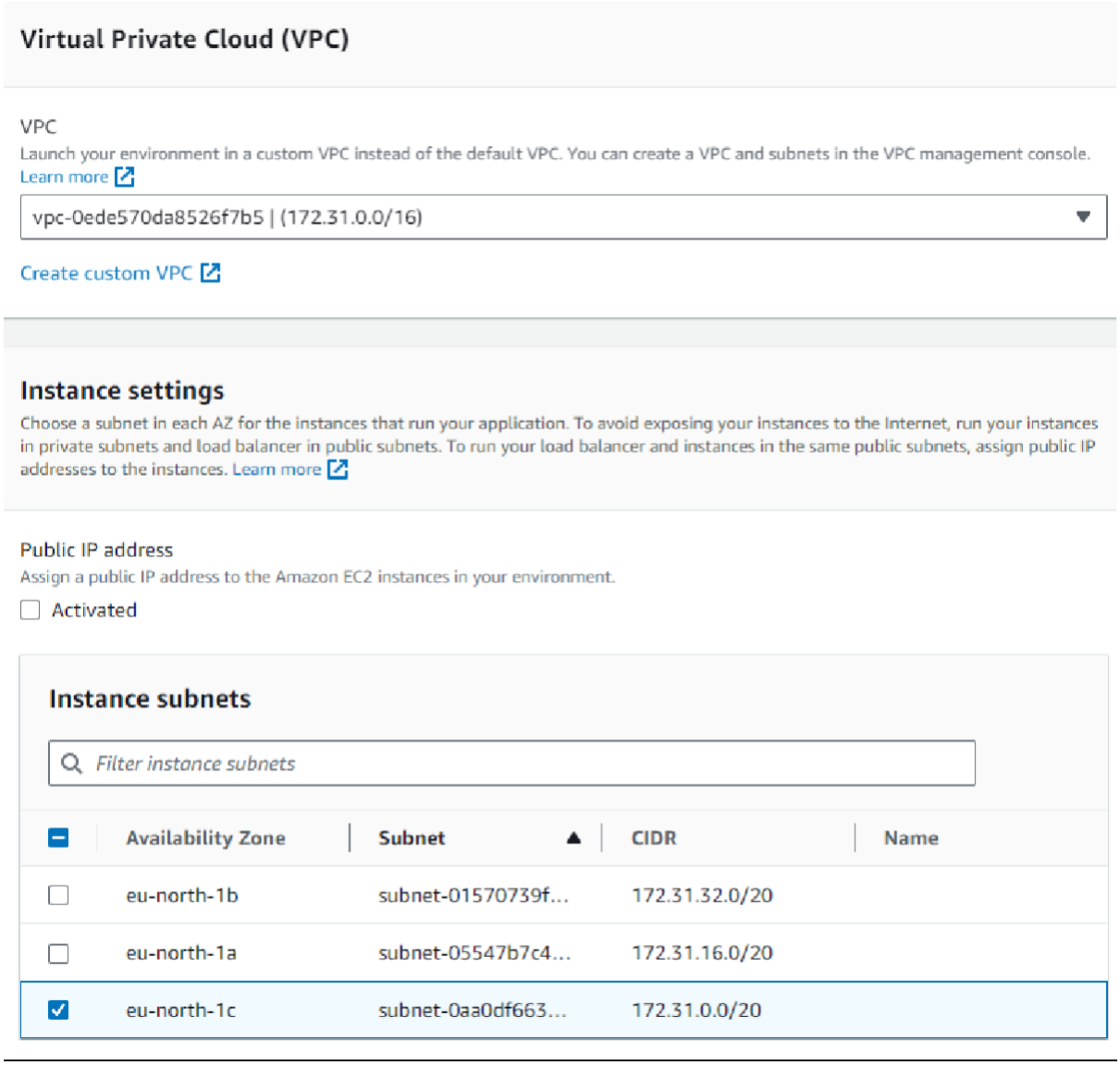


Click on create role

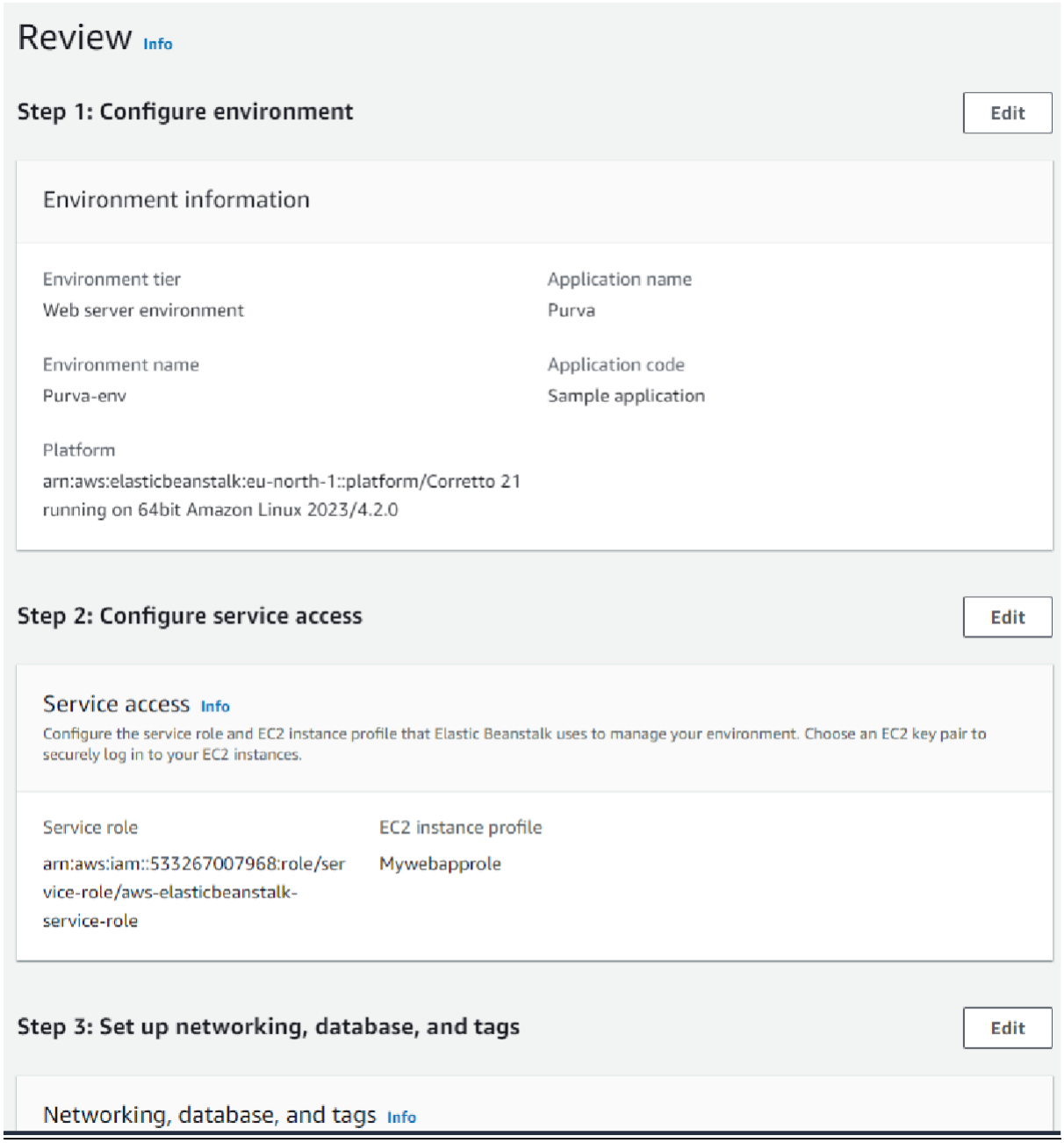


Refresh the instance profile cyclic button

Refresh the instance profile cyclic button

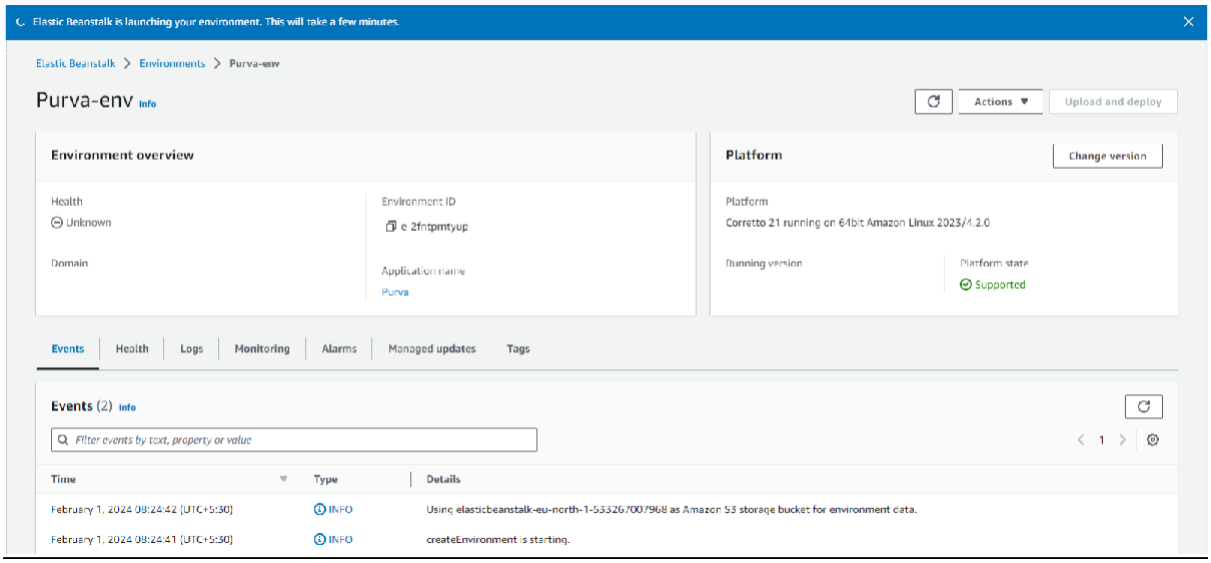


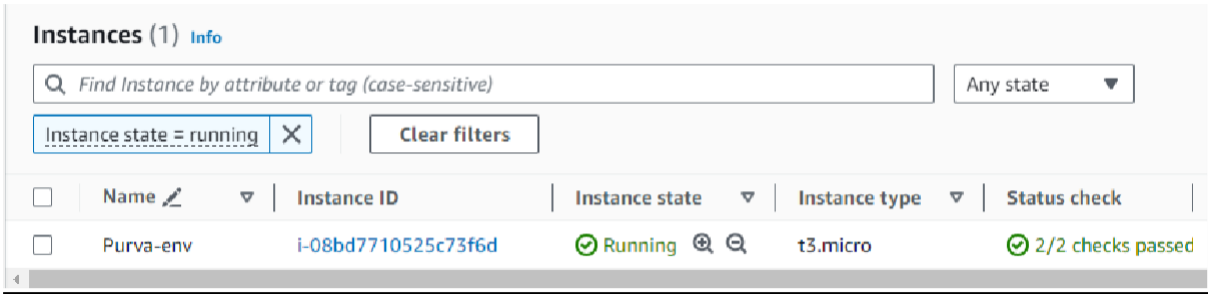
Three times next

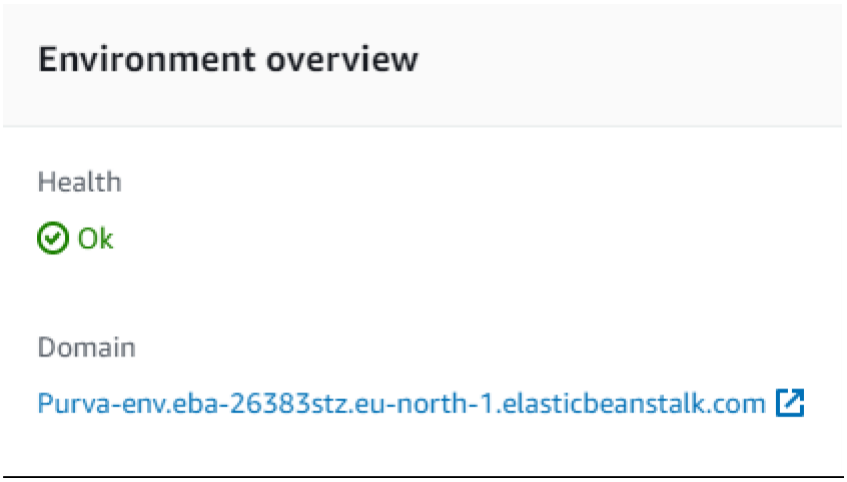


Click next next next

At last will get this window



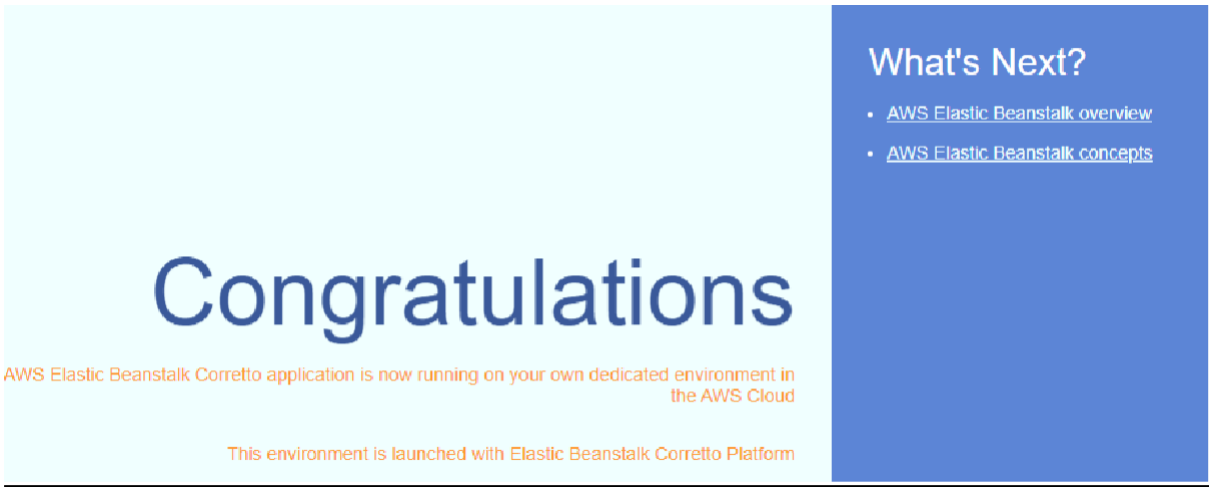




Click on domain url in environment overview

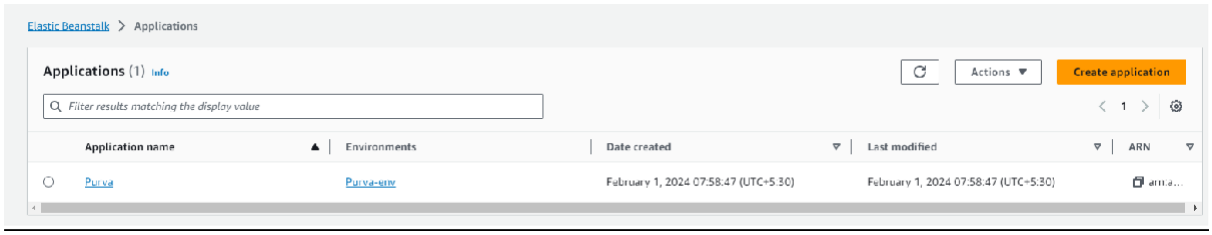
Click submit
Go to EC2 and check if running

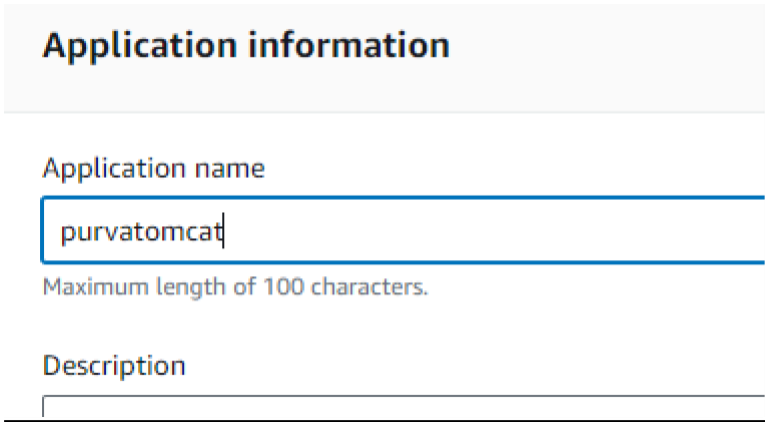
Should get this message

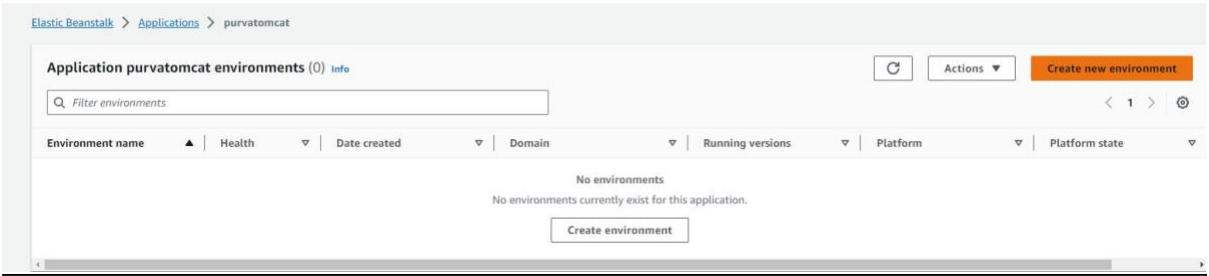


Creating TOMCAT server

Configure an elastic beanstalk in AWS
GO TO ELASTIC BEANSTALK HOME PAGE
GET started page
Create application







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

purvatomcat

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

Purvatomcat-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

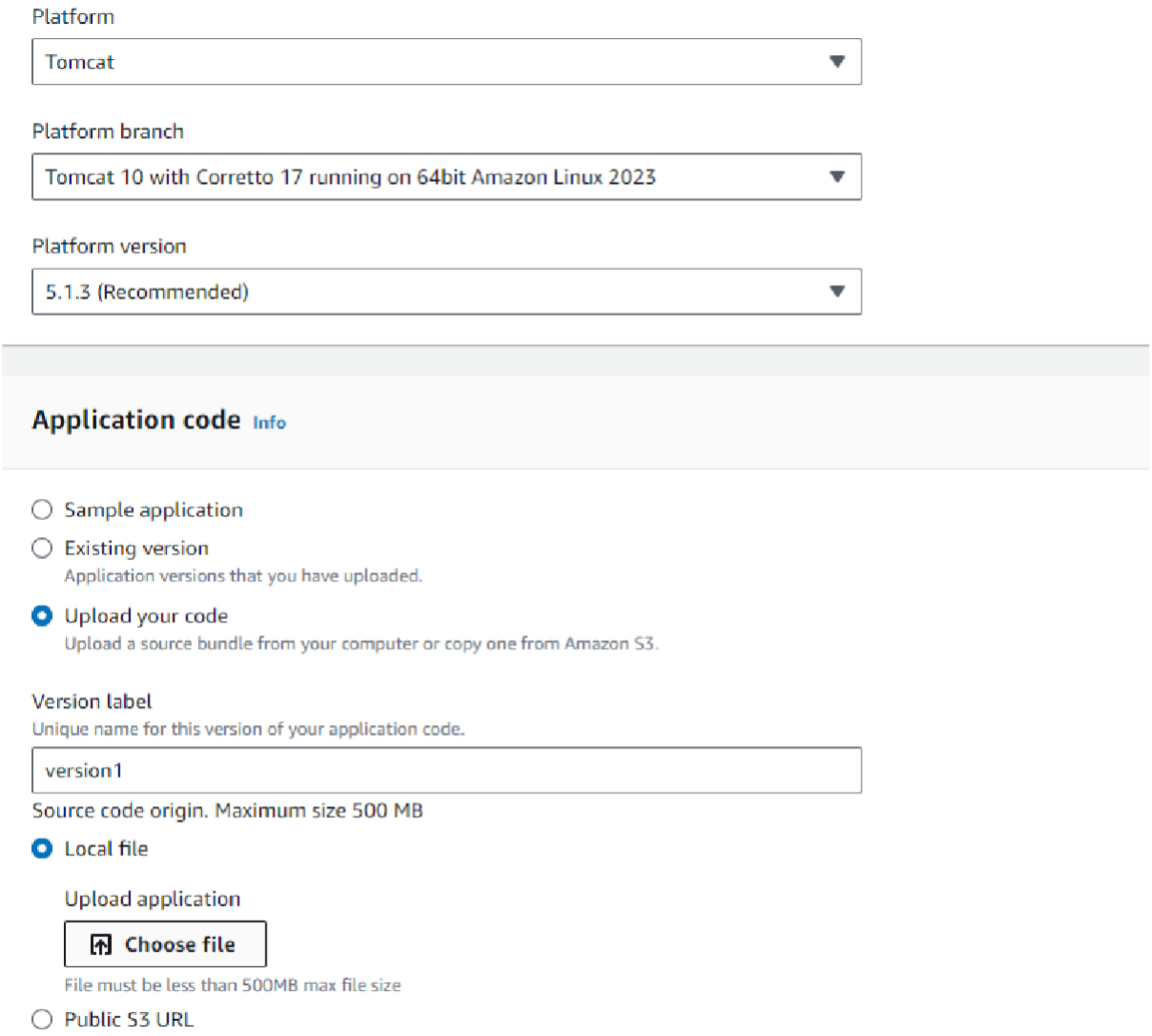
Leave blank for autogenerated value .eu-north-1.elasticbeanstalk.com Check availability

Environment description

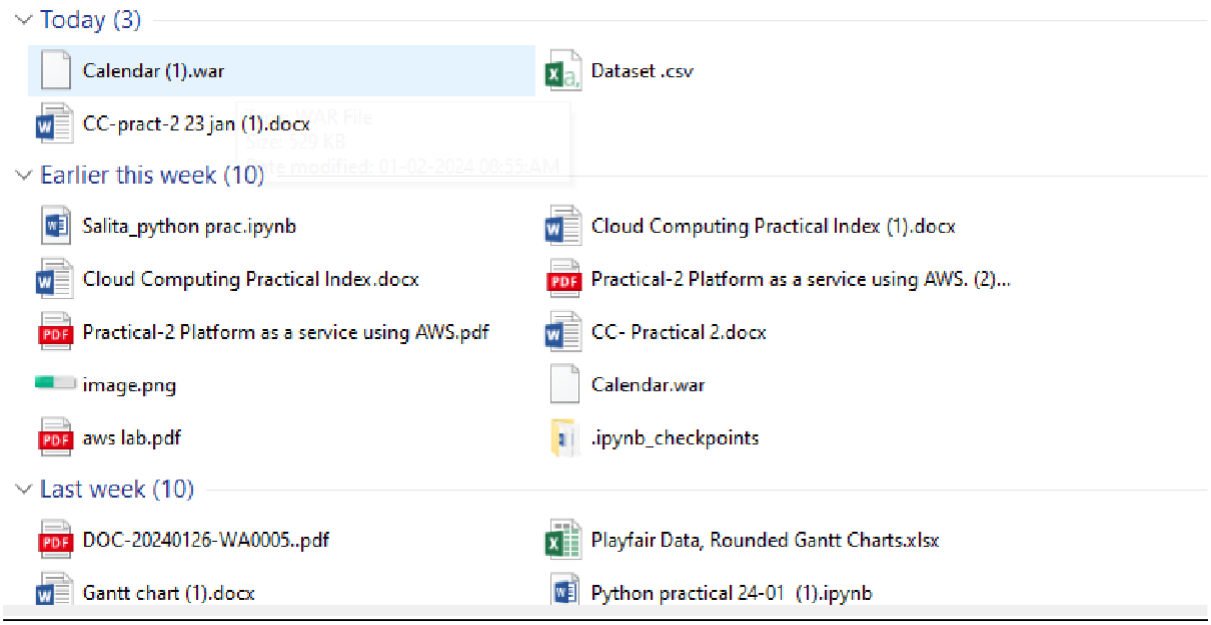
This is an app which will execute tomcat

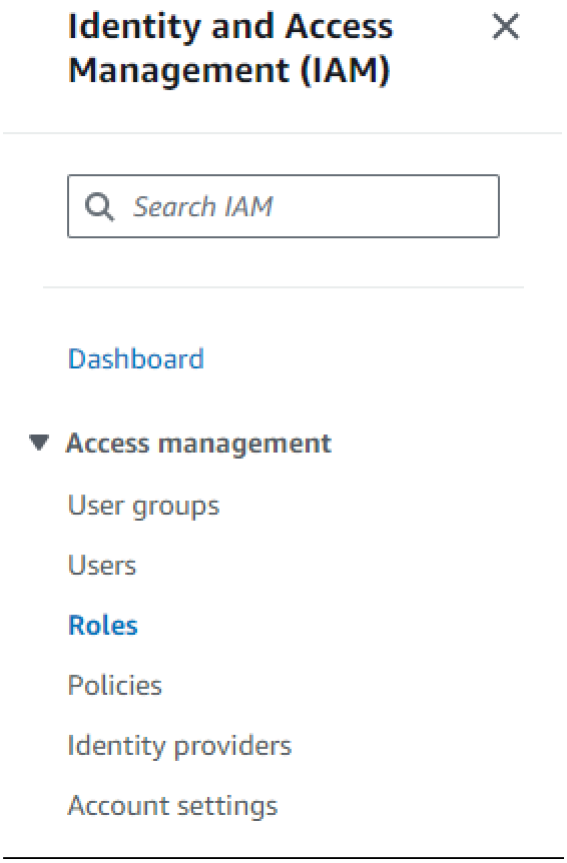
From the web download calendar.war file from github https://github.com/manulachathurika/Apache_Stratos_Tomcat_Applications/blob/master/Calendar.war

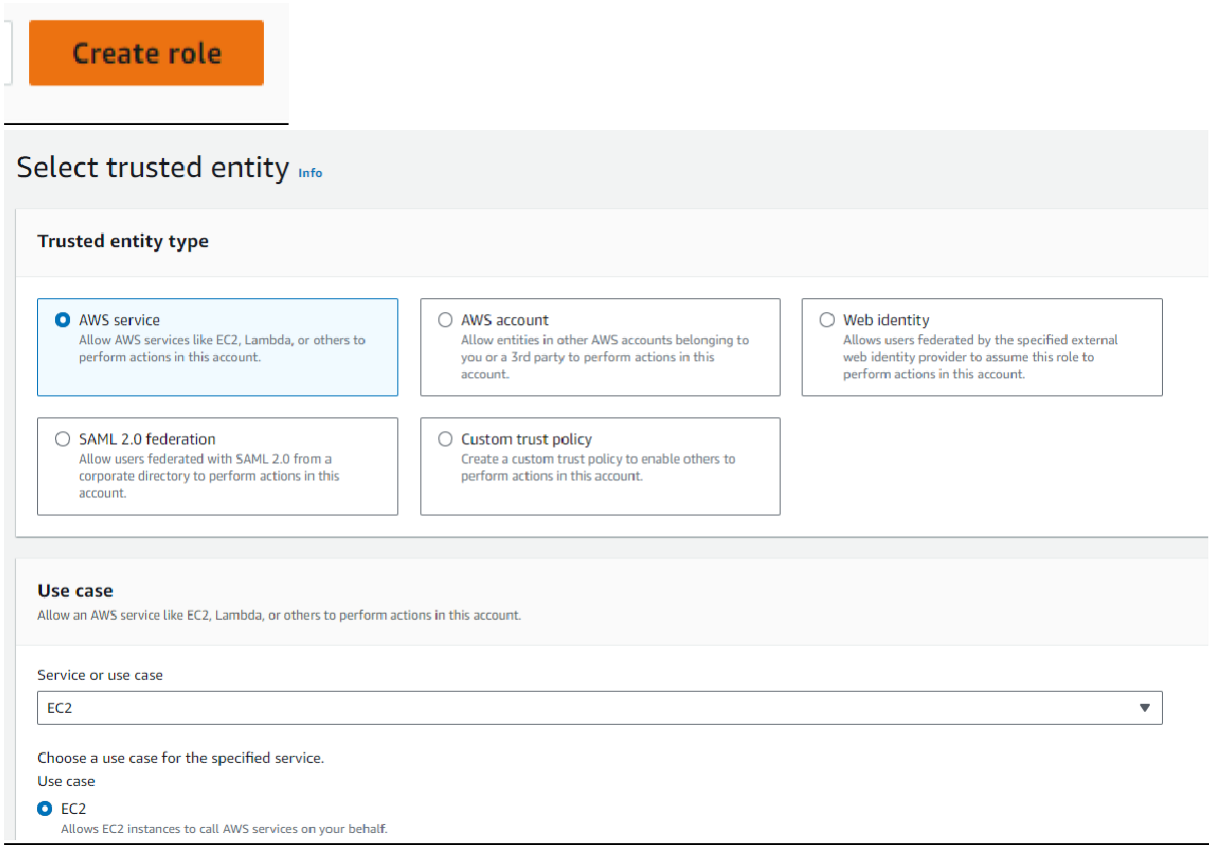
Select local file from option and choose file from your device

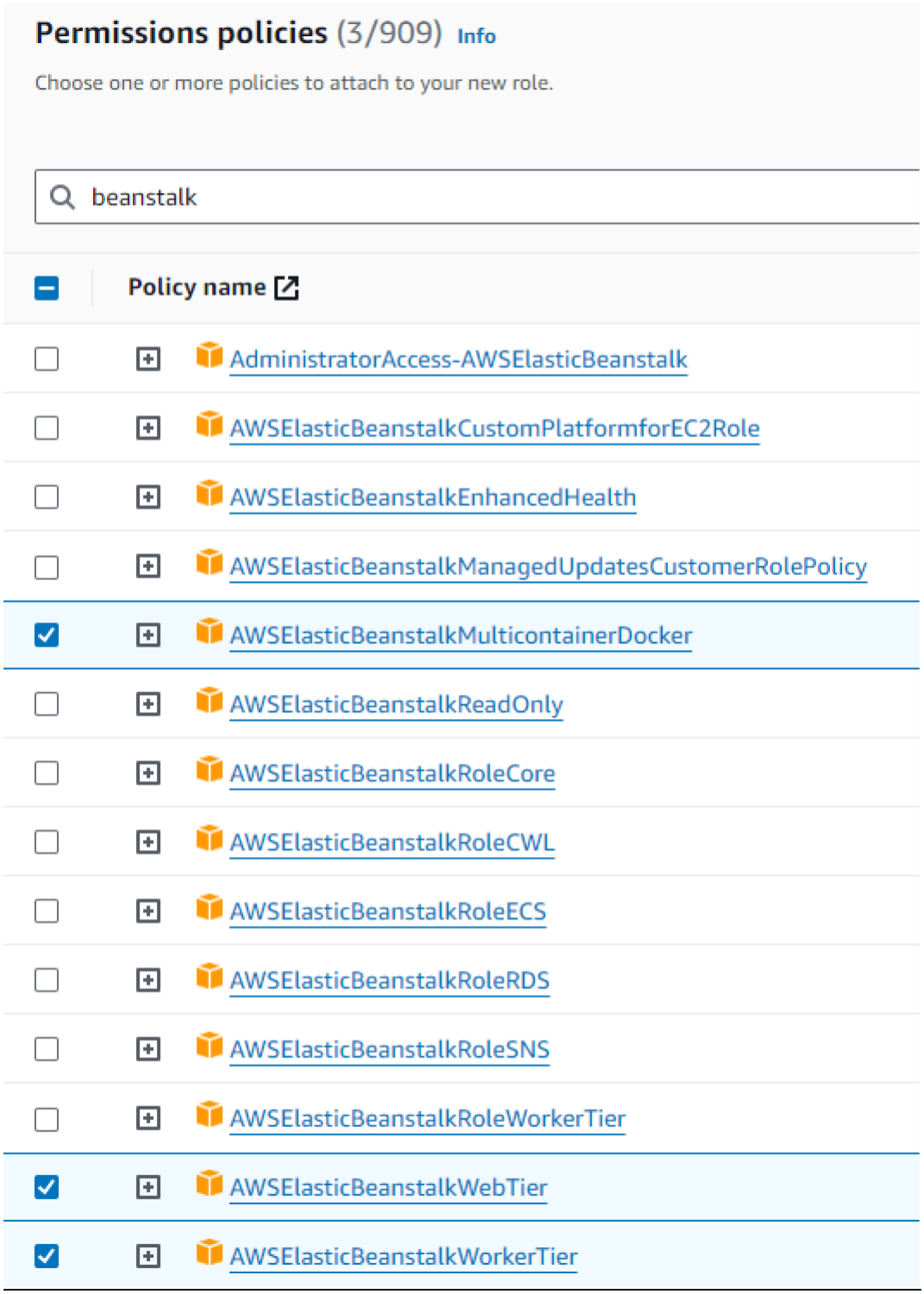


Click next
Go to IAM – Roles- Create new role
Use case EC2

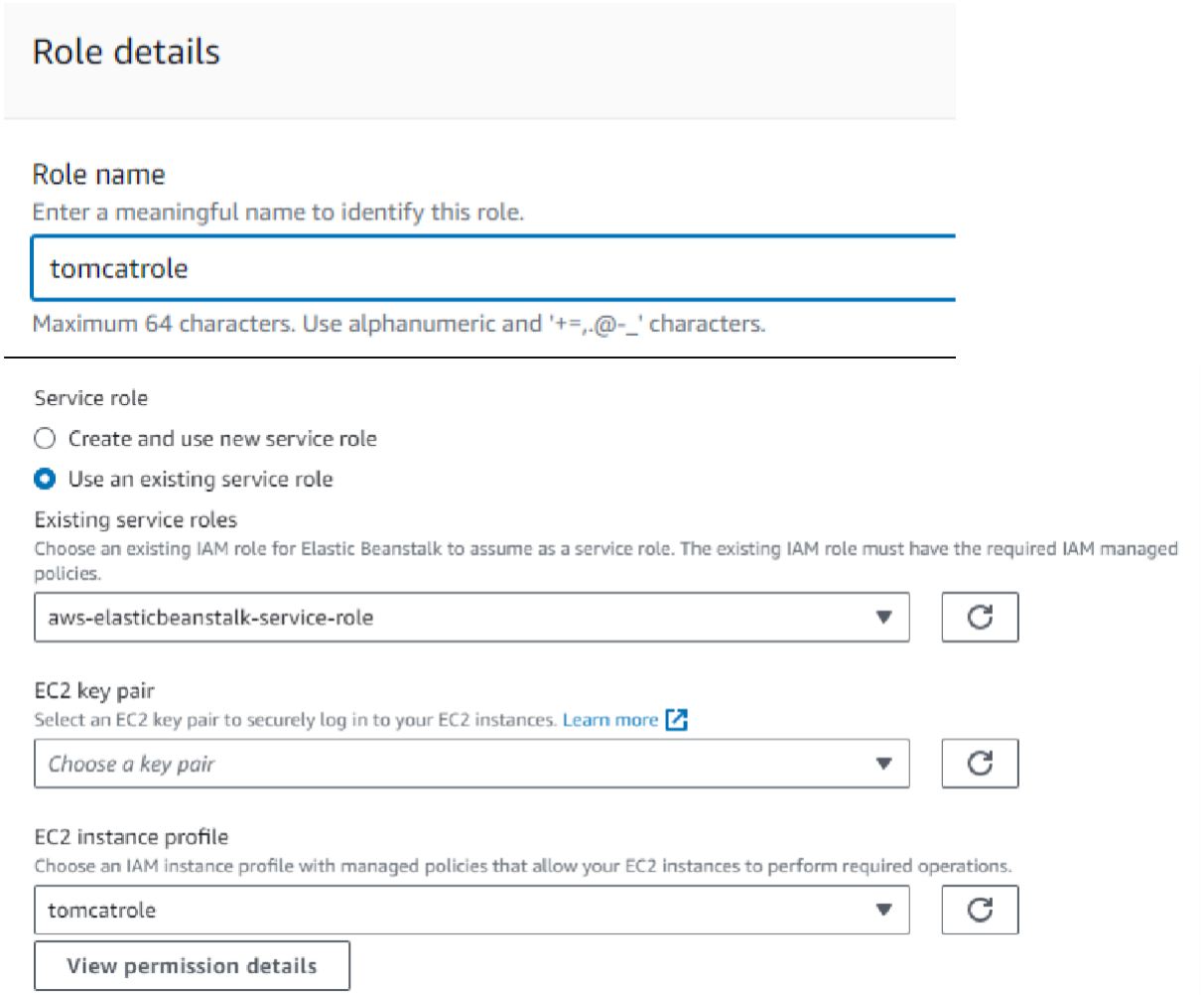

Today (3)
Calendar (1).war
Dataset .csv
CC-pract-2 23 jan (1).docx
Earlier this week (10)
Salita_python prac.ipynb
Cloud Computing Practical Index (1).docx
Cloud Computing Practical Index.docx
Practical-2 Platform as a service using AWS. (2)...
Practical-2 Platform as a service using AWS.pdf
CC- Practical 2.docx
image.png
Calendar.war
aws lab.pdf
.ipynb_checkpoints
Last week (10)
DOC-20240126-WA0005..pdf
Playfair Data, Rounded Gantt Charts.xlsx
Gantt chart (1).docx
Python practical 24-01 (1).ipynb


Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings


Create role
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
Choose a use case for the specified service.
Use case
EC2
Allows EC2 instances to call AWS services on your behalf.

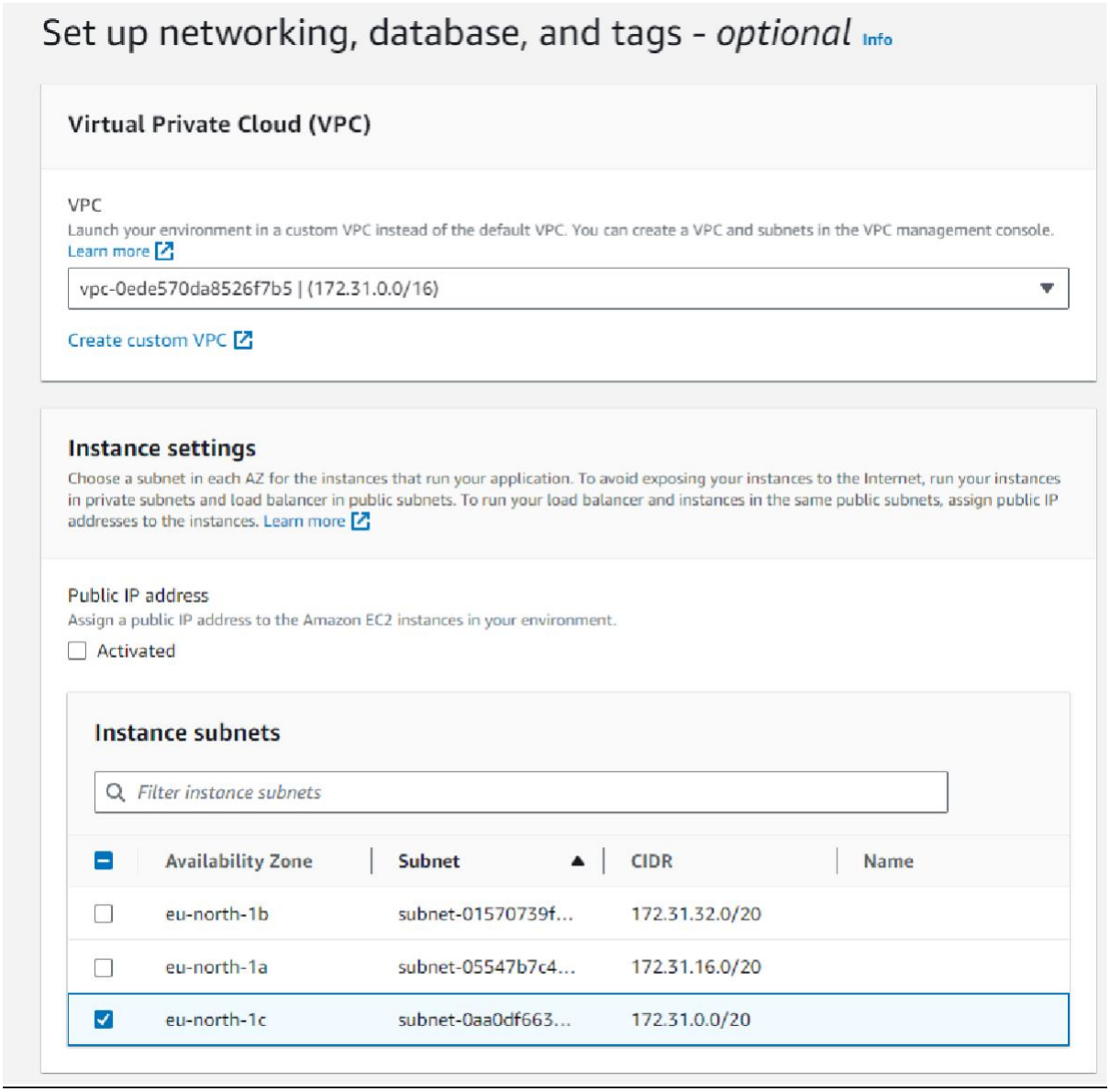


Give role name



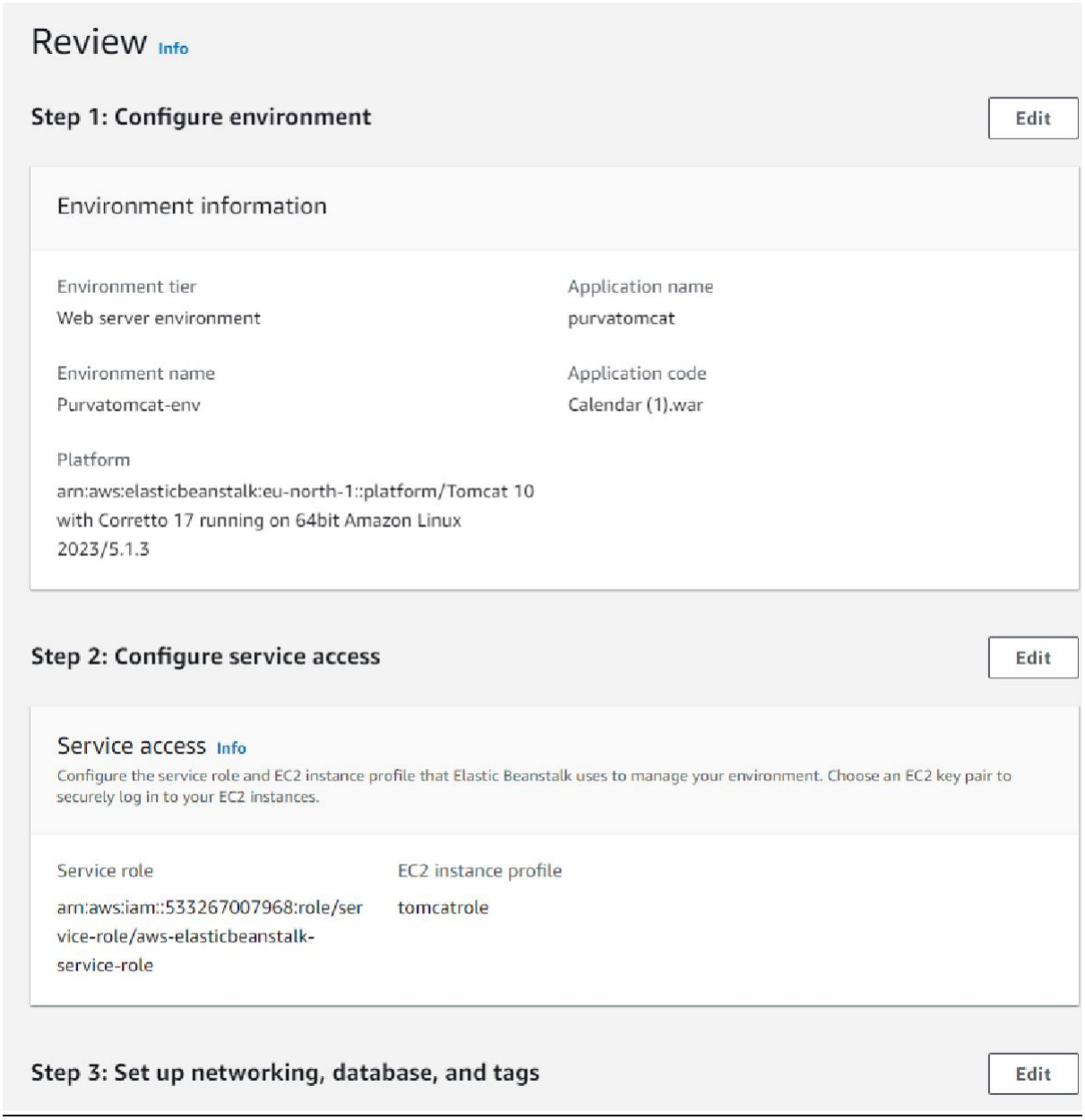
In configure services refresh instance profile button

In virtual private cloud select



Now 3 times next-> the submit

After launched, click on the domain name



Must get this

GWT Calendar

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

<	February	>	<	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		