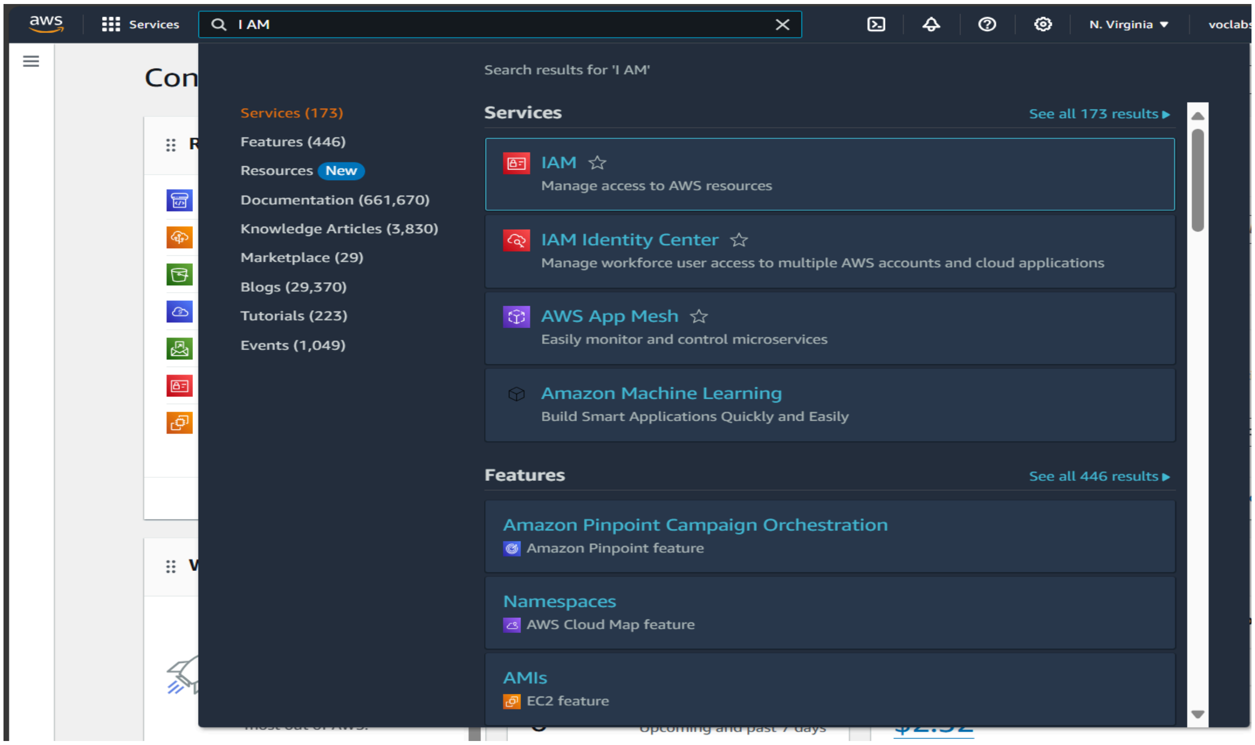
Experiment No: 2

Step1:- Creation of role:-

1. Login to your AWS account and search for IAM



Search results for 'I AM'

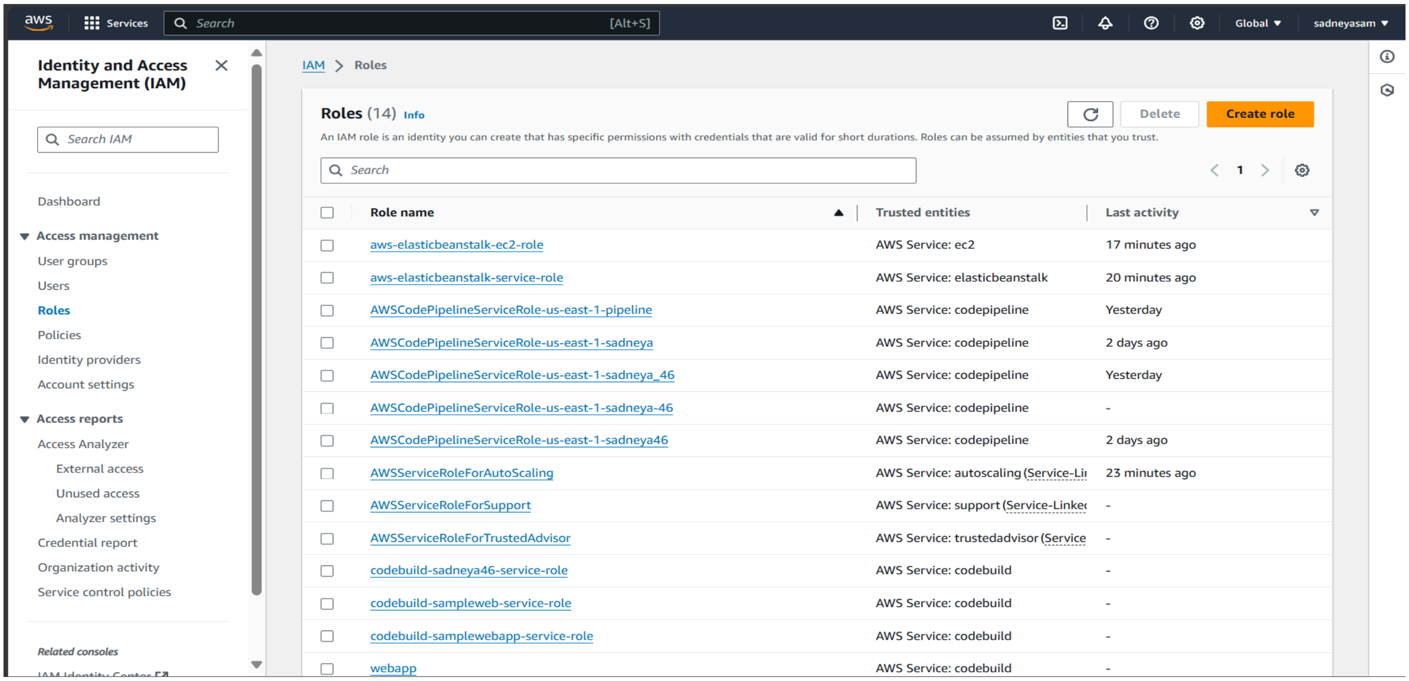
Services See all 173 results ▸

- IAM ☆ Manage access to AWS resources
- IAM Identity Center ☆ Manage workforce user access to multiple AWS accounts and cloud applications
- AWS App Mesh ☆ Easily monitor and control microservices
- Amazon Machine Learning Build Smart Applications Quickly and Easily

Features See all 446 results ▸

- Amazon Pinpoint Campaign Orchestration Amazon Pinpoint feature
- Namespaces AWS Cloud Map feature
- AMIs EC2 feature

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

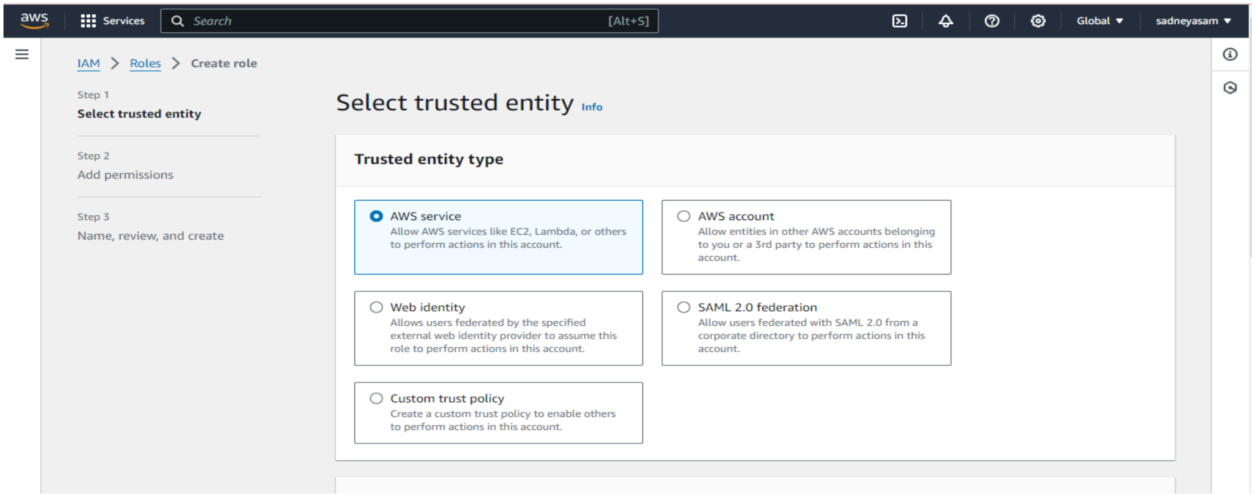


Roles (14) Info

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	Last activity
□	aws-elasticbeanstalk-ec2-role	AWS Service: ec2	17 minutes ago
□	aws-elasticbeanstalk-service-role	AWS Service: elasticbeanstalk	20 minutes ago
□	AWSCodePipelineServiceRole-us-east-1-pipeline	AWS Service: codepipeline	Yesterday
□	AWSCodePipelineServiceRole-us-east-1-sadneya	AWS Service: codepipeline	2 days ago
□	AWSCodePipelineServiceRole-us-east-1-sadneya_46	AWS Service: codepipeline	Yesterday
□	AWSCodePipelineServiceRole-us-east-1-sadneya-46	AWS Service: codepipeline	-
□	AWSCodePipelineServiceRole-us-east-1-sadneya46	AWS Service: codepipeline	2 days ago
□	AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Li	23 minutes ago
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
□	codebuild-sadneya46-service-role	AWS Service: codebuild	-
□	codebuild-sampleweb-service-role	AWS Service: codebuild	-
□	codebuild-samplewebapp-service-role	AWS Service: codebuild	-
□	webapp	AWS Service: codebuild	-

3. Then select a trusted entity as AWS service.



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.

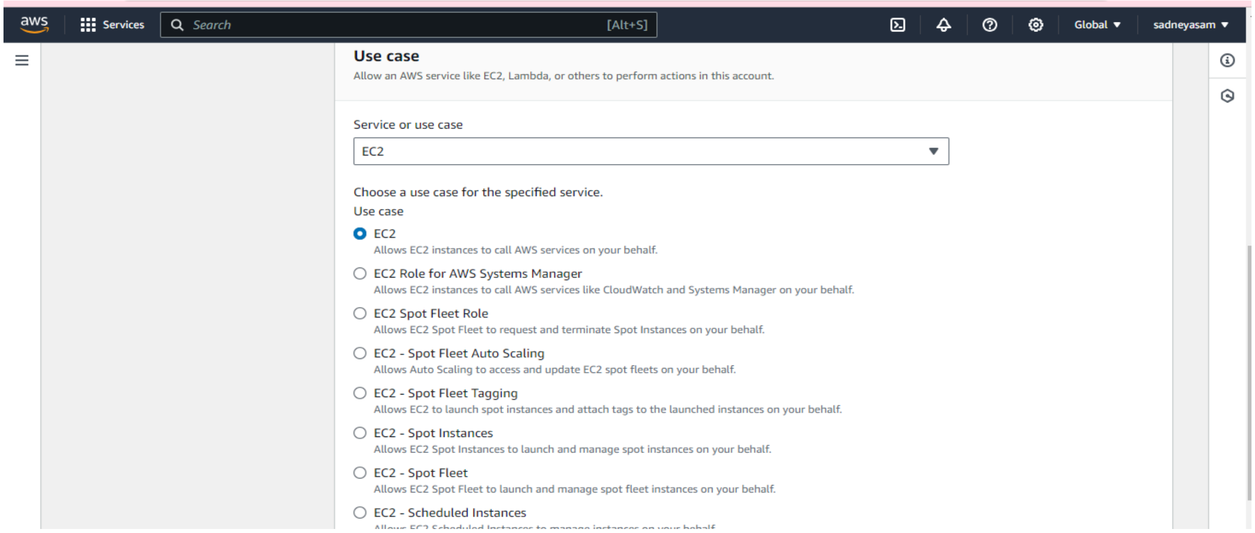
○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ 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.

4. Select use case as EC2.



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.

○ 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.

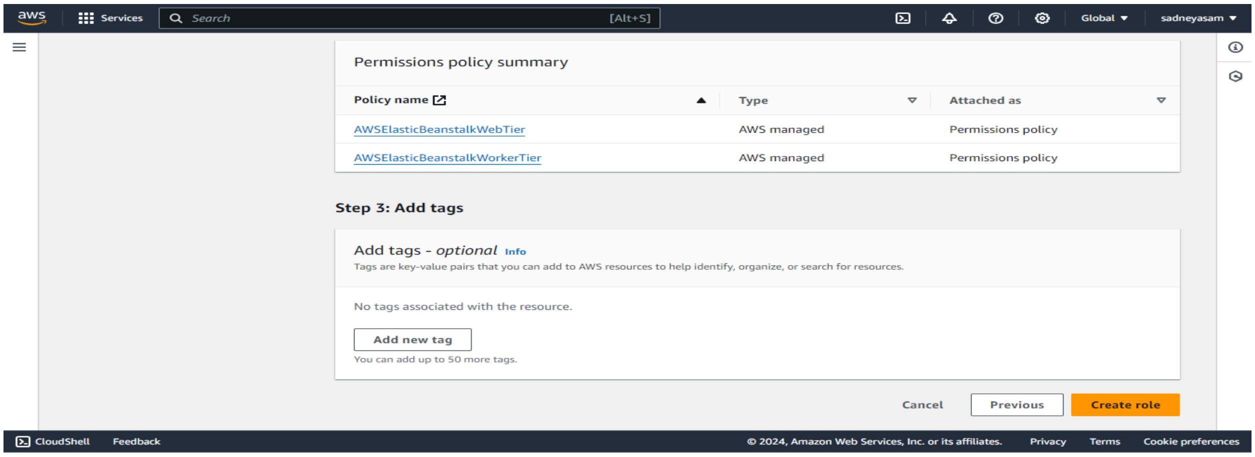
○ EC2 - Spot Fleet Tagging
Allows EC2 to launch spot instances and attach tags to the launched instances on your behalf.

○ 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 Benstalk Web Tier.and AWS elastic Benstalk worker tier.



Permissions policy summary

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

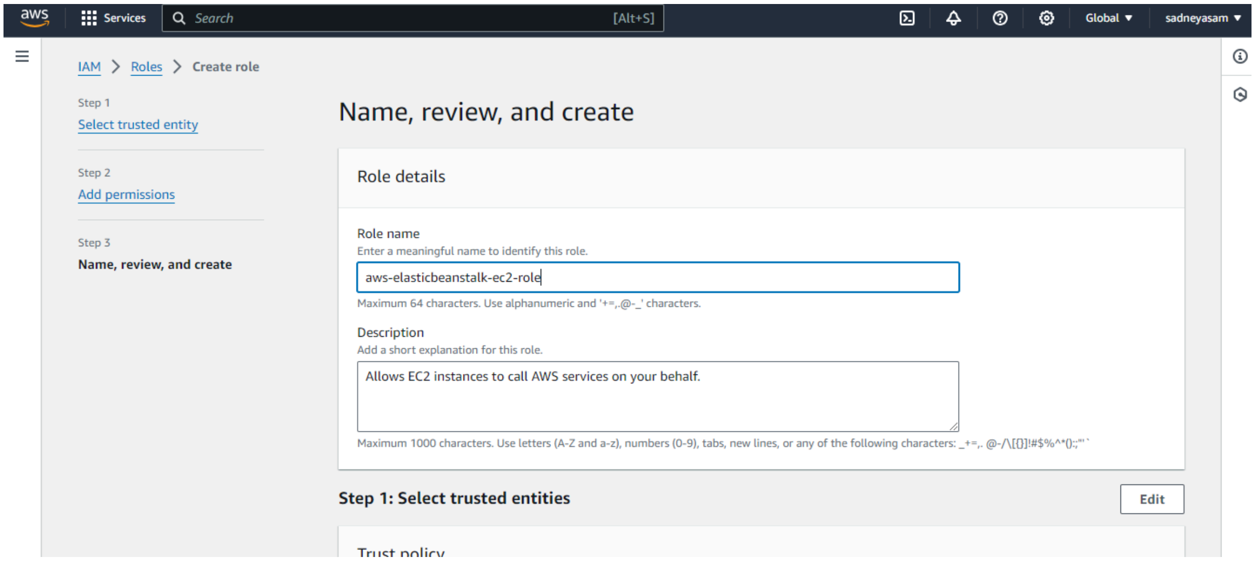
Step 3: Add tags

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.

You can add up to 50 more tags.

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



Name, review, and create

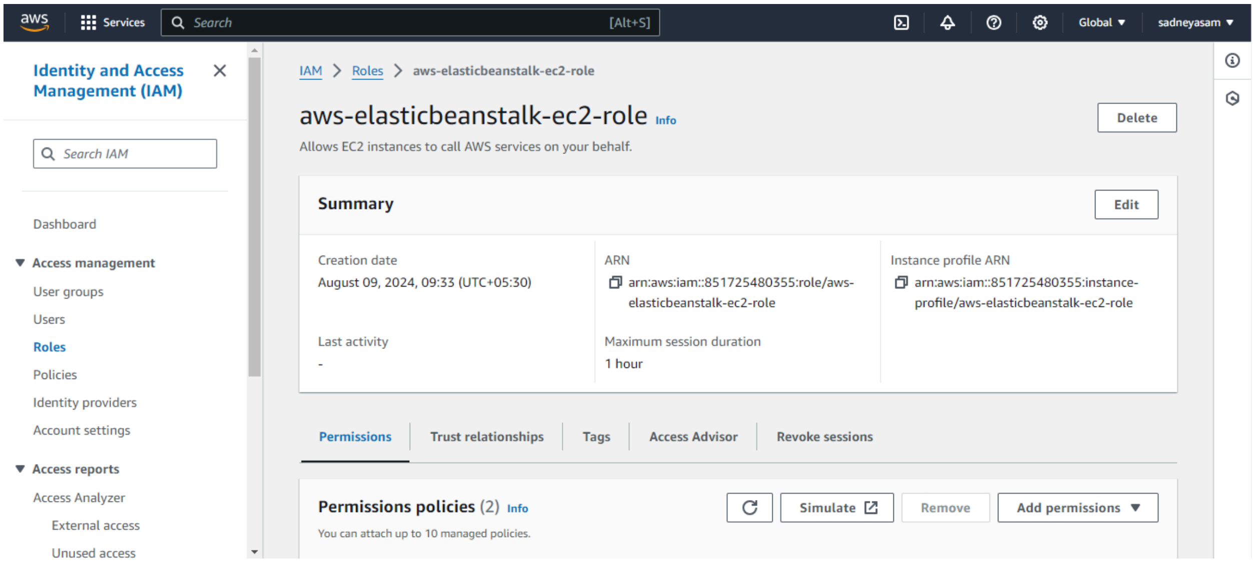
Role details

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

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date
August 09, 2024, 09:33 (UTC+05:30)

ARN
arn:aws:iam::851725480355:role/aws-elasticbeanstalk-ec2-role

Instance profile ARN
arn:aws:iam::851725480355:instance-profile/aws-elasticbeanstalk-ec2-role

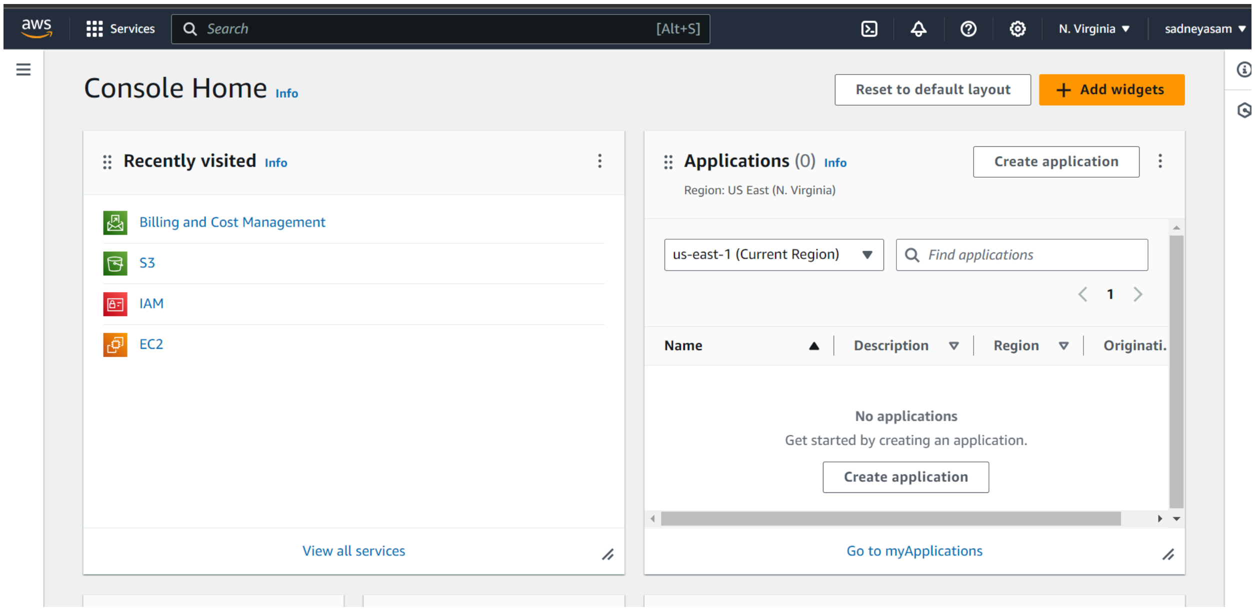
Last activity
-

Maximum session duration
1 hour

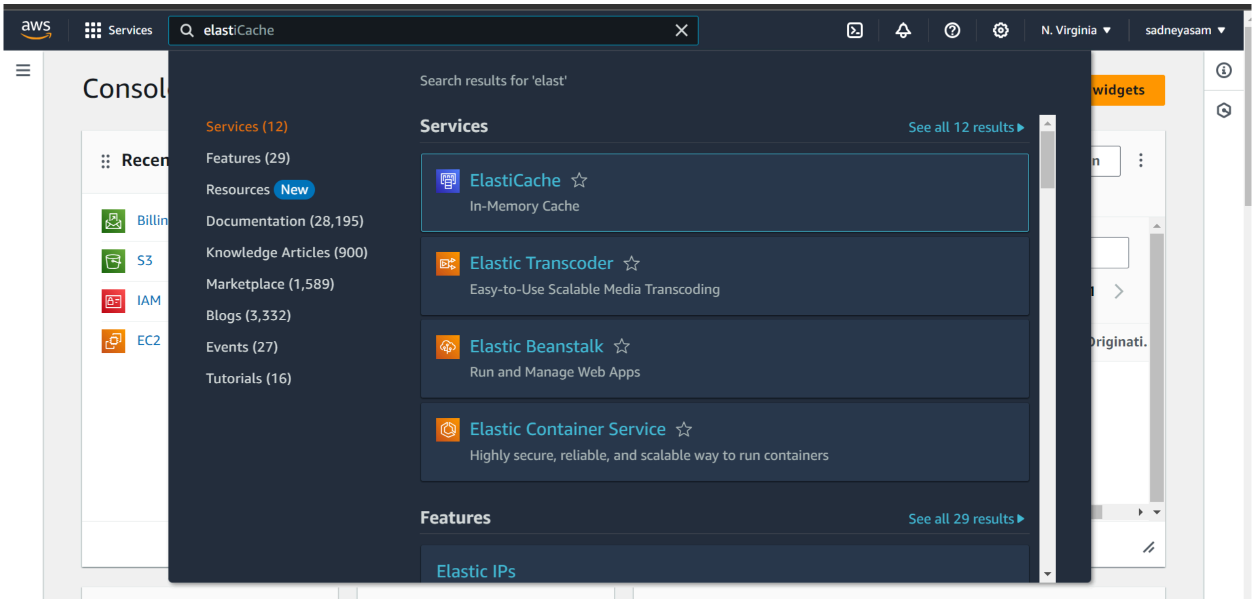
Permissions policies (2)
You can attach up to 10 managed policies.

Step2:- Creation Elastic Benstalk Environment

1. search for Elastic Beanstalk in the search box.

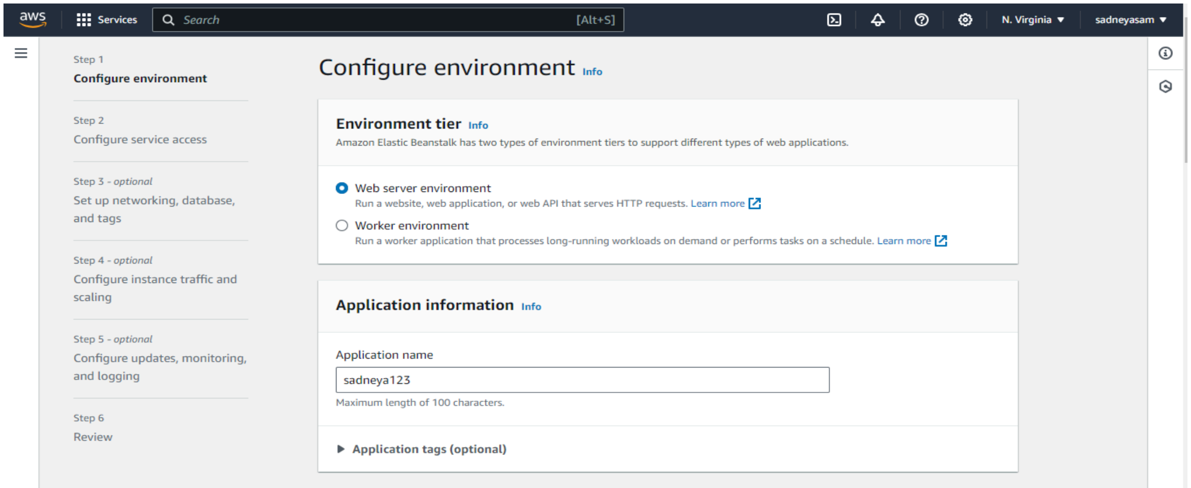


No applications
Get started by creating an application.


Search: elastiCache

Search results for 'elast'

2. Open up Elastic Benstalk and name your web app. (here I have given name sadneya123)



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

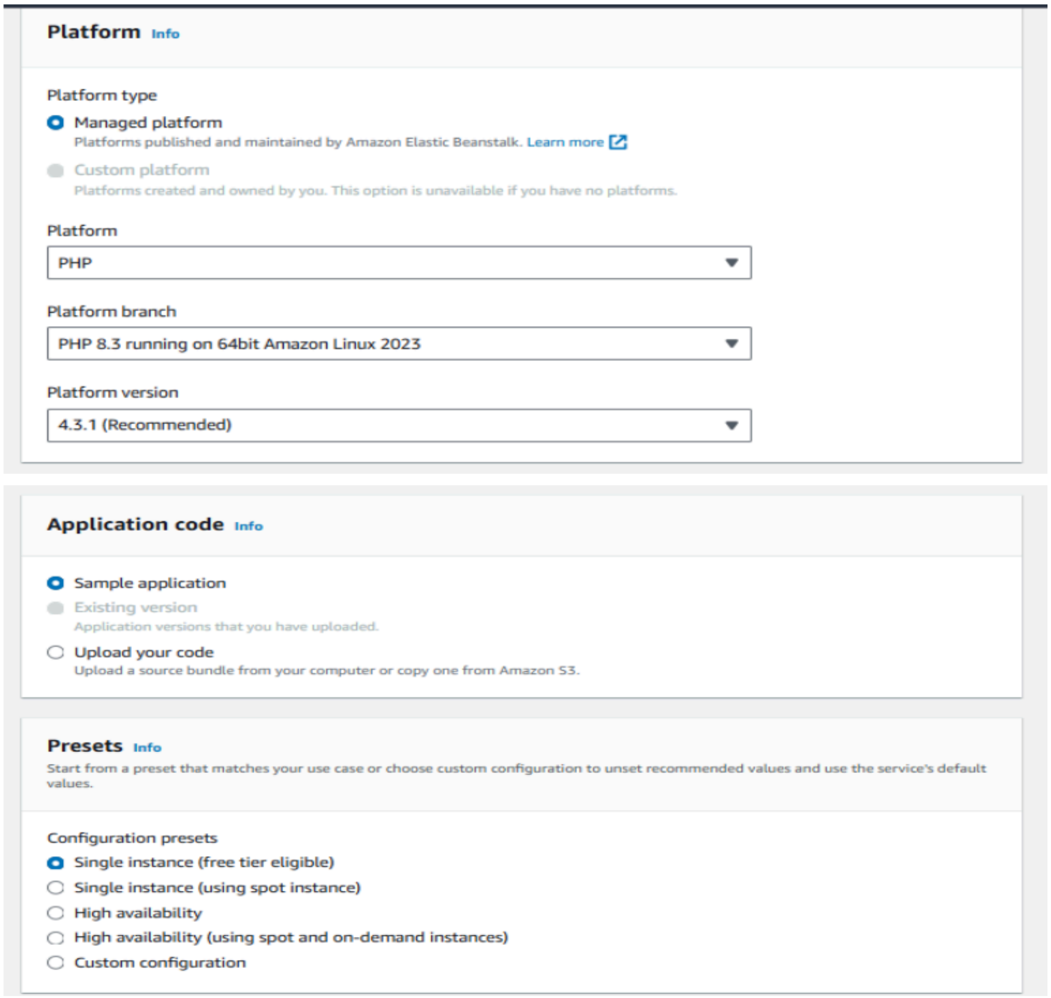
○ Worker environment
Run a worker application that processes long-running workloads on demand or performs tasks on a schedule. Learn more

Application information Info

Application name
sadneya123
Maximum length of 100 characters.

▶ Application tags (optional)

3. Select platform as PHP.



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)

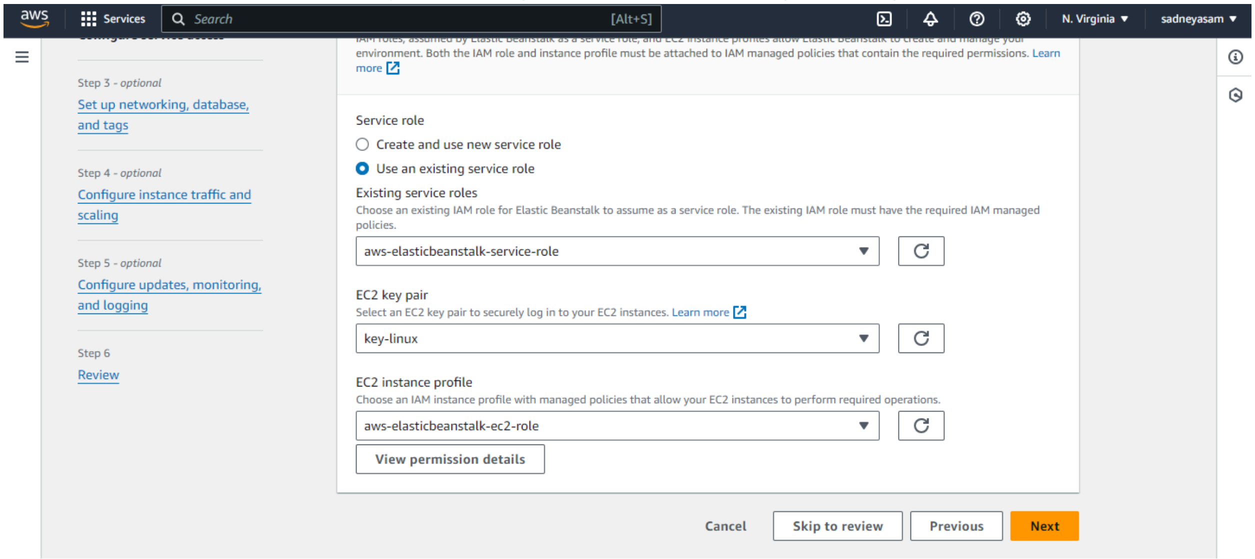
Application code Info

● 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

4. After clicking on next u need to select the use existing role. Then you will see the existing role select it like here it is aws-elasticbeanstalk-service-role. Which we created in 1st part.Select role,then select key you have created then profile will be automatically selected according to role.then click on create application by keeping all the remaining settings as it is.



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

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.

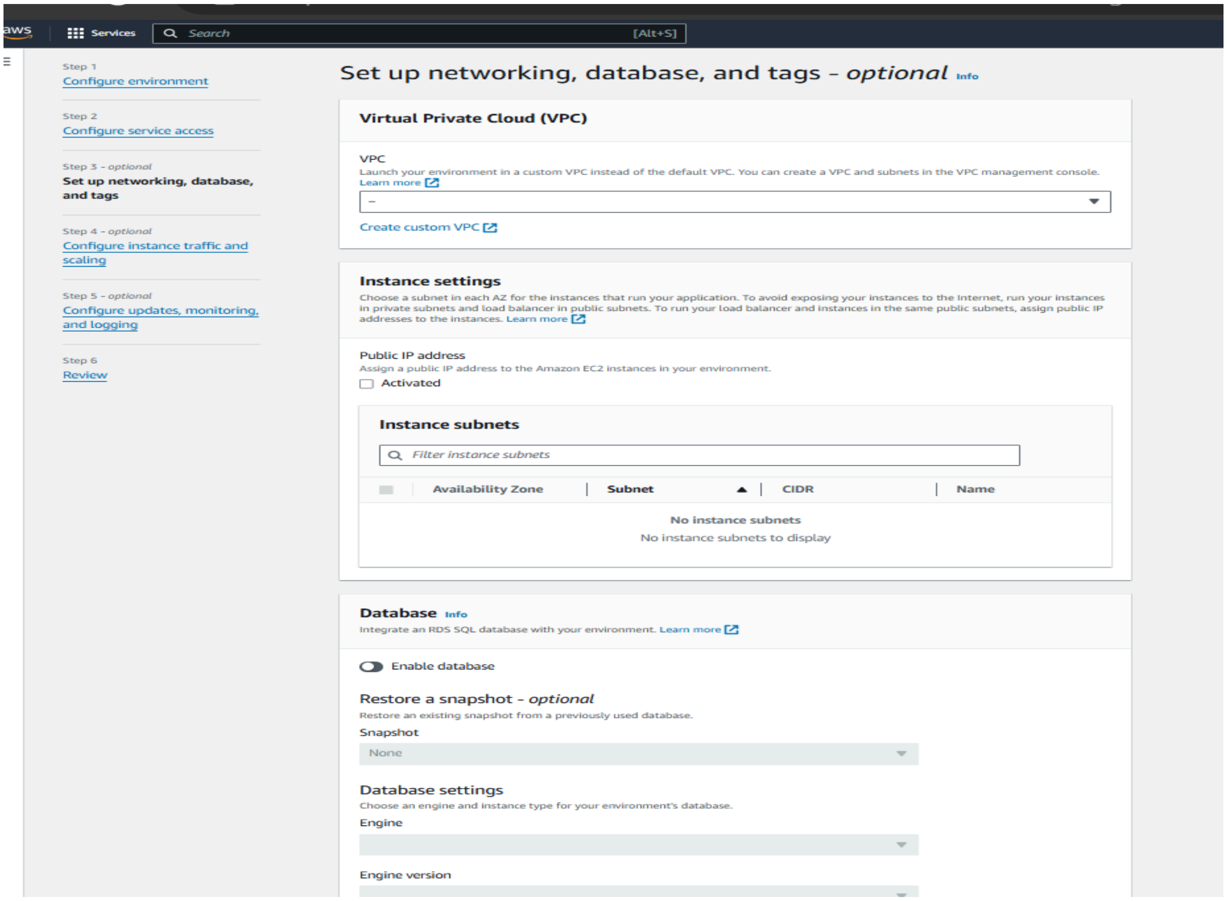
aws-elasticbeanstalk-service-role

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

key-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



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

Set up networking, database, and tags - *optional* Info

Virtual Private Cloud (VPC)

VPC
Launch your environment in a custom VPC instead of the default VPC. You can create a VPC and subnets in the VPC management console. Learn more

–

Create custom VPC

Instance settings
Choose a subnet in each AZ for the instances that run your application. To avoid exposing your instances to the Internet, run your instances in private subnets and load balancer in public subnets. To run your load balancer and instances in the same public subnets, assign public IP addresses to the instances. Learn more

Public IP address
Assign a public IP address to the Amazon EC2 instances in your environment.
□ Activated

Instance subnets

Filter instance subnets

	Availability Zone	Subnet ▲	CIDR	Name

No instance subnets
No instance subnets to display

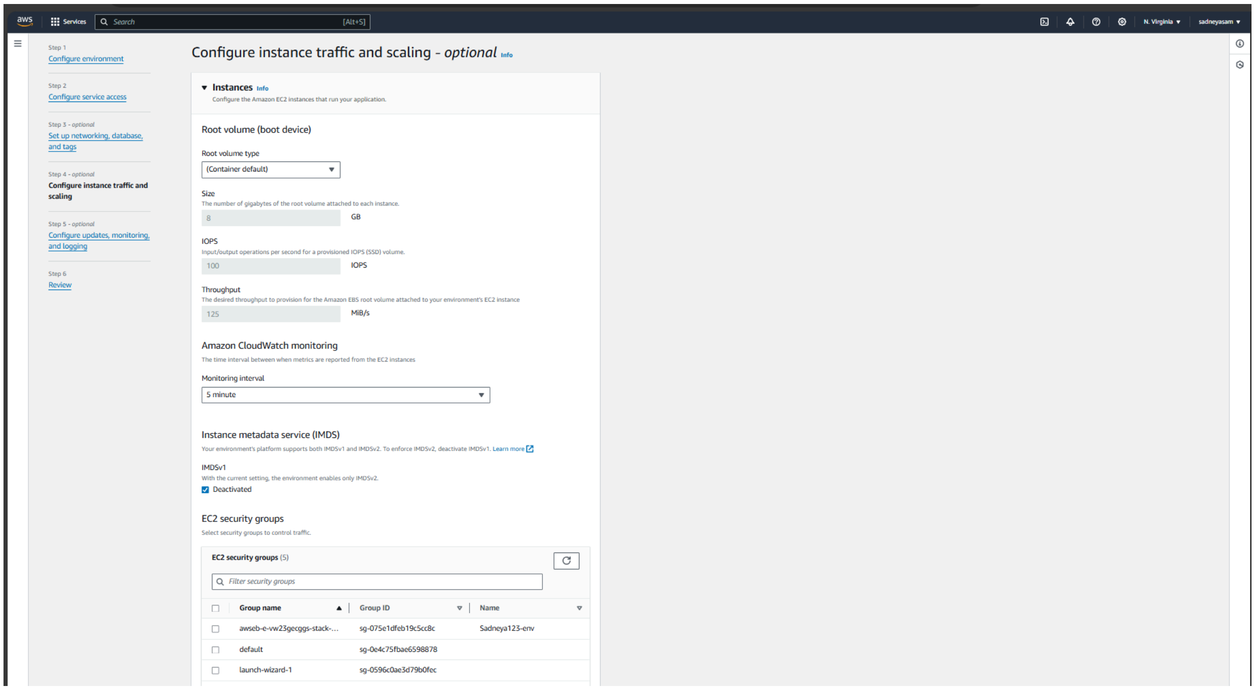
Database Info
Integrate an RDS SQL database with your environment. Learn more

Enable database

Restore a snapshot - *optional*
Restore an existing snapshot from a previously used database.
Snapshot
None

Database settings
Choose an engine and instance type for your environment's database.
Engine

Engine version



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 ▼

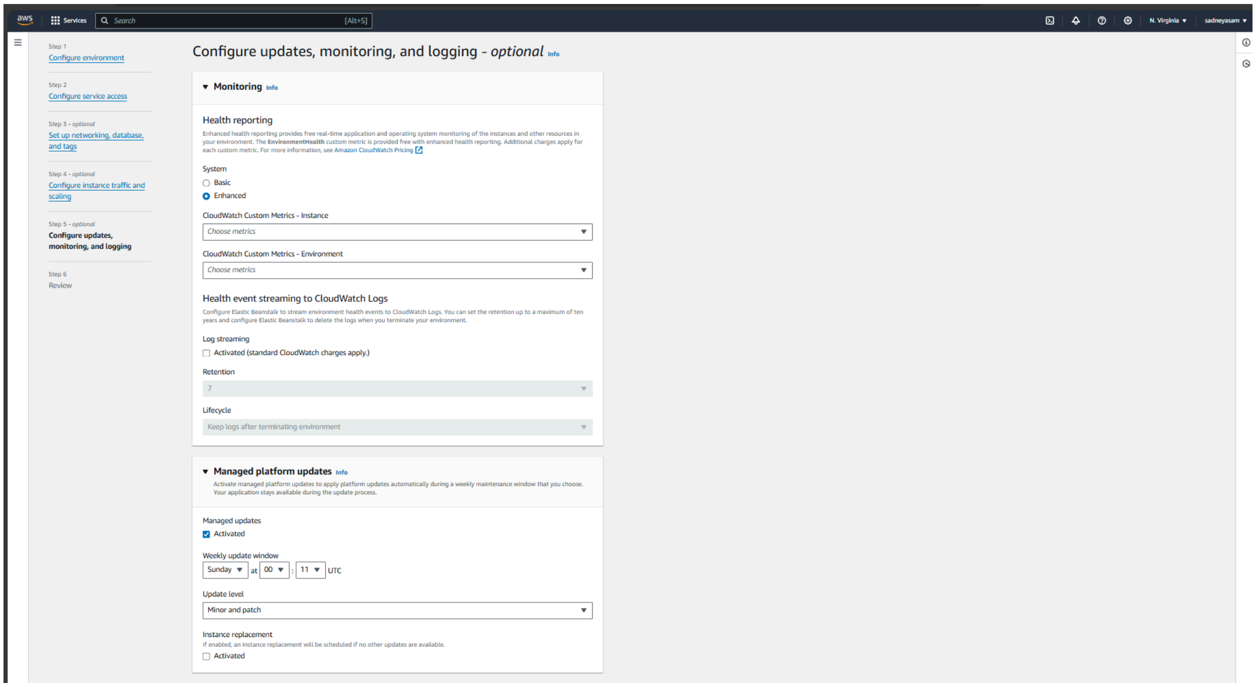
Instance metadata service (IMDS)
Your environment's platform supports both IMDSv1 and IMDSv2. To enforce IMDSv2, deactivate IMDSv1. Learn more

IMDSv1
With the current setting, the environment enables only IMDSv2.
☑ Deactivated

EC2 security groups
Select security groups to control traffic.

EC2 security groups (5)
Filter security groups

□	Group name	Group ID	Name
□	awseb-e-vw23gecggs-stack-...	sg-075e1dfeb19c5cc8c	Sadneya123-env
□	default	sg-0e4c75fbae6598878	
□	launch-wizard-1	sg-0596c0ae3d79b0fec	



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

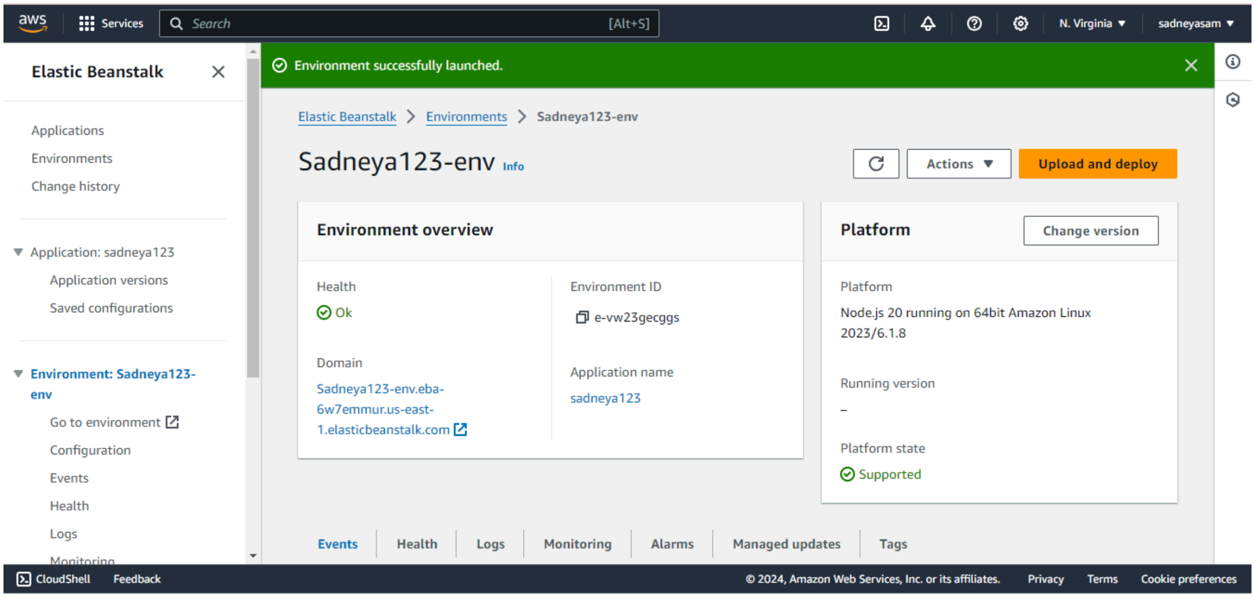
Weekly update window
Sunday ▼ at 00 ▼ : 11 ▼ UTC

Update level
Minor and patch ▼

Instance replacement
If enabled, an instance replacement will be scheduled if no other updates are available.
□ Activated

Keep Set up networking, database, and tags ,Configure instance traffic and scaling,Configure updates, monitoring, and logging all these default.

5. Beanstalk creates a sample environment for you to deploy your application. By default, it creates an EC2 instance, a security group, an Auto Scaling group, an Amazon S3 Bucket, Amazon CloudWatch alarms and a domain name for your Application.



Environment successfully launched.

Sadneya123-env

Environment overview

Health: Ok

Environment ID: e-vw23gecggs

Domain: Sadneya123-env.eba-6w7emmur.us-east-1.elasticbeanstalk.com

Application name: sadneya123

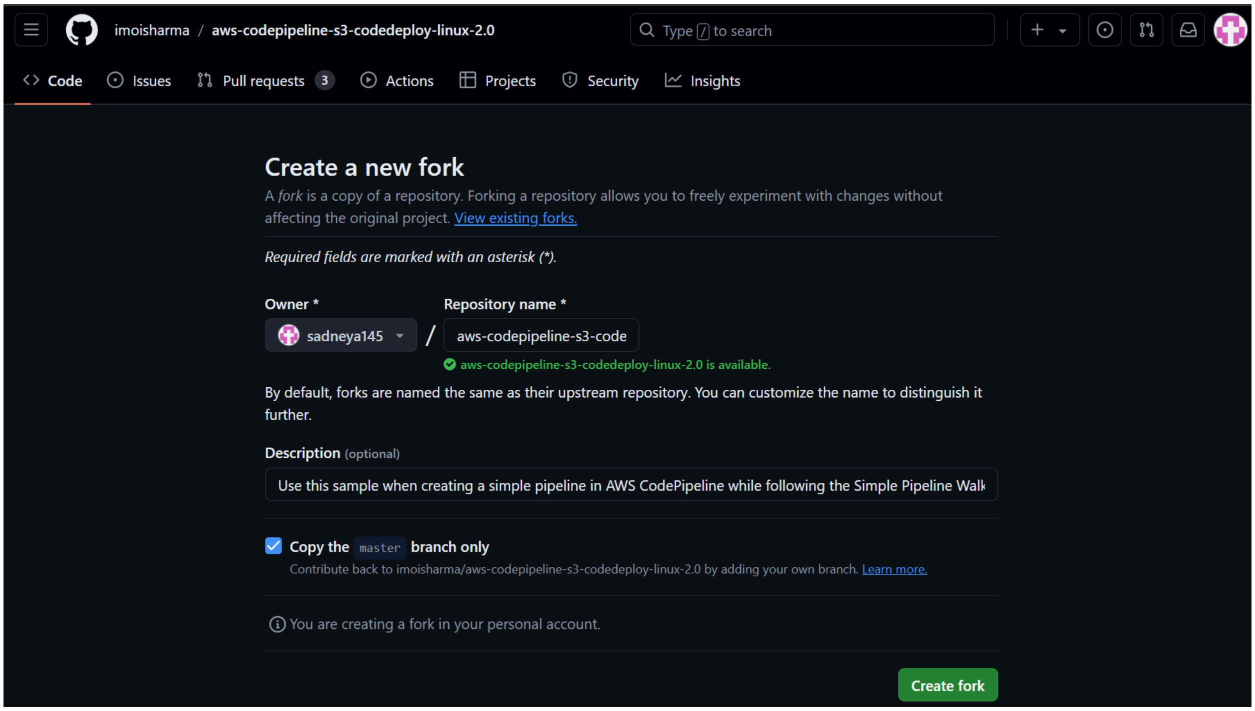
Platform

Platform: Node.js 20 running on 64bit Amazon Linux 2023/6.1.8

Running version: –

Platform state: Supported

Step 3: Get a copy of your sample code



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 *: sadneya145 / Repository name *: aws-codepipeline-s3-code

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 Walk

☑ Copy the `master` branch only

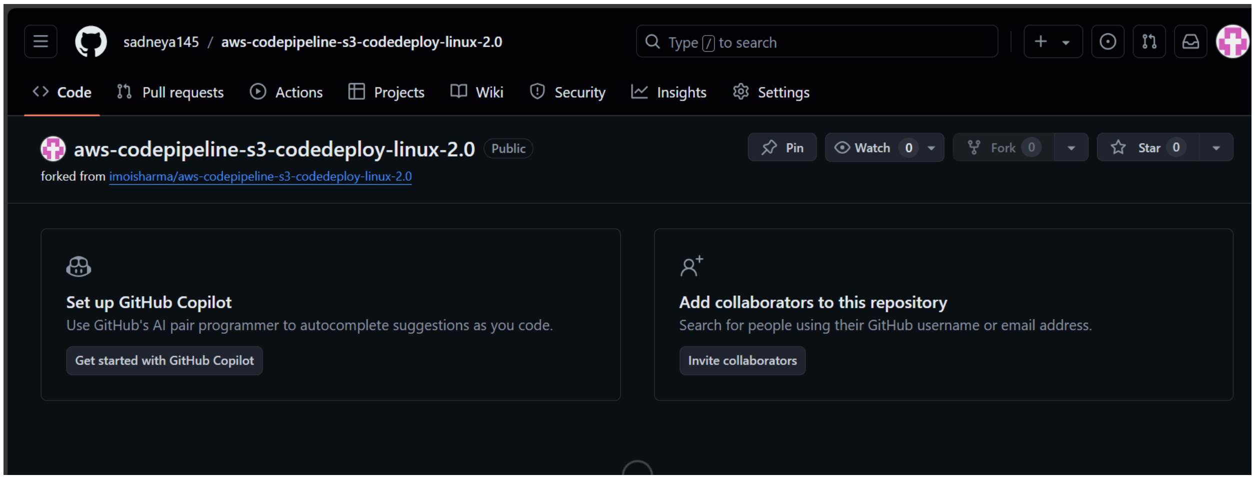
Contribute back to imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0 by adding your own branch. Learn more.

You are creating a fork in your personal account.

In this step, we will get the sample code from this GitHub Repository to later host it. The pipeline takes code from the source and then performs actions on it.

For this experiment, as a source, we will use this forked GitHub repository.
We can alternatively also use Amazon S3 and AWS CodeCommit.

Go to the repository shared above and simply fork it.


sadneya145 / aws-codepipeline-s3-codedeploy-linux-2.0
Type / to search
Code
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
aws-codepipeline-s3-codedeploy-linux-2.0
Public
Pin
Watch 0
Fork 0
Star 0

forked from imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0

Set up GitHub Copilot

Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Get started with GitHub Copilot

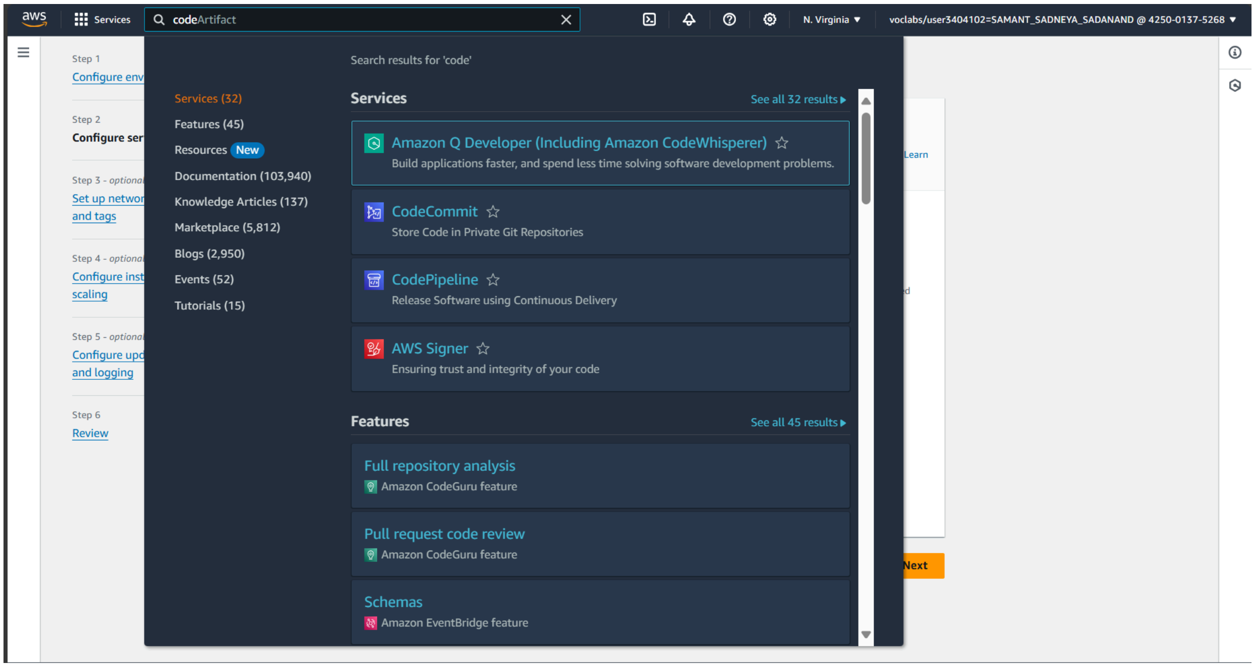
Add collaborators to this repository

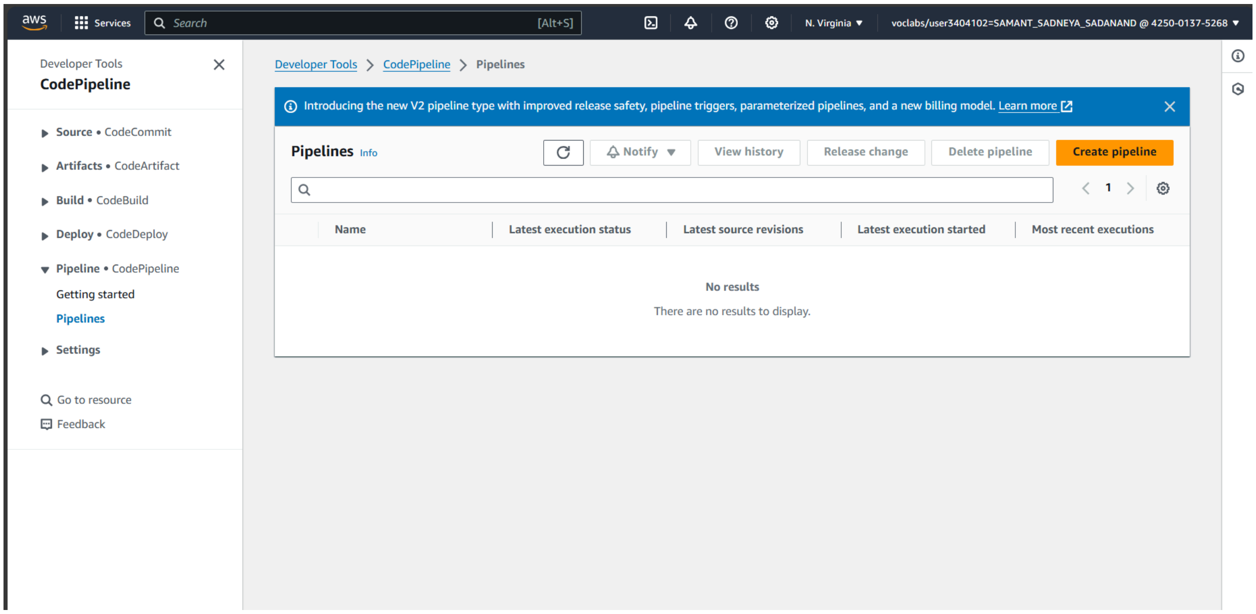
Search for people using their GitHub username or email address.

Step 4: Creating a CodePipeline

In this step, we'll create a simple pipeline that has its source and deployment information. In this case, however, we will skip the build stage where you get to plug in our preferred build provider.

1. Search CodePipeline in the search bar and click on create a new Pipeline.





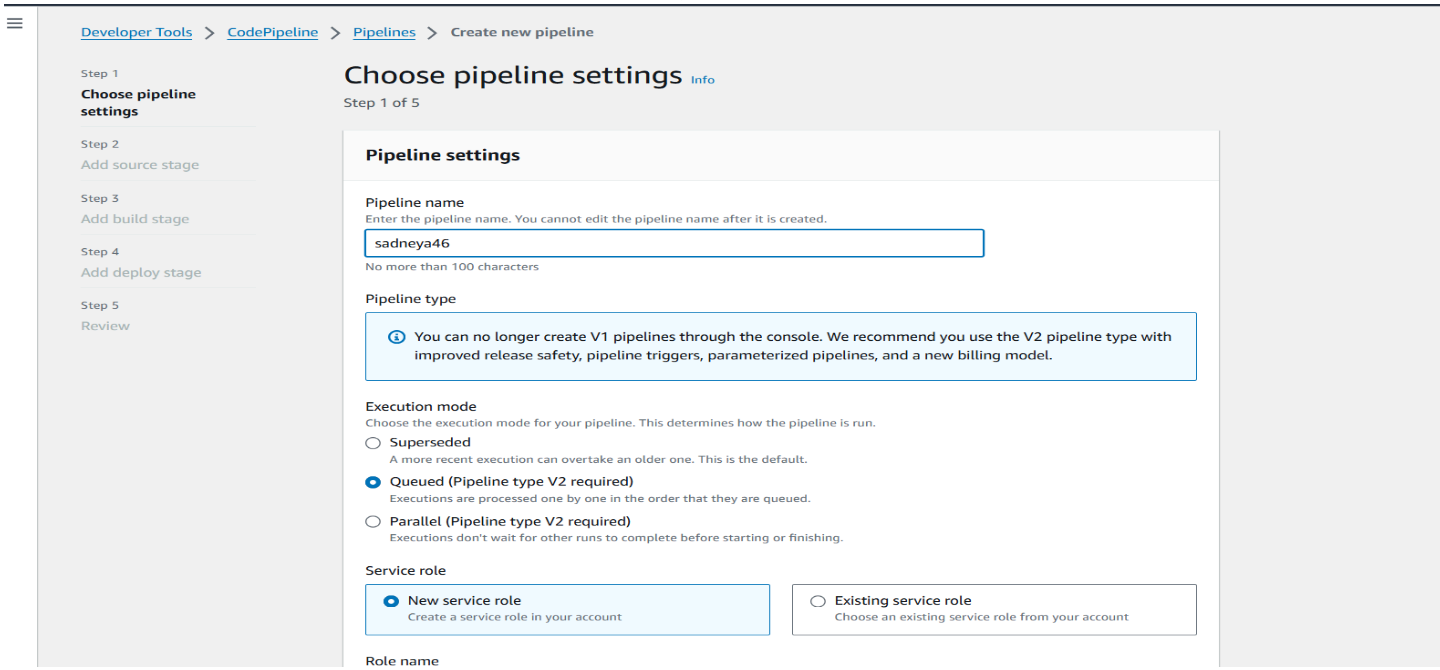
Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. Learn more

Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions

No results

There are no results to display.

2. Give a name to your pipeline. Here I have given name as sadenya_46



Choose pipeline settings

Step 1 of 5

Pipeline settings

Pipeline name

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

sadneya46

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.

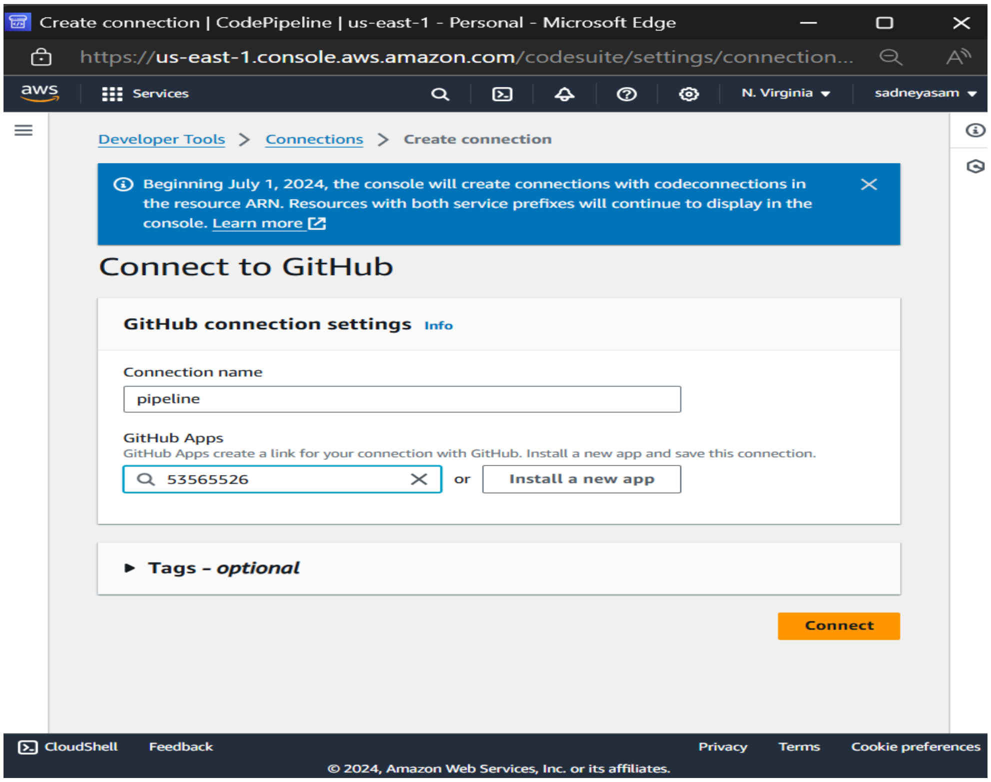
Service role

● New service role
Create a service role in your account

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

Role name

3. In the source stage, choose GitHub v2 as the provider, then connect your GitHub account to AWS by creating a connection. You'd need your GitHub credentials and then you'd need to authorize and install AWS on the forked GitHub Repository.



Create connection | CodePipeline | us-east-1 - Personal - Microsoft Edge

https://us-east-1.console.aws.amazon.com/codesuite/settings/connection...

Developer Tools > Connections > Create connection

Beginning July 1, 2024, the console will create connections with codeconnections in the resource ARN. Resources with both service prefixes will continue to display in the console. Learn more

Connect to GitHub

GitHub connection settings Info

Connection name

pipeline

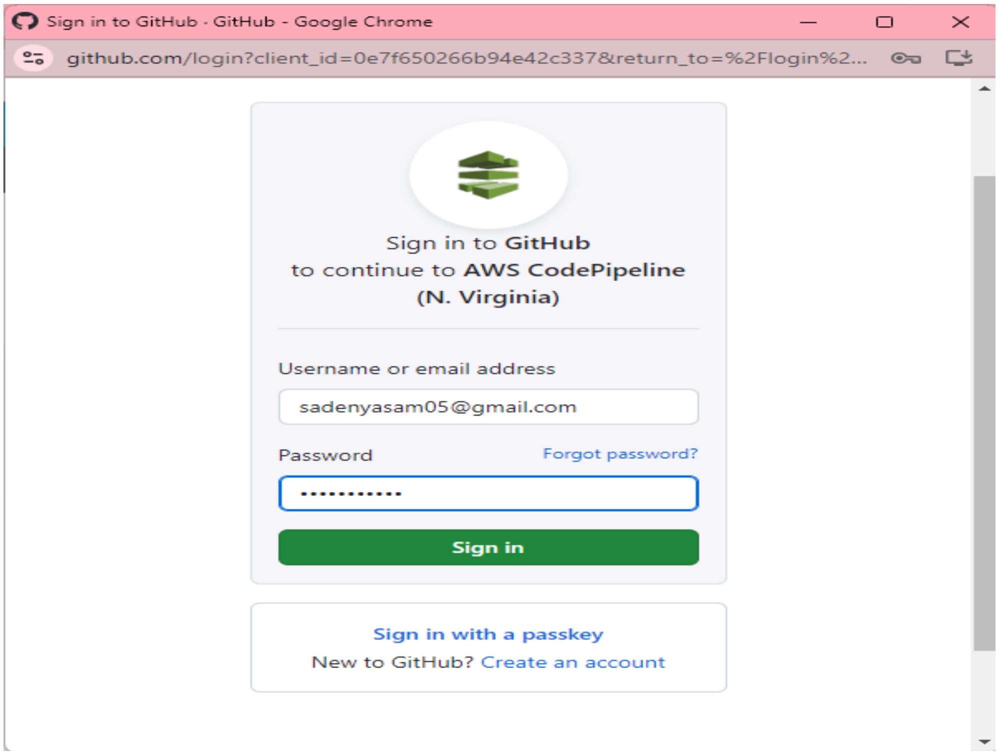
GitHub Apps

GitHub Apps create a link for your connection with GitHub. Install a new app and save this connection.

53565526 or Install a new app

▸ Tags - *optional*

© 2024, Amazon Web Services, Inc. or its affiliates.



Sign in to GitHub · GitHub - Google Chrome

github.com/login?client_id=0e7f650266b94e42c337&return_to=%2Flogin%2...

Sign in to GitHub
to continue to **AWS CodePipeline (N. Virginia)**

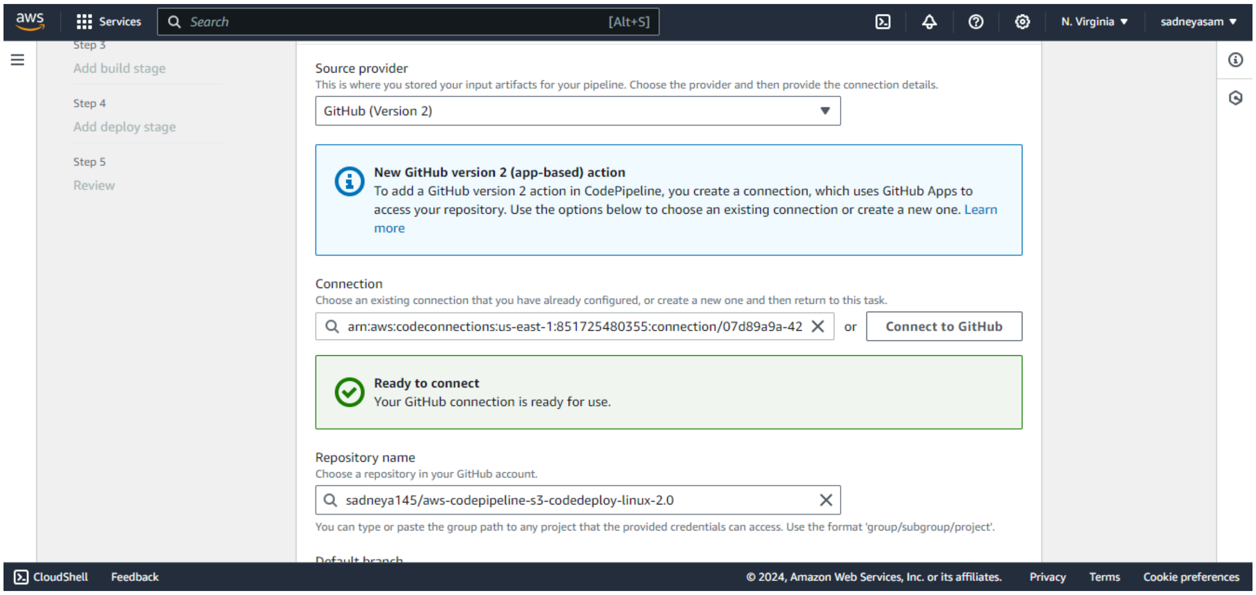
Username or email address

sadenyasam05@gmail.com

Password

••••••••••••

New to GitHub? Create an account



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

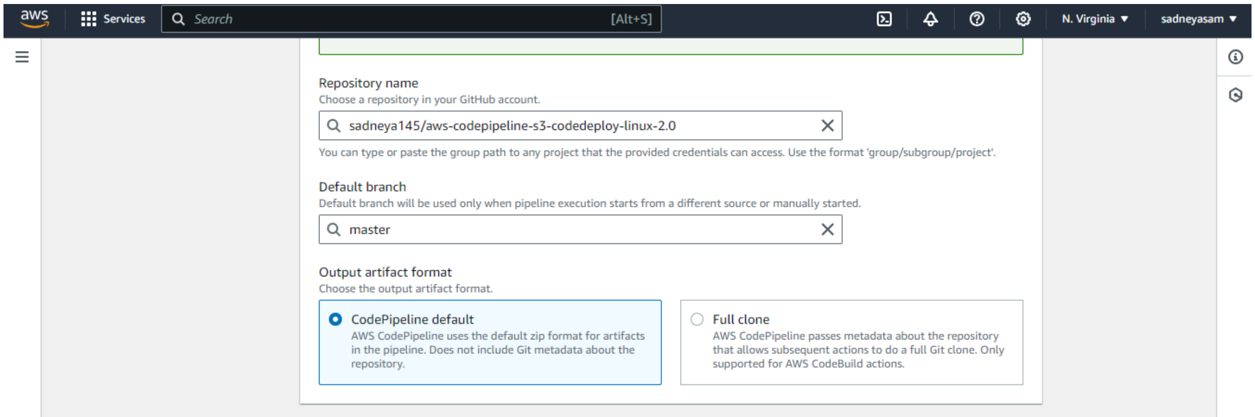
Connection
Choose an existing connection that you have already configured, or create a new one and then return to this task.
arn:aws:codeconnections:us-east-1:851725480355:connection/07d89a9a-42 ✕ or Connect to GitHub

Ready to connect
Your GitHub connection is ready for use.

Repository name
Choose a repository in your GitHub account.
sadneya145/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

4. select the forked repository then select the master branch.



Repository name
Choose a repository in your GitHub account.
sadneya145/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'.

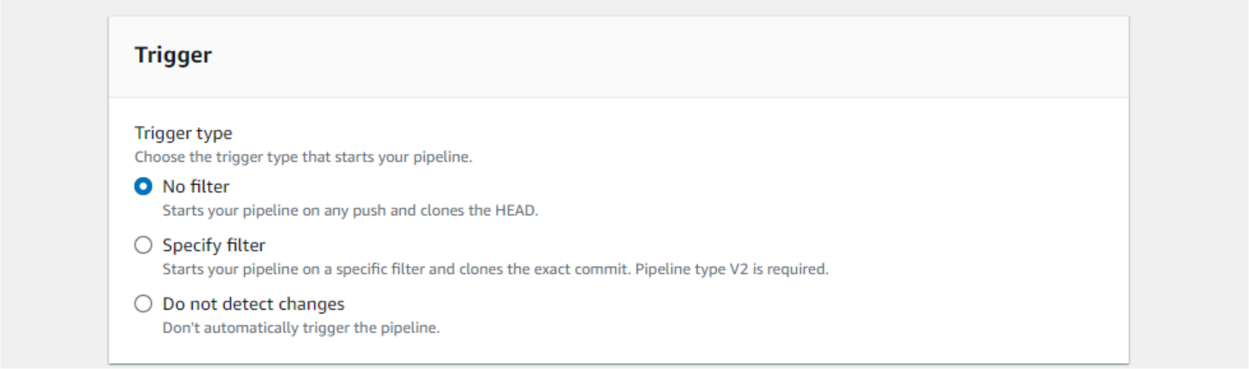
Default branch
Default branch will be used only when pipeline execution starts from a different source or manually started.
master ✕

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.

5. Then select trigger type none.



Trigger

Trigger type
Choose the trigger type that starts your pipeline.

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

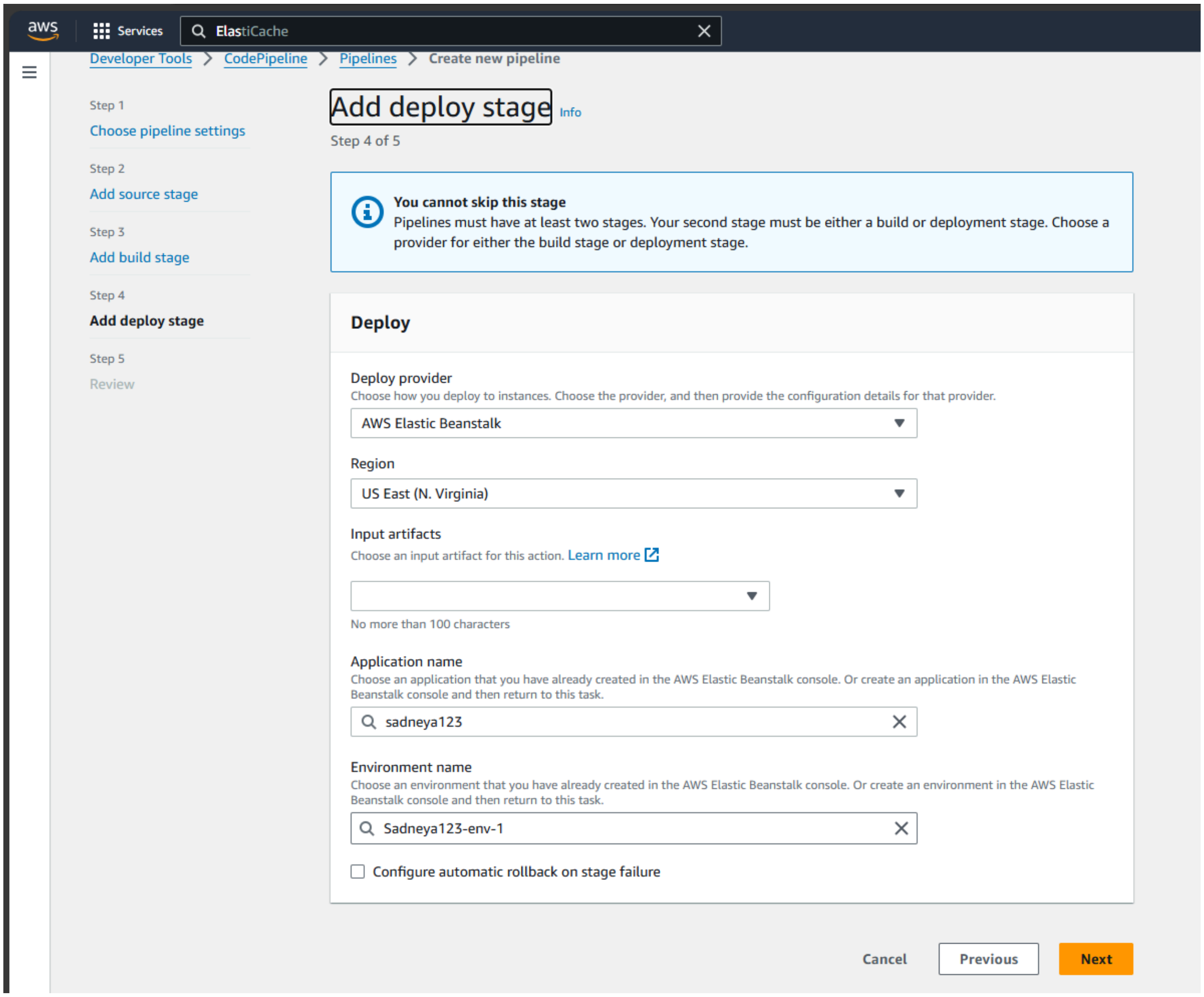
○ Specify filter
Starts your pipeline on a specific filter and clones the exact commit. Pipeline type V2 is required.

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

After that, click Continue and skip the build stage. Proceed to the Deployment stage.

Step 5: Deployment

1. Choose Beanstalk as the Deploy Provider, same region as the Bucket and Beanstalk, name and environment name.



Add deploy stage

Step 4 of 5

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

US East (N. Virginia)

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.

sadneya123

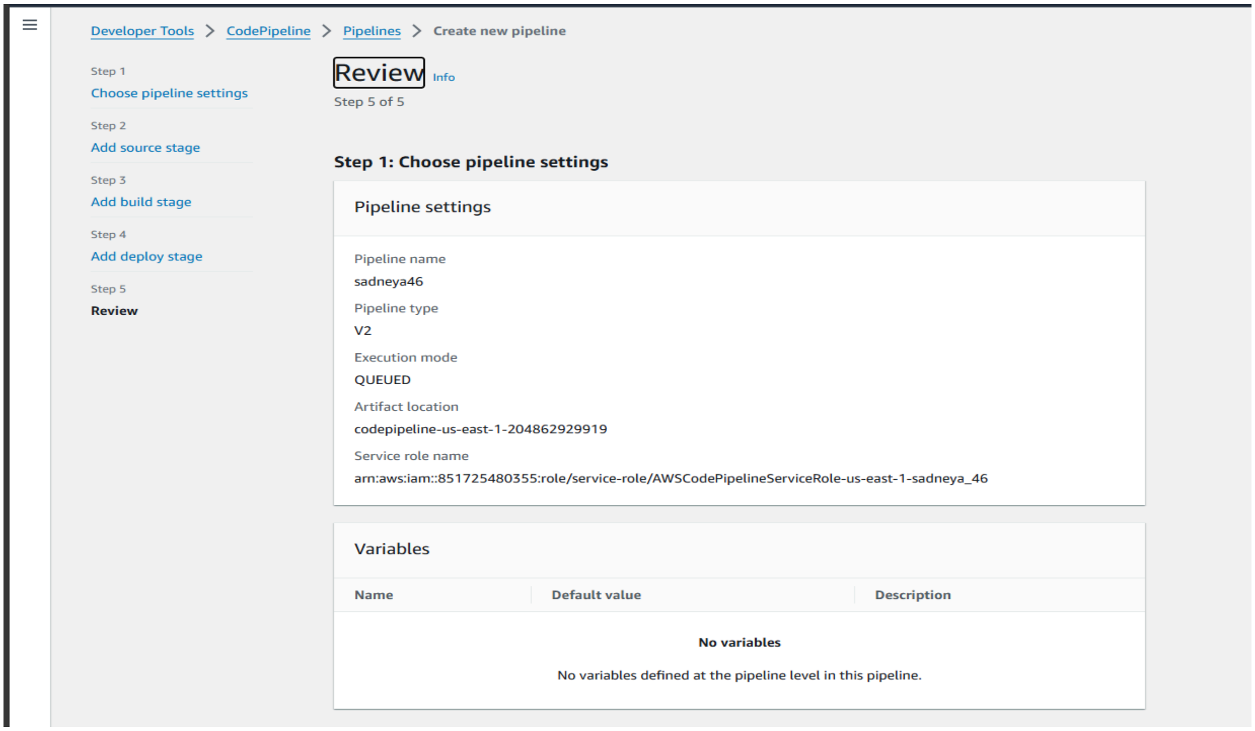
Environment name

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.

Sadneya123-env-1

☐ Configure automatic rollback on stage failure

2.Click Next, Review and create the pipeline.



Review

Step 5 of 5

Step 1: Choose pipeline settings

Pipeline settings

Pipeline name
sadneya46

Pipeline type
V2

Execution mode
QUEUED

Artifact location
codepipeline-us-east-1-204862929919

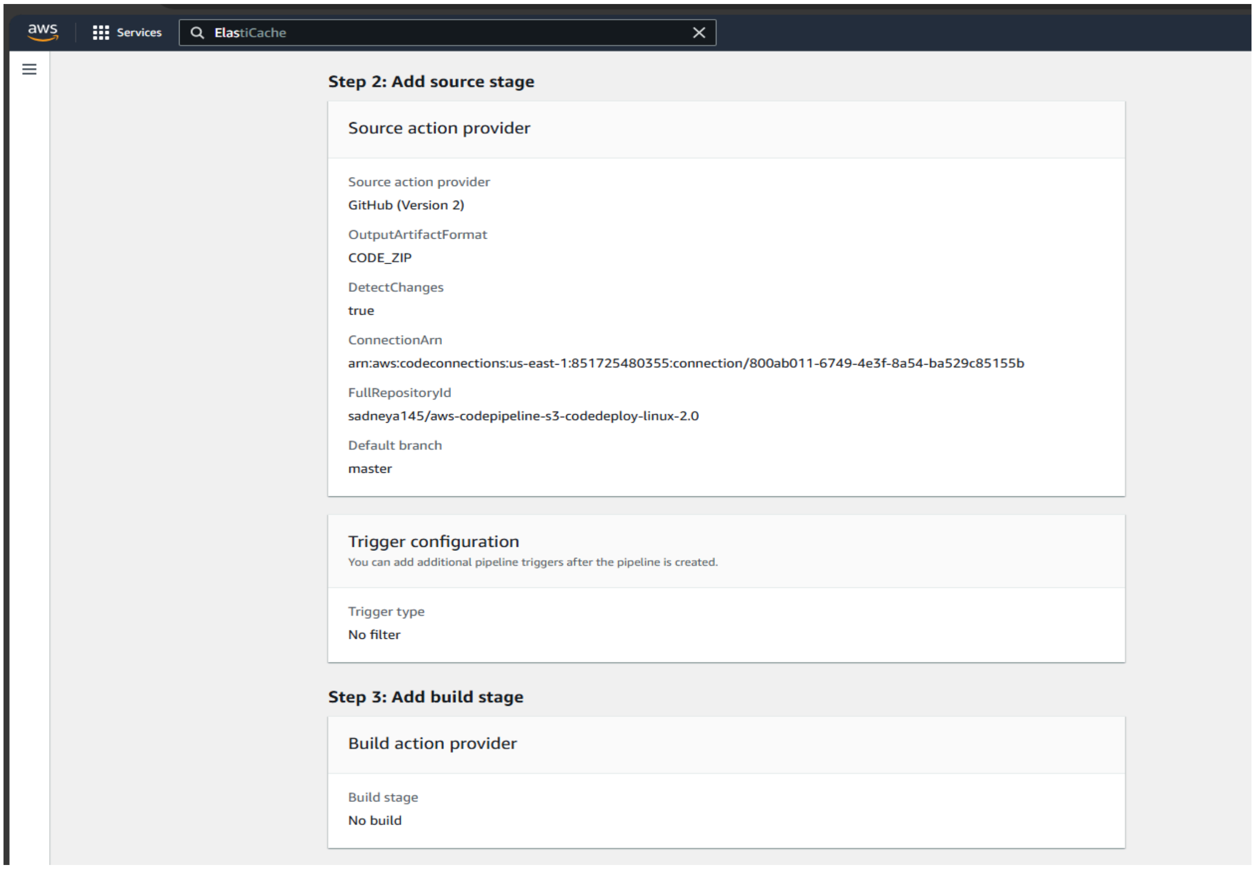
Service role name
arn:aws:iam::851725480355:role/service-role/AWSCodePipelineServiceRole-us-east-1-sadneya_46

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
GitHub (Version 2)

OutputArtifactFormat
CODE_ZIP

DetectChanges
true

ConnectionArn
arn:aws:codeconnections:us-east-1:851725480355:connection/800ab011-6749-4e3f-8a54-ba529c85155b

FullRepositoryId
sadneya145/aws-codepipeline-s3-codedeploy-linux-2.0

Default branch
master

Trigger configuration

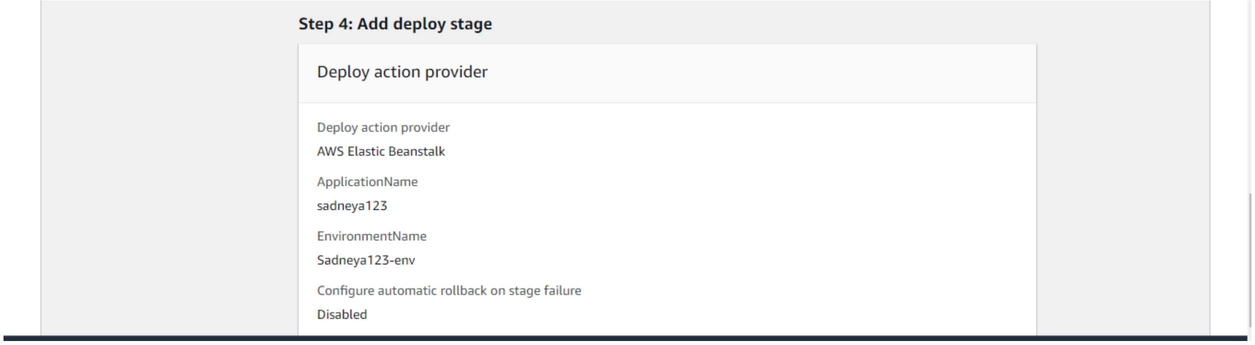
You can add additional pipeline triggers after the pipeline is created.

Trigger type
No filter

Step 3: Add build stage

Build action provider

Build stage
No build



Step 4: Add deploy stage

Deploy action provider

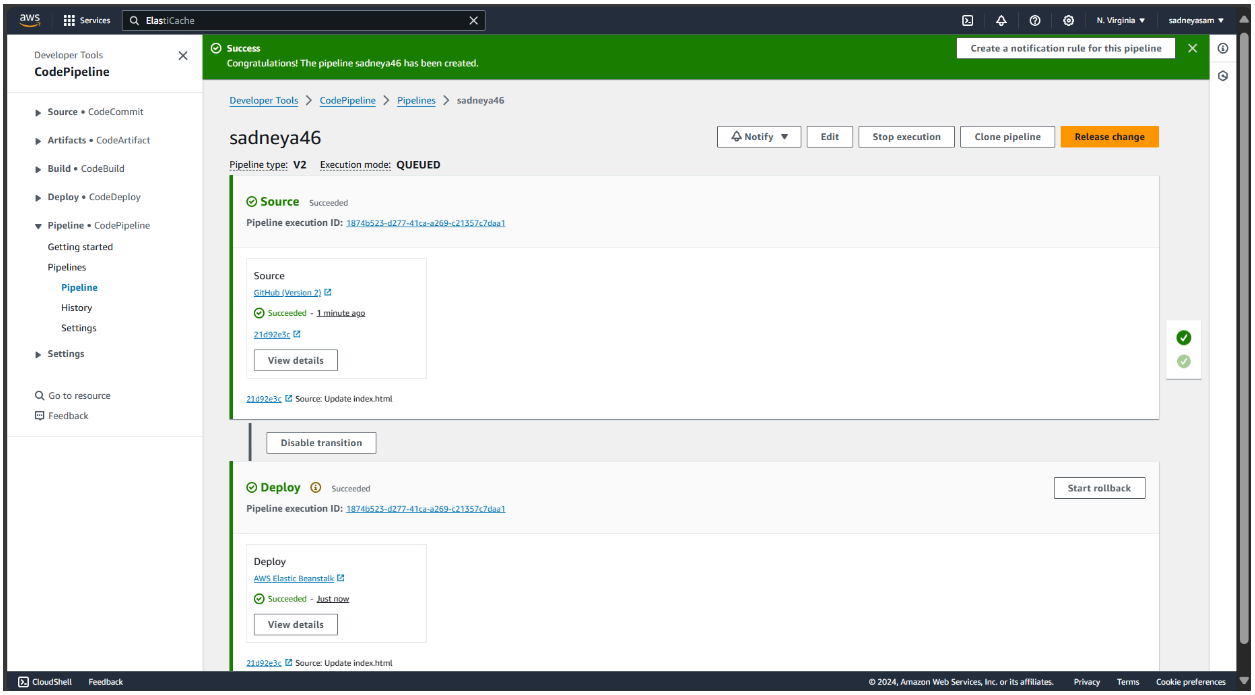
Deploy action provider
AWS Elastic Beanstalk

ApplicationName
sadneya123

EnvironmentName
Sadneya123-env

Configure automatic rollback on stage failure
Disabled

3. Then it will give you this result on screen. i.e. deployed successfully.



Success
Congratulations! The pipeline sadneya46 has been created.

Developer Tools > CodePipeline > Pipelines > sadneya46

sadneya46

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded
Pipeline execution ID: 1874b523-d277-41ca-a269-c21357c7daa1

Source
GitHub (Version 2)
Succeeded - 1 minute ago
21d92e3c

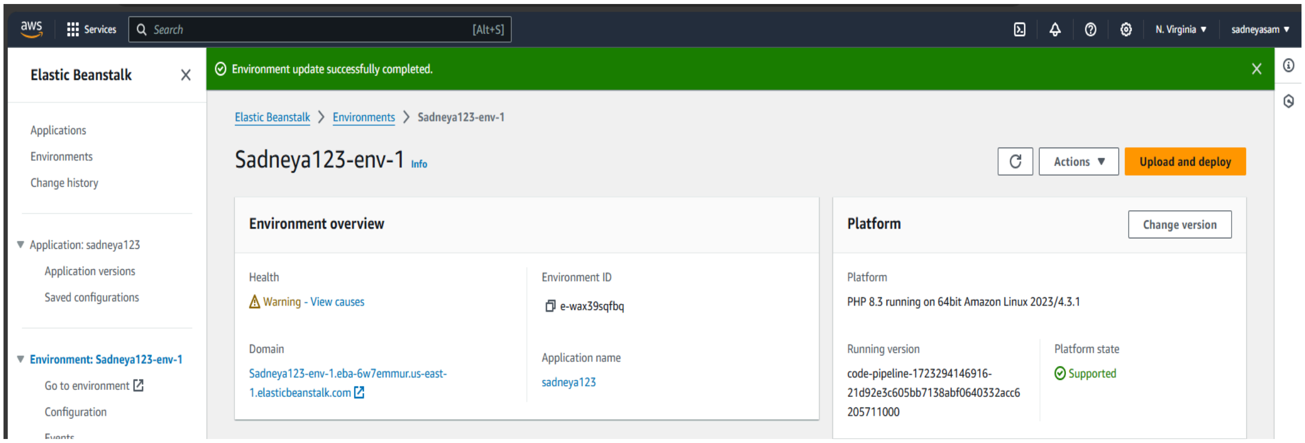
21d92e3c Source: Update index.html

Deploy Succeeded
Pipeline execution ID: 1874b523-d277-41ca-a269-c21357c7daa1

Deploy
AWS Elastic Beanstalk
Succeeded - Just now

21d92e3c Source: Update index.html

4. In a few minutes the website will get hosted successfully. Then click on the url present over the environment created on Elastic Benstalk.



Environment update successfully completed.

Elastic Beanstalk > Environments > Sadneya123-env-1

Sadneya123-env-1

Environment overview

Health
Warning - View causes

Environment ID
e-wax39sqfbq

Domain
Sadneya123-env-1.eba-6w7emmur.us-east-1.elasticbeanstalk.com

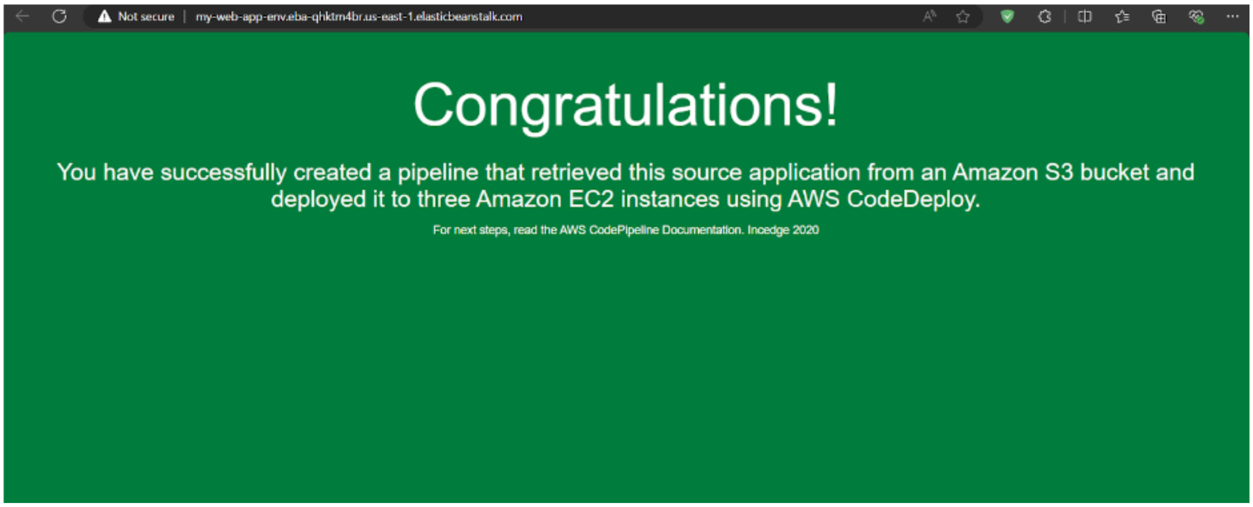
Application name
sadneya123

Platform

Platform
PHP 8.3 running on 64bit Amazon Linux 2023/4.3.1

Running version
code-pipeline-1723294146916-21d92e3c605bb7138abf0640332acc6205711000

Platform state
Supported



Not secure | my-web-app-env.eba-qhktm4br.us-east-1.elasticbeanstalk.com

Congratulations!

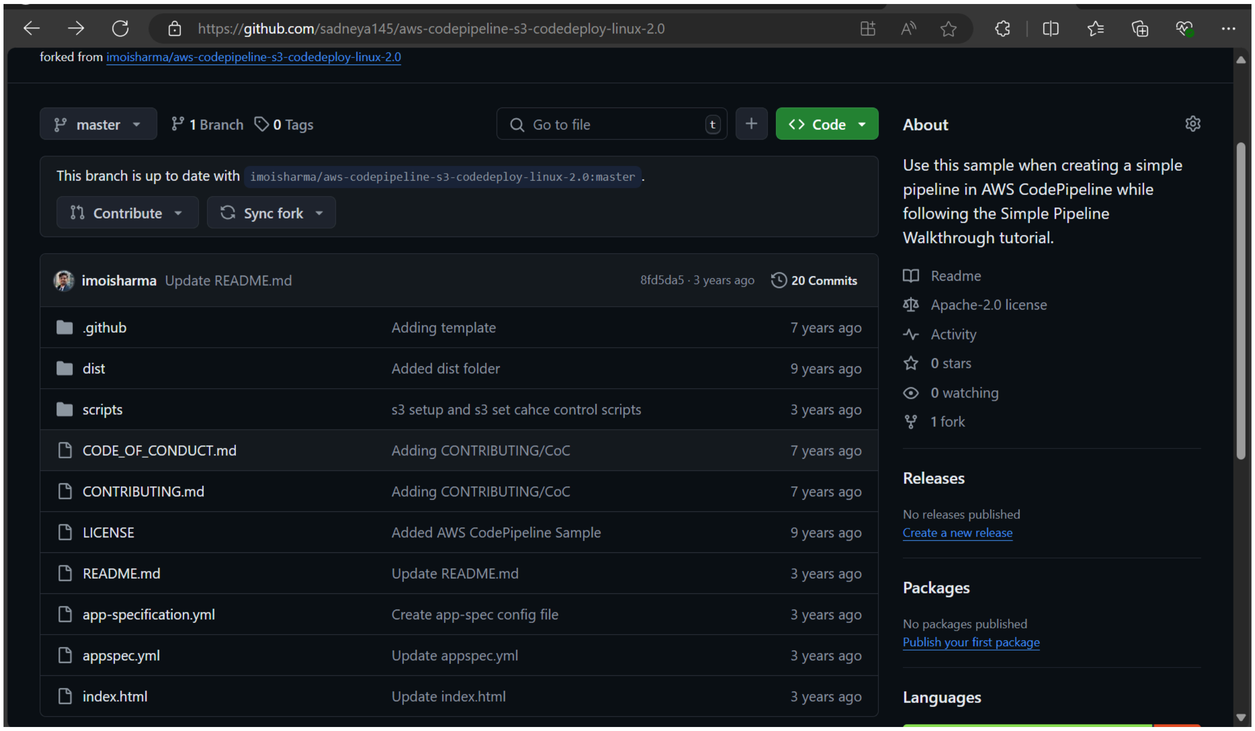
You have successfully created a pipeline that retrieved this source application from an Amazon S3 bucket and deployed it to three Amazon EC2 instances using AWS CodeDeploy.

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.

Step 6: Committing changes to update app

1. In this we make some changes in the file.Open github.com then open the forked repository. Then update the changes in the index.html file and finally commit those changes.



forked from imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0

This branch is up to date with `imoisharma/aws-codepipeline-s3-codedeploy-linux-2.0:master`.

imoisharma Update README.md 8fd5da5 · 3 years ago 20 Commits

.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

About

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

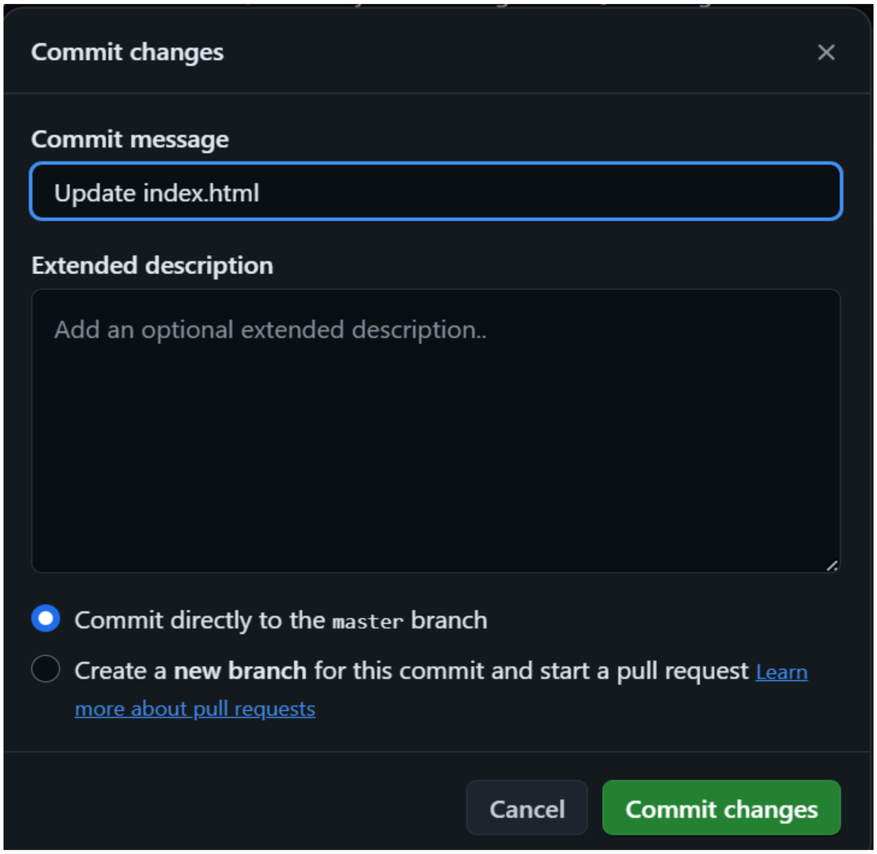
Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages



Commit changes

Commit message

Update index.html

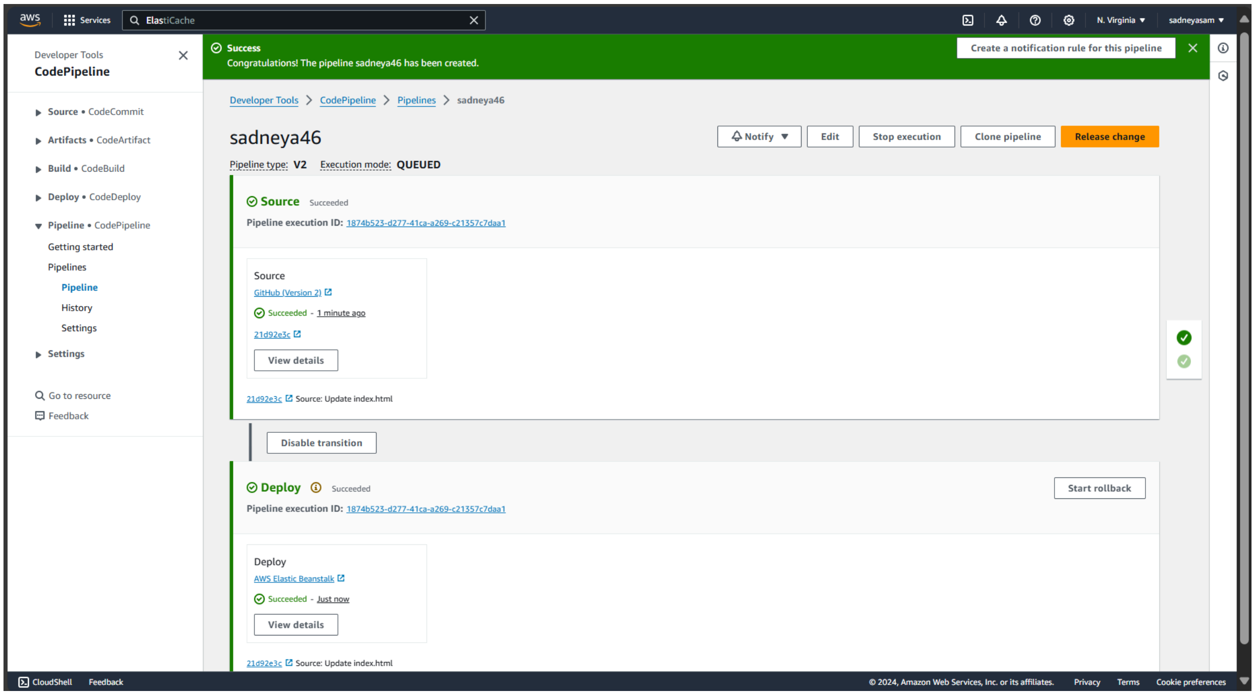
Extended description

Add an optional extended description..

● Commit directly to the `master` branch

○ Create a **new branch** for this commit and start a pull request [Learn more about pull requests]

2. Then again start the deployment of the pipeline.



Success
Congratulations! The pipeline sadneya46 has been created.

Developer Tools > CodePipeline > Pipelines > sadneya46

sadneya46

Pipeline type: V2   Execution mode: QUEUED

Source   Succeeded
Pipeline execution ID: 1874b523-d277-41ca-a269-c21357c7daa1

Source
GitHub (Version 2)
Succeeded - 1 minute ago
21d92e3c

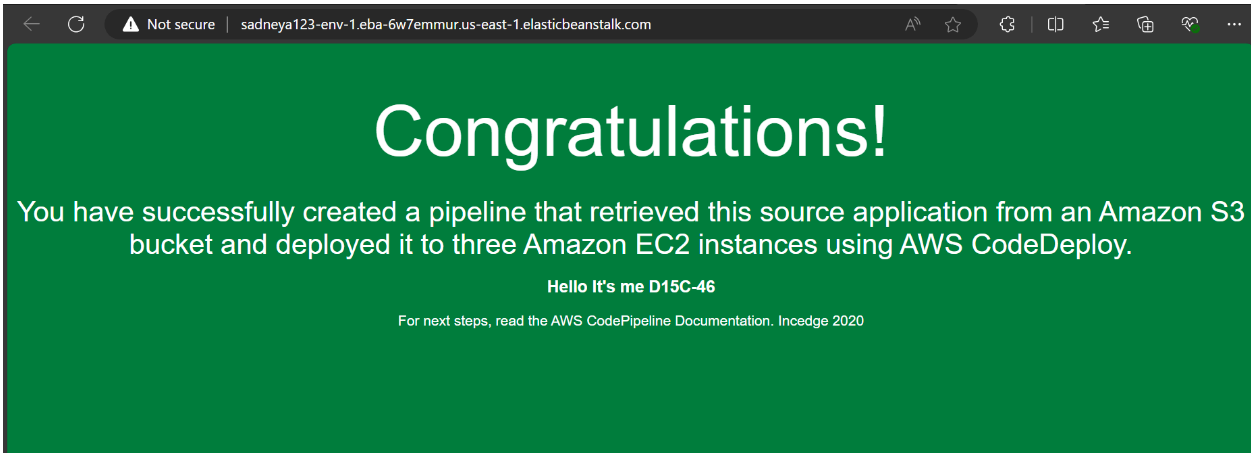
21d92e3c Source: Update index.html

Deploy   Succeeded
Pipeline execution ID: 1874b523-d277-41ca-a269-c21357c7daa1

Deploy
AWS Elastic Beanstalk
Succeeded - Just now

21d92e3c Source: Update index.html

3. Check the changes in the website , here I have added a message in h3 tag.



Not secure
sadneya123-env-1.eba-6w7emmur.us-east-1.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.

Hello It's me D15C-46

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