



Manchester
Metropolitan
University

CLOUD COMPUTING

/PLATFORM as a SERVICE (PaaS)
/AWS ELASTIC BEANSTALK

Dr Mohammed Kaleem

 m.kaleem@mmu.ac.uk



CONTENTS

/01 UNDERSTANDING THE DIFFERENCE BETWEEN LAMBDA AND ELASTIC BEANSTALK

/02 DEPLOYING JAVA APPS TO AWS BEANSTALK

- How to deploy a java app to AWS elastic beanstalk
 - Creating a service role
 - Configuring AWS elastic beanstalk and deploying a WAR file application

AWS LAMBDA Vs AWS ELASTIC BEANSTALK

Both Lambda and Elastic Beanstalk are popular services offered by Amazon Web Services (AWS) for deploying and managing applications, but they differ significantly in their architecture and use cases.



Elastic Beanstalk



Lambda

AWS LAMBDA

AWS Lambda is a serverless compute service that allows you to run code without provisioning or managing servers.

Key Features:

- **Event-driven:** Executes code in **response to events** such as changes to data in Amazon S3, DynamoDB, or invoking HTTP endpoints via API Gateway.
- **Pay-per-use pricing:** Billed based on the number of requests and the duration of code execution, with no charge when the code is not running. **2m free request.**
- **Supports multiple programming languages** including Node.js, Python, Java, and more.
- **Good Use Cases:**
 - Real-time file processing
 - IoT data processing
 - Backend for mobile or web applications
 - Chatbots and voice-enabled applications



AWS ELASTIC BEANSTALK

AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with popular programming languages, frameworks, and containers.

Key Features:

- **Fully managed platform:** AWS handles infrastructure provisioning, deployment, and scaling, **allowing developers to focus on application development.**
- **Support for multiple platforms:** Offers support for **Java**, .NET, Node.js, Python, Ruby, Go, Docker, and more.
- **Customizable environment configurations:** Allows customization of underlying resources such as EC2 instance types, auto-scaling configurations, and load balancers.
- **Use Cases:**
 - **Web application hosting**
 - API services
 - Microservices architectures
 - Development and testing environments



AWS LAMBDA Vs AWS ELASTIC BEANSTALK

Deployment Model:

- AWS Lambda: Serverless, event-driven model.
- AWS Elastic Beanstalk: Managed platform-as-a-service (PaaS) model.

Scaling:

- AWS Lambda: Scales automatically based on the number of incoming requests.
- AWS Elastic Beanstalk: Offers auto-scaling capabilities based on defined metrics such as CPU utilization or request count.

Management Complexity:

- AWS Lambda: Minimal management required as AWS handles infrastructure provisioning and scaling.
- AWS Elastic Beanstalk: Requires more configuration and management compared to Lambda, but less than managing raw infrastructure.

Pricing:

- AWS Lambda: Pay-per-use pricing based on the number of requests and duration of execution.
- AWS Elastic Beanstalk: Pay for underlying resources such as EC2 instances, load balancers, and storage, with additional charges for data transfer and other services.

TYPICAL USE CASES

AWS Lambda:

Ideal for event-driven applications with short-lived, stateless functions.

Suitable for applications with unpredictable or sporadic workloads.



AWS Elastic Beanstalk:

Best suited for long-running applications requiring consistent performance.

Provides more control over underlying infrastructure and configuration.

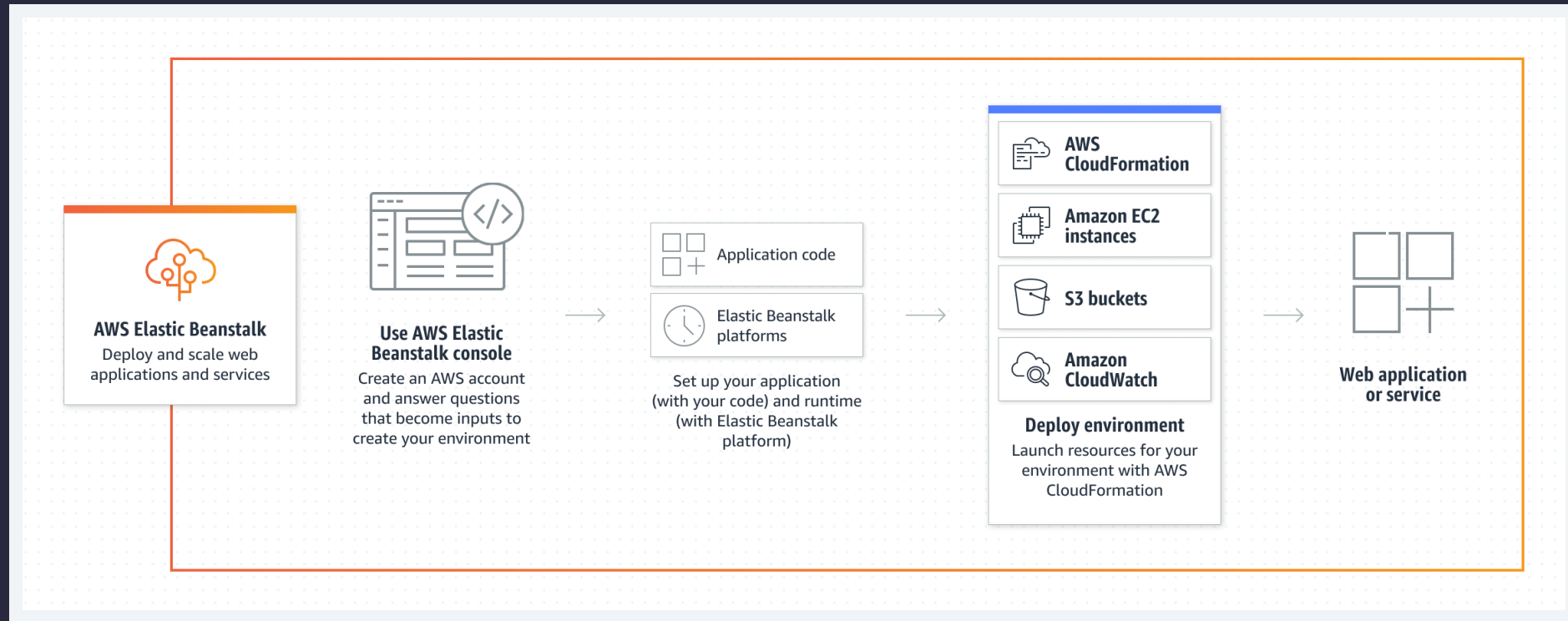


Choosing between Lambda and Elastic Beanstalk depends on factors such as application architecture, expected workload, and management preferences.

ALL DEPENDS ON USE CASE

- AWS Lambda and AWS Elastic Beanstalk **are both powerful services** offered by AWS for deploying and managing applications.
- Choose AWS **Lambda for event-driven**, serverless architectures with **short-lived functions** and unpredictable workloads.
- Opt for AWS **Elastic Beanstalk when deploying long-running applications** that require more control over underlying infrastructure and configuration.
- Consider your specific use case, workload requirements, and management preferences when selecting between Lambda and Elastic Beanstalk.

DEPLOYING AN APP TO AWS ELASTIC BEANSTALK

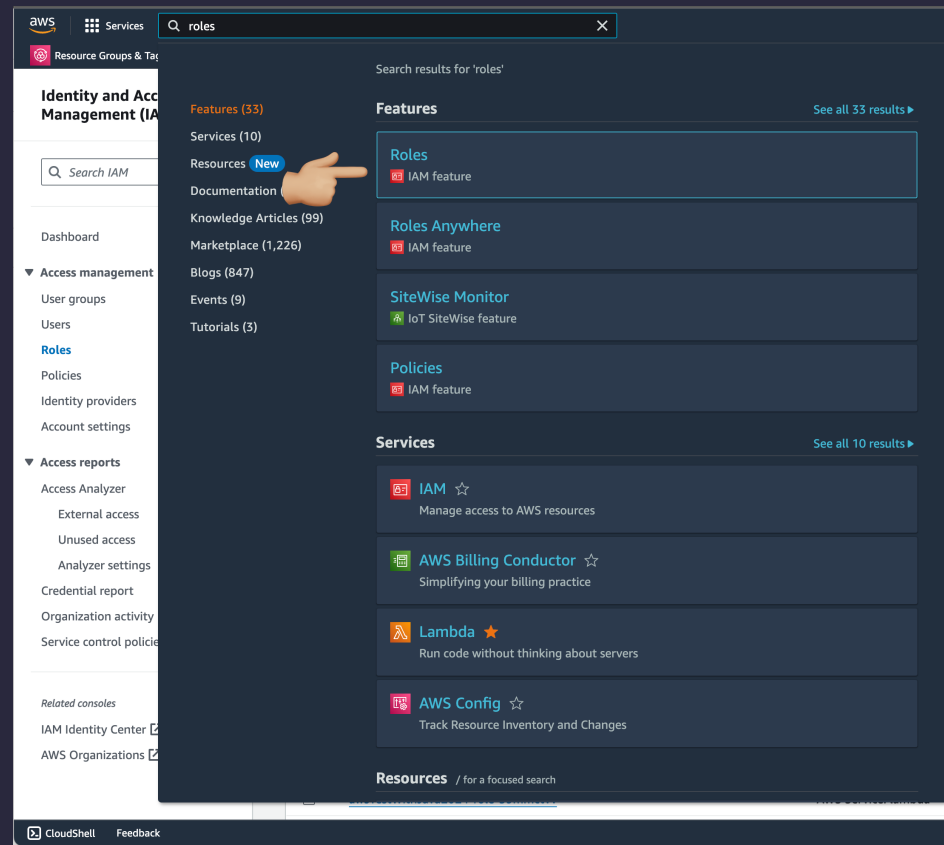


STEP 1: CREATING A SERVICE ROLE

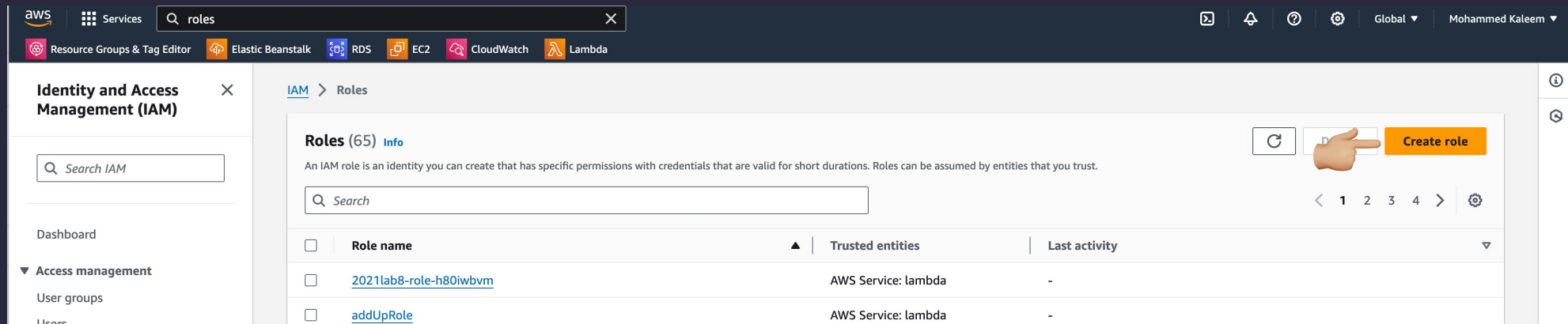
- A service role is the AWS Identity and Access Management role (**IAM role**) that Elastic Beanstalk assumes when calling other services on your behalf.
- For example, Elastic Beanstalk uses a service role when it calls Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing, and Amazon EC2 Auto Scaling APIs to gather information.
- The service role that Elastic Beanstalk uses is the one that you specified when you create the Elastic Beanstalk environment.
- AWS Elastic Beanstalk takes away all EC2 VM configuration and does all that for you using the service role.



CREATING A SERVICE ROLE



CREATING A SERVICE ROLE



The screenshot shows the AWS IAM console interface. The top navigation bar includes the AWS logo, a 'Services' menu, a search bar with 'roles' entered, and user information for 'Mohammed Kaleem'. The left sidebar shows the 'Identity and Access Management (IAM)' section with options for 'Dashboard', 'Access management', 'User groups', and 'Users'. The main content area is titled 'IAM > Roles' and displays 'Roles (65)' with an 'Info' link. Below this, a description states: '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.' A search bar is provided for filtering roles. A table lists existing roles with columns for 'Role name', 'Trusted entities', and 'Last activity'. Two roles are visible: '2021lab8-role-h80iwsvm' and 'addUpRole', both with 'AWS Service: lambda' as the trusted entity. In the top right corner of the main content area, there are buttons for 'Refresh', 'Filter', and a prominent orange 'Create role' button, which is pointed to by a hand icon.

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>	2021lab8-role-h80iwsvm	AWS Service: lambda	-
<input type="checkbox"/>	addUpRole	AWS Service: lambda	-

CREATING A SERVICE ROLE

aws Services

Resource Groups & Tag Editor Elastic Beanstalk RDS EC2 CloudWatch Lambda

[IAM](#) > [Roles](#) > Create role

Step 1
Select trusted entity

Step 2
[Add permissions](#)

Step 3
Name, review, and create

Select trusted entity [Info](#)

Trusted entity type

- ☒ **AWS service**
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- ☐ **AWS account**
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- ☐ **Web identity**
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- ☐ **SAML 2.0 federation**
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- ☐ **Custom trust policy**
Create a custom trust policy to enable others to perform actions in this account.

Use case

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

Service or use case

EC2

Choose a use case for the specified service.

Use case

- ☒ **EC2**
Allows EC2 instances to call AWS services on your behalf.


CREATING A SERVICE ROLE - add permissions

[IAM](#) > [Roles](#) > aws-elasticbeanstalk-ec2-role

aws-elasticbeanstalk-ec2-role [Info](#)

[Delete](#)

Summary [Edit](#)

Creation date March 22, 2023, 10:18 (UTC)	ARN arn:aws:iam::271900167803:role/aws-elasticbeanstalk-ec2-role	Instance profile ARN arn:aws:iam::271900167803:instance-profile/aws-elasticbeanstalk-ec2-role
Last activity  24 minutes ago	Maximum session duration 1 hour	




[Permissions](#) | [Trust relationships](#) | [Tags](#) | [Access Advisor](#) | [Revoke sessions](#)

Permissions policies (3) [Info](#)

You can attach up to 10 managed policies.

[Refresh](#) [Simulate](#) [Remove](#) [Add permissions](#)

Search Filter by Type [All types](#) [<](#) [1](#) [>](#) [Settings](#)

<input type="checkbox"/>	Policy name Info	Type	Attached entities
<input type="checkbox"/>	 AWS Elastic Beanstalk Multi-Container Docker	AWS managed	1
<input type="checkbox"/>	 AWS Elastic Beanstalk WebTier	AWS managed	1
<input type="checkbox"/>	 AWS Elastic Beanstalk WorkerTier	AWS managed	1

CREATING A SERVICE ROLE – name and save

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.

Maximum 1000 characters. Use alphanumeric and '+=, @- _ ' characters.

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.


CancelPreviousCreate role


Dr Mohammed Kaleem

STEP 2: CREATING A BEANSTALK APP

[Elastic Beanstalk](#) > Applications


Applications (3) [Info](#)



Actions 

[Create application](#)

< 1 >



	Application name ▲	Environments	Date created ▼	Last modified
<input type="radio"/>	hello-world	Hello-world-env	March 14, 2024 11:17:5...	March 14, 2024 11:17:5...
<input type="radio"/>	new-app	New-app-env	March 14, 2024 11:11:1...	March 14, 2024 11:11:1...
<input type="radio"/>	time-app	Time-app-env	March 14, 2024 11:01:2...	March 14, 2024 11:01:2...

CREATING A BEANSTALK APP – name your application

aws Services Search [Option+S] London Mohammed Kaleem

Resource Groups & Tag Editor Elastic Beanstalk RDS EC2 CloudWatch Lambda

Environment successfully launched.

Elastic Beanstalk > Create application

Create new application [Info](#)

Application information

Application name

lecture-demo-app

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.

[Add new tag](#)



You can add 50 more tags.


[Cancel](#) [Create](#)

CREATING A BEANSTALK APP – create environment

[Elastic Beanstalk](#) > [Applications](#) > lecture-demo-app

Application lecture-demo-app environments (0) [Info](#)

 **Actions** ▼  **Create new environment**


< 1 > 

Enviro... ▲	Health ▼	Date cr... ▼	Domain ▼	Runnin... ▼	Platform ▼
<p>No environments</p> <p>No environments currently exist for this application.</p> <p>Create environment</p>					

CREATING A BEANSTALK APP – select your platform


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 9 with Corretto 17 running on 64bit Amazon Linux 2023 ▼

Platform version

 5.1.4 (Recommended) ▼

CREATING A BEANSTALK APP – upload code

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.

1.0

Source code origin. Maximum size 500 MB

☒ Local file

Upload application

 Choose file

✔ File name: **demo-hello-world.war**
File must be less than 500MB max file size

☐ Public S3 URL

CREATING A BEANSTALK APP

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

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-lecture

[View permission details](#)

EC2 key pair

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

march-2024

[View permission details](#)

EC2 instance profile

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


kaleem-elasticbs-role-2024

[View permission details](#)

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

CREATING A BEANSTALK APP – review and submit

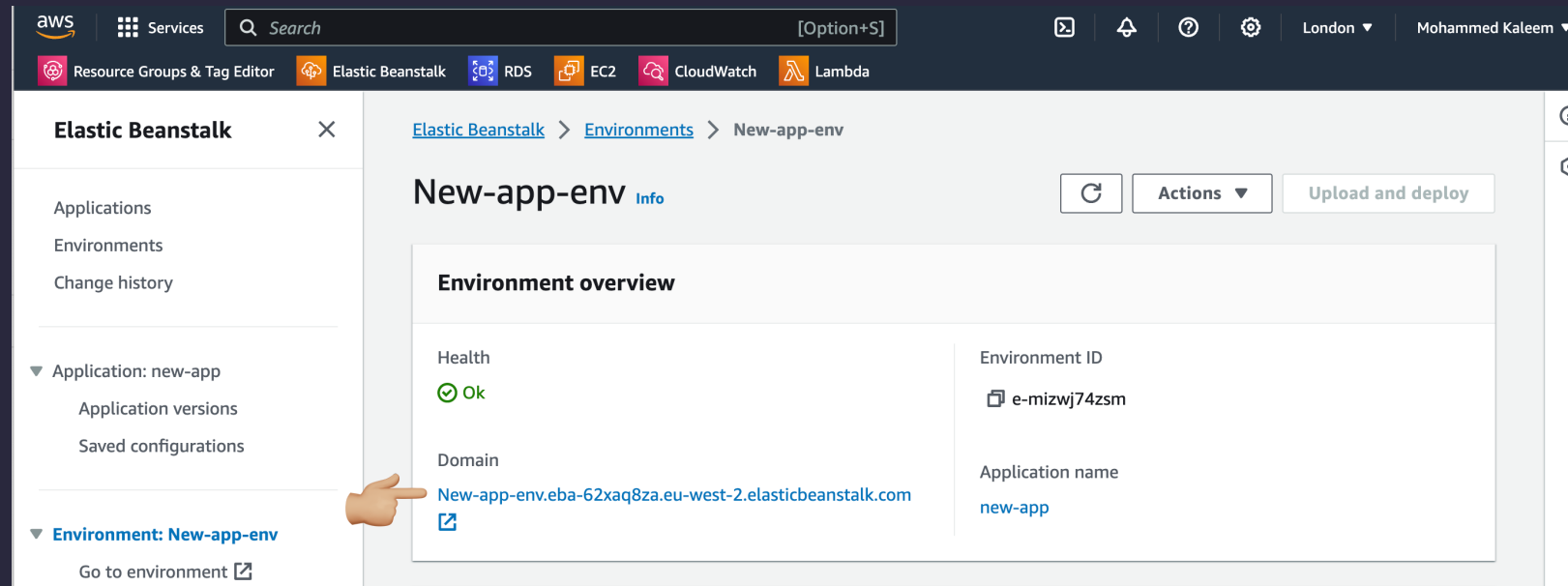
Key ▲	Value ▼
JDBC_CONNECTION_STRING	

Cancel Previo  Submit



Then wait...

CREATING A BEANSTALK APP – wait for the URL



A few minutes later you will get the URL to your web application.

Summary

- Create the IAM Role
- Export application from eclipse as WAR or JAR depending on how you implemented the application.
 - Dynamic Web Applications should be packaged as WAR files.
 - Spring boot applications should be packaged as JAR files.
- Configure new elastic beanstalk application/environment.
- Wait a while...
- App will be deployed, all set up and deployment handled by AWS, you'll get a URL to your app at the end of the process.