SVKM'S NMIM'S Nilkamal School of Mathematics, Applied Statistics & Analytics

Master of Science (Data Science)

Practical-2 Platform as a service using AWS.

NAME: SANCIA FERNANDES

ROLL.NO: A012

Date:-23/01/2024 **Submission Date:- 30/01/2024**

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 cloudbased 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

Q1: CREATE VIEW

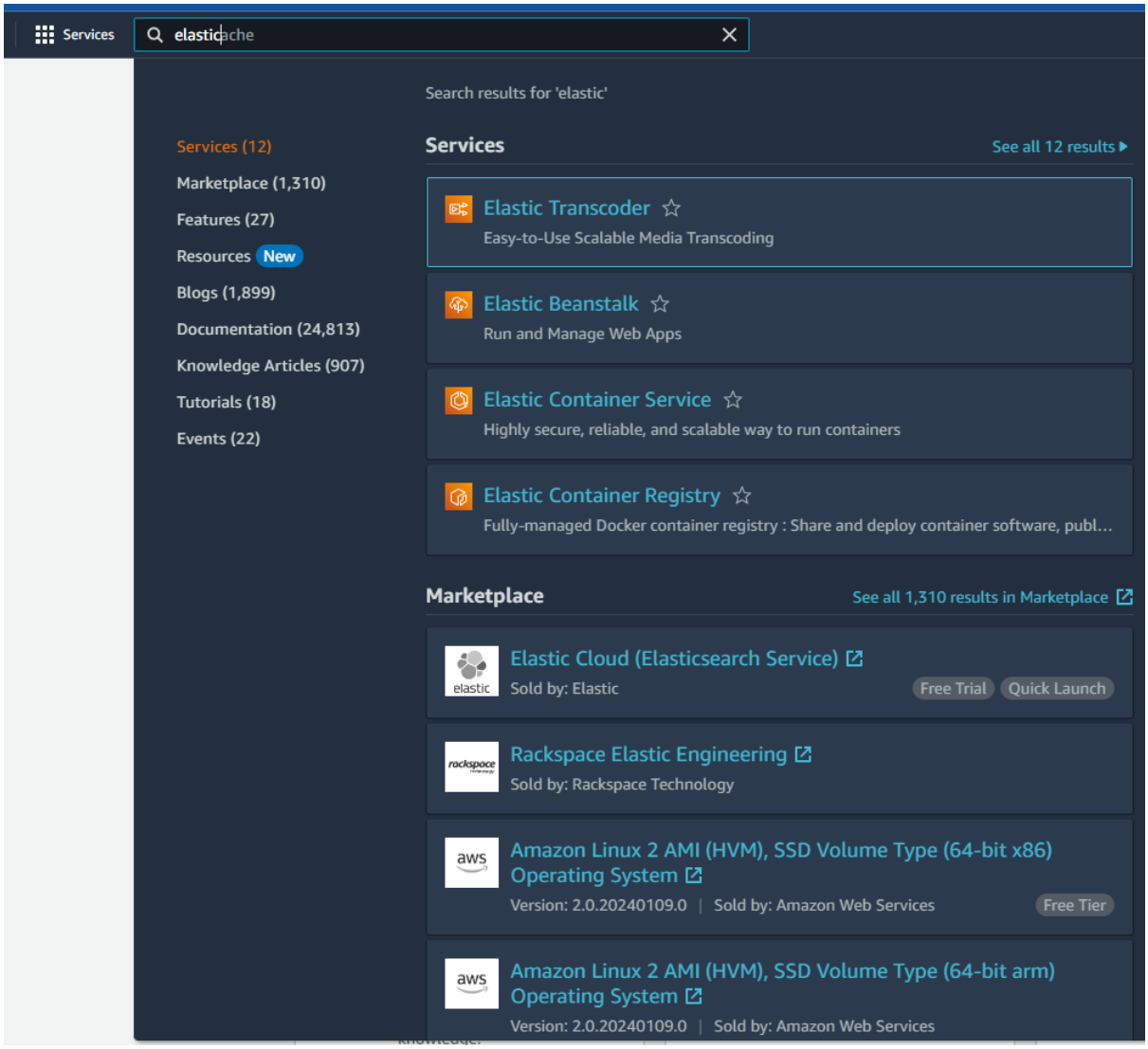
In Elastic Beanstalk

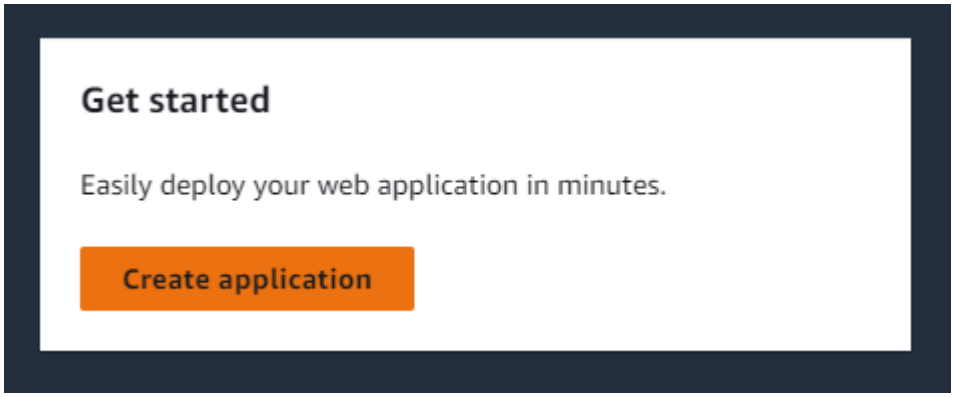
• EXECUTING APPLICATIONS

• UPLOADING APPLICATIONS

PAAS

STEPS SCREENSHOTS





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

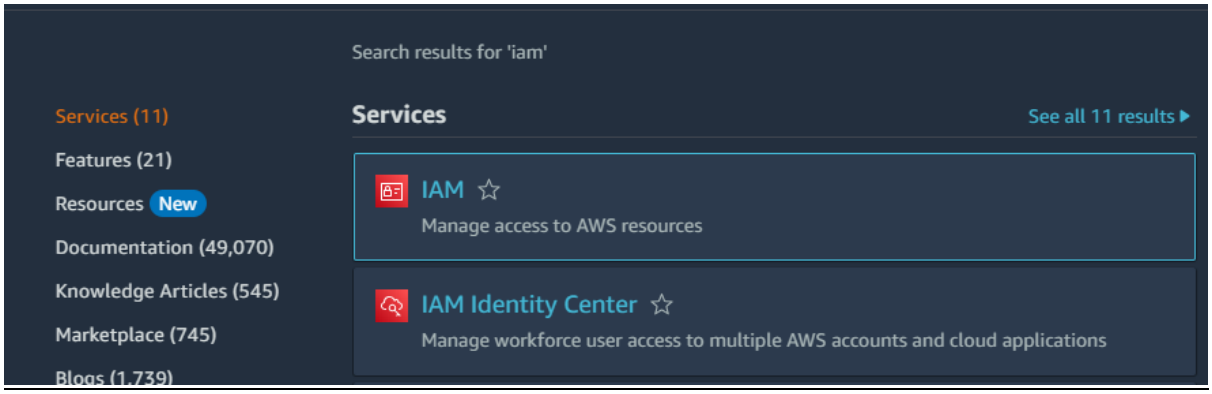
Platform version

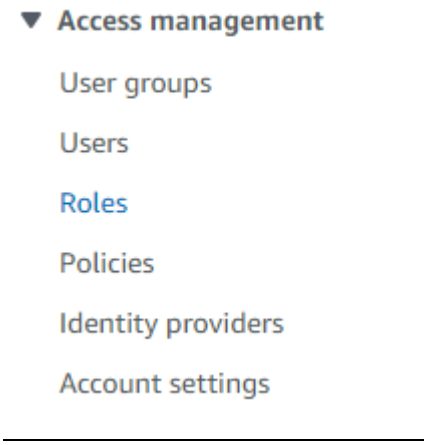
4.2.0 (Recommended)

Application code Info

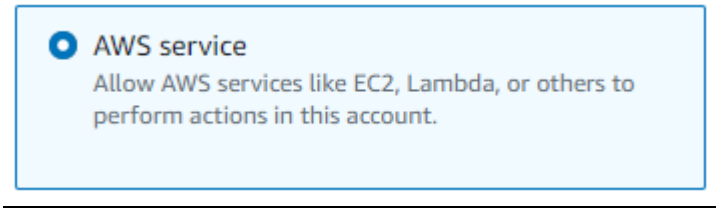
- (•) Sample application
- ( ) Existing version
  Application versions that you have uploaded.
- ( ) Upload your code
  Upload a source bundle from your computer or copy one from Amazon S3.

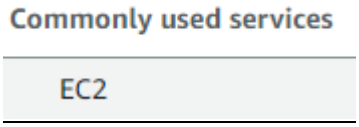
Creating a role for an application.

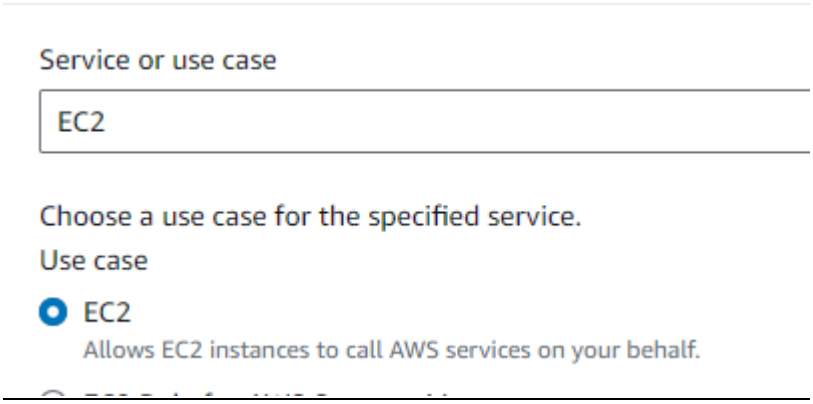



Access management
User groups
Users
Roles
Policies
Identity providers
Account settings

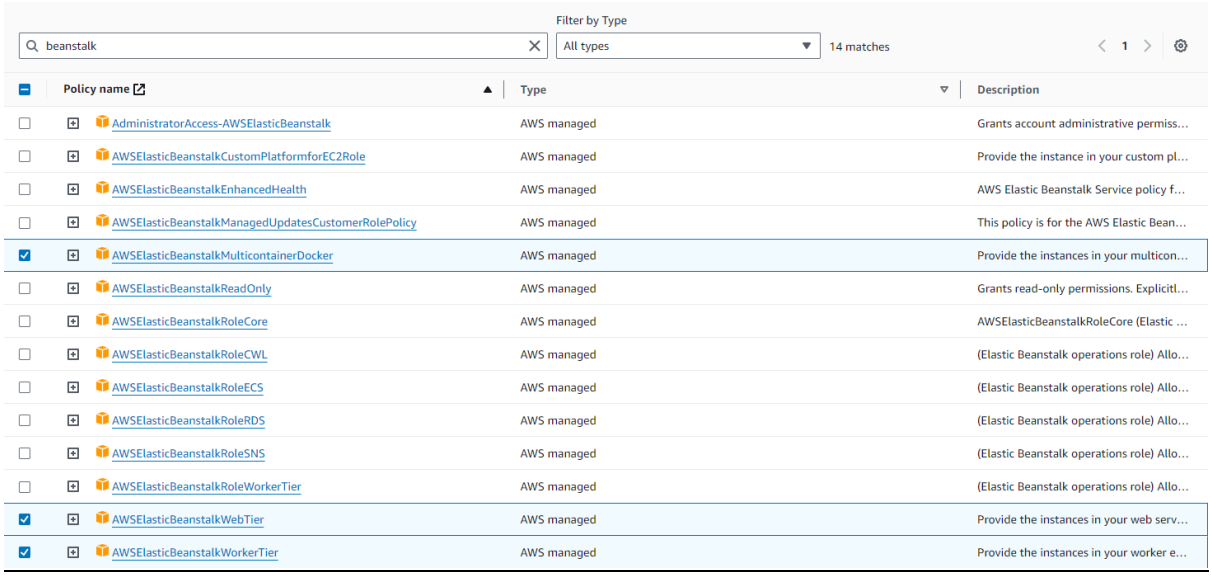

Create role


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.

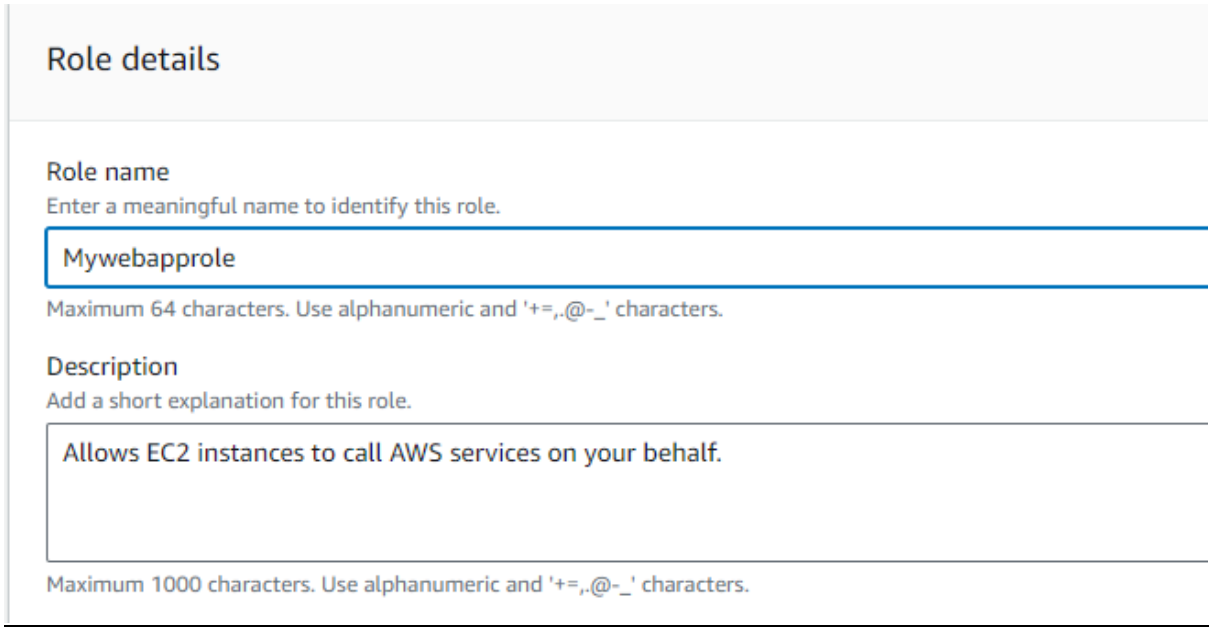
Click next

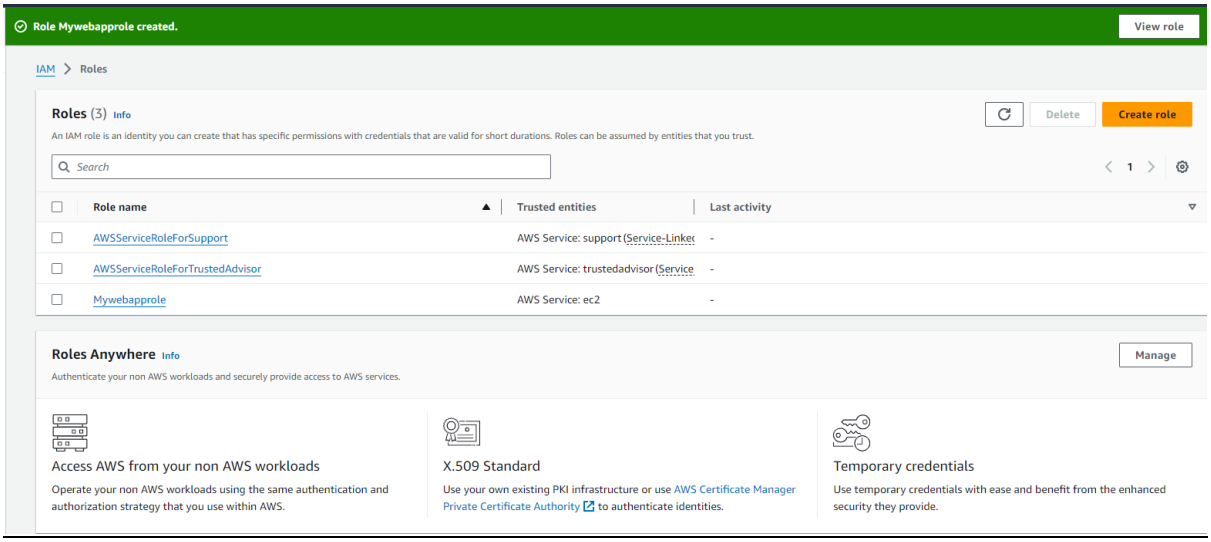


Worker tier-applications that you run

Multicontainer -webserver

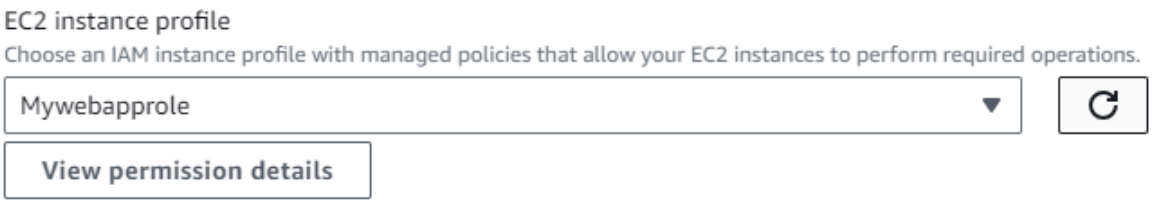
Click next

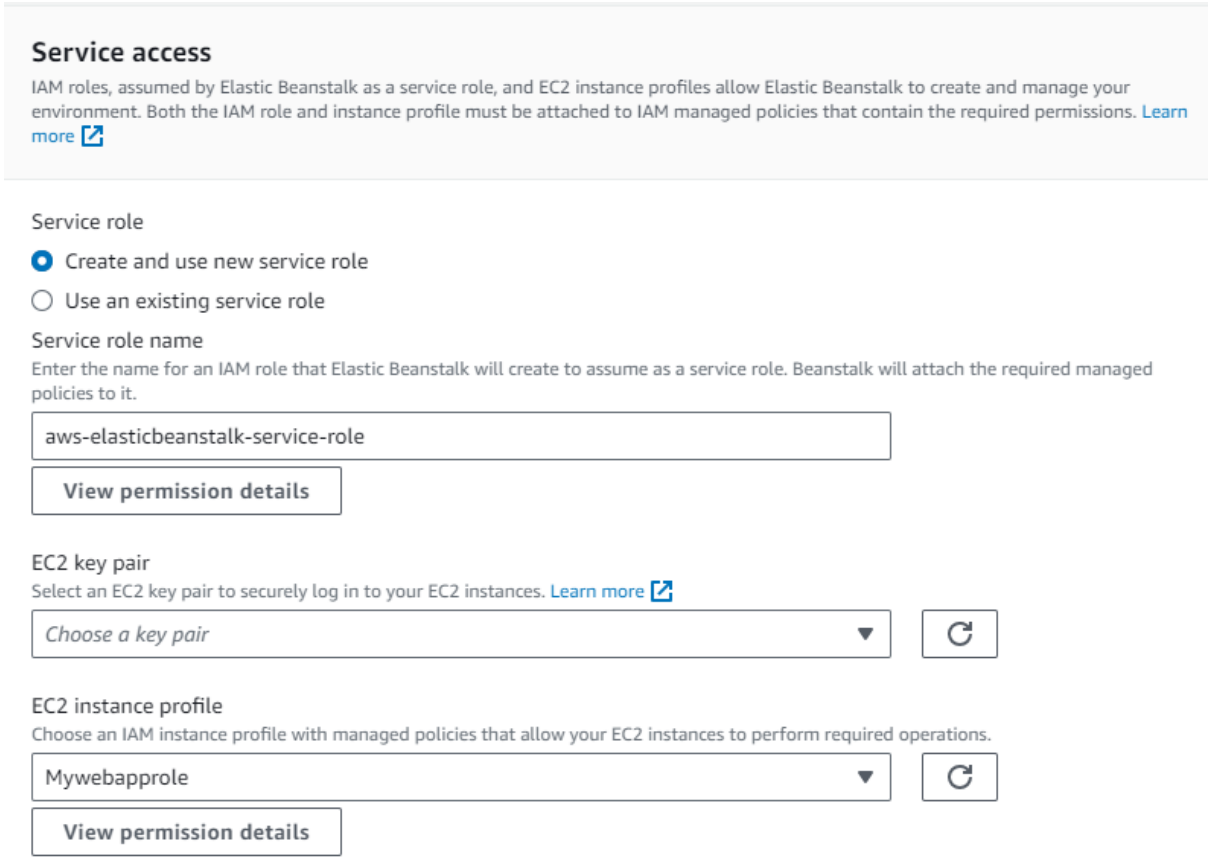




Go to the previous window

Refresh the instance profile cyclic button




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
Service role name
Enter the name for an IAM role that Elastic Beanstalk will create to assume as a service role. Beanstalk will attach the required managed policies to it.
aws-elasticbeanstalk-service-role
View permission details
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.
Mywebapprole
View permission details

Click next

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-0ede570da8526f7b5 | (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

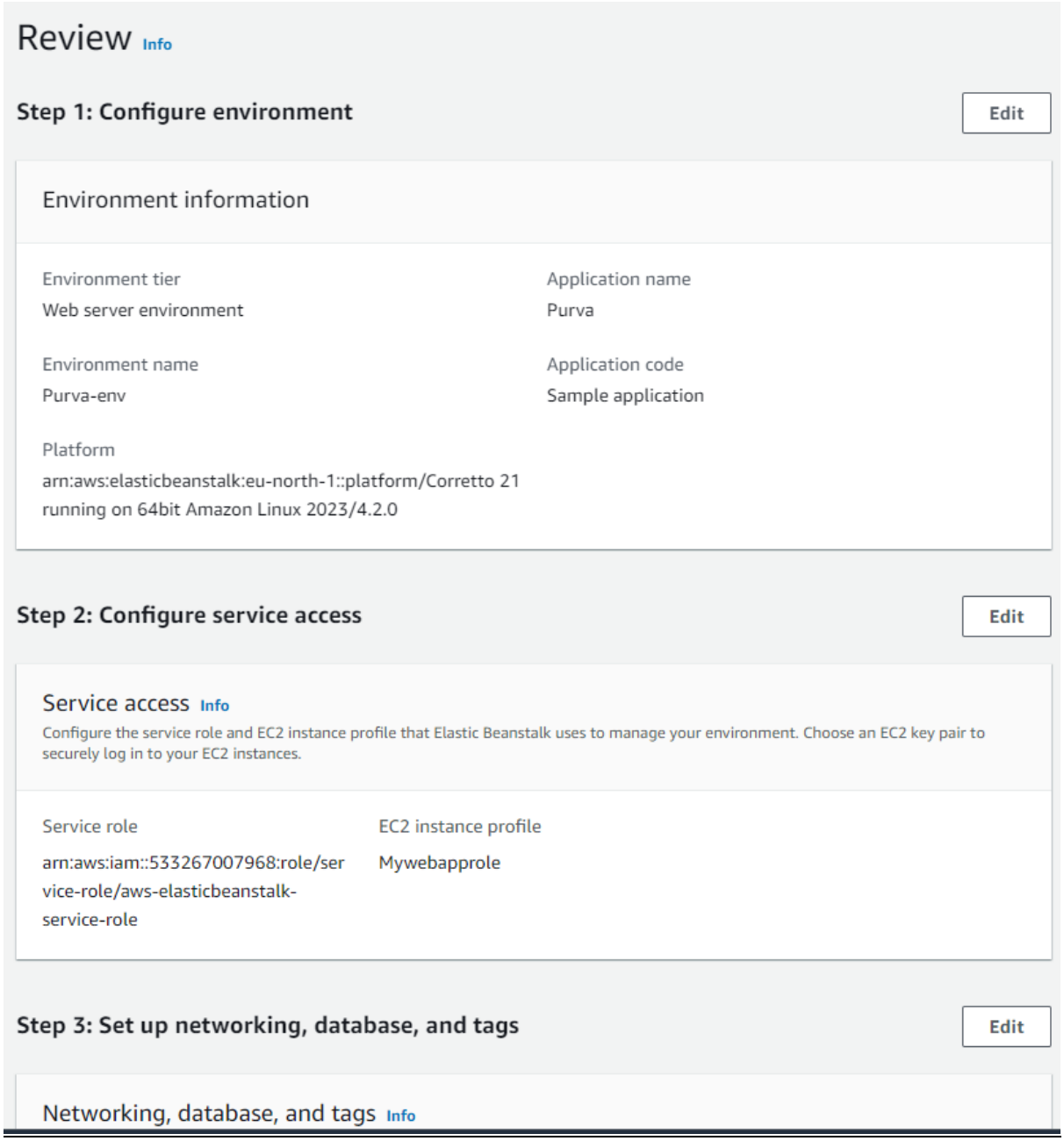
Filter instance subnets

	Availability Zone	Subnet ▲	CIDR	Name
☐	eu-north-1b	subnet-01570739f...	172.31.32.0/20	
☐	eu-north-1a	subnet-05547b7c4...	172.31.16.0/20	
☑	eu-north-1c	subnet-0aa0df663...	172.31.0.0/20	

Three times next

Click next next next

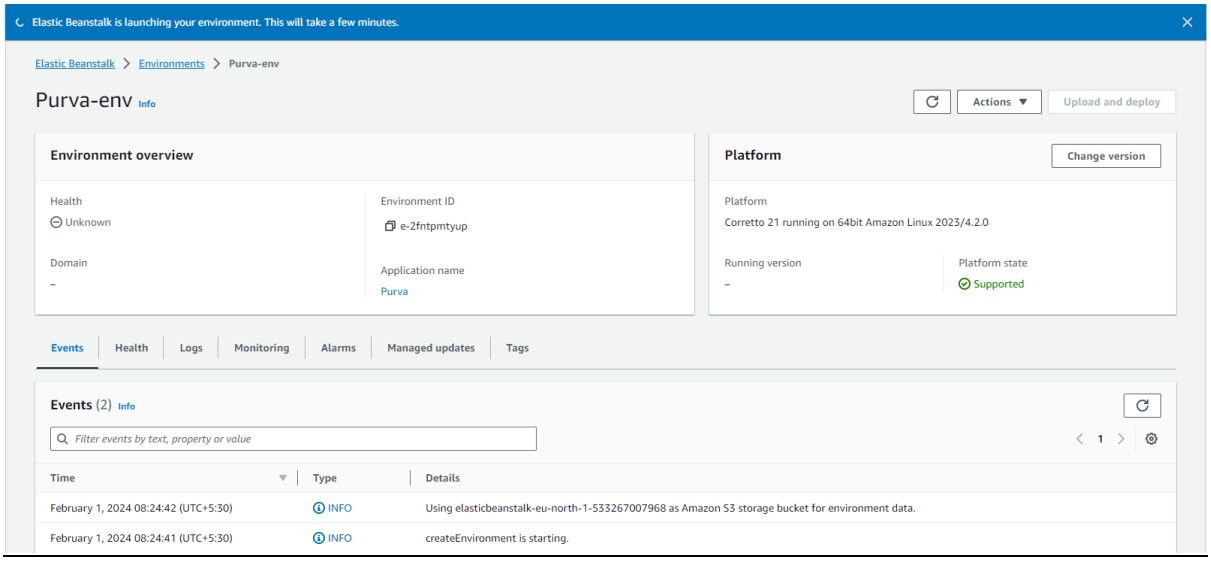
At last will get this window

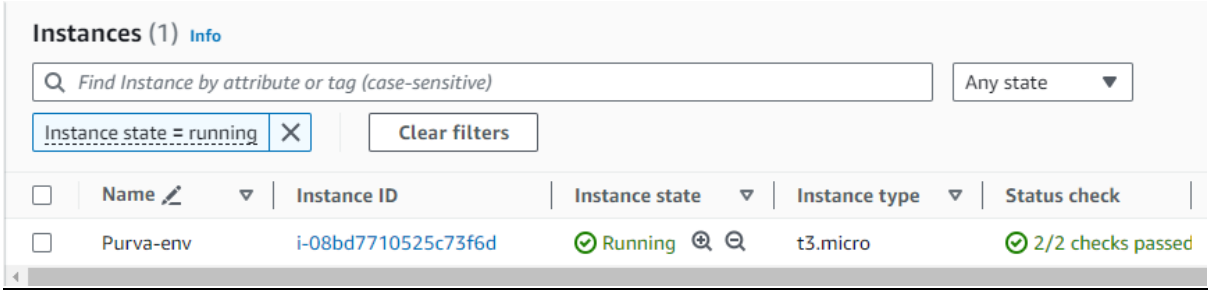


Click submit

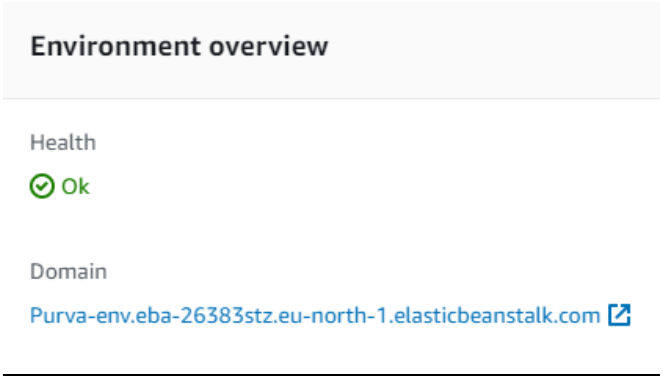
Go to EC2 and check if running

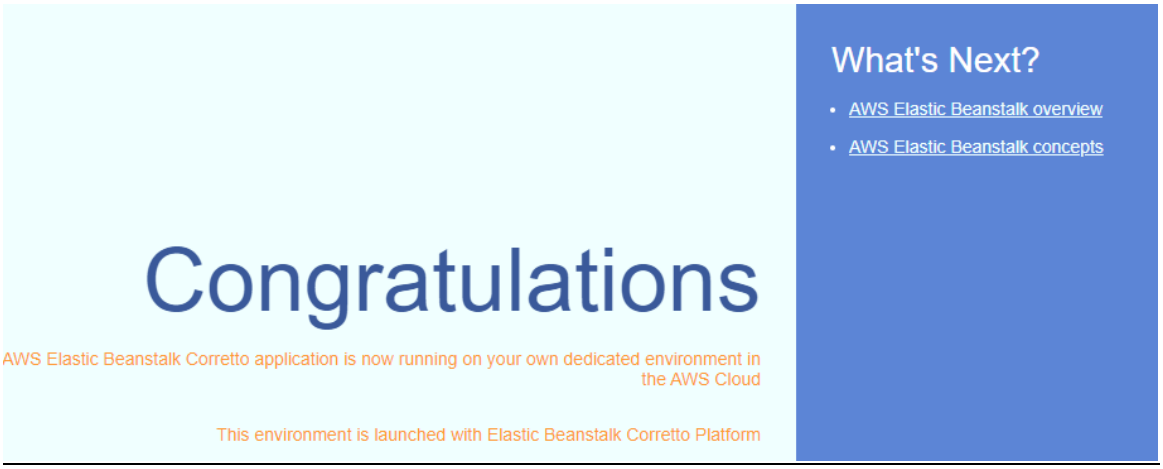
Click on domain url in environment overview





Should get this message





Creating TOMCAT server

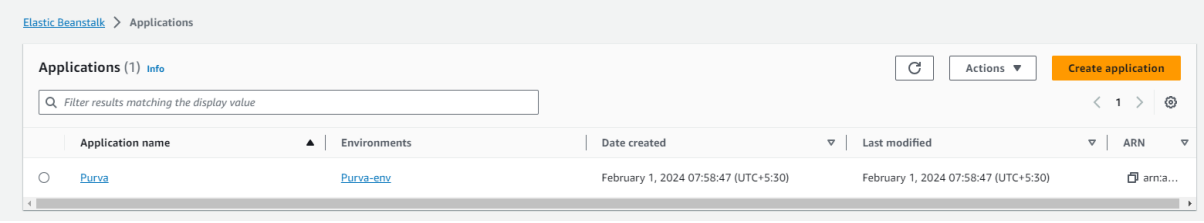
Q NO2

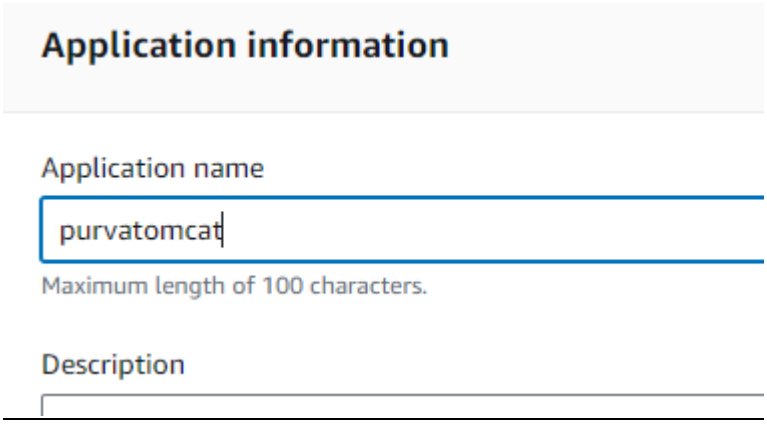
Configure an elastic beanstalk in AWS

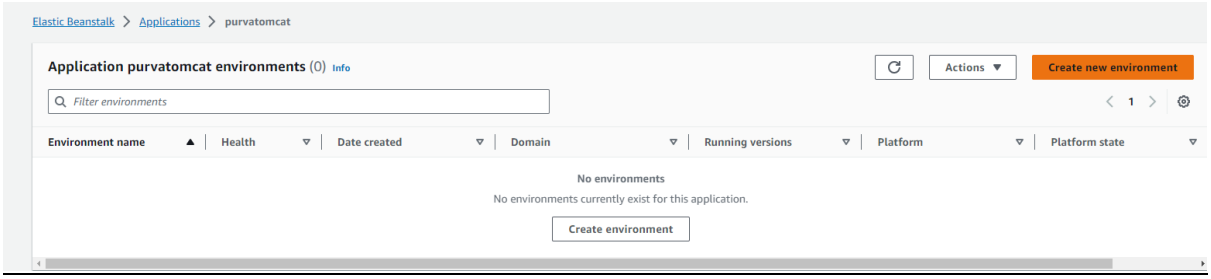
GO TO ELASTIC BEANSTALK HOME PAGE

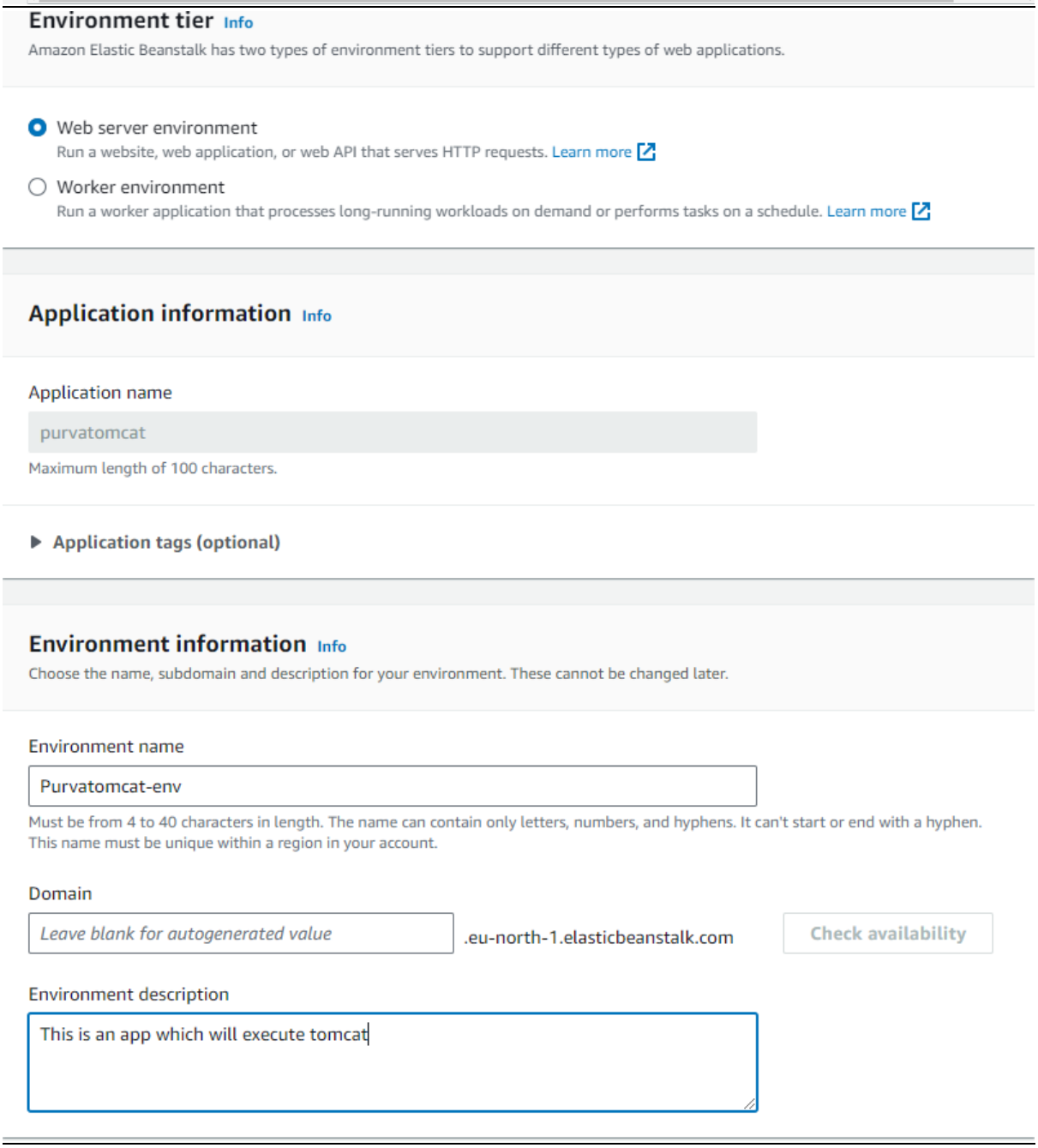
GET started page

Create application



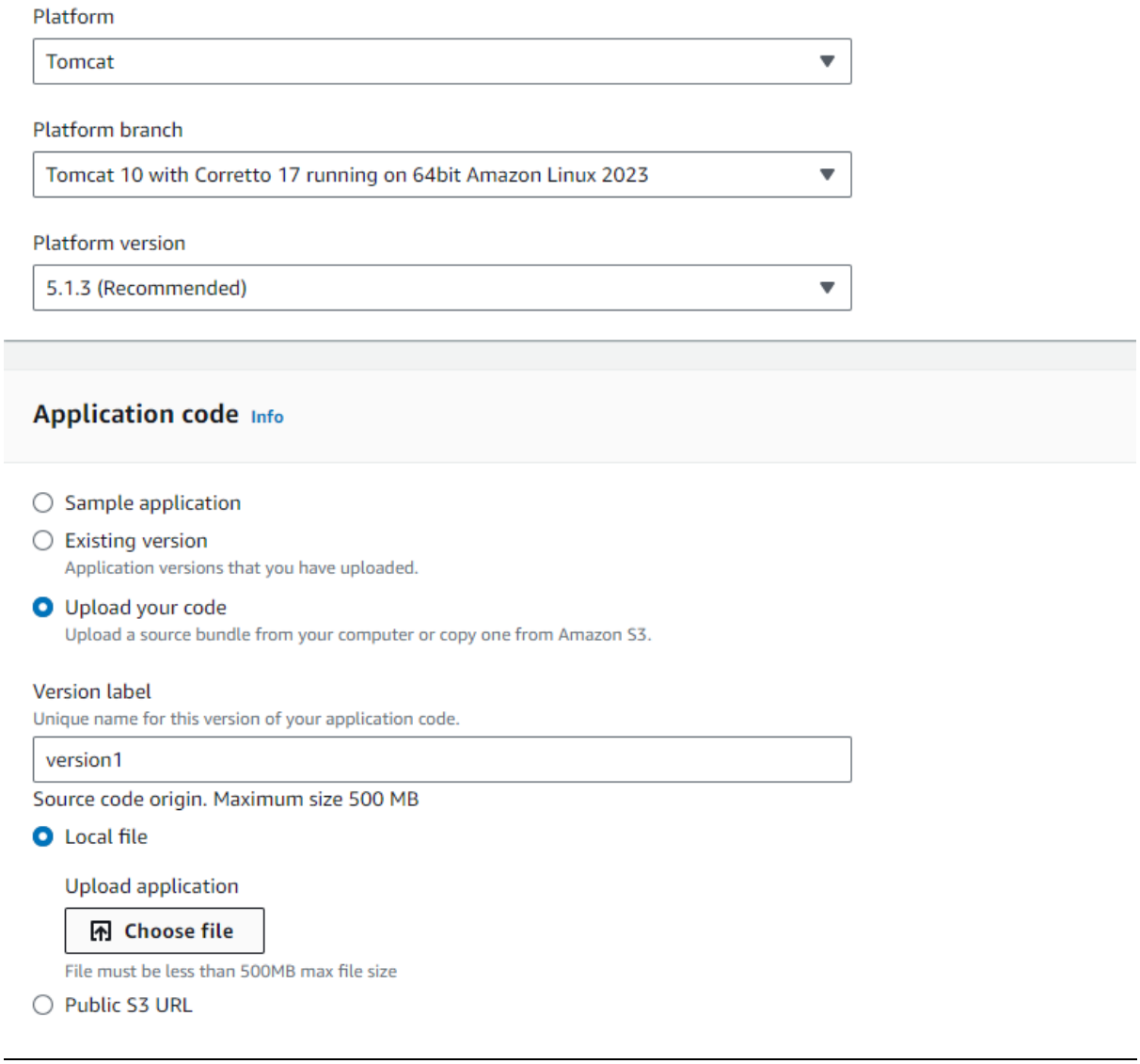


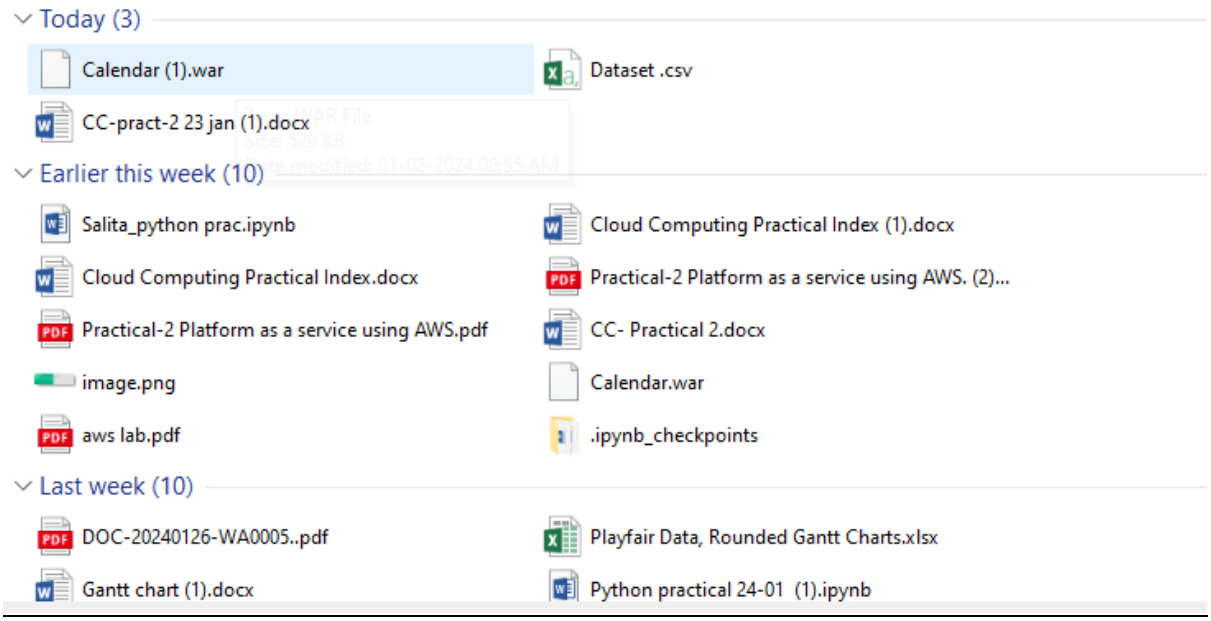


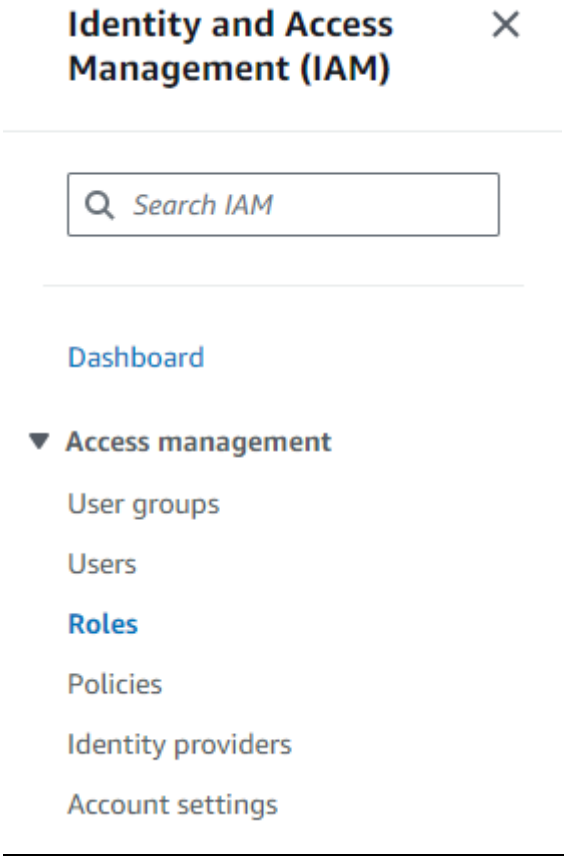


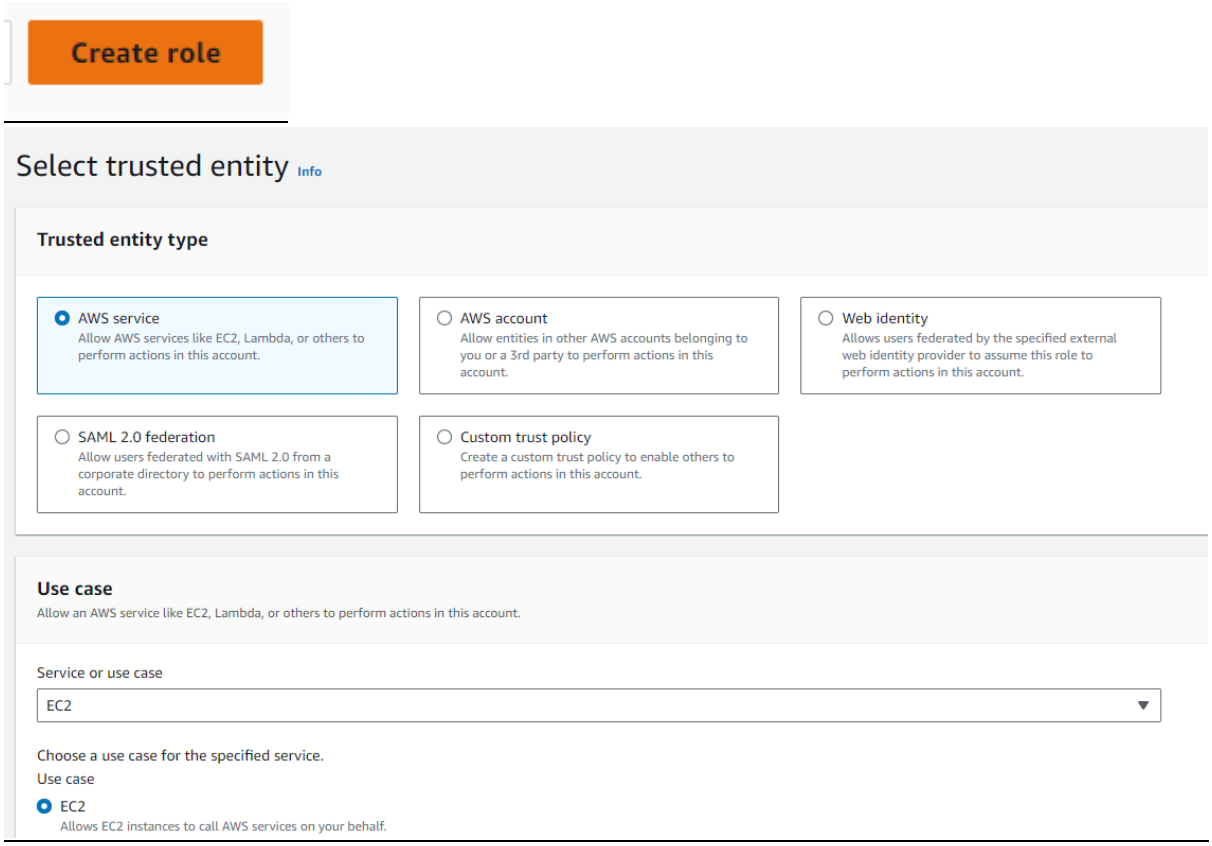
From the web download calendar.war file from github https://github.com/manulachathurika/Apache_Stratos_Tomcat_Applications/blob/master/Calendar.war

Select local file from option and choose file from your device

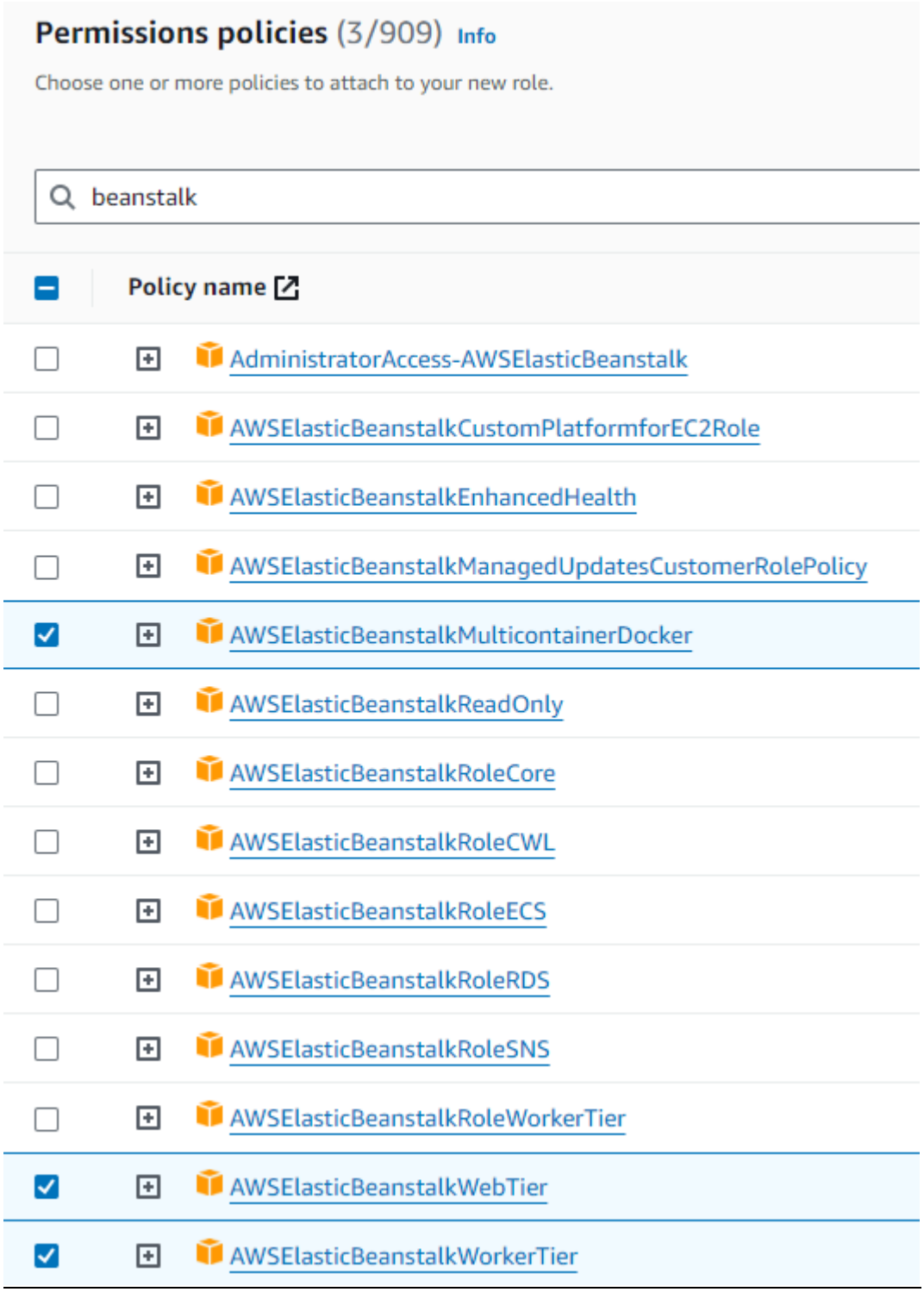



Today (3)
Calendar (1).war
Dataset .csv
CC-pract-2 23 jan (1).docx
Earlier this week (10)
Salita_python prac.ipynb
Cloud Computing Practical Index (1).docx
Cloud Computing Practical Index.docx
Practical-2 Platform as a service using AWS. (2)...
Practical-2 Platform as a service using AWS.pdf
CC- Practical 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

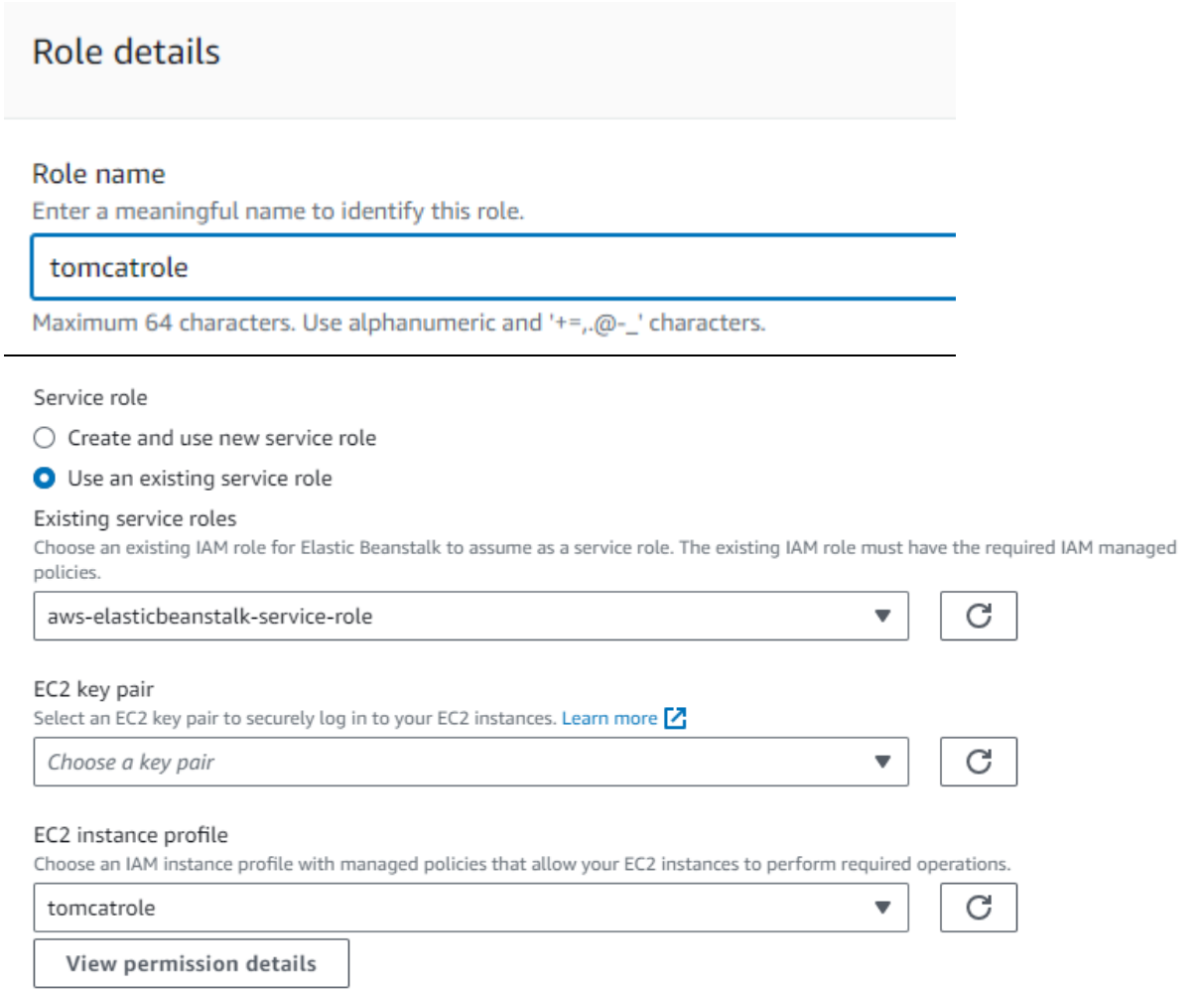

Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings



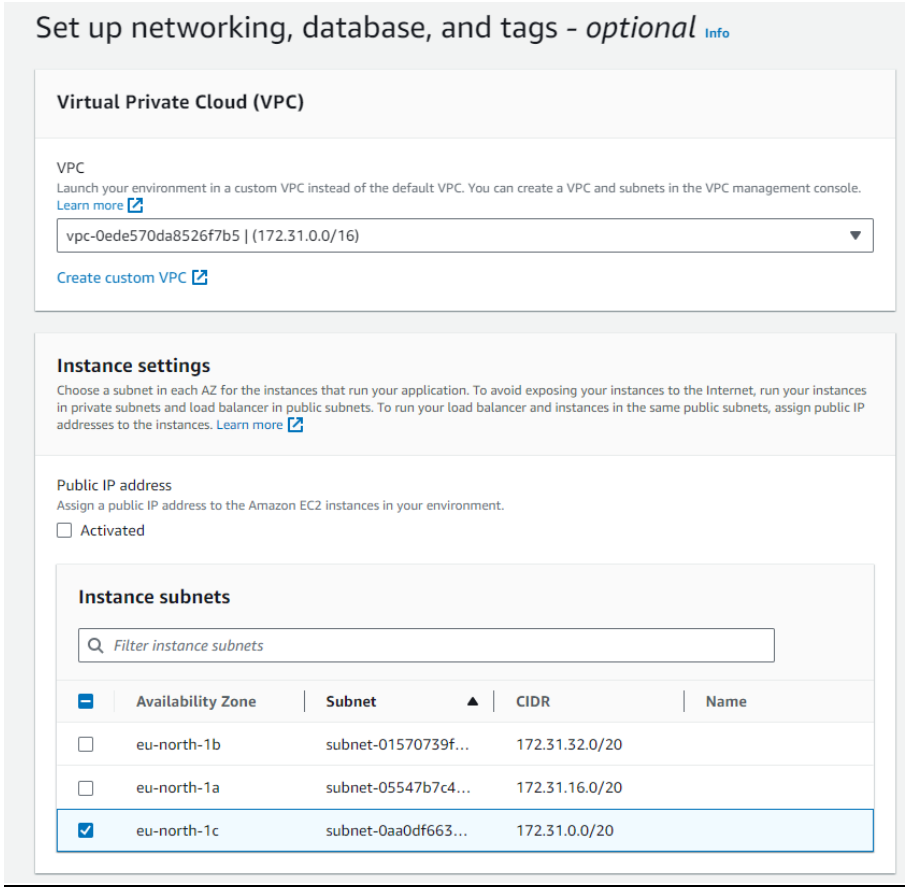
In permissions select same 3 things

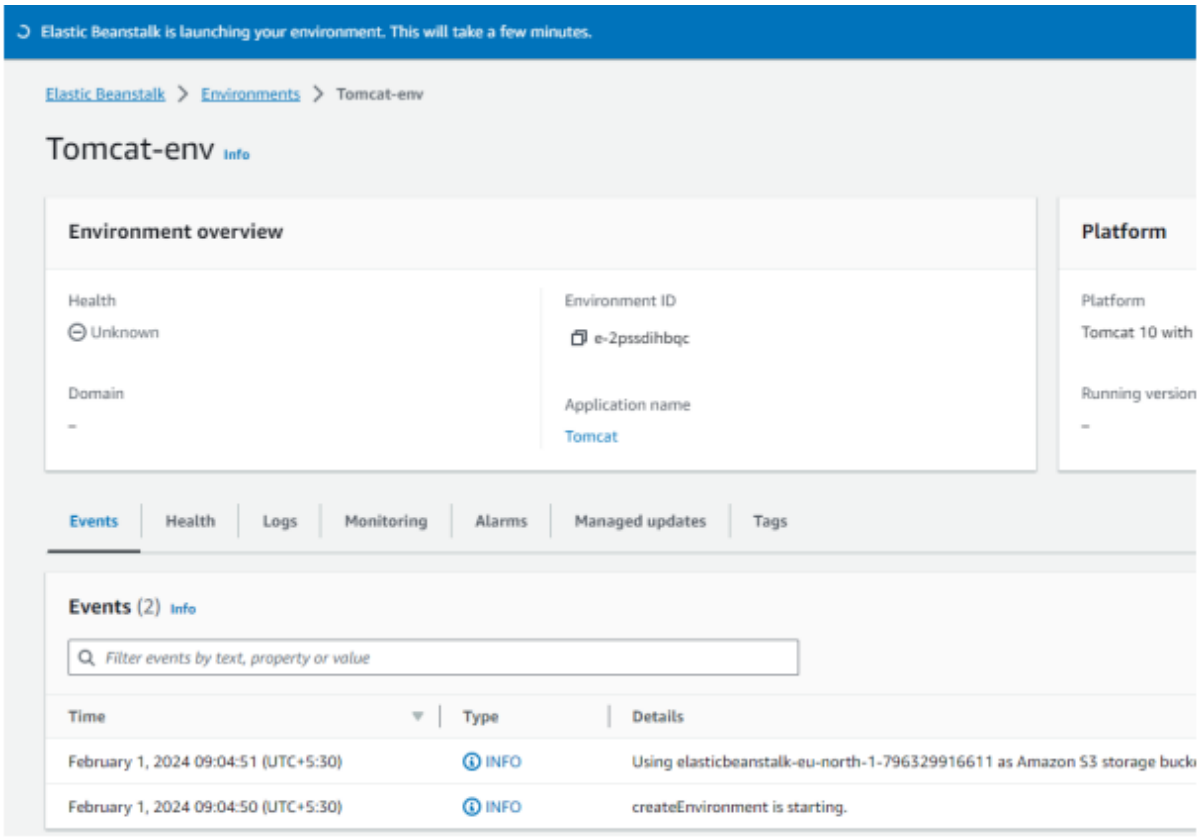


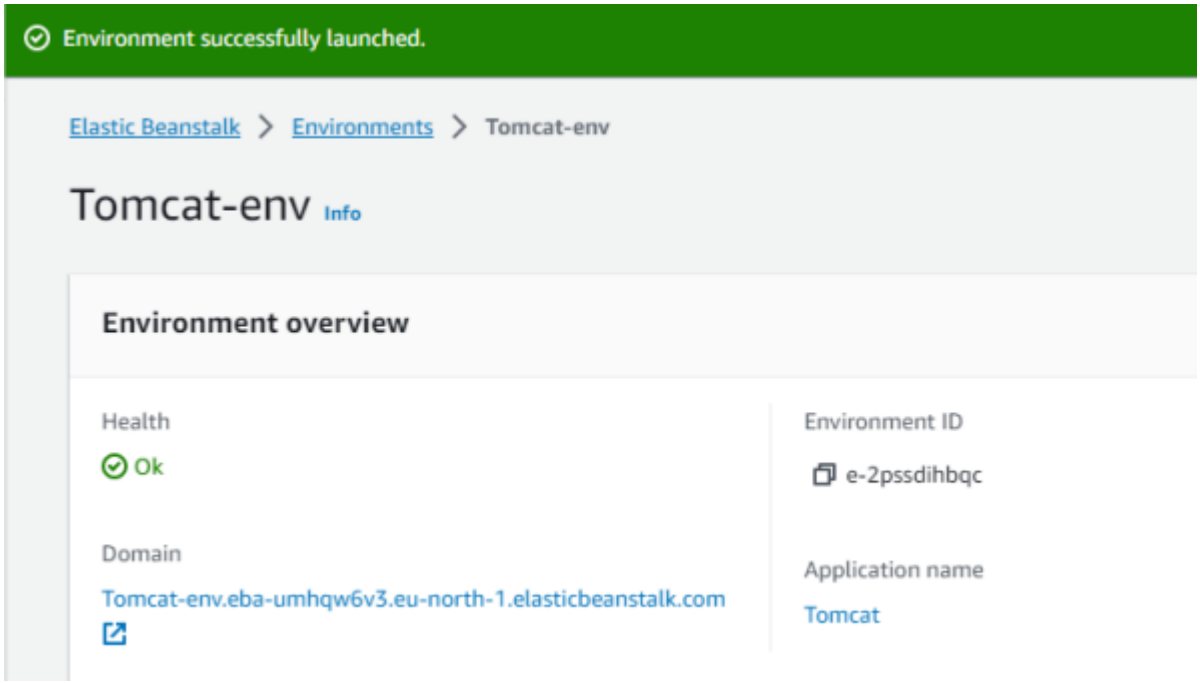
Give role name



In virtual private cloud select


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-0ede570da8526f7b5 | (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
eu-north-1b
subnet-01570739f...
172.31.32.0/20
eu-north-1a
subnet-05547b7c4...
172.31.16.0/20
eu-north-1c
subnet-0aa0df663...
172.31.0.0/20


Elastic Beanstalk is launching your environment. This will take a few minutes.
Elastic Beanstalk > Environments > Tomcat-env
Tomcat-env Info
Environment overview
Health
Unknown
Environment ID
e-2pssdihbqc
Domain
-
Application name
Tomcat
Platform
Platform
Tomcat 10 with
Running version
-
Events
Health
Logs
Monitoring
Alarms
Managed updates
Tags
Events (2) Info
Filter events by text, property or value
Time
Type
Details
February 1, 2024 09:04:51 (UTC+5:30)
INFO
Using elasticbeanstalk-eu-north-1-796329916611 as Amazon S3 storage buck
February 1, 2024 09:04:50 (UTC+5:30)
INFO
createEnvironment is starting.


Environment successfully launched.
Elastic Beanstalk
Environments
Tomcat-env
Tomcat-env
Info
Environment overview
Health
Ok
Domain
Tomcat-env.eba-umhqw6v3.eu-north-1.elasticbeanstalk.com
Environment ID
e-2pssdihbqc
Application name
Tomcat

Review Info

Step 1: Configure environment

Edit

Environment information

Environment tier
Web server environment

Application name
purvatomcat

Environment name
Purvatomcat-env

Application code
Calendar (1).war

Platform
arn:aws:elasticbeanstalk:eu-north-1::platform/Tomcat 10 with Corretto 17 running on 64bit Amazon Linux 2023/5.1.3

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::533267007968:role/service-role/aws-elasticbeanstalk-service-role

EC2 instance profile
tomcatrole

Step 3: Set up networking, database, and tags

Edit

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		