



PARSHVANATH CHARITABLE TRUST'S

A. P. SHAH INSTITUTE OF TECHNOLOGY

Department of Information Technology

(NBA Accredited)



Semester: V

Academic Year: 2021-22

Class / Branch: TE IT

Subject: Advanced Devops Lab (ADL)

Name of Instructor: Prof. Manjusha K.

Name of Student:

Student ID:

EXPERIMENT NO. 02

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: Create a deployment environment

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

Elastic Beanstalk > Getting started

Create a web app

Create a new application and environment with a sample application or your own code. By creating an environment, you allow Amazon Elastic Beanstalk to manage Amazon Web Services resources and permissions on your behalf. [Learn more](#)

Application information

Application name

MyEBS

Up to 100 Unicode characters, not including forward slash (/).

Application 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](#)

Key

EBS

Value

CICD

Remove tag

Add tag

49 remaining

Platform

Platform

PHP

Platform branch

PHP 7.4 running on 64bit Amazon Linux 2

Platform version

3.3.4 (Recommended)

Application code

☒ Sample application

Get started right away with sample code.

☐ Upload your code

Upload a source bundle from your computer or copy one from Amazon S3.

Cancel

Configure more options

Create application

Step2: Get a copy of the sample code

In this step, you will retrieve a copy of the sample app's code and choose a source to host the code.

The pipeline takes code from the source and then performs actions on it.

You can use one of three options as your source: a GitHub repository, an Amazon S3 bucket, or an AWS CodeCommit repository. Select your preference and follow the steps below:

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](#)

Pipeline settings

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

awspipeline

No more than 100 characters

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

AWSCodePipelineServiceRole-ap-south-1-awspipeline

Type your service role name

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

► **Advanced settings**

Cancel **Next**

a. If you plan to use Amazon S3 as your source, you will retrieve the sample code from the AWS GitHub repository, save it to your computer, and upload it to an Amazon S3 bucket.

- Visit our GitHub repository containing the sample code at <https://github.com/imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0>
- Click the dist folder.

b. Save the source files to your computer:

- Click the file named aws-codepipeline-s3-aws-codedeploy_linux.zip
- Click View Raw.
- Save the sample file to your local computer.

c. open the Amazon S3 console and create your Amazon S3 bucket:

- Click Create Bucket

- **Bucket Name:** type a unique name for your bucket, such as `awscodepipeline-demobucket-variables`. All bucket names in Amazon S3 must be unique, so use one of your own, not one with the name shown in the example.
- **Region:** In the drop-down, select the region where you will create your pipeline, such as `ap-South-1`
- Click **Create**.

d. The console displays the newly created bucket, which is empty.

- Click **Properties**.
- Expand **Versioning** and select **Enable Versioning**. When versioning is enabled, Amazon S3 saves every version of every object in the bucket.

e. You will now upload the sample code to the Amazon S3 bucket:

- Click **Upload**.
- Follow the on-screen directions to upload the `.zip` file containing the sample code you downloaded from GitHub.

Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#) 

General configuration

Bucket name

awscodepipeline-demobucket-variables1

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#) 

AWS Region


Asia Pacific (Mumbai) ap-south-1 ▼

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#) 

☒ Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☒ Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access

Amazon S3 > awscodepipeline-demobucket-variables11

awscodepipeline-demobucket-variables11 [Info](#)

Objects **Properties** Permissions Metrics Management Access Points

Bucket overview

| | | |
|--|--|---|
| AWS Region Asia Pacific (Mumbai) ap-south-1 | Amazon Resource Name (ARN) arn:aws:s3::awscodepipeline-demobucket-variables11 | Creation date August 2, 2021, 09:45:02 (UTC+05:30) |
|--|--|---|

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

[Edit](#)

Bucket Versioning
Disabled

Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

you can upload directly zip file here from <https://github.com/imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0>

Upload succeeded
View details below.

[View details below](#)

Summary

| | | |
|--|---|-----------------------------|
| Destination s3://awscodepipeline-demobucket-variables11 | Succeeded 7 files, 12.2 KB (100.00%) | Failed 0 files, 0 B (0%) |
|--|---|-----------------------------|

Files and folders Configuration

Files and folders (7 Total, 12.2 KB)

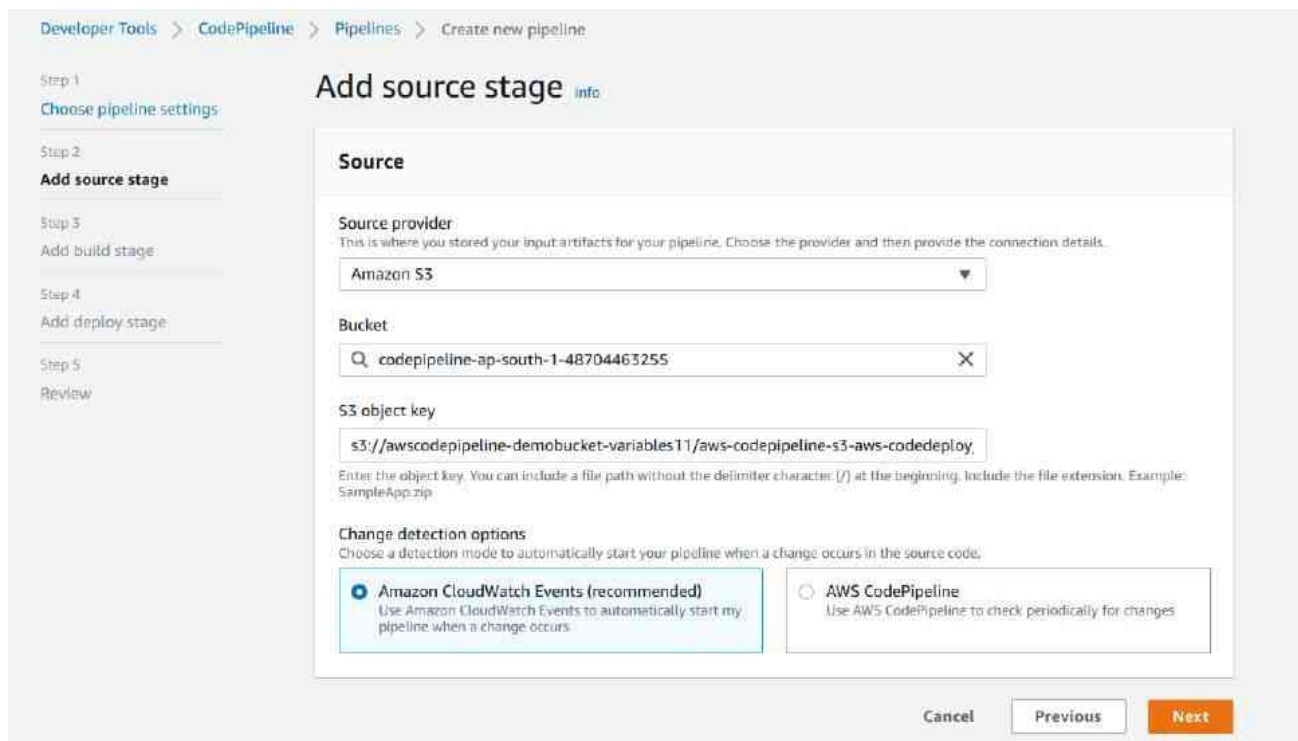
| Name | Folder | Type | Size | Status |
|----------------------|---|--------------------|---------|-----------|
| LICENSE | aws-codepipeline-s3-aws-codedeploy-linux/ | - | 10.6 KB | Succeeded |
| README.md | aws-codepipeline-s3-aws-codedeploy-linux/ | text/markdown | 249.0 B | Succeeded |
| appspec.yml | aws-codepipeline-s3-aws-codedeploy-linux/ | application/x-yaml | 359.0 B | Succeeded |
| index.html | aws-codepipeline-s3-aws-codedeploy-linux/ | text/html | 782.0 B | Succeeded |
| install_dependencies | aws-codepipeline-s3-aws-codedeploy-linux/scripts/ | - | 34.0 B | Succeeded |
| start_server | aws-codepipeline-s3-aws-codedeploy-linux/scripts/ | - | 33.0 B | Succeeded |
| stop_server | aws-codepipeline-s3-aws-codedeploy-linux/scripts/ | - | 105.0 B | Succeeded |

Step3: Create your Pipeline

In this step, you will create and configure a simple pipeline with two actions: source and deploy. You will provide CodePipeline with the locations of your source repository and deployment environment.

A true continuous deployment pipeline requires a build stage, where code is compiled and unit tested. CodePipeline lets you plug your preferred build provider into your pipeline. However, in this we will skip the build stage.

Goto Pipeline again and create it



The screenshot shows the 'Add source stage' configuration page in the AWS CodePipeline console. The breadcrumb trail at the top reads: Developer Tools > CodePipeline > Pipelines > Create new pipeline. On the left, a sidebar lists the steps: Step 1: Choose pipeline settings (active), Step 2: Add source stage, Step 3: Add build stage, Step 4: Add deploy stage, Step 5: Review. The main heading is 'Add source stage' with an 'info' link. The configuration form includes: 'Source provider' set to 'Amazon S3'; 'Bucket' set to 'codepipeline-ap-south-1-48704463255'; 'S3 object key' set to 's3://awscodepipeline-demobucket-variables11/aws-codepipeline-s3-aws-codedeploy'; and 'Change detection options' with 'Amazon CloudWatch Events (recommended)' selected. At the bottom are 'Cancel', 'Previous', and 'Next' buttons.

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

Add source stage [info](#)

Source

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

Amazon S3

Bucket

codepipeline-ap-south-1-48704463255

S3 object key

s3://awscodepipeline-demobucket-variables11/aws-codepipeline-s3-aws-codedeploy

Enter the object key. You can include a file path without the delimiter character (/) at the beginning. Include the file extension. Example: SampleApp.zip

Change detection options
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☒ **Amazon CloudWatch Events (recommended)**
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs.

☐ **AWS CodePipeline**
Use AWS CodePipeline to check periodically for changes.

Cancel Previous Next

In above you can give zip file name in S3 object Key and choose bucket name which you created

In Step 4: Deploy Stage:

- Deployment provider: Click AWS Elastic Beanstalk.
- Application name: MYEBS.
- Environment name: Click Myebs-env.
- Click Next step.

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

Add deploy stage [Info](#)

 **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) ▼

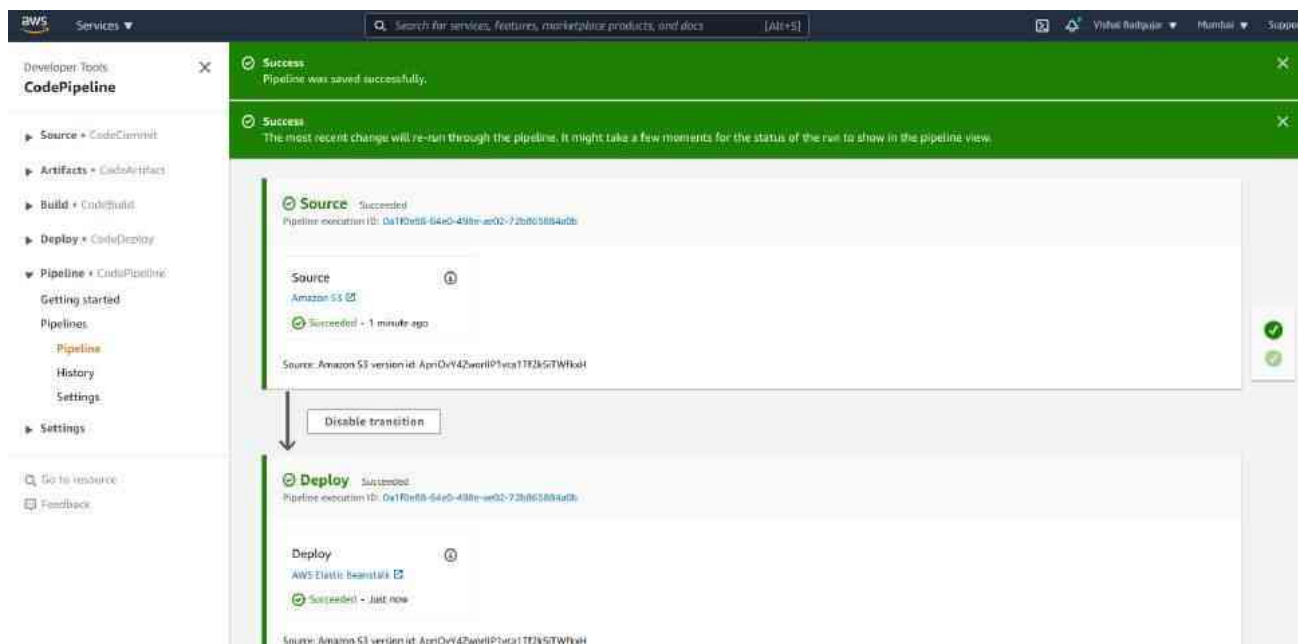
Application name
Choose an application that you have already created in the AWS Elastic Beanstalk console. Or create an application in the AWS Elastic Beanstalk console and then return to this task.

MyEBS 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.

Myebs-env X

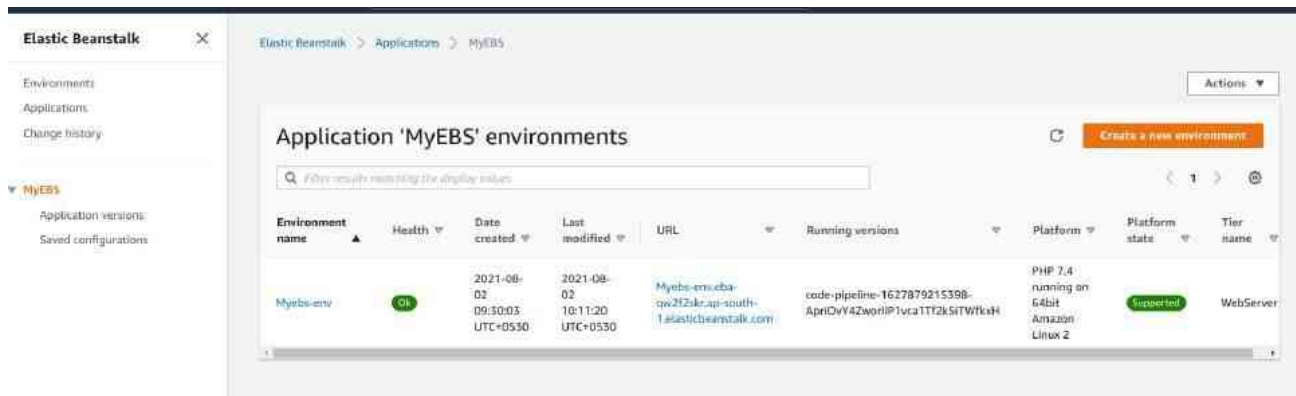
Cancel Previous **Next**



After your pipeline is created, the pipeline status page appears and the pipeline automatically starts to run. You can view progress as well as success and failure messages as the pipeline perform each action.

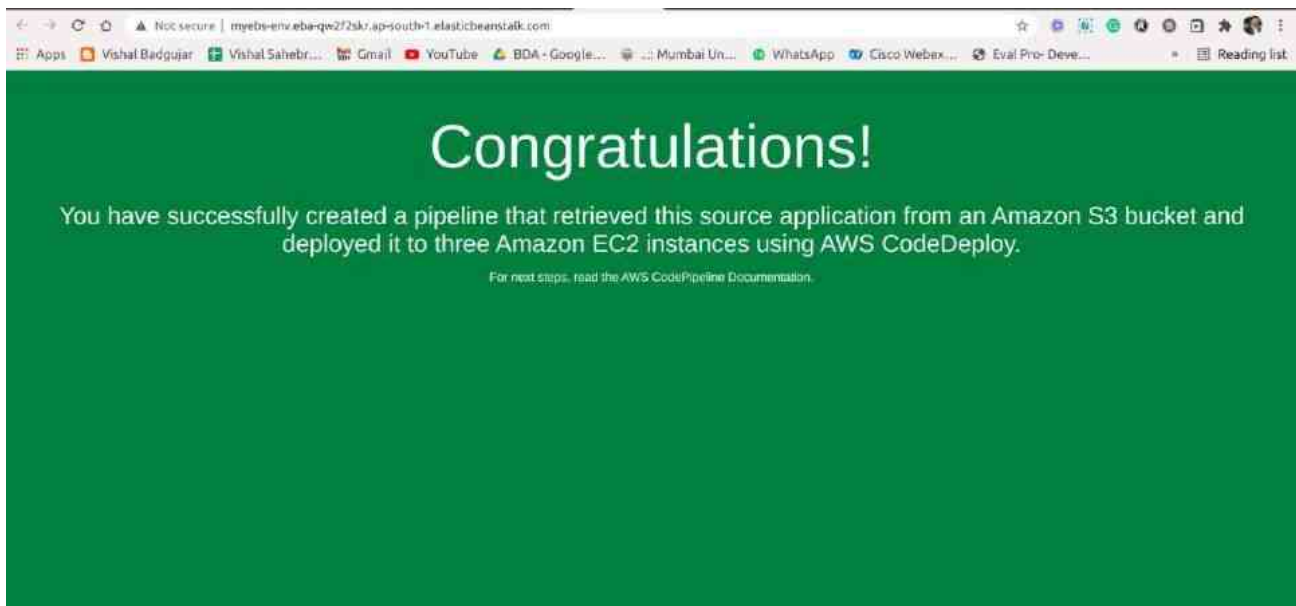
To verify your pipeline ran successfully, monitor the progress of the pipeline as it moves through each stage. The status of each stage will change from No executions yet to In Progress, and then to either Succeeded or Failed. The pipeline should complete the first run within a few minutes.

Now go to your EBS environment and click on the URL to view the sample website you deployed.



You have successfully created an automated software release pipeline using AWS CodePipeline!

Using CodePipeline, you created a pipeline that uses GitHub, Amazon S3, or AWS CodeCommit as the source location for application code and then deploys the code to an Amazon EC2 instance managed by AWS Elastic Beanstalk.



Step 5: Commit a change and then update your app

Step 6: Clean up your resources

To avoid future charges, you will delete all the resources you launched throughout this tutorial, which includes the pipeline, the Elastic Beanstalk application, and the source you set up to host the code.

- First, you will delete your pipeline:
- Second, delete your Elastic Beanstalk application:

OUTPUT:

The top screenshot shows the AWS S3 console. A green banner at the top states: "Successfully created bucket 'soham-bucket2'. To upload files and folders, or to configure additional bucket settings choose [View details](#)." Below this, the page title is "Amazon S3 > Buckets". There is an "Account snapshot" section and a "Buckets (2)" section. A table lists the buckets:

| Name | AWS Region | Access | Creation date |
|---------------|---------------------------------|-------------------------------|---------------------------------------|
| soham-bucket1 | US East (N. Virginia) us-east-1 | Bucket and objects not public | August 11, 2023, 14:05:34 (UTC+05:30) |
| soham-bucket2 | US East (N. Virginia) us-east-1 | Objects can be public | August 11, 2023, 14:06:06 (UTC+05:30) |

The bottom screenshot shows the AWS CodeBuild console. The page title is "Developer Tools > CodeBuild > Build projects". There is a "Build projects" section with a table listing the build projects:

| Name | Source provider | Repository | Latest build status | Description | Last Modified |
|-------|-----------------|--|---------------------|-------------|---------------|
| proj1 | GitHub | Soham01011/Car-pwa-deploy-on-aws | - | - | 6 days ago |

[Alt+S]

N. Virginia

Soham Dalvi

Source provider

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

GitHub (Version 1)

Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.

Connected

✔ You have successfully configured the action with the provider.

The GitHub (Version 1) action is not recommended

The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. [Learn more](#)

Repository

🔍 Soham01011/Car-pwa-deploy-on-aws

Branch

🔍 master

Change detection options

Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

🔵 GitHub webhooks (recommended)

Use webhooks in GitHub to automatically start my pipeline when a change occurs

⚪ AWS CodePipeline

Use AWS CodePipeline to check periodically for changes

aws

Services

Search

[Alt+S]

N. Virginia

Soham Dalvi

Developer Tools

CodePipeline

▶ Source • CodeCommit

▶ Artifacts • CodeArtifact

▶ Build • CodeBuild

▶ Deploy • CodeDeploy

▼ Pipeline • CodePipeline

Getting started

Pipelines

Pipeline

History

Settings

▶ Settings

🔍 Go to resource

🗨 Feedback

✔ Success

Create a notification rule for this pipeline

Congratulations! The pipeline pipeline-1 has been created.

Developer Tools > CodePipeline > Pipelines > pipeline-1

pipeline-1

🔔 Notify

Edit

Stop execution

Clone pipeline

Release change

✔ Source Succeeded

Pipeline execution ID: 9ea76575-fc88-4007-b773-5cb7f1d71293

Source

GitHub (Version 1)

✔ Succeeded - Just now

e9696332

e9696332 Source: Delete CNAME

Disable transition

🔄 Build In progress

Pipeline execution ID: 9ea76575-fc88-4007-b773-5cb7f1d71293

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

- User groups
- Users
- Roles
- Policies
- Identity providers
- Account settings

Access reports

- Access analyzer
 - Archive rules
 - Analyzers
 - Settings
- Credential report
- Organization activity

Policy was successfully attached to role.

IAM > Roles > codebuild-p-service-role

codebuild-p-service-role

Delete

Summary

Edit

Creation date

August 04, 2023, 15:36 (UTC+05:30)

ARN

arn:aws:iam::027642185607:role/service-role/codebuild-p-service-role

Last activity

None

Maximum session duration

1 hour

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (2)

Info

You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

Policy name

Type

Description

Developer Tools

CodePipeline

Source • CodeCommit

Artifacts • CodeArtifact

Build • CodeBuild

Deploy • CodeDeploy

Pipeline • CodePipeline

- Getting started
- Pipelines
- Pipeline
- History
- Settings

Settings

Go to resource

Feedback

Success

Stage Build successfully retried

Build

Succeeded

Pipeline execution ID: 9ea76575-fc88-4007-b773-5cb7f1d71293

Build

AWS CodeBuild

Succeeded - Just now

Details

e9696332 Source: Delete CNAME

Disable transition

Deploy

Succeeded

Pipeline execution ID: 9ea76575-fc88-4007-b773-5cb7f1d71293

Deploy

Amazon S3

Succeeded - Just now

e9696332 Source: Delete CNAME

CloudShell

Feedback

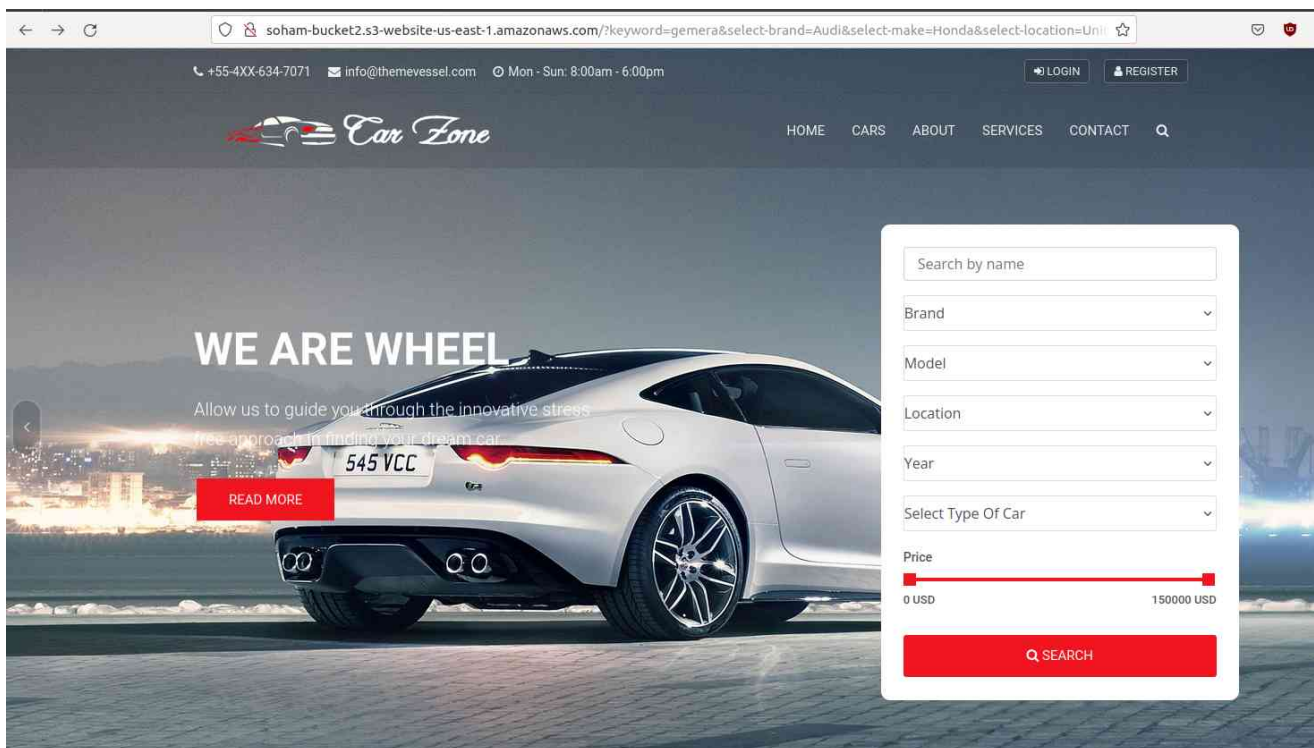
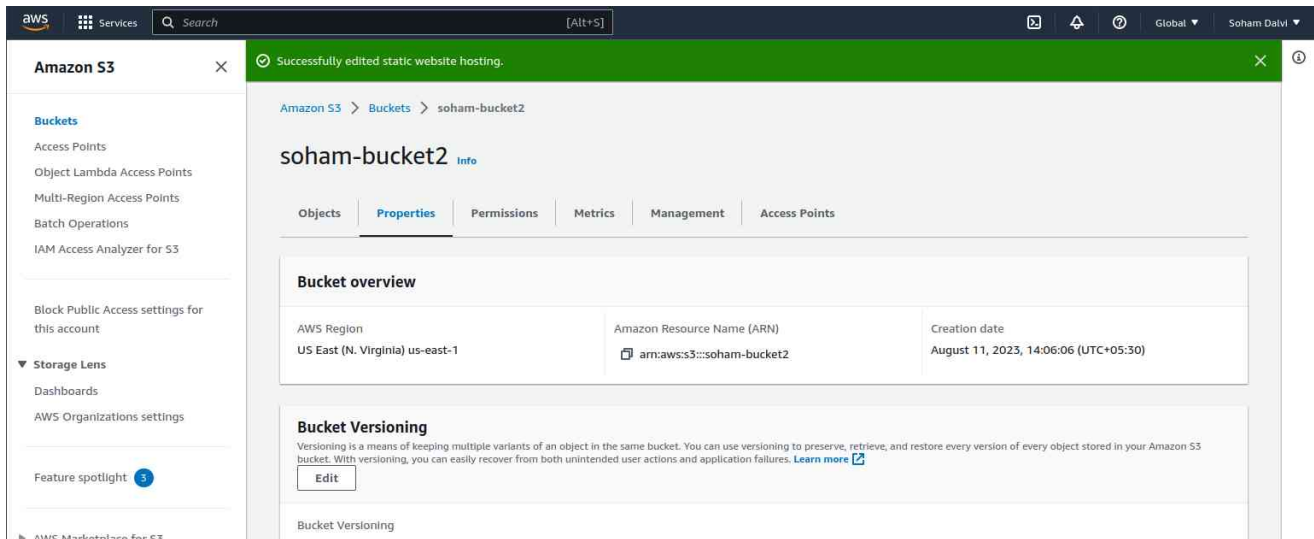
Language

© 2023, Amazon Web Services India Private Limited or its affiliates.

Privacy

Terms

Cookie preferences



Conclusion:Hence we understood how to host a website with S3 bucket instance on AWS.