

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

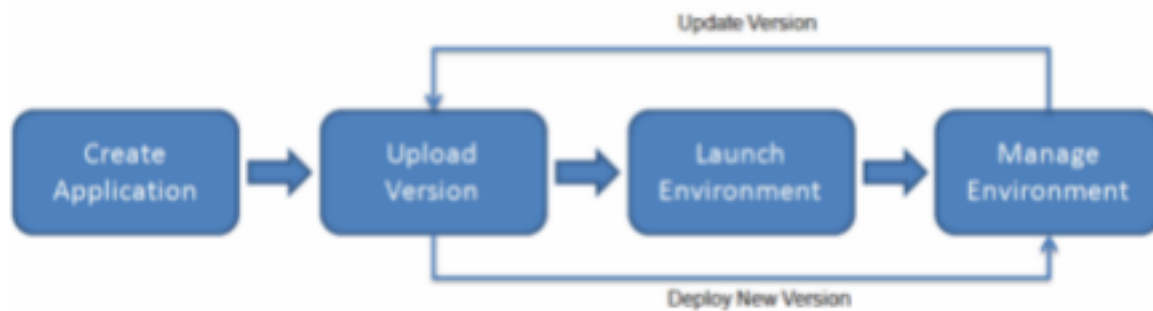
Deploy scalable web applications in minutes without the complexity of provisioning and managing underlying infrastructure.

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

X

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

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.

Domain

.eu-north-1.elasticbeanstalk.com

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

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.

Search results for 'iam'

Services (11)

Features (21)

Resources **New**


Documentation (49,070)


Knowledge Articles (545)

Marketplace (745)

Blogs (1,739)

Services [See all 11 results](#)

 **IAM** ☆
Manage access to AWS resources

 **IAM Identity Center** ☆
Manage workforce user access to multiple AWS accounts and cloud applications

▼ 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

Filter by Type			
Search: elastic	X	All types	14 matches
Policy name	Type	Description	
<input type="checkbox"/> AdministratorAccess-AWSManagedElastic	AWS managed	Grants account administrative permissions...	
<input type="checkbox"/> AWS-ElasticBeanstalkCustomPlatformforEC2Role	AWS managed	Provide the instance in your custom pl...	
<input type="checkbox"/> AWS-ElasticBeanstalkEnhancedHealth	AWS managed	AWS Elastic Beanstalk Service policy f...	
<input type="checkbox"/> AWS-ElasticBeanstalkManagedUpdatesContainerRolePolicy	AWS managed	This policy is for the AWS Elastic Bean...	
<input checked="" type="checkbox"/> AWS-ElasticBeanstalkMulticontainerDocker	AWS managed	Provide the instances in your multicon...	
<input type="checkbox"/> AWS-ElasticBeanstalkReadOnly	AWS managed	Grants read-only permissions. Explicit...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleCore	AWS managed	AWS-ElasticBeanstalkRoleCore (Elastic...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleCH	AWS managed	(Elastic Beanstalk operations role) Allo...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleEC2	AWS managed	(Elastic Beanstalk operations role) Allo...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleEFS	AWS managed	(Elastic Beanstalk operations role) Allo...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleSNS	AWS managed	(Elastic Beanstalk operations role) Allo...	
<input type="checkbox"/> AWS-ElasticBeanstalkRoleWorkerTier	AWS managed	(Elastic Beanstalk operations role) Allo...	
<input checked="" type="checkbox"/> AWS-ElasticBeanstalkWebTier	AWS managed	Provide the instances in your web serv...	
<input checked="" type="checkbox"/> AWS-ElasticBeanstalkWorkerTier	AWS managed	Provide the instances in your worker s...	

Worker tier-applications that you run

Multicontainer -webserver

Click next

Role details

Role name

Enter a meaningful name to identify this role.

Mywebapprole


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.

 Role Mywebapprole created. [View role](#)

[Roles](#) > Roles

Roles (3) [info](#)

AssumeRole 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.

☐

[AWS ServiceRoleForSupport](#)

AWS Service: support (Service:Support)

--

☐

[AWS ServiceRoleForTrustedAdvisor](#)

AWS Service: trustedadvisor (Service:TrustedAdvisor)

--

☐

[Mywebapprole](#)

AWS Service: ec2

--

Roles Anywhere [info](#)

Authenticate your non-AWS workloads and securely provide access to AWS services.



Access AWS from your non-AWS workloads

Operate your non-AWS workloads using the same authentication and authorization strategy that you use within AWS.



X.509 Standard

Use your own existing PKI infrastructure or use [AWS Certificate Manager Private Certificate Authority](#) to authenticate identities.



Temporary credentials

Use temporary credentials with ease and benefit from the enhanced security they provide.

[Manage](#)

[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





From the web download calendar.war file from github

https://github.com/manulachathurika/Apache_Stratos_Tomcat_Applications/blob/master/Calendar.war

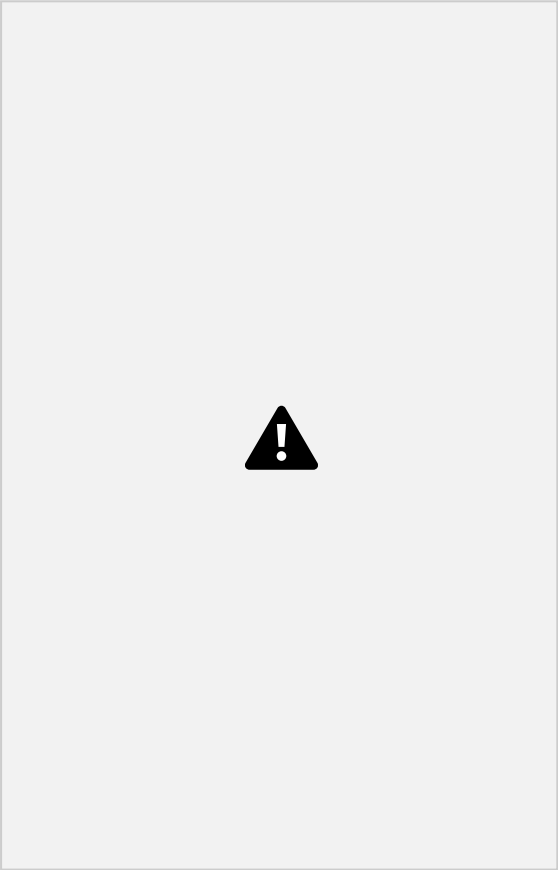
Select local file from option and choose file from your device



Click next

Go to IAM – Roles- Create new role

Use case EC2







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

