

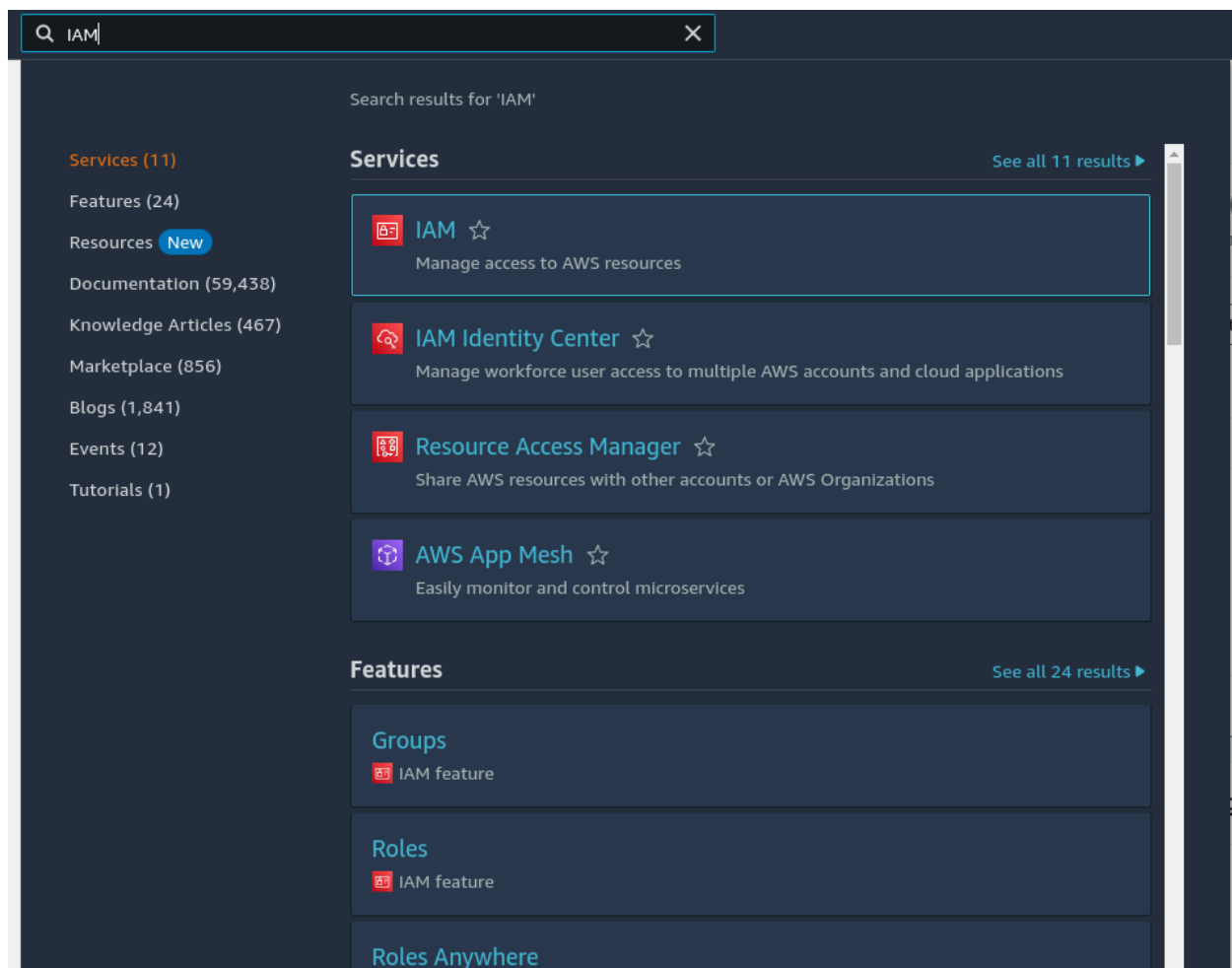
Experiment No. 2

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.

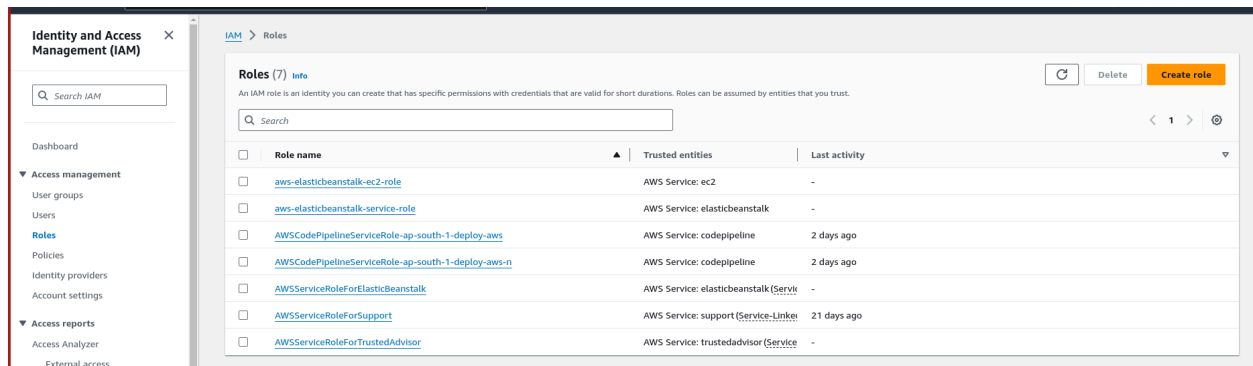
Steps:

Initially we will create a new Role in IAM

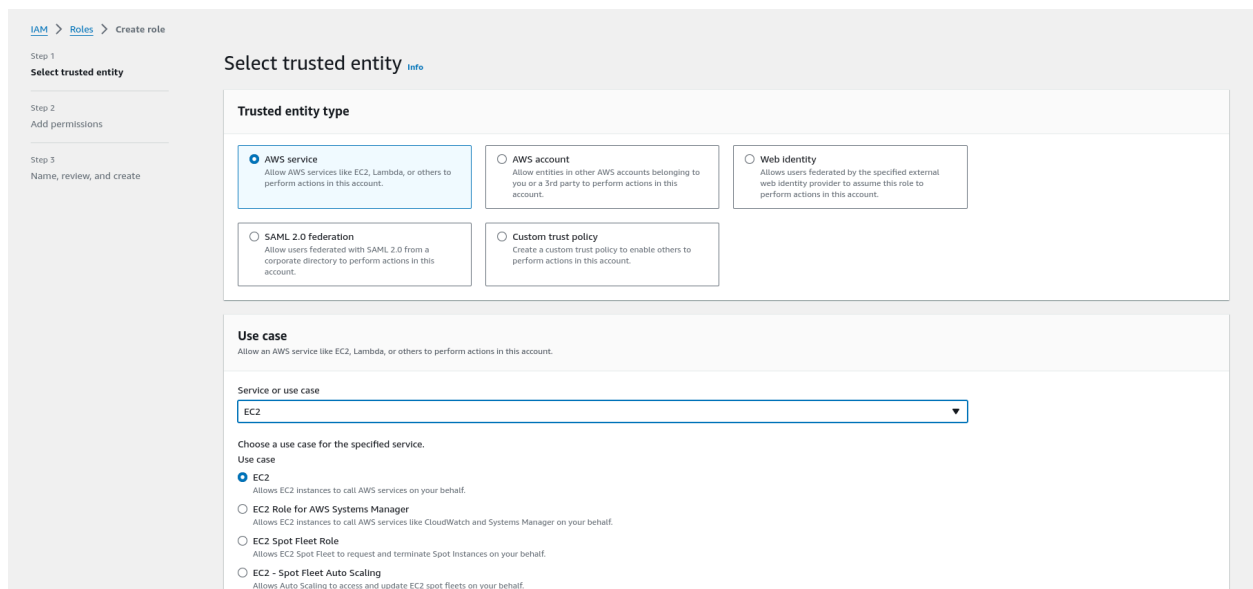
1. Search for IAM in search box



2. Go to Roles, and select AWSElasticBeanstalkWebtier and AWSElasticBeanstalkWorkertier, and create a role



3. Select entity type as AWS Service



The screenshot shows the 'Step 2: Add permissions' section of the AWS IAM console. It features a table titled 'Permissions policy summary' with two rows of AWS managed policies. Below the table is the 'Step 3: Add tags' section, which includes a link for 'Add tags - optional info' and a button to 'Add new tag'. At the bottom right, there are 'Cancel', 'Previous', and 'Create role' buttons.

Policy name	Type	Attached as
AWSElasticBeanstalkWebTier	AWS managed	Permissions policy
AWSElasticBeanstalkWorkerTier	AWS managed	Permissions policy

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.

Cancel Previous **Create role**

4. Give name to newly created role such as **aws-elasticbeanstalk-ec2-role**

The screenshot shows the 'Name, review, and create' step of the AWS IAM console. It includes a sidebar with navigation links for 'IAM > Roles > Create role', 'Step 1: Select trusted entity', 'Step 2: Add permissions', and 'Step 3: Name, review, and create'. The main area contains 'Role details' with a 'Role name' field (containing 'aws-elasticbeanstalk-ec2-role') and a 'Description' field (containing 'Allows EC2 instances to call AWS services on your behalf.').

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

Step 1
[Select trusted entity](#)

Step 2
[Add permissions](#)

Step 3
Name, review, and create

Name, review, and create

Role details

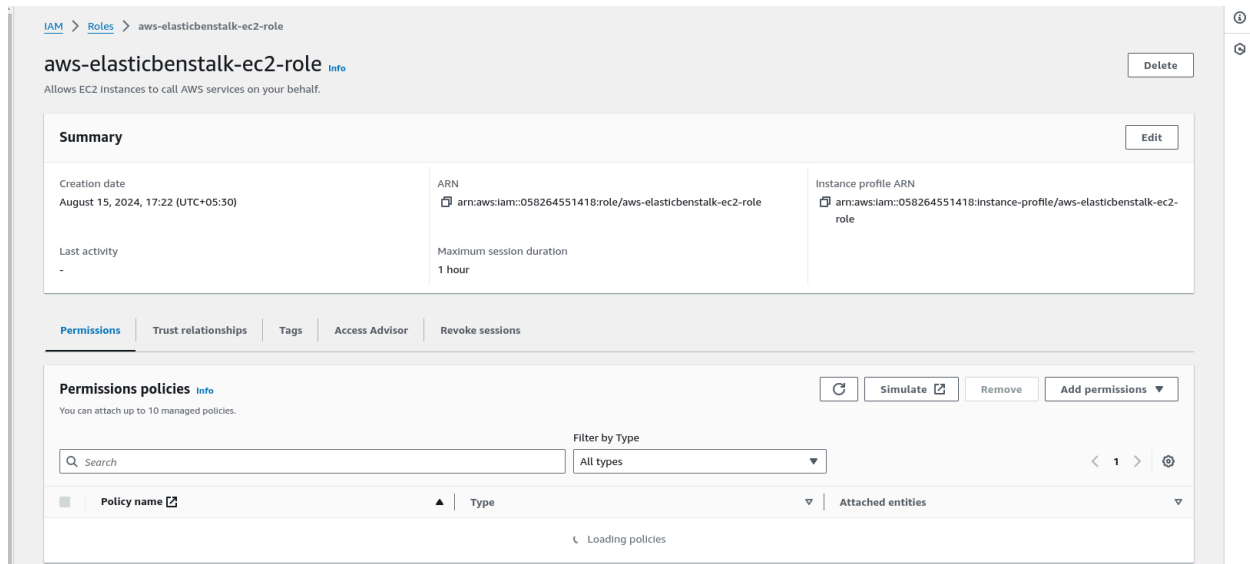
Role name
Enter a meaningful name to identify this role.

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

Description
Add a short explanation for this role.

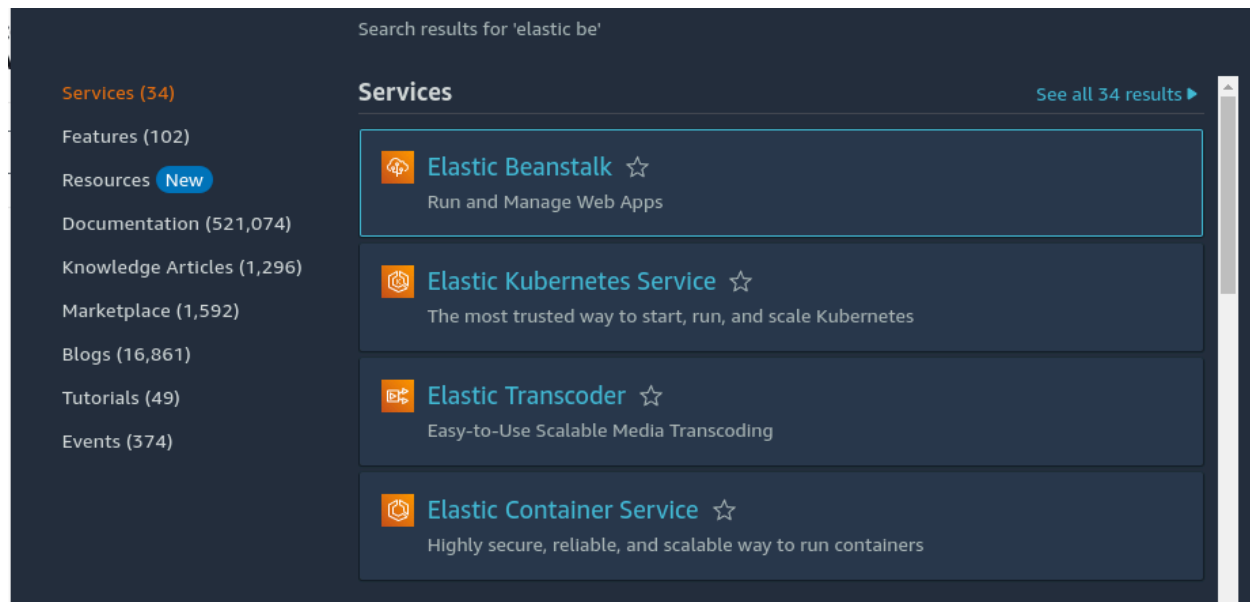
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+*, @-/[]!#\$%^&*()~`"

5. Summary of newly created role



Now we have to create Elastic Beanstalk Environment

6. Search **Elastic Beanstalk** and proceed with it



7. Create a new Environment and name it.

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

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

.ap-south-1.elasticbeanstalk.com

8. Select Platform as PHP, Application code as **Sample Application**, presets **Single Instance**

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

4.3.2 (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.

Presets [Info](#)

Start from a preset that matches your use case or choose custom configuration to unset recommended values and use the service's default values.

Configuration presets

☒ **Single instance (free tier eligible)**

☐ Single instance (using spot instance)

☐ High availability

☐ High availability (using spot and on-demand instances)

☐ Custom configuration

9. Under Service access settings, select **Use an existing service role**.
Name service role as **aws-elasticbeanstalk-service-role** and EC2

instance profile as **aws-elasticbeanstalk-ec2-role** which we had created earlier

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 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

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-service-role

EC2 key pair

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

ubuntu-linux

EC2 instance profile

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

aws-elasticbeanstalk-ec2-role

View permission details

Cancel

Skip to review

Previous

Next

10. Keep below settings as it as

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 instance traffic and scaling - *optional* [Info](#)

▼ Instances [Info](#)

Configure the Amazon EC2 instances that run your application.

Root volume (boot device)

Root volume type

(Container default) ▼

Size

The number of gigabytes of the root volume attached to each instance.

8

GB

IOPS

Input/output operations per second for a provisioned IOPS (SSD) volume.

100

IOPS

Throughput

The desired throughput to provision for the Amazon EBS root volume attached to your environment's EC2 instance

125

MIB/s

Amazon CloudWatch monitoring

The time interval between when metrics are reported from the EC2 instances

Monitoring interval

5 minute ▼

Instance metadata service (IMDS)

Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. [Learn more](#) [↗](#)

IMDSv1

With the current setting, the environment enables only IMDSv2.

☒ Deactivated

EC2 security groups

Select security groups to control traffic.

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 updates, monitoring, and logging - *optional* [Info](#)

▼ **Monitoring** [Info](#)

Health reporting

Enhanced health reporting provides free real-time application and operating system monitoring of the instances and other resources in your environment. The **EnvironmentHealth** custom metric is provided free with enhanced health reporting. Additional charges apply for each custom metric. For more information, see [Amazon CloudWatch Pricing](#).

System

☐ Basic

☒ Enhanced

CloudWatch Custom Metrics - Instance

Choose metrics ▼

CloudWatch Custom Metrics - Environment

Choose metrics ▼

Health event streaming to CloudWatch Logs

Configure Elastic Beanstalk to stream environment health events to CloudWatch Logs. You can set the retention up to a maximum of ten years and configure Elastic Beanstalk to delete the logs when you terminate your environment.

Log streaming

☐ Activated (standard CloudWatch charges apply.)

Retention

7 ▼

Lifecycle

Keep logs after terminating environment ▼

▼ **Managed platform updates** [Info](#)

Activate managed platform updates to apply platform updates automatically during a weekly maintenance window that you choose. Your application stays available during the update process.

Managed updates

☒ Activated

11. Our environment is created successfully

Elastic Beanstalk

Applications

Environments

Change history

▼ Application: alok-app

Application versions

Saved configurations

▼ Environment: Alok-app-env

Go to environment

Configuration

Events

Environment successfully launched.

Elastic Beanstalk > Environments > Alok-app-env

Alok-app-env [Info](#)

Environment overview

Health

Ok

Domain

Alok-app-env.eba-wcjuvikb.ap-south-1.elasticbeanstalk.com

Environment ID

e-qsf6h4zm4q

Application name

alok-app

Platform

Change version

Platform

PHP 8.3 running on 64bit Amazon Linux 2023/4.3.2

Running version

-

Platform state

Supported

Events

Health

Logs

Monitoring

Alarms

Managed updates

Tags

12. Now we would deploy a codepipeline. For below repo in your github
Fork this repository in your github.

The screenshot shows the GitHub repository page for 'aws-codepipeline-s3-codedeploy-linux-2.0' by 'imoisharma'. The repository is public and has 20 commits. The file list includes:

File	Description	Time
.github	Adding template	7 years ago
dist	Added dist folder	9 years ago
scripts	s3 setup and s3 set cahce control scripts	3 years ago
CODE_OF_CONDUCT.md	Adding CONTRIBUTING/CoC	7 years ago
CONTRIBUTING.md	Adding CONTRIBUTING/CoC	7 years ago
LICENSE	Added AWS CodePipeline Sample	9 years ago
README.md	Update README.md	3 years ago
app-specification.yml	Create app-spec config file	3 years ago
appspec.yml	Update appspec.yml	3 years ago
index.html	Update index.html	3 years ago

Below the repository page, the 'Create a new fork' section is visible. It includes a description of forking, a form to create a fork, and a 'Create fork' button.

Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (*).

Owner *

aaaaalok

Repository name *

aws-codepipeline-s3-cod

aws-codepipeline-s3-codedeploy-linux-2.0 is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Use this sample when creating a simple pipeline in AWS CodePipeline while following the Simple Pipeline Wa

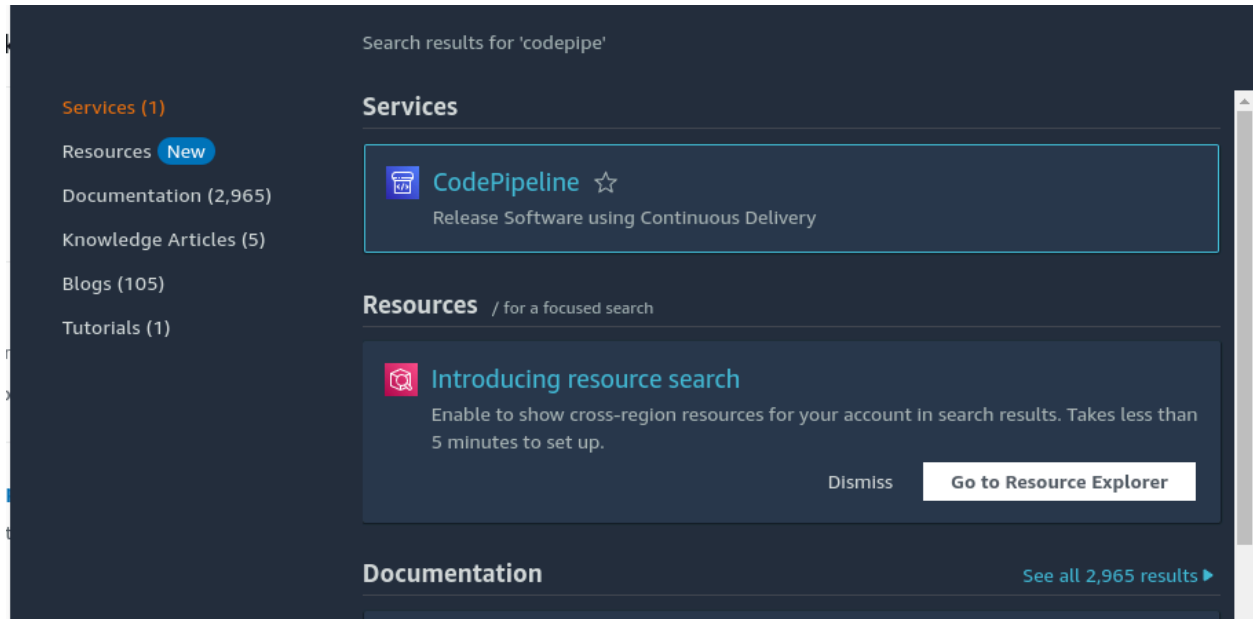
☒ Copy the master branch only

Contribute back to imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0 by adding your own branch. [Learn more.](#)

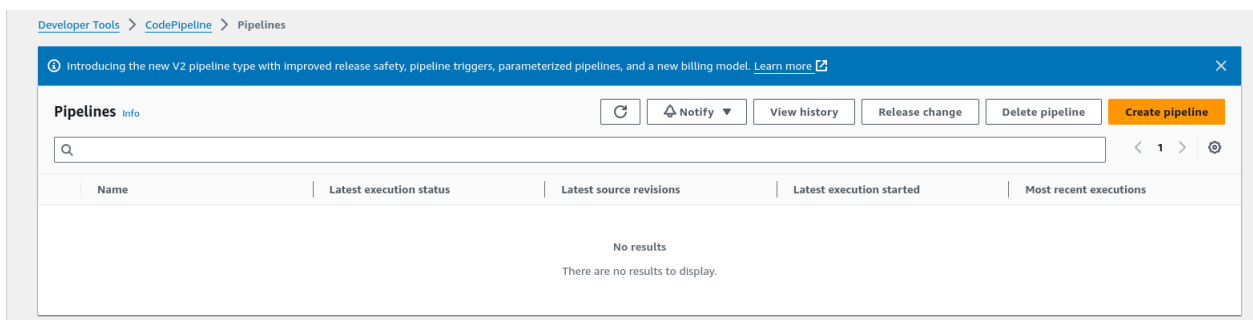
☐ You are creating a fork in your personal account.

Create fork

13. After forking, Now we would create a CodePipeline. Goto CodePipeline under services section



14. Create a new Pipeline



15. Name the pipeline leaving rest settings to its default

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

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.

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.

☐ Parallel (Pipeline type V2 required)
Executions don't wait for other runs to complete before starting or finishing.

Service role

☒ New service role
Create a service role in your account

☐ Existing service role
Choose an existing service role from your account

Role name


Type your service role name

☒ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

16. Under Source stage, select **Source Provider** as **GitHub (Version 2)**


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.

X or

 **Ready to connect**
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.

X

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.

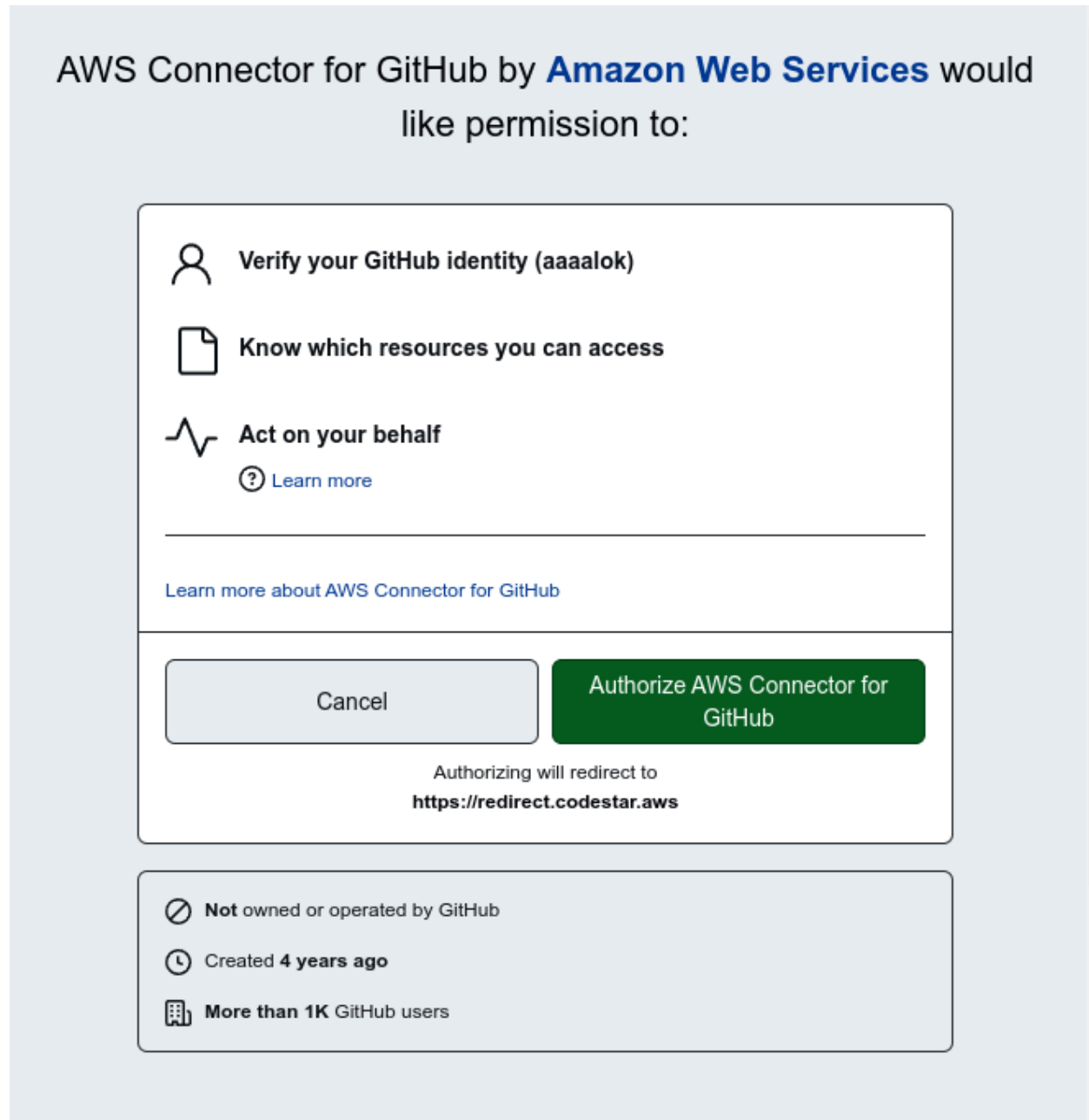
X

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.

17. Connect your GitHub account to AWS for it to build and deploy and track changes on repo



18. As of Now Skip **Build stage** under **Deploy Stage** enter AWS Elastic Beanstalk as Deploy Provider

Step 2

Add source stage

Step 3

Add build stage

Step 4

Add deploy stage

Step 5

Review

You cannot skip this stage

Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

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 (Mumbai)

Input artifacts

Choose an input artifact for this action. [Learn more](#)

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.

Q alok-app

X

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.

Q Alok-app-env

X

Alok-app-env

Cancel

Previous

Next

19. Review the summary of Pipeline created and click on **Create Pipeline**

[Choose pipeline settings](#)

Step 5 of 5

Step 2

Add source stage

Step 3

Add build stage

Step 4

Add deploy stage

Step 5

Review

Step 1: Choose pipeline settings

Pipeline settings

Pipeline name

aloky

Pipeline type

V2

Execution mode

QUEUED

Artifact location

codepipeline-ap-south-1-13487781303

Service role name

AWSCodePipelineServiceRole-ap-south-1-aloky

Variables

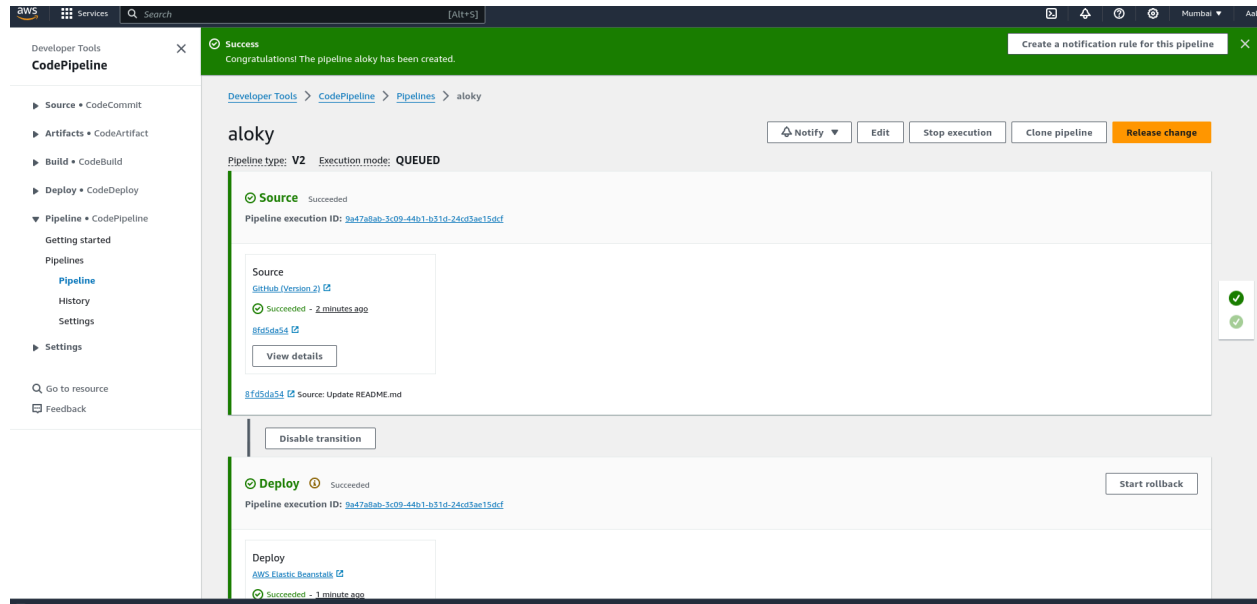
Name	Default value	Description
No variables		
No variables defined at the pipeline level in this pipeline.		

Step 2: Add source stage

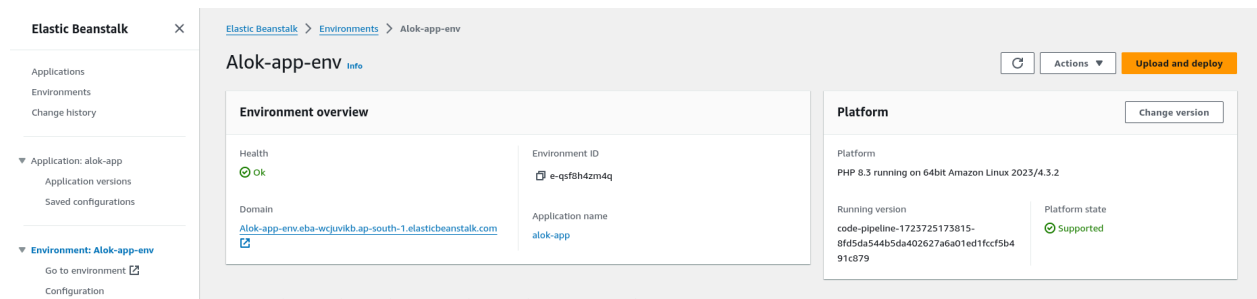
Source action provider

Source action provider
Source action provider GitHub (Version 2) OutputArtifactFormat CODE_ZIP DetectChanges true ConnectionArn arn:aws:codeconnections:ap-south-1:058264551418:connection/142a38a1-c5e6-4700-ab3f-73ffa5543ea4 FullRepositoryId aaaalok/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

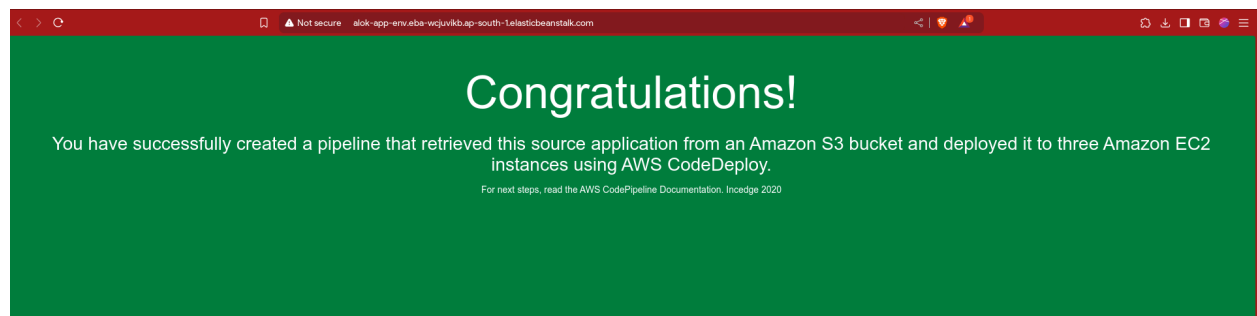
20. Our pipeline will be created and deployed in few minutes



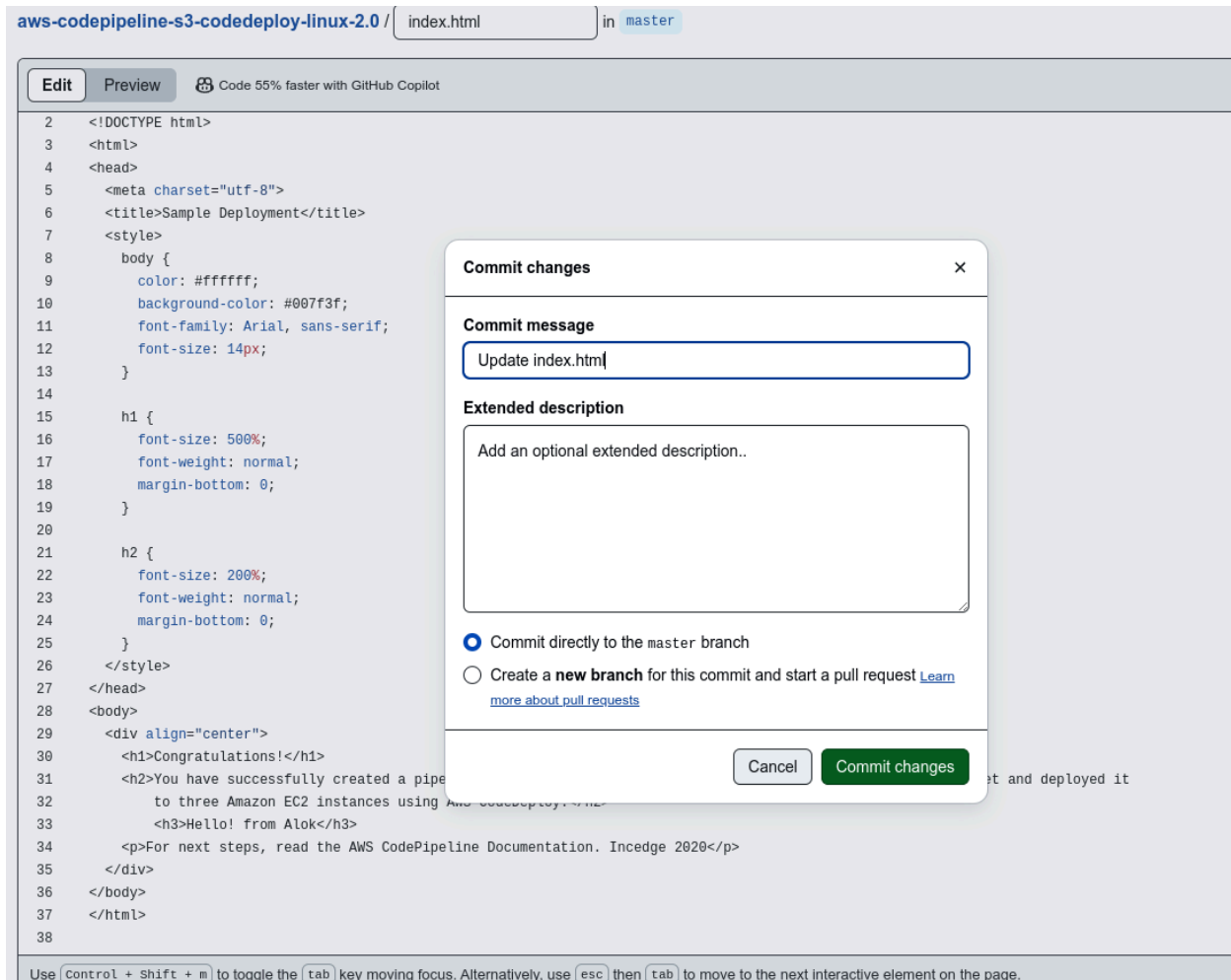
21. Go to Elastic Beanstalk environment that we had created previously and click on the URL given



22. Our pipeline is successfully deployed



23. Now to test our pipeline, we would do some changes in files in the repository



The screenshot shows the AWS CodePipeline console interface. At the top, the breadcrumb navigation indicates the path: `aws-codepipeline-s3-codedeploy-linux-2.0 / index.html` in the `master` branch. Below this, there are tabs for `Edit` and `Preview`, and a notification that says "Code 55% faster with GitHub Copilot".

The main area displays the content of `index.html` with line numbers from 2 to 38. The HTML code includes a DOCTYPE declaration, a head section with a UTF-8 charset and a title "Sample Deployment", and a body section with a style block and a div containing three paragraphs of text.

A "Commit changes" dialog box is open in the foreground. It has a title bar with a close button. The dialog contains the following fields and options:

- Commit message:** A text input field containing "Update index.html".
- Extended description:** A text area with the placeholder text "Add an optional extended description..".
- Commit options:** Two radio buttons. The first is selected and labeled "Commit directly to the master branch". The second is labeled "Create a new branch for this commit and start a pull request" with links to "Learn" and "more about pull requests".
- Buttons:** "Cancel" and "Commit changes".

At the bottom of the console, a footer note reads: "Use `Control + Shift + m` to toggle the `tab` key moving focus. Alternatively, use `esc` then `tab` to move to the next interactive element on the page."

24. Deploy process is automatically started on detecting change in repository

UPDATE APPLICATION

Deploy

Succeeded

Pipeline execution ID: [6a15e9f8-e87b-44fa-bf9d-5c3c2bdee80b](#)

Deploy

[AWS Elastic Beanstalk](#)

Succeeded

Aug 15, 2024 6:09 PM (UTC+5:30)

View details

[71437e83](#) [Source: Update index.html](#)

Start rollback

25. A new text **Hello! from Alok** is shown in the website.

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.

Hello! from Alok

For next steps, read the [AWS CodePipeline Documentation](#), Incedge 2020