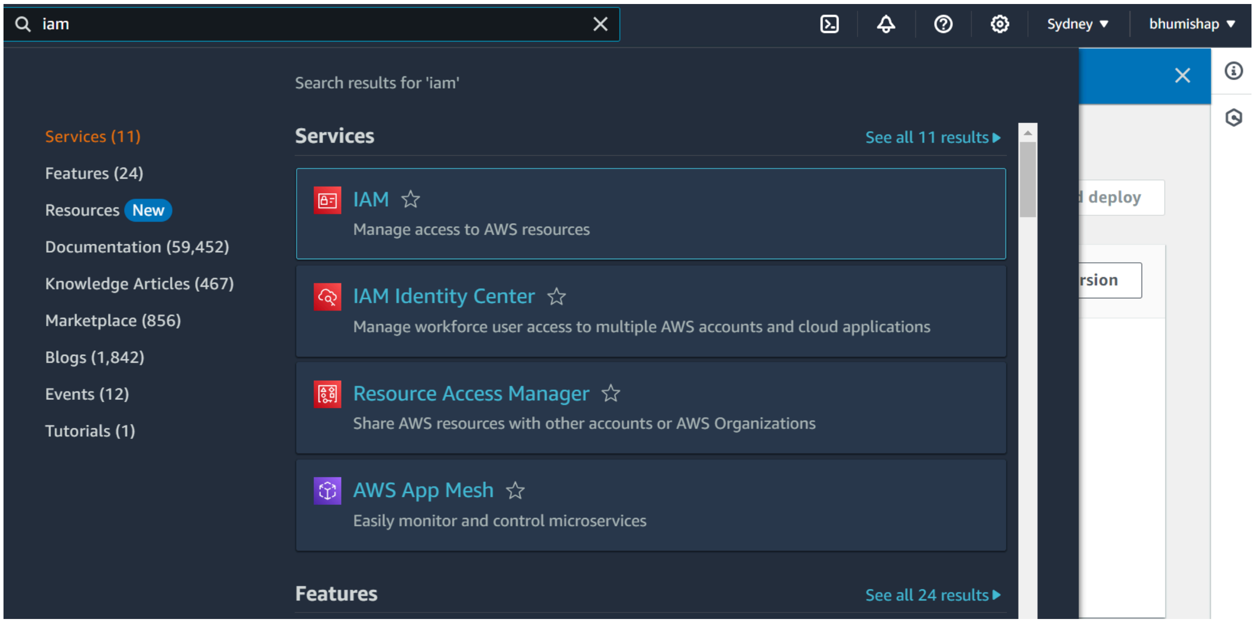
Experiment No. 2

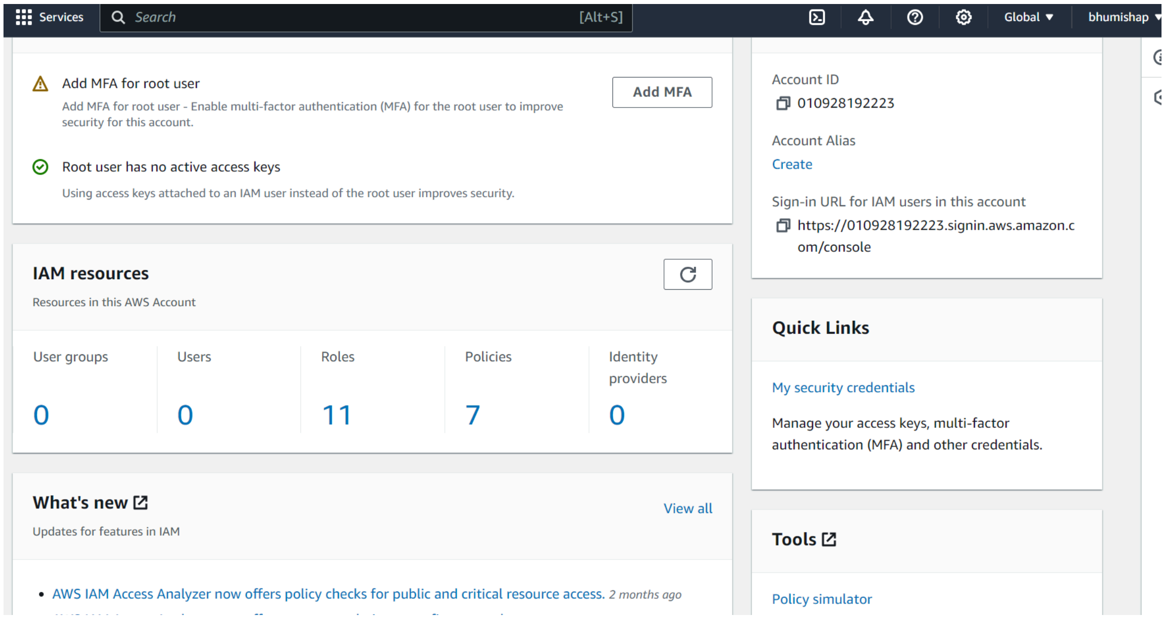
Aim: To Build Your Application using AWS CodeBuild and Deploy on S3 / SEBS using AWS CodePipeline, deploy Sample Application on EC2 instance using AWS CodeDeploy.

Step1:- Creation of role:-

1. Login to your AWS account and search for IAM



2. Then go into the role section and click on create role.



Add MFA for root user

Add MFA for root user - Enable multi-factor authentication (MFA) for the root user to improve security for this account.

Root user has no active access keys

Using access keys attached to an IAM user instead of the root user improves security.

Account ID
010928192223

Account Alias
Create

Sign-in URL for IAM users in this account
https://010928192223.signin.aws.amazon.com/console

IAM resources

Resources in this AWS Account

User groups	Users	Roles	Policies	Identity providers
0	0	11	7	0

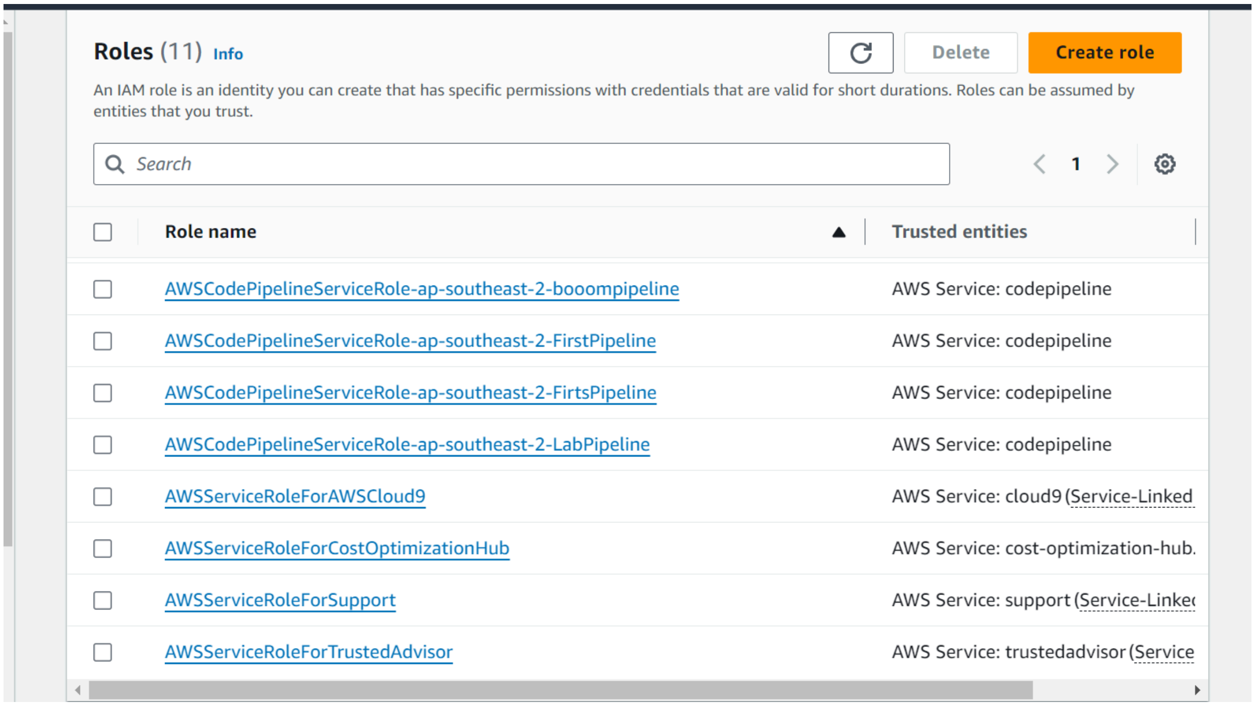
My security credentials

Manage your access keys, multi-factor authentication (MFA) and other credentials.

What's new

Updates for features in IAM

- AWS IAM Access Analyzer now offers policy checks for public and critical resource access. 2 months ago

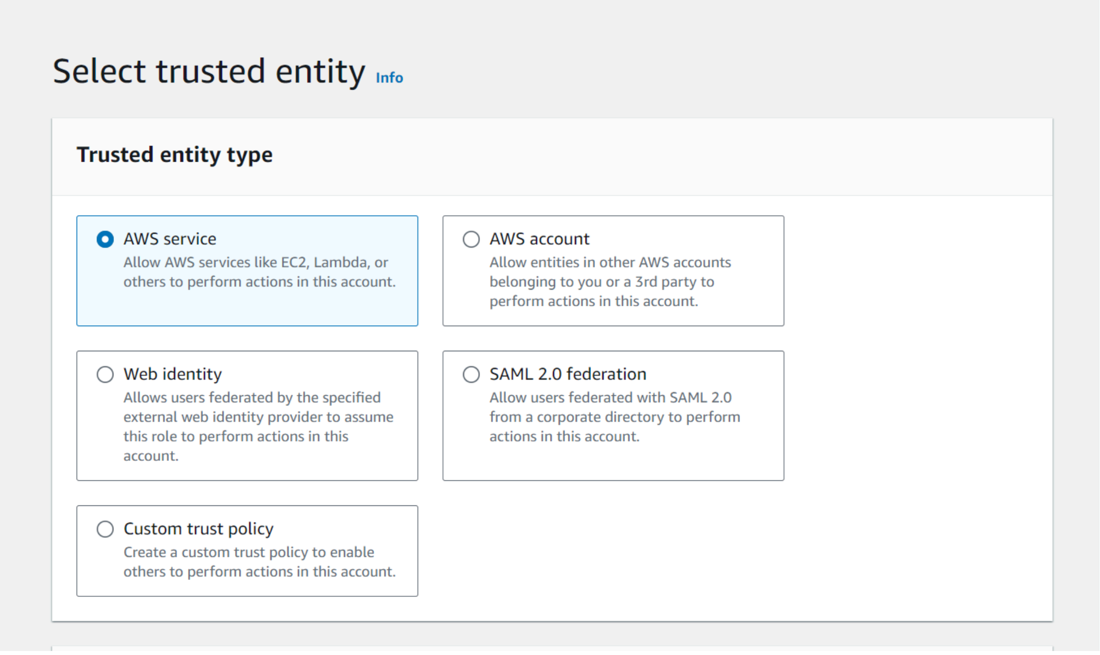


Roles (11) Info

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.

□	Role name ▲	Trusted entities
□	AWSCodePipelineServiceRole-ap-southeast-2-booompipeline	AWS Service: codepipeline
□	AWSCodePipelineServiceRole-ap-southeast-2-FirstPipeline	AWS Service: codepipeline
□	AWSCodePipelineServiceRole-ap-southeast-2-FirtsPipeline	AWS Service: codepipeline
□	AWSCodePipelineServiceRole-ap-southeast-2-LabPipeline	AWS Service: codepipeline
□	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked
□	AWSServiceRoleForCostOptimizationHub	AWS Service: cost-optimization-hub.
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linke
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service

3. Then select a trusted entity as AWS service.



Select trusted entity Info

Trusted entity type

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

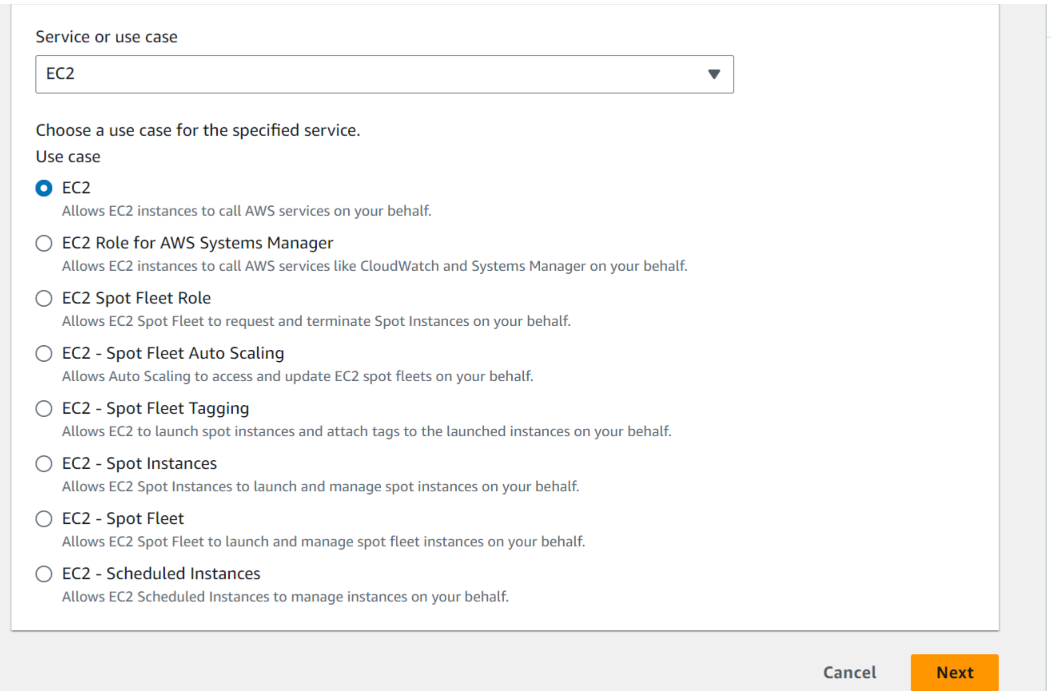
○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ 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.

4. Select use case as 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.

○ EC2 Role for AWS Systems Manager
Allows EC2 instances to call AWS services like CloudWatch and Systems Manager on your behalf.

○ EC2 Spot Fleet Role
Allows EC2 Spot Fleet to request and terminate Spot Instances on your behalf.

○ EC2 - Spot Fleet Auto Scaling
Allows Auto Scaling to access and update EC2 spot fleets on your behalf.

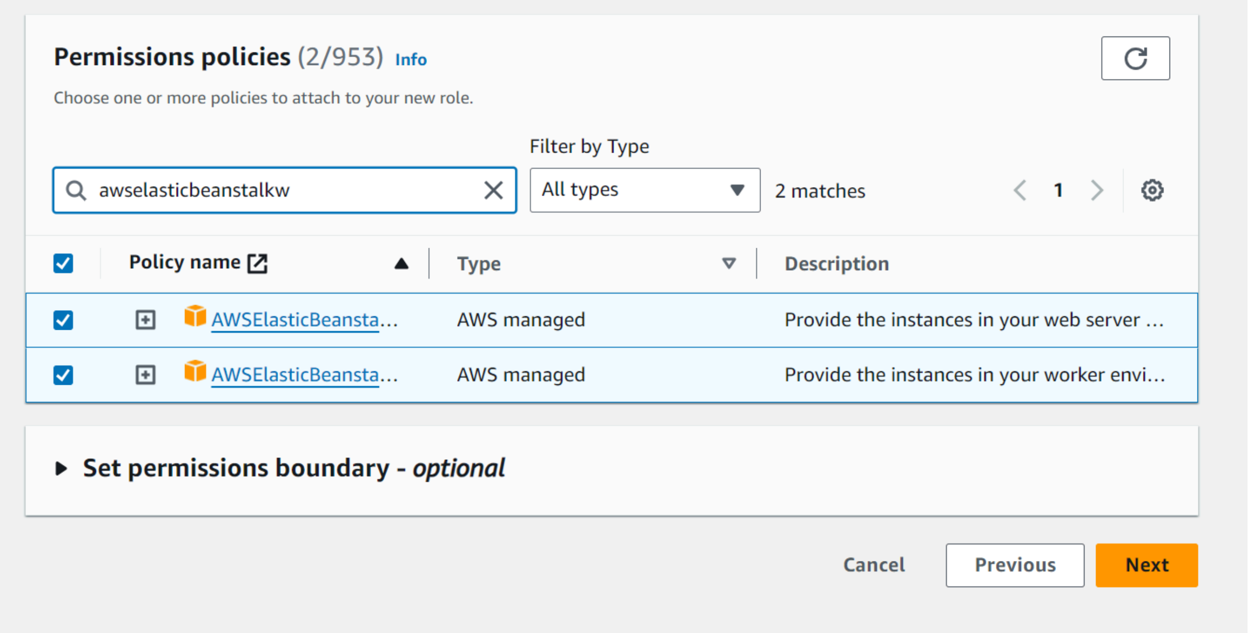
○ EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.

○ EC2 - Spot Instances
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.

○ EC2 - Spot Fleet
Allows EC2 Spot Fleet to launch and manage spot fleet instances on your behalf.

○ EC2 - Scheduled Instances
Allows EC2 Scheduled Instances to manage instances on your behalf.

5. Select permissions as AWS Elastic Beanstalk Web Tier.and AWS elastic Beanstalk worker tier.



Permissions policies (2/953) Info

Choose one or more policies to attach to your new role.

Search: awselasticbeanstalkw

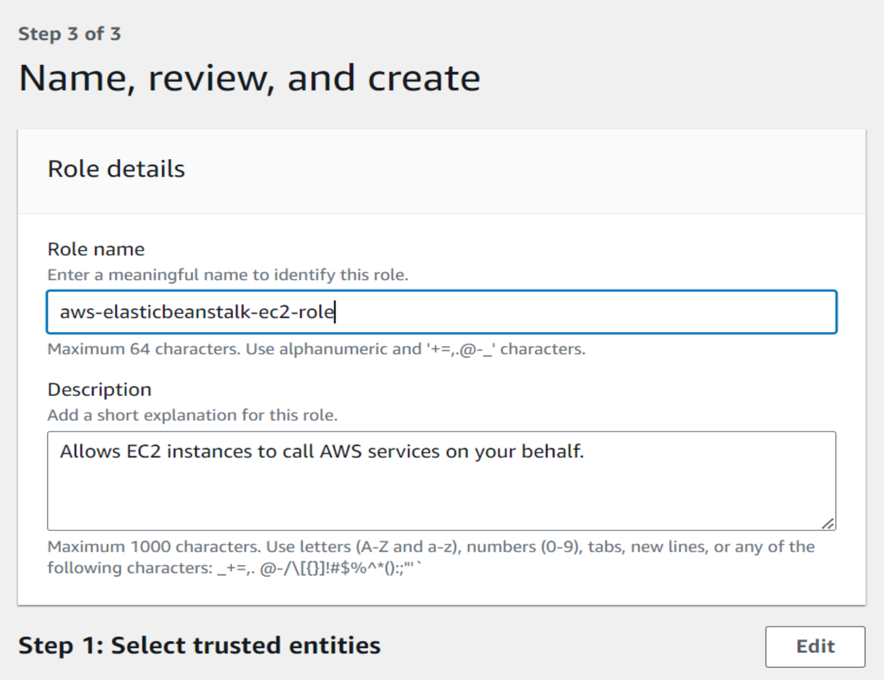
Filter by Type: All types ▼

2 matches

☑	Policy name	Type	Description
☑	⊞ AWSElasticBeansta…	AWS managed	Provide the instances in your web server …
☑	⊞ AWSElasticBeansta…	AWS managed	Provide the instances in your worker envi…

▶ **Set permissions boundary - *optional***

6. Give a name to Role. Here I have given my role name as aws -elasticbenstalk -ec2 role.



Step 3 of 3

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

aws-elasticbeanstalk-ec2-role

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

Description

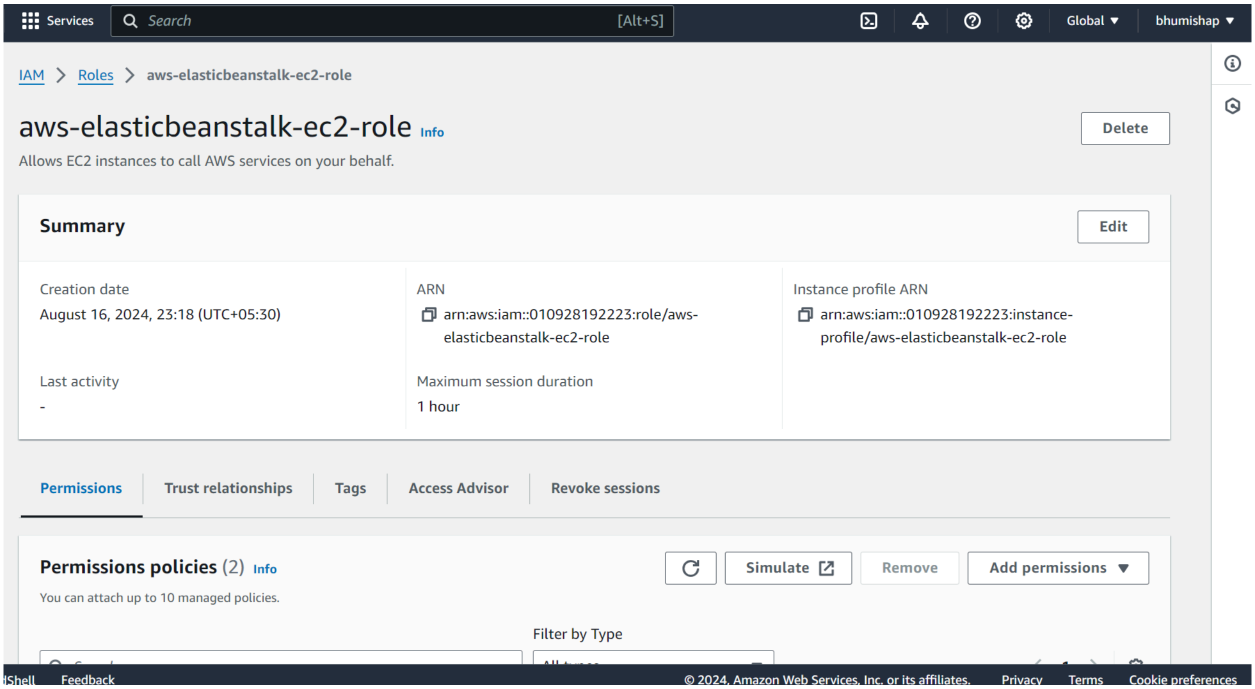
Add a short explanation for this role.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/\[{}]!#$%^*();:"'`

Step 1: Select trusted entities

7. Then the role gets created.



aws-elasticbeanstalk-ec2-role Info

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date: August 16, 2024, 23:18 (UTC+05:30)

ARN: arn:aws:iam::010928192223:role/aws-elasticbeanstalk-ec2-role

Instance profile ARN: arn:aws:iam::010928192223:instance-profile/aws-elasticbeanstalk-ec2-role

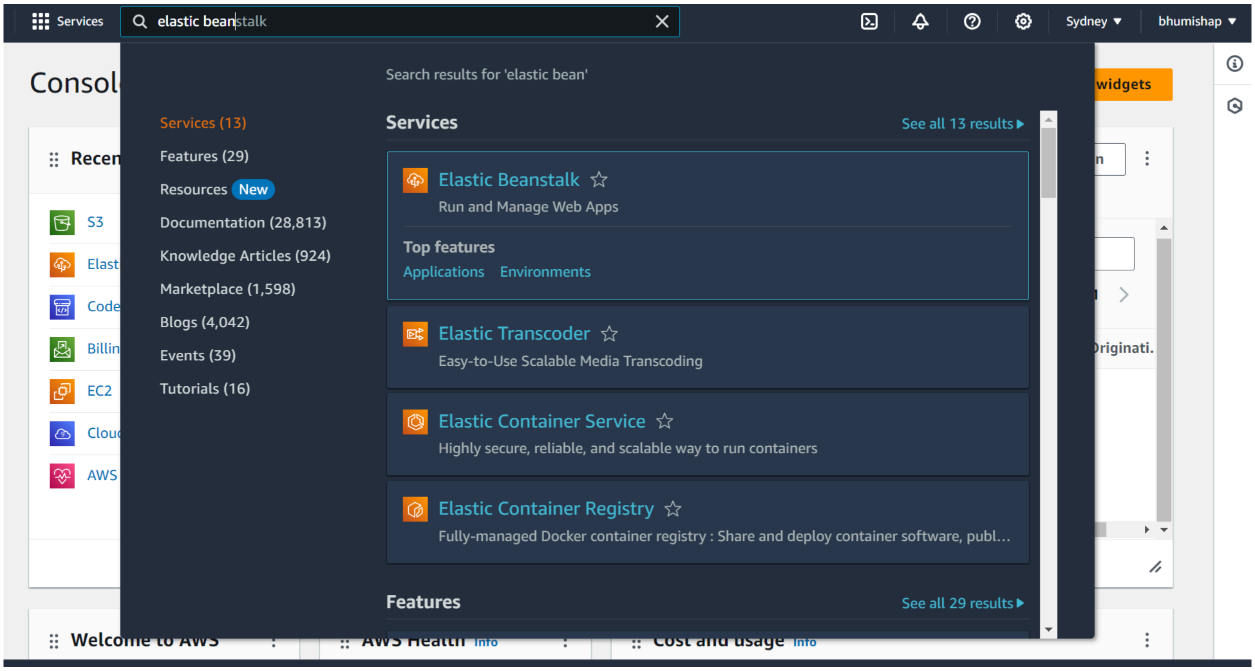
Last activity: -

Maximum session duration: 1 hour

Permissions policies (2) Info

You can attach up to 10 managed policies.

Step 2: Search for Elastic Beanstalk in the searchbar near services



Search results for 'elastic bean'

- Services (13)
- Features (29)
- Resources New
- Documentation (28,813)
- Knowledge Articles (924)
- Marketplace (1,598)
- Blogs (4,042)
- Events (39)
- Tutorials (16)

Services See all 13 results ▸

Elastic Beanstalk ☆
Run and Manage Web Apps

Top features
Applications Environments

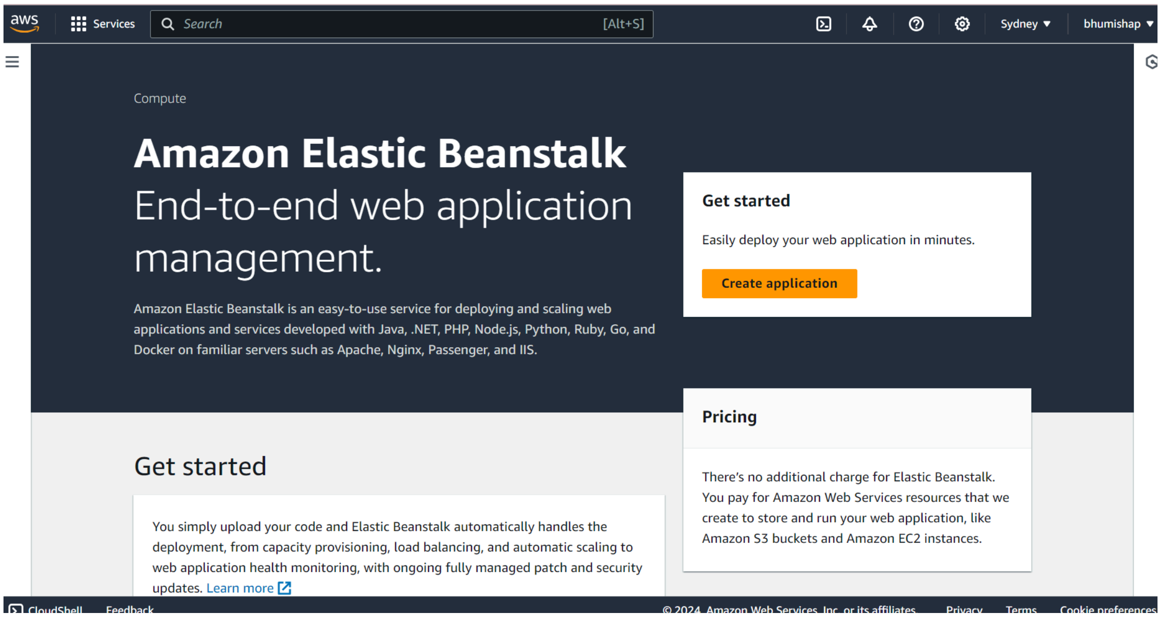
Elastic Transcoder ☆
Easy-to-Use Scalable Media Transcoding

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…

Features See all 29 results ▸

Step 3: Go to Elastic Beanstalk and click on Create Application



Compute

Amazon Elastic Beanstalk
End-to-end web application management.

Amazon Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.

Get started

Easily deploy your web application in minutes.

Create application

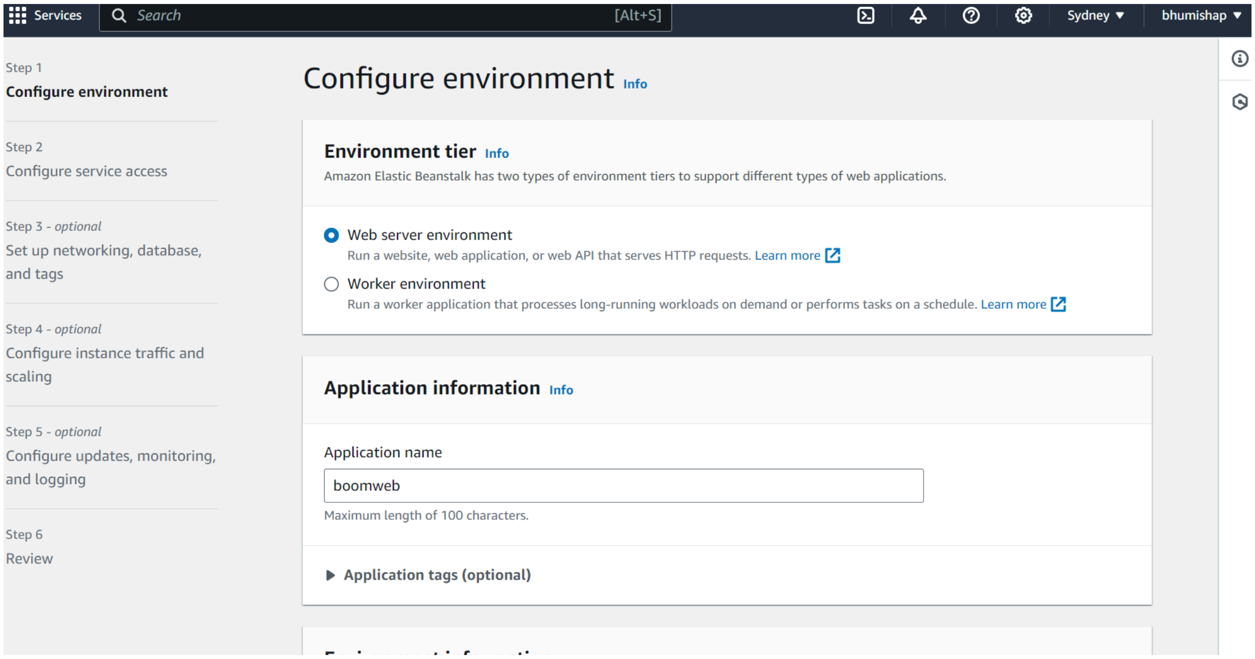
Pricing

There's no additional charge for Elastic Beanstalk. You pay for Amazon Web Services resources that we create to store and run your web application, like Amazon S3 buckets and Amazon EC2 instances.

Get started

You simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, and automatic scaling to web application health monitoring, with ongoing fully managed patch and security updates. Learn more

Step 4: Enter the name of your application. Scroll down and in the platform, select platform as PHP. Keep the application code as Sample Application. Set the instance to a single instance. Click on NEXT



Step 1
Configure environment

Step 2
Configure service access

Step 3 - *optional*
Set up networking, database, and tags

Step 4 - *optional*
Configure instance traffic and scaling

Step 5 - *optional*
Configure updates, monitoring, and logging

Step 6
Review

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

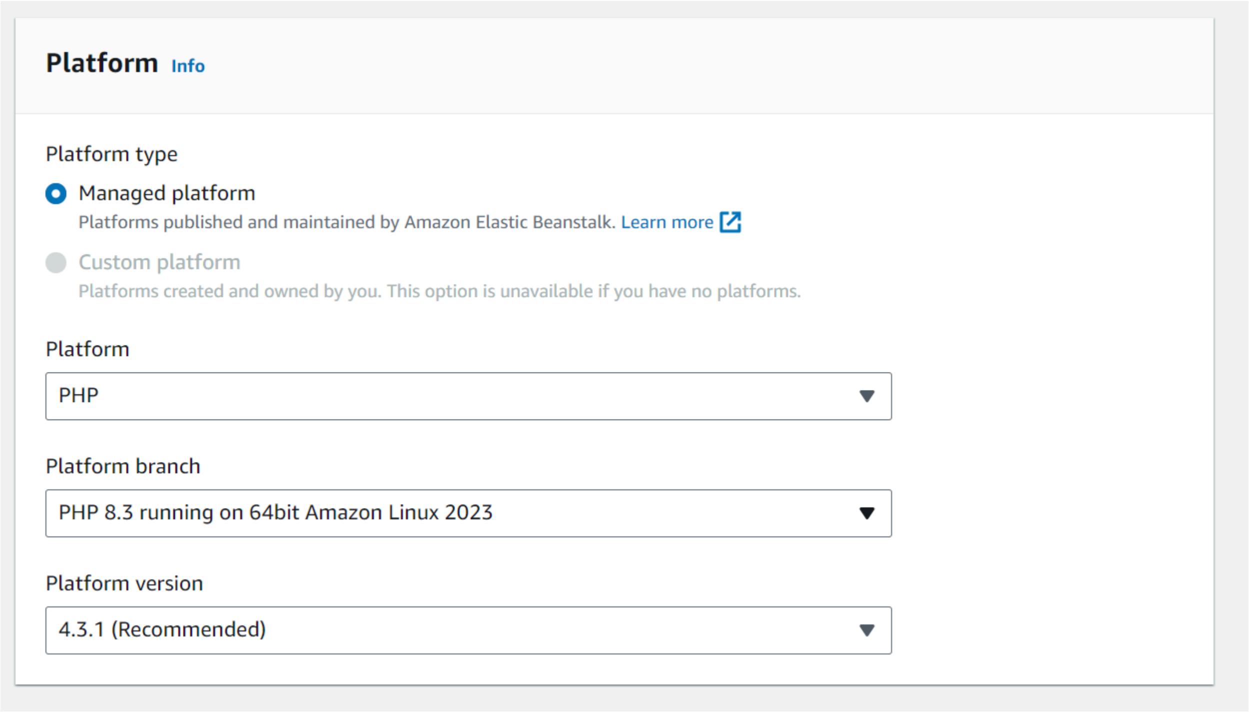
○ Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more

Application information Info

Application name: boomweb

Maximum length of 100 characters.

▶ Application tags (optional)



Platform Info

Platform type

● Managed platform
Platforms published and maintained by Amazon Elastic Beanstalk. Learn more

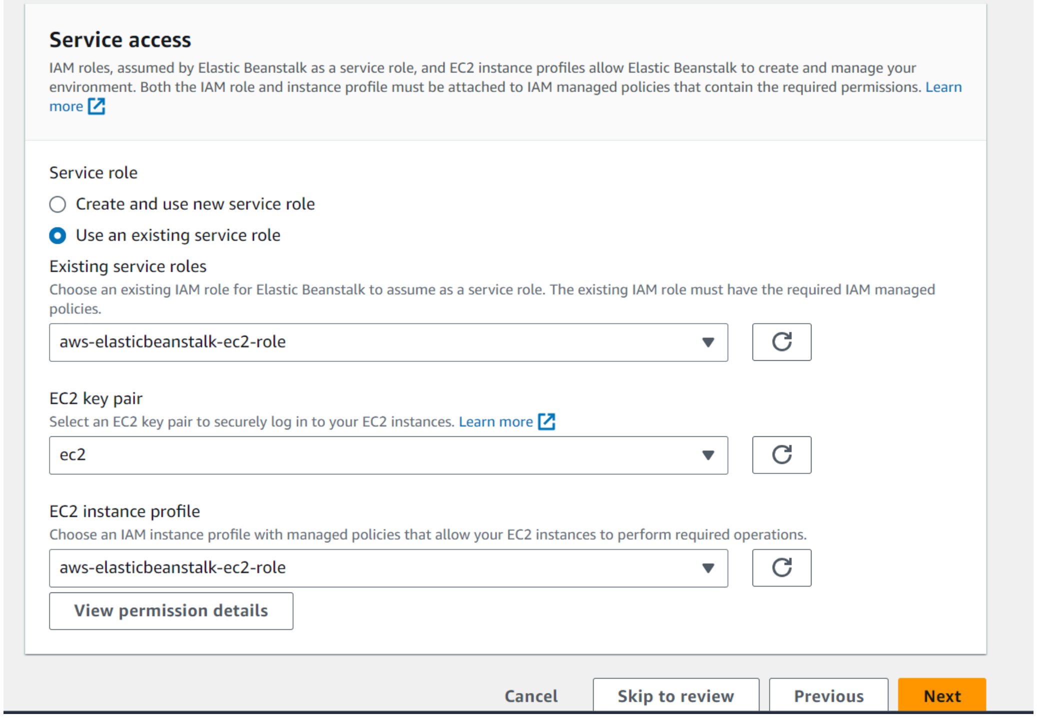
○ Custom platform
Platforms created and owned by you. This option is unavailable if you have no platforms.

Platform: PHP

Platform branch: PHP 8.3 running on 64bit Amazon Linux 2023

Platform version: 4.3.1 (Recommended)

Step 5: Use an existing service role and choose whatever service role is available on your account.



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-ec2-role

EC2 key pair

Select an EC2 key pair to securely log in to your EC2 instances. Learn more

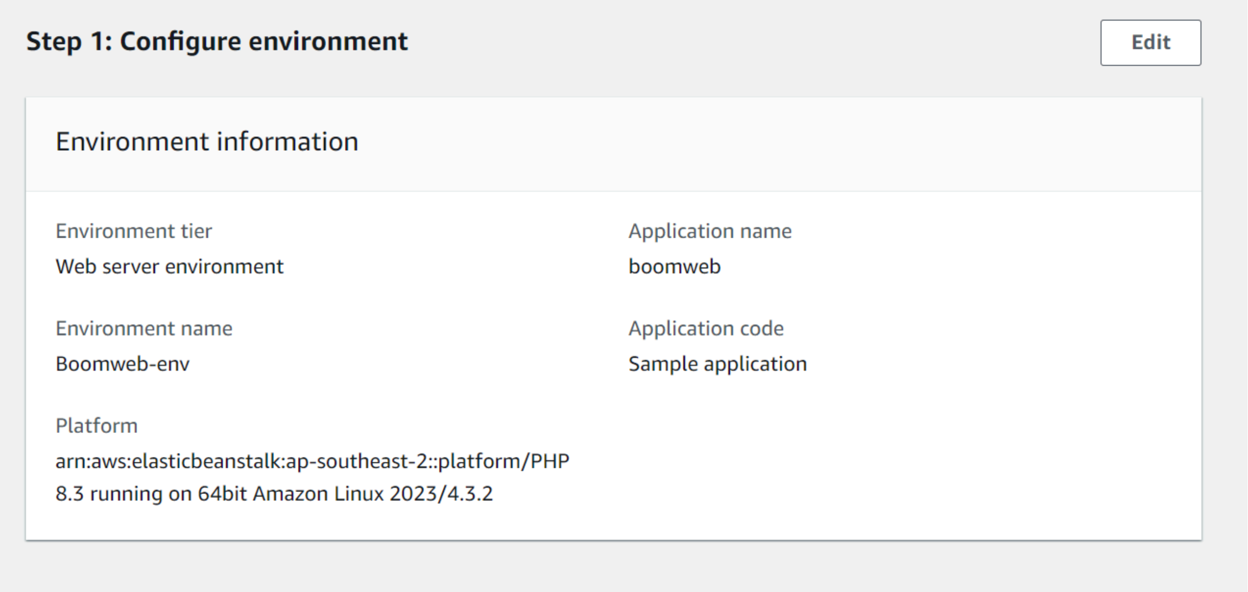
ec2

EC2 instance profile

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

aws-elasticbeanstalk-ec2-role

Step 6: Review the settings that you have set up for your application and submit your application.



Step 1: Configure environment

Environment information

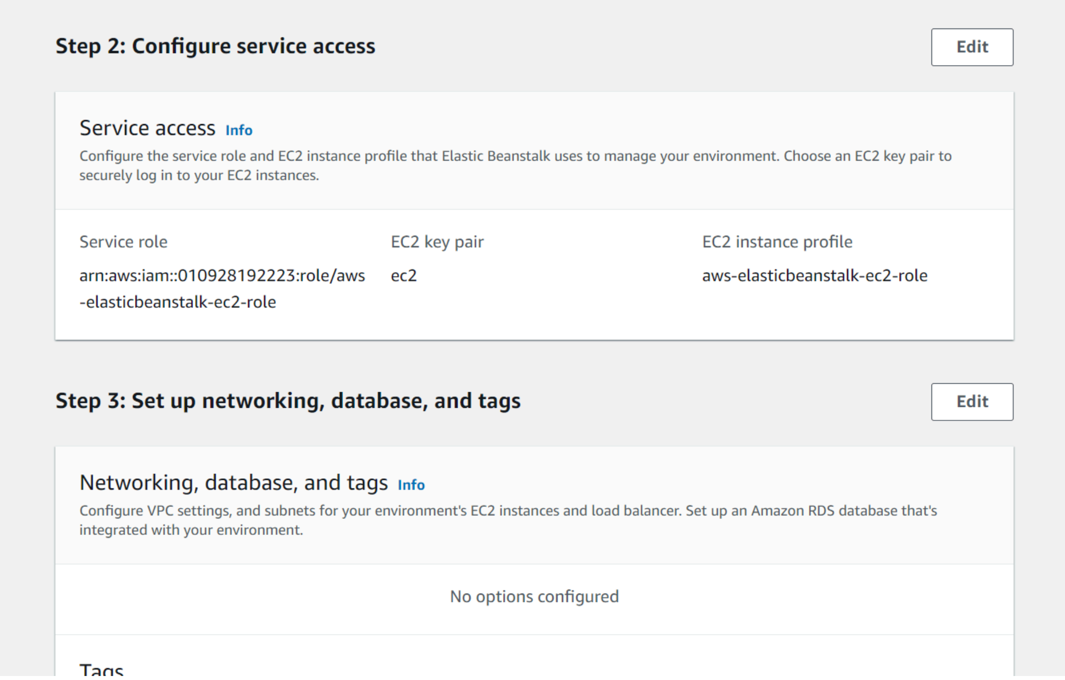
Environment tier
Web server environment

Application name
boomweb

Environment name
Boomweb-env

Application code
Sample application

Platform
arn:aws:elasticbeanstalk:ap-southeast-2::platform/PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2



Step 2: Configure service access

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 key pair	EC2 instance profile
arn:aws:iam::010928192223:role/aws-elasticbeanstalk-ec2-role	ec2	aws-elasticbeanstalk-ec2-role

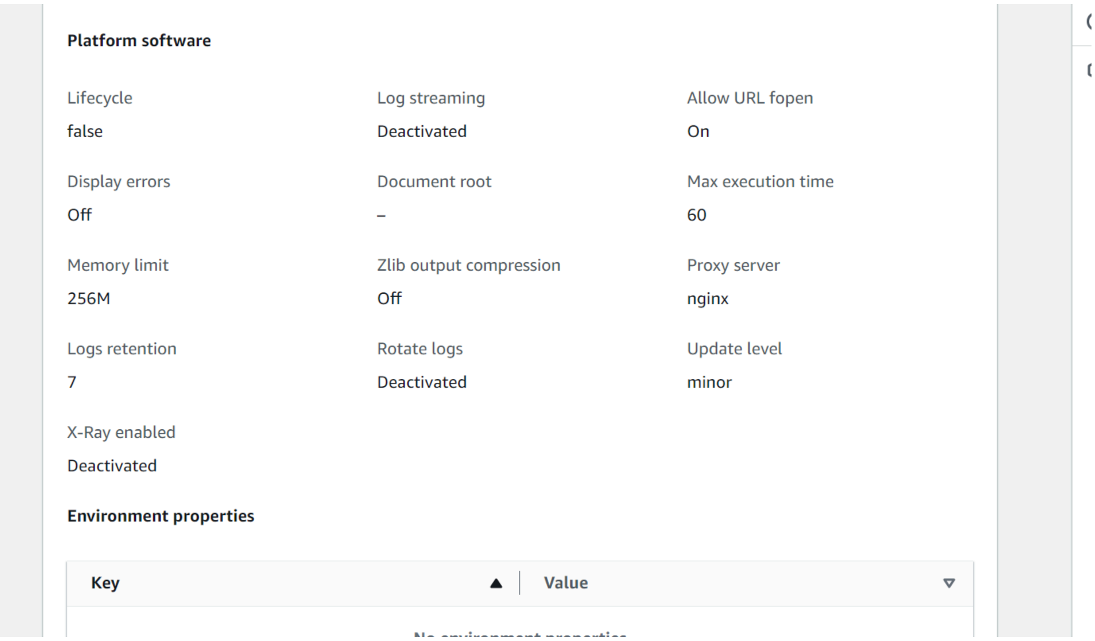
Step 3: Set up networking, database, and tags

Networking, database, and tags Info

Configure VPC settings, and subnets for your environment's EC2 instances and load balancer. Set up an Amazon RDS database that's integrated with your environment.

No options configured

Tags

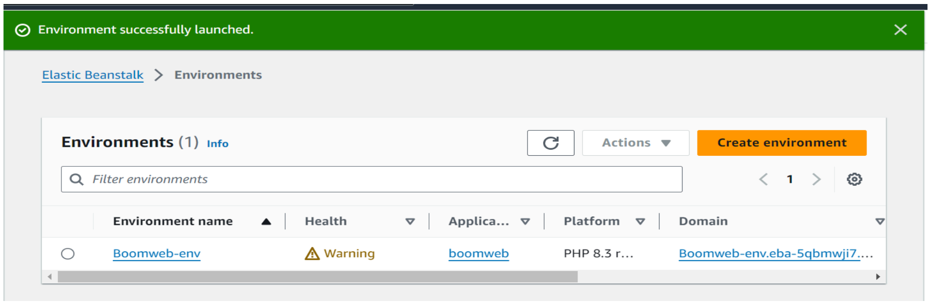


Platform software

Lifecycle	Log streaming	Allow URL fopen
false	Deactivated	On
Display errors	**Document root**	**Max execution time**
Off	–	60
Memory limit	**Zlib output compression**	**Proxy server**
256M	Off	nginx
Logs retention	**Rotate logs**	**Update level**
7	Deactivated	minor
X-Ray enabled		
Deactivated		

Environment properties

Key ▲	Value ▽
No environment properties	



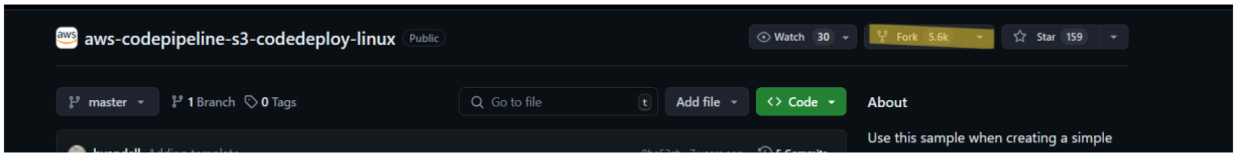
Environment successfully launched.

Elastic Beanstalk > Environments

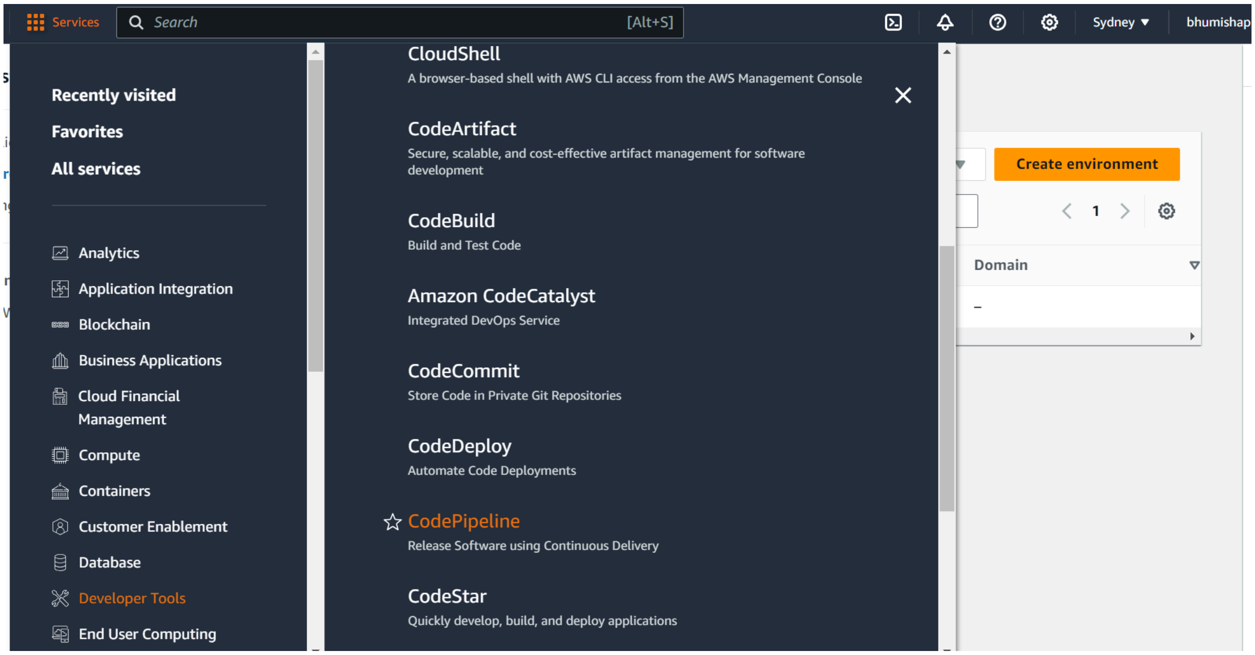
Environments (1) Info

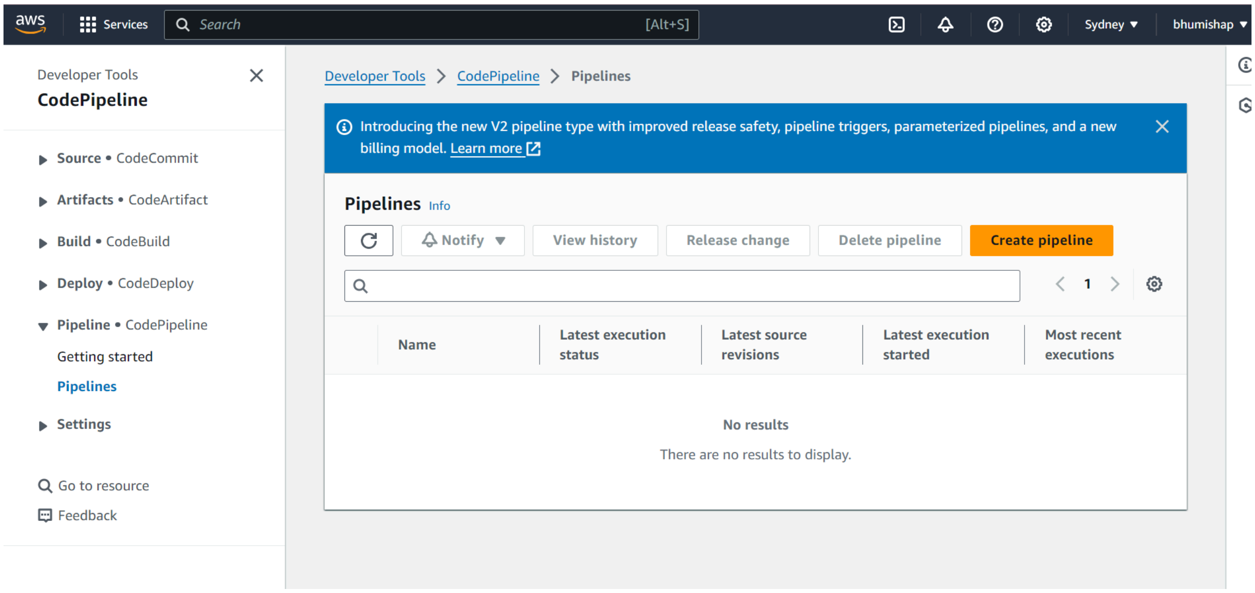
	Environment name ▲	Health ▽	Applica... ▽	Platform ▽	Domain ▽
○	Boomweb-env	⚠ Warning	boomweb	PHP 8.3 r...	Boomweb-env.eba-5qbmwji7....

Step 7: Go to the github link below. This is a github with a sample code for deploying a file on AWS CodePipeline. Fork this repository into your personal github.
https://github.com/aws-samples/aws-codepipeline-s3-codedeploy-linux

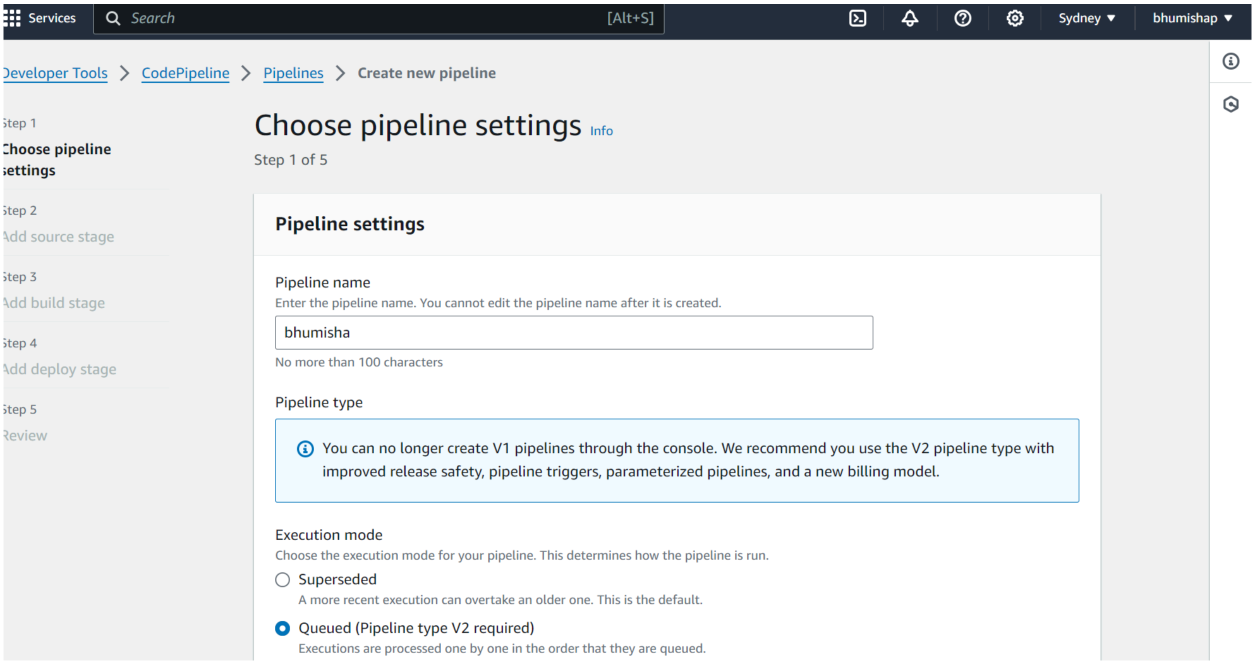


Step 8: Click on Create Pipeline.


Services
Search
[Alt+S]
Sydney
bhumishap
Recently visited
Favorites
All services
Analytics
Application Integration
Blockchain
Business Applications
Cloud Financial Management
Compute
Containers
Customer Enablement
Database
Developer Tools
End User Computing
CloudShell
A browser-based shell with AWS CLI access from the AWS Management Console
CodeArtifact
Secure, scalable, and cost-effective artifact management for software development
CodeBuild
Build and Test Code
Amazon CodeCatalyst
Integrated DevOps Service
CodeCommit
Store Code in Private Git Repositories
CodeDeploy
Automate Code Deployments
CodePipeline
Release Software using Continuous Delivery
CodeStar
Quickly develop, build, and deploy applications
Create environment
1
Domain
–



Step 9: Give a name to your Pipeline. A new service role would be created with the name of the Pipeline.



Choose pipeline settings Info

Step 1 of 5

Pipeline settings

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

bhumisha

No more than 100 characters

Pipeline type

You can no longer create V1 pipelines through the console. We recommend you use the V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model.

Execution mode

Choose the execution mode for your pipeline. This determines how the pipeline is run.

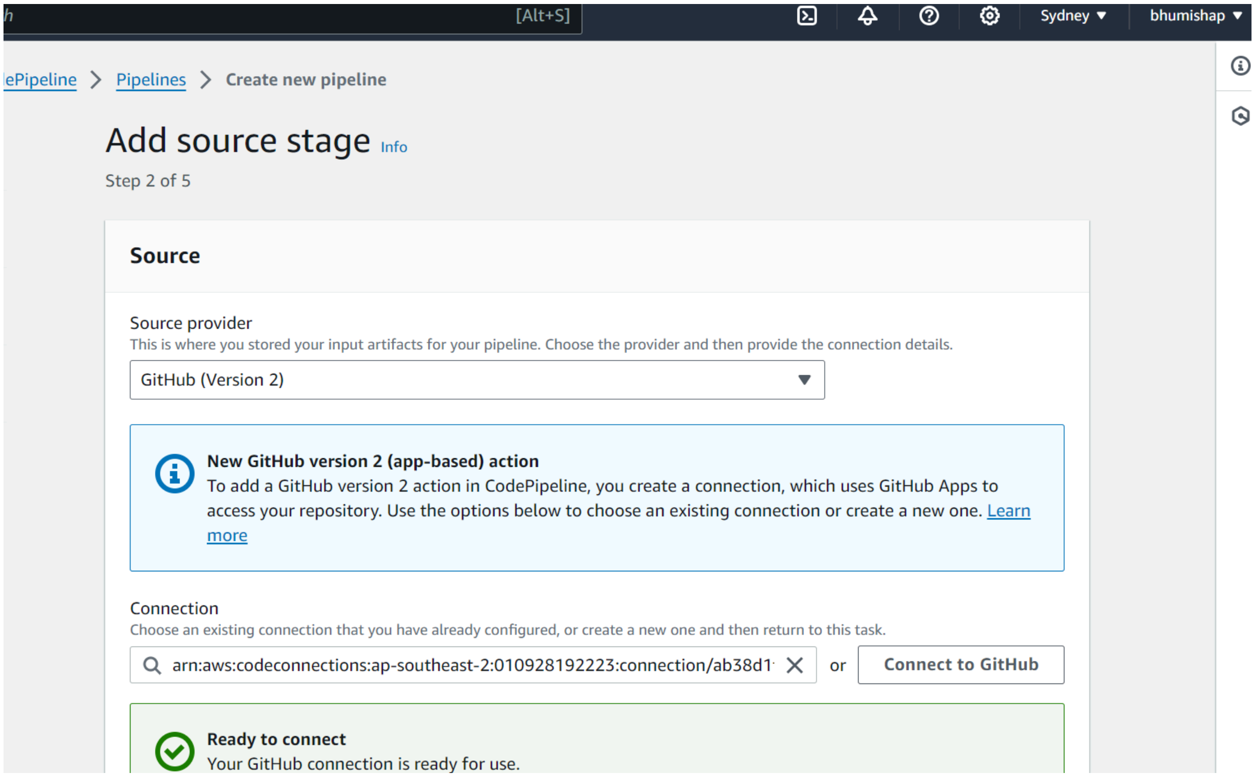
○ Superseded

A more recent execution can overtake an older one. This is the default.

● Queued (Pipeline type V2 required)

Executions are processed one by one in the order that they are queued.

Step 10: Select a source provider (as Github (Version 2)). Click on Connect to Github to connect your github.



Add source stage Info

Step 2 of 5

Source

Source provider

This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 2) ▼

New GitHub version 2 (app-based) action

To add a GitHub version 2 action in CodePipeline, you create a connection, which uses GitHub Apps to access your repository. Use the options below to choose an existing connection or create a new one. Learn more

Connection

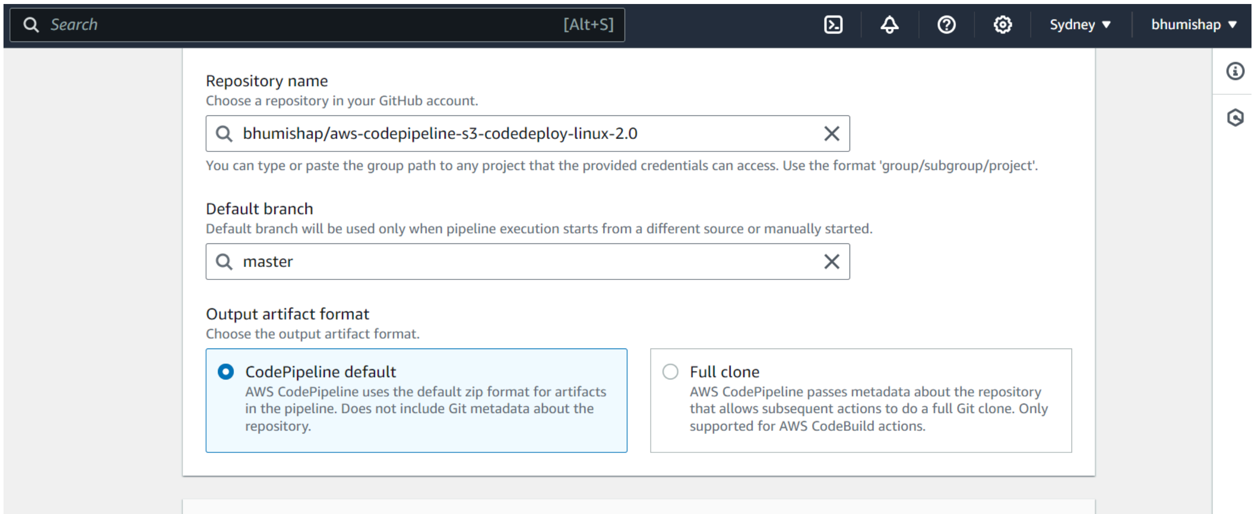
Choose an existing connection that you have already configured, or create a new one and then return to this task.

arn:aws:codeconnections:ap-southeast-2:010928192223:connection/ab38d1 ✕ or Connect to GitHub

Ready to connect

Your GitHub connection is ready for use.

Step 11: Select the repository that you had forked to your GitHub. After that select the branch on which the files are present (default is Master).



Repository name
Choose a repository in your GitHub account.
bhumishap/aws-codepipeline-s3-codedeploy-linux-2.0
You can type or paste the group path to any project that the provided credentials can access. Use the format 'group/subgroup/project'.

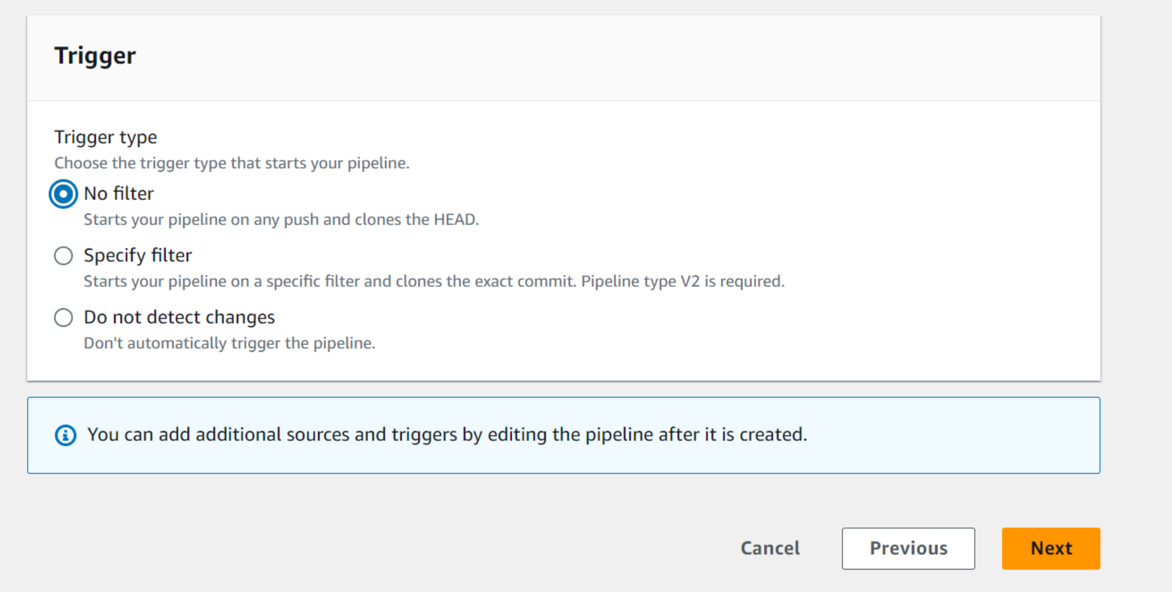
Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.
master

Output artifact format
Choose the output artifact format.

● CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

○ Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

Step 12: Set the Trigger type as no filter. This would allow it to the website to update as soon as some change is made in the github.



Trigger

Trigger type
Choose the trigger type that starts your pipeline.

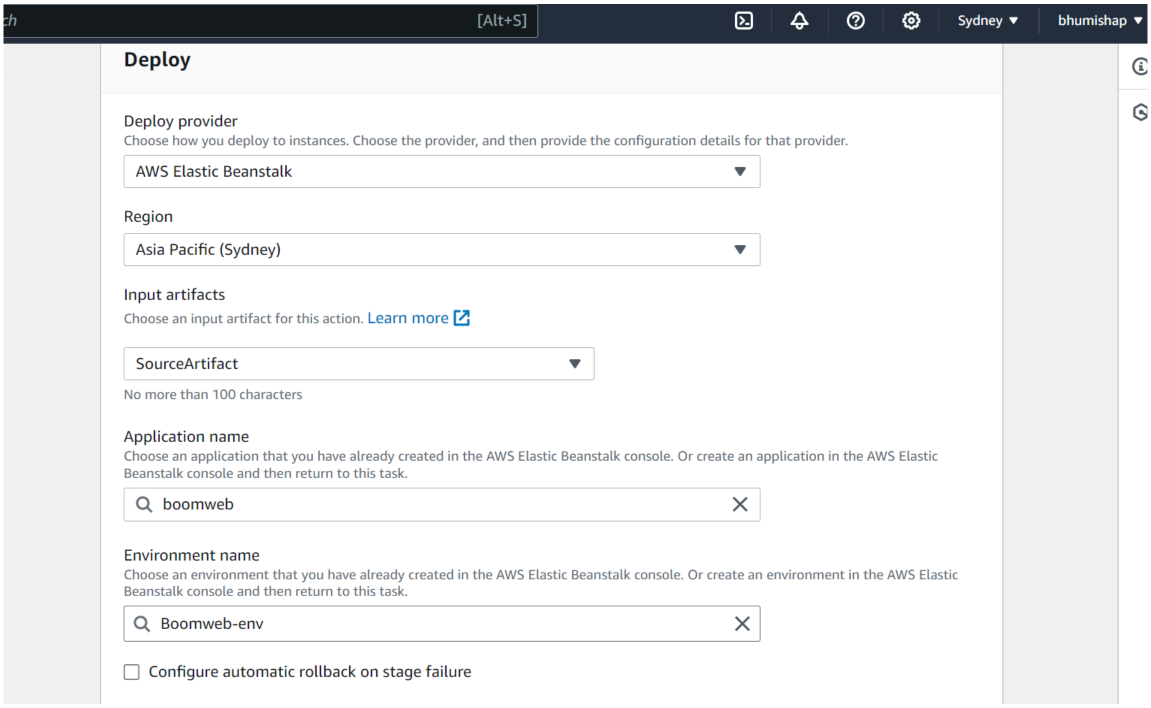
● No filter
Starts your pipeline on any push and clones the HEAD.

○ Specify filter
Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

○ Do not detect changes
Don't automatically trigger the pipeline.

You can add additional sources and triggers by editing the pipeline after it is created.

Step 13: Skip the build stage and go to Deploy. Select the deploy provider as AWS Elastic Beanstalk and Input Artifact as SourceArtifact. The application name would be the name of your Elastic Beanstalk. Then click on next.



Deploy

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
AWS Elastic Beanstalk

Region
Asia Pacific (Sydney)

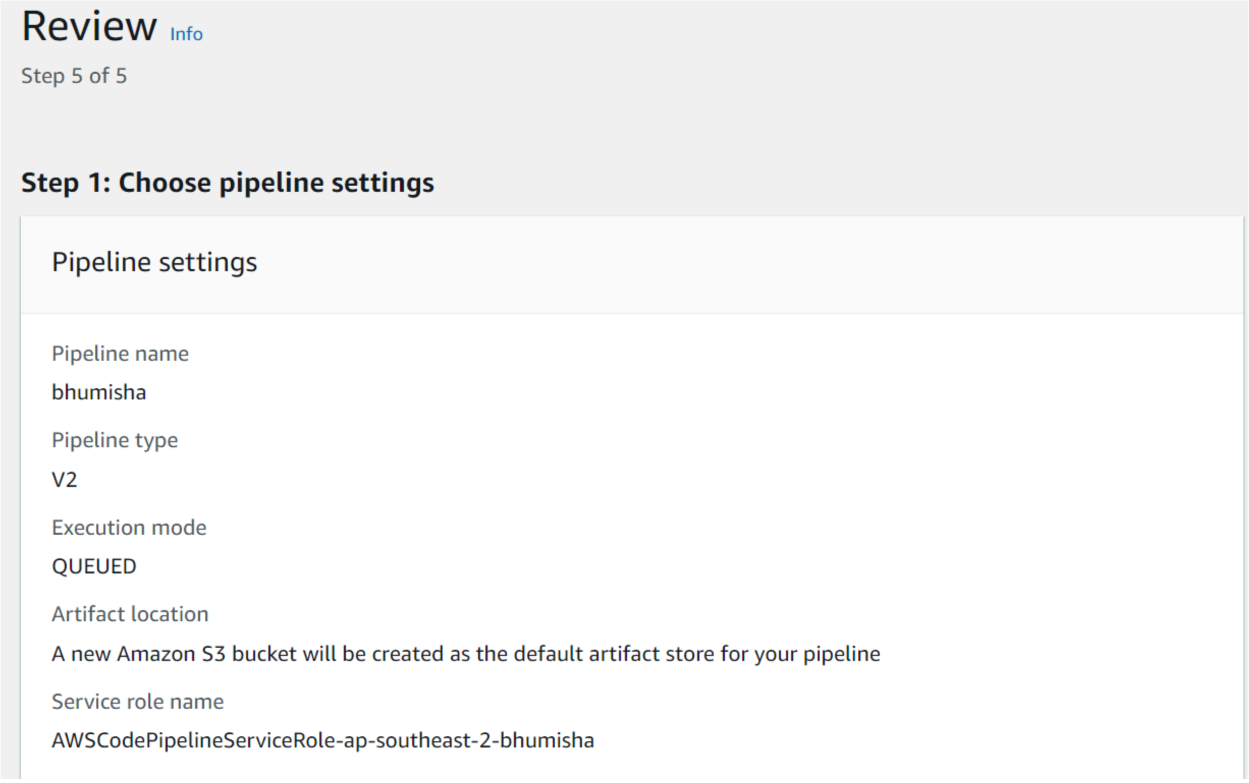
Input artifacts
Choose an input artifact for this action. Learn more
SourceArtifact
No more than 100 characters

Application name
Choose an application that you have already created in the AWS Elastic Beanstalk console. Or create an application in the AWS Elastic Beanstalk console and then return to this task.
boomweb

Environment name
Choose an environment that you have already created in the AWS Elastic Beanstalk console. Or create an environment in the AWS Elastic Beanstalk console and then return to this task.
Boomweb-env

□ Configure automatic rollback on stage failure

Step 14: Check all the information and click on create Pipeline.


Review Info
Step 5 of 5

Step 1: Choose pipeline settings

Pipeline settings

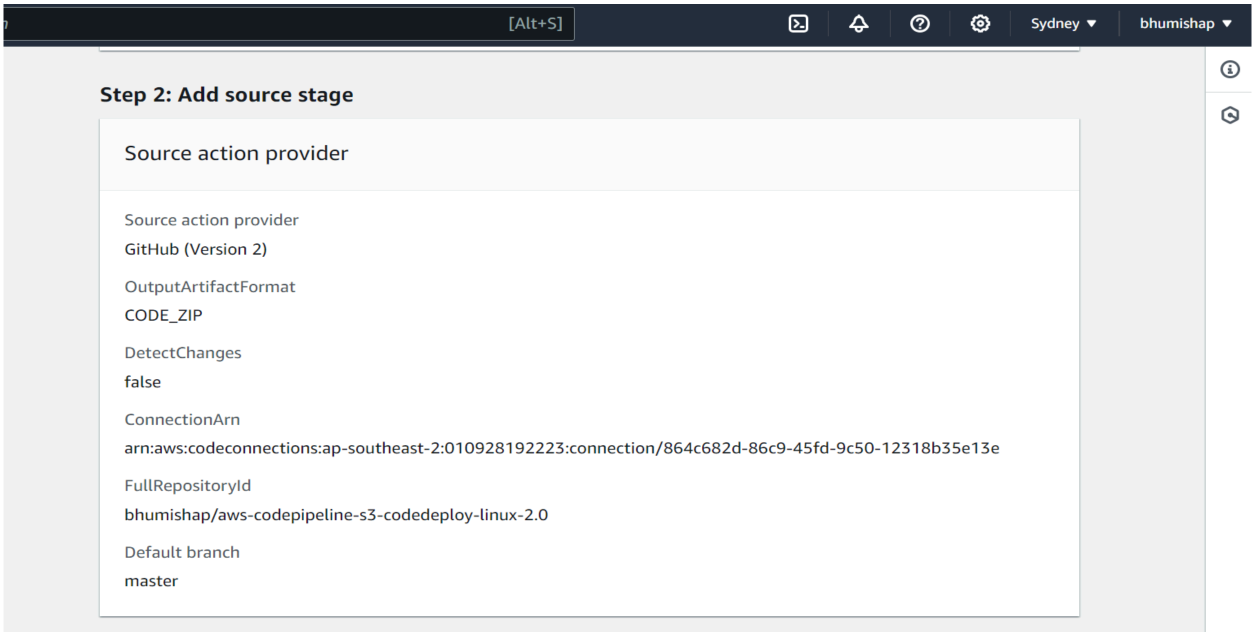
Pipeline name
bhumisha

Pipeline type
V2

Execution mode
QUEUED

Artifact location
A new Amazon S3 bucket will be created as the default artifact store for your pipeline

Service role name
AWSCodePipelineServiceRole-ap-southeast-2-bhumisha



Step 2: Add source stage

Source action provider

Source action provider
GitHub (Version 2)

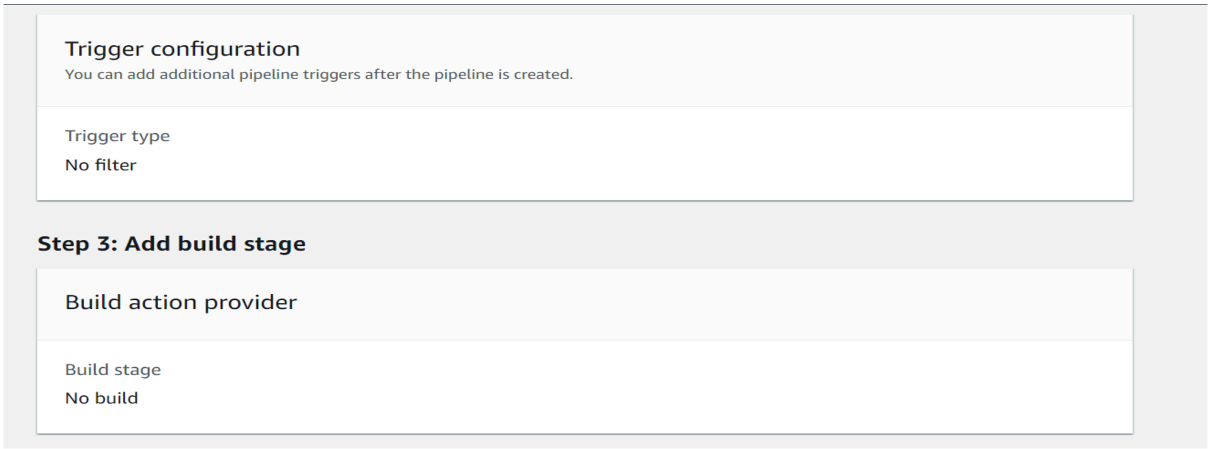
OutputArtifactFormat
CODE_ZIP

DetectChanges
false

ConnectionArn
arn:aws:codeconnections:ap-southeast-2:010928192223:connection/864c682d-86c9-45fd-9c50-12318b35e13e

FullRepositoryId
bhumishap/aws-codepipeline-s3-codedeploy-linux-2.0

Default branch
master



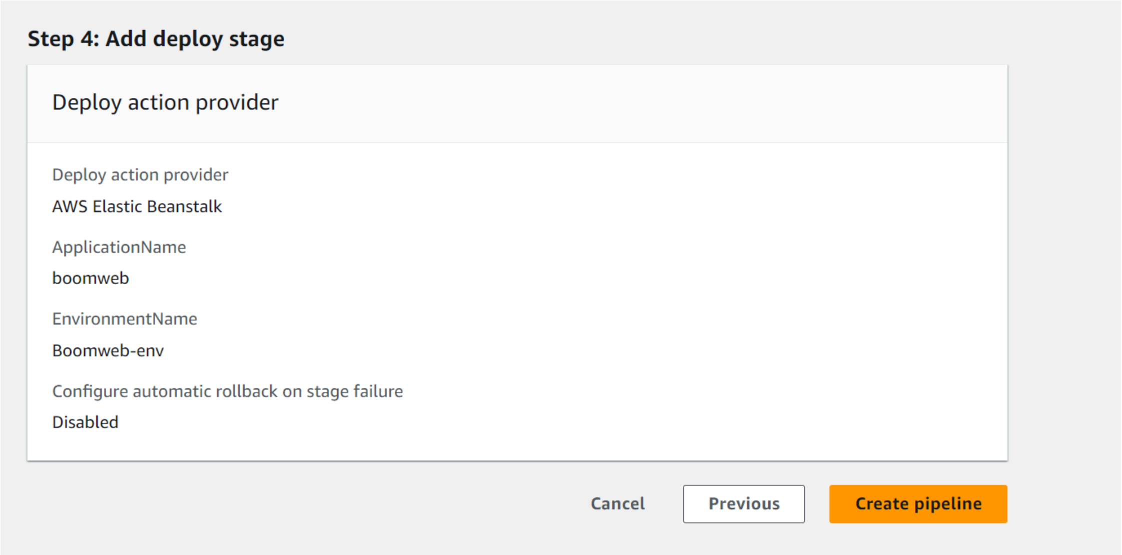
Trigger configuration
You can add additional pipeline triggers after the pipeline is created.

Trigger type
No filter

Step 3: Add build stage

Build action provider

Build stage
No build



Step 4: Add deploy stage

Deploy action provider

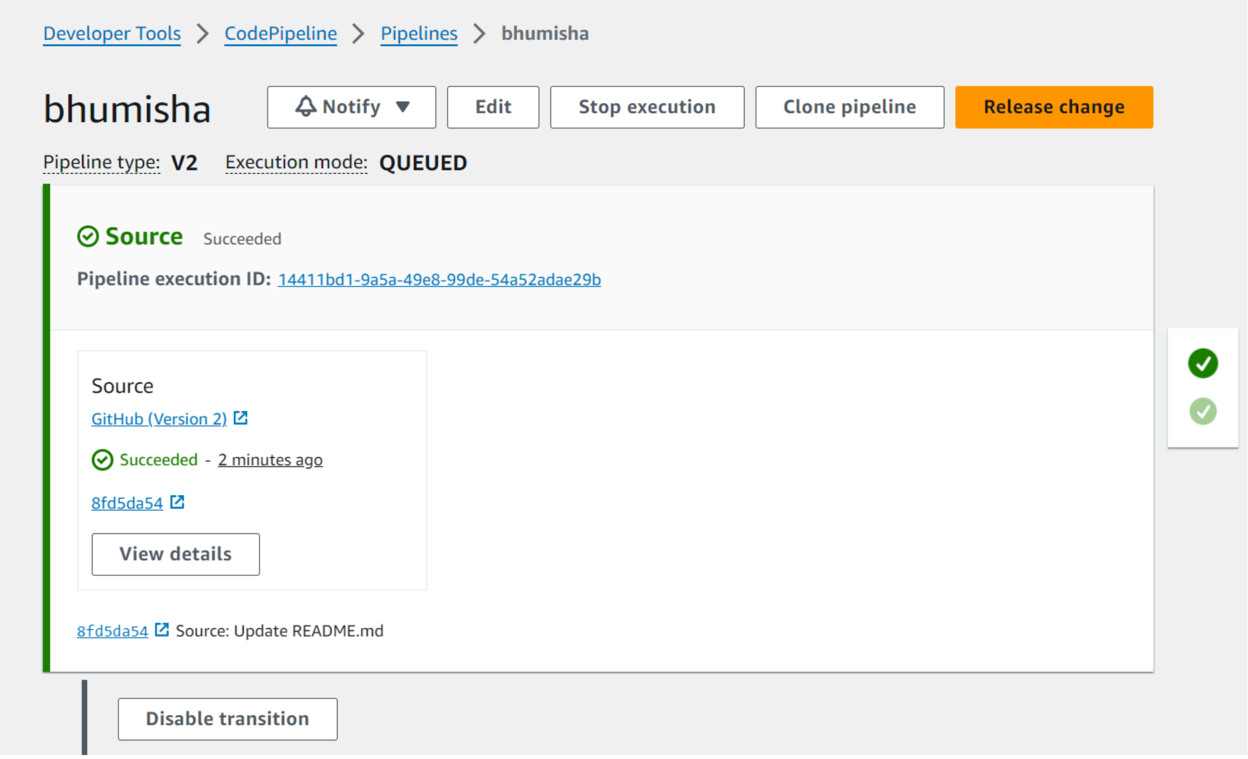
Deploy action provider
AWS Elastic Beanstalk

ApplicationName
boomweb

EnvironmentName
Boomweb-env

Configure automatic rollback on stage failure
Disabled

Step 15: If the pipeline is successfully deployed, this screen comes up where the source is set up and then it is transitioned to deploy.



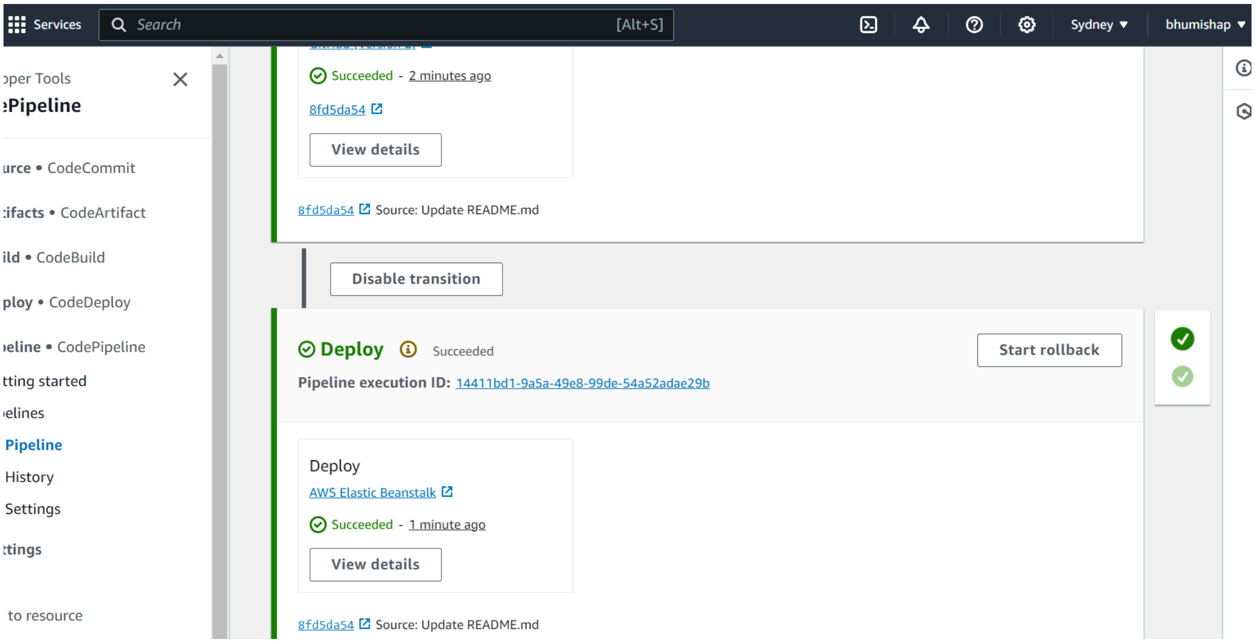
Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

Pipeline execution ID: 14411bd1-9a5a-49e8-99de-54a52adae29b

Source
GitHub (Version 2)
Succeeded - 2 minutes ago
8fd5da54

8fd5da54 Source: Update README.md



Succeeded - 2 minutes ago
8fd5da54

8fd5da54 Source: Update README.md

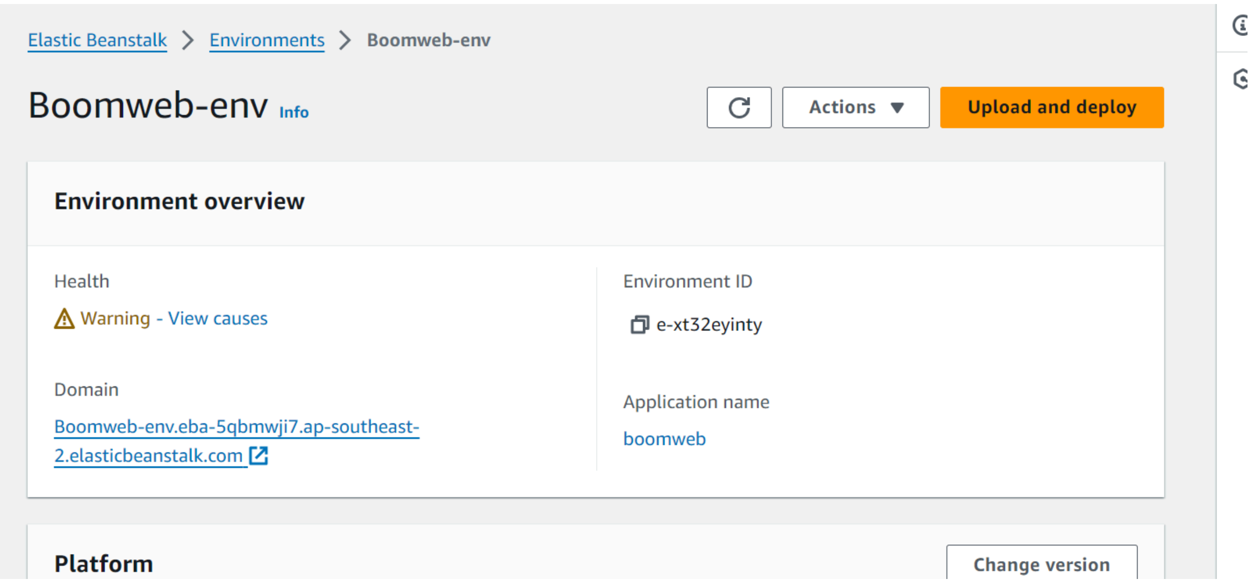
Deploy Succeeded

Pipeline execution ID: 14411bd1-9a5a-49e8-99de-54a52adae29b

Deploy
AWS Elastic Beanstalk
Succeeded - 1 minute ago

8fd5da54 Source: Update README.md

Step 16: In a few minutes the website will get hosted successfully. Then click on the url present over the environment created on Elastic Beanstalk.



Elastic Beanstalk > Environments > Boomweb-env

Boomweb-env Info

Environment overview

Health: Warning - View causes

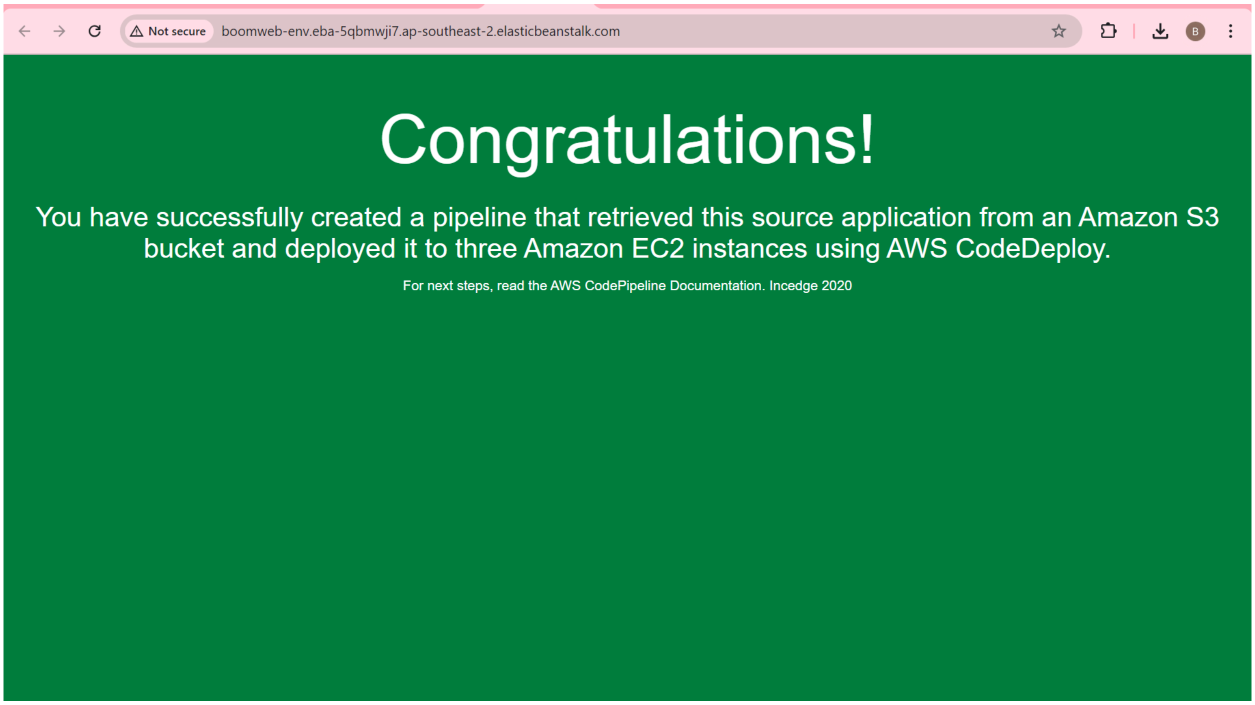
Environment ID: e-xt32eyinty

Domain: Boomweb-env.eba-5qbmwji7.ap-southeast-2.elasticbeanstalk.com

Application name: boomweb

Platform

Step 17: This will successfully show the sample website hosted.



Not secure boomweb-env.eba-5qbmwji7.ap-southeast-2.elasticbeanstalk.com

Congratulations!

You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

For next steps, read the AWS CodePipeline Documentation. Incedge 2020

If you can see this, that means that you successfully created an automated software using CodePipeline.