VASANTDADA PATIL PRATISHTHAN'S COLLEGE OF ENGINEERING AND VISUAL ARTS

ISO 9001:2015 Certified Institute

Department of Information Technology

NBA Accredited Course (Dated 01/07/2024 to 30/06/2027)

# EXPERIMENT - 2

**Aim:** To Build Your Application using AWS Code Build and Deploy on S3 / SEBS using AWS Code Pipeline, deploy Sample Application on EC2 instance using AWS Code Deploy.

## Theory:

AWS Code Pipeline:

Continuous deployment allows you to deploy revisions to a production environment automatically without explicit approval from a developer, making the entire software release process automated. You will create the pipeline using AWS CodePipeline, a service that builds, tests, and deploys your code every time there is a code change. You will use your GitHub account, an Amazon Simple Storage Service (S3) bucket, or an AWS CodeCommit repository as the source location for the sample app's code. You will also use AWS Elastic Beanstalk as the deployment target for the sample app. Your completed pipeline will be able to detect changes made to the source repository containing the sample app and then automatically update your live sample app.

AWS Code Build:

AWS CodeBuild is a fully managed build service in the cloud. CodeBuild compiles your source code, runs unit tests, and produces artifacts that are ready to deploy. CodeBuild eliminates the need to provision, manage, and scale your own build servers. It provides prepackaged build environments for popular programming languages and build tools such as Apache Maven, Gradle, and more. You can also customize build environments in CodeBuild to use your own build tools. CodeBuild scales automatically to meet peak build requests.

You can add CodeBuild as a build or test action to the build or test stage of a pipeline in AWS CodePipeline. AWS CodePipeline is a continuous delivery service that you can use to model, visualize, and automate the steps required to release your code. This includes building your code. A pipeline is a workflow construct that describes how code changes go through a release process.In this Experiment, you use AWS CodeBuild to build a collection of sample source code input files (build input artifacts or build input) into a deployable version of the source code (build output artifact or build output). Specifically, you instruct CodeBuild to use Apache Maven, a common build tool, to build a set of Java class files into a Java Archive (JAR) file.

AWS S3:

Amazon S3 (Simple Storage Service) is a highly scalable, durable, and secure object storage service provided by AWS. It is designed to store and retrieve any amount of data from anywhere on the web. Here's an overview of some key concepts and features of AWS S3

## Steps:

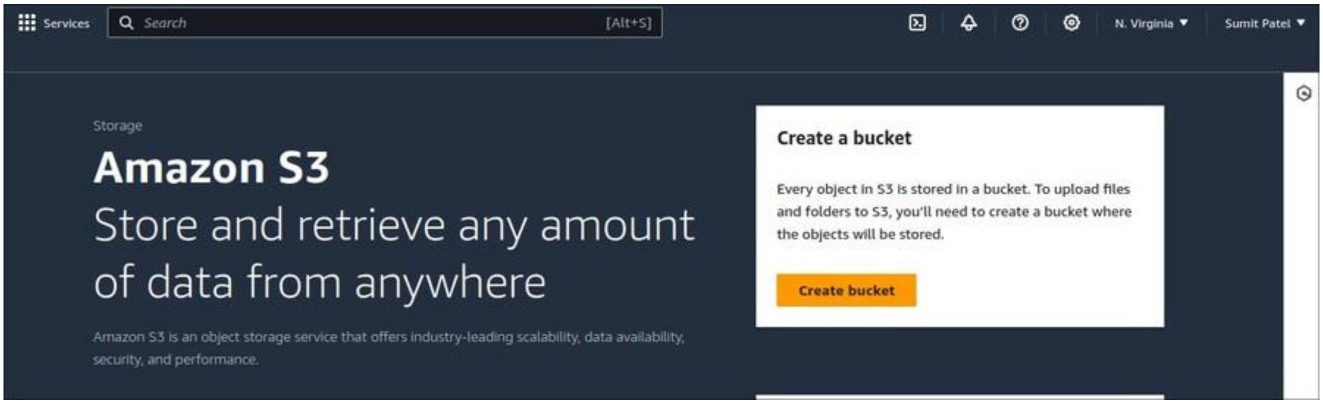
1. Create your project / source code. / Get your any project.

In this step you can need to choose any existing project you wanted to deploy, or create new project / source code. You can push to any git server.

2. Create a buildspec.yml file

In this step, you create a build specification (build spec) file. A buildspec is a collection of build commands and related settings, in YAML format, that CodeBuild uses to run a build. Without a build spec, CodeBuild cannot successfully convert your build input into build output or locate the build output artifact in the build environment to upload to your output bucket. Create this file, name it buildspec.yml, and then save it in the root (top level) directory.

3.



Storage

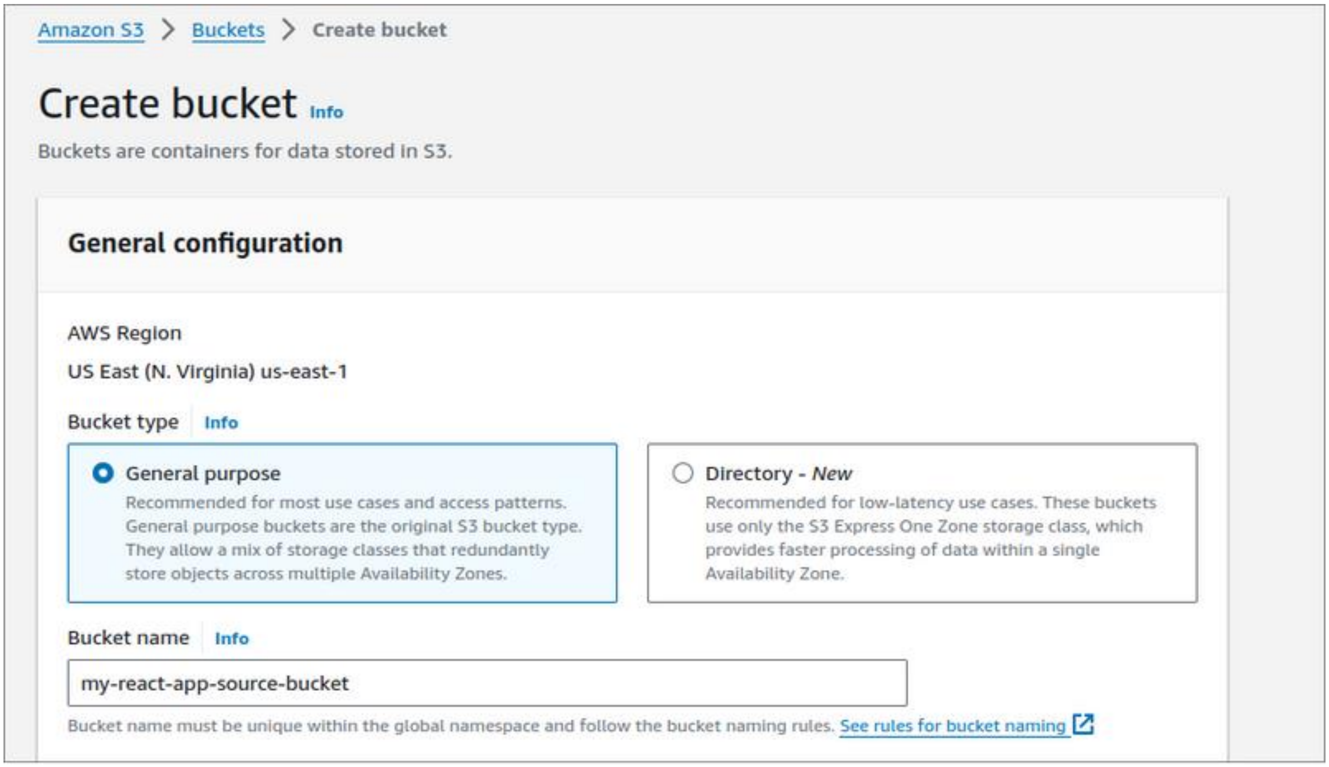
Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

US East (N. Virginia) us-east-1

Bucket type Info

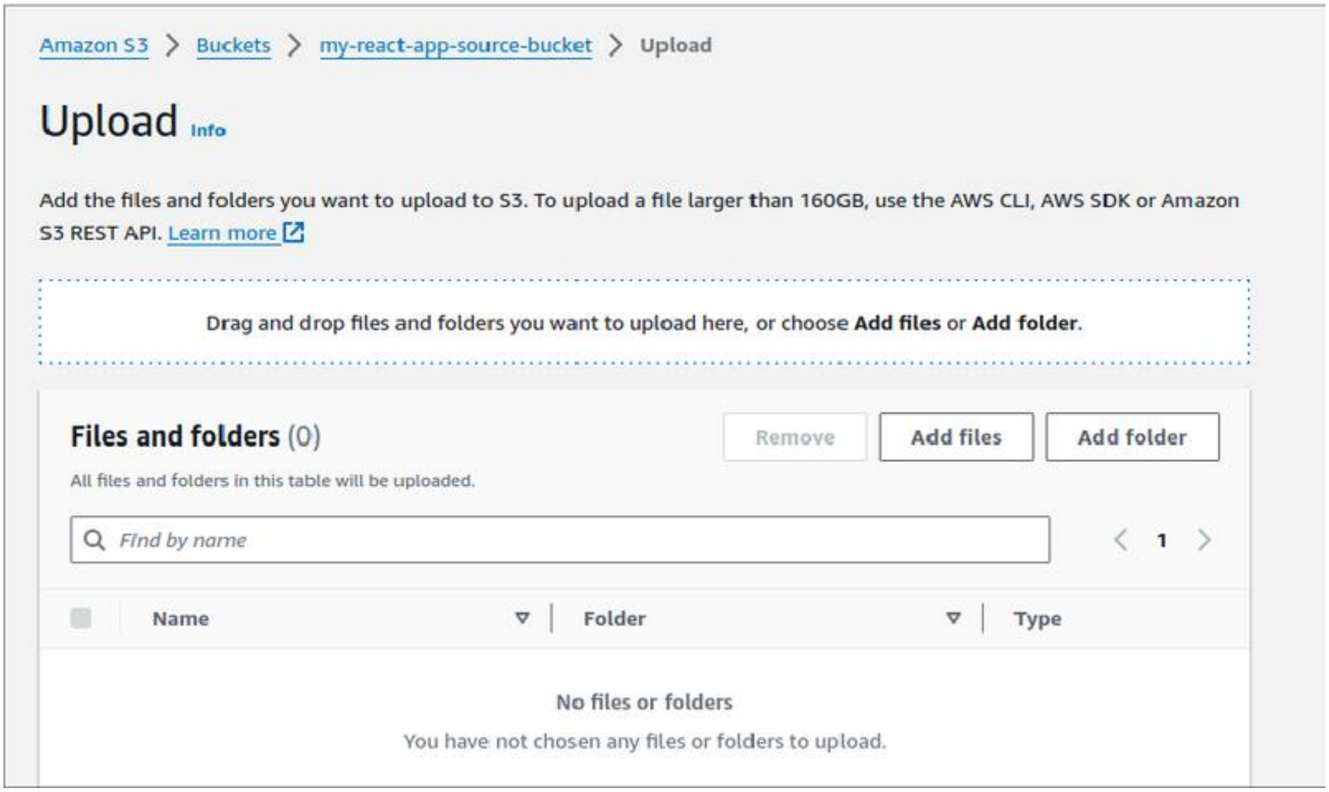
● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory - *New*
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info

my-react-app-source-bucket

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming



Upload Info

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

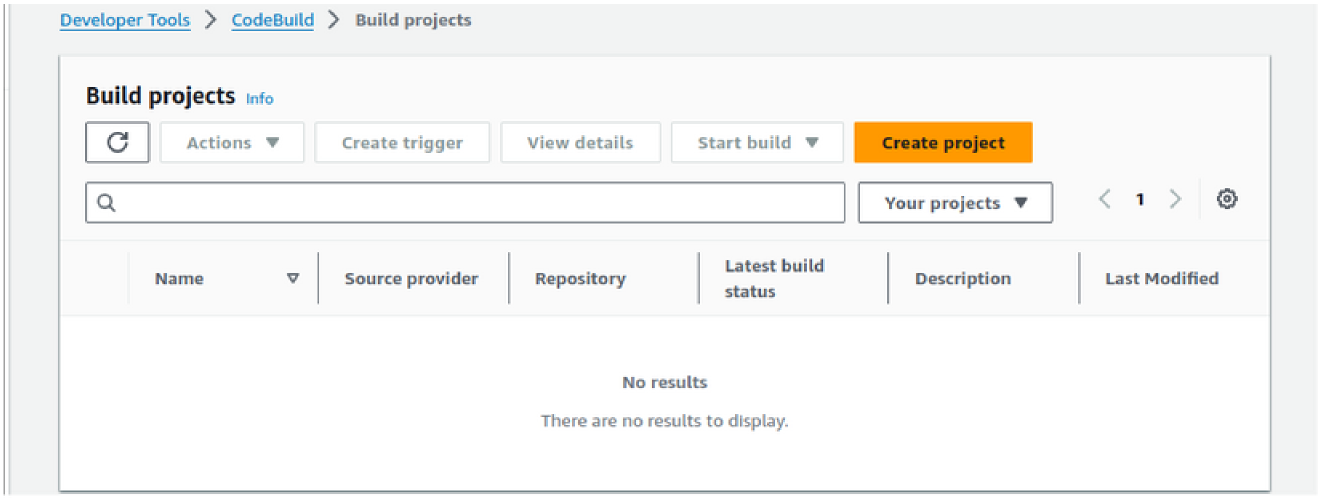
Files and folders (0)

All files and folders in this table will be uploaded.

| □ | Name | Folder | Type |
|---|---|---|---|

No files or folders

You have not chosen any files or folders to upload.



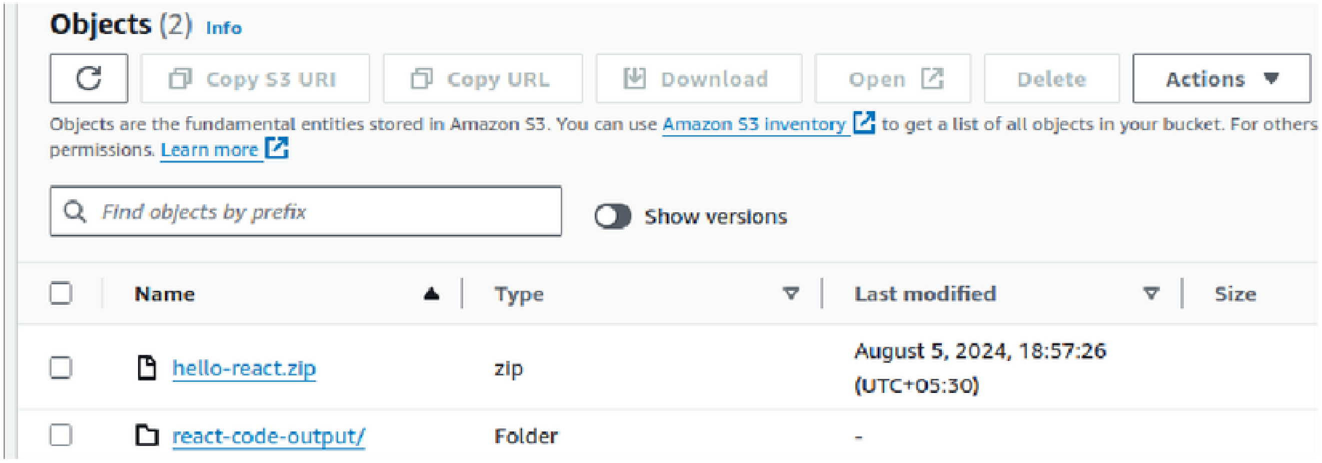
Developer Tools > CodeBuild > Build projects

Build projects Info

| Name | Source provider | Repository | Latest build status | Description | Last Modified |
|---|---|---|---|---|---|

No results

There are no results to display.



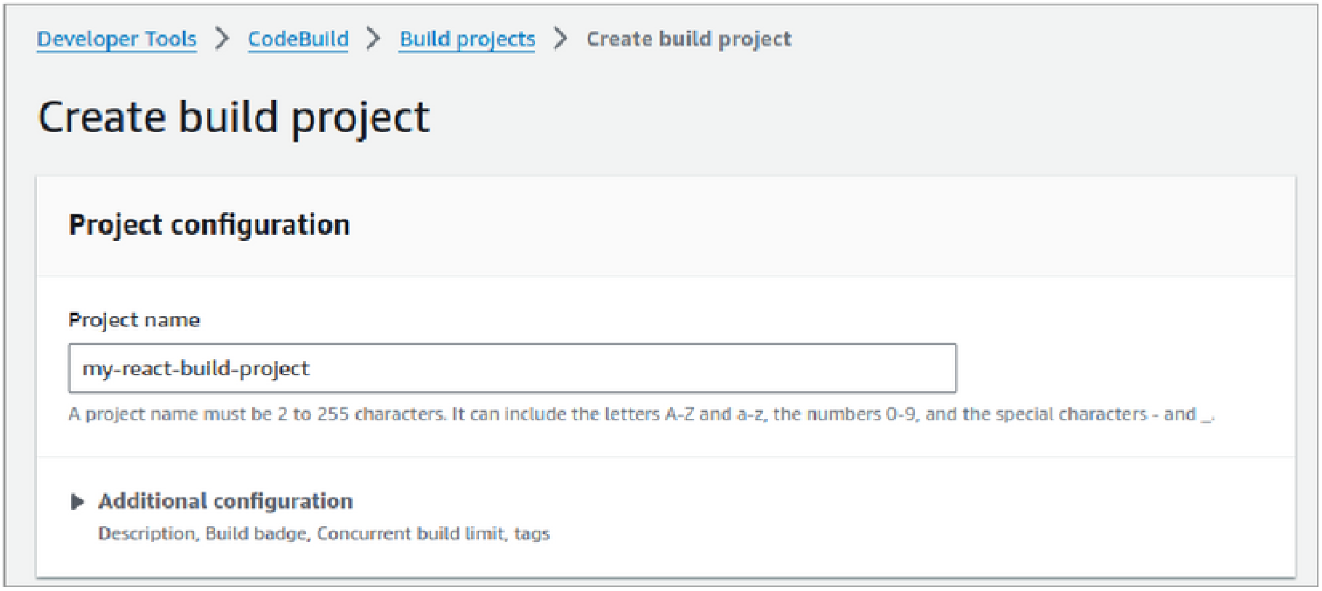
Objects (2) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others permissions. Learn more

Find objects by prefix

Show versions

| □ | Name | Type | Last modified | Size |
|---|---|---|---|---|
| □ | hello-react.zip | zip | August 5, 2024, 18:57:26 (UTC+05:30) | |
| □ | react-code-output/ | Folder | - | |



Developer Tools > CodeBuild > Build projects > Create build project

Create build project

Project configuration

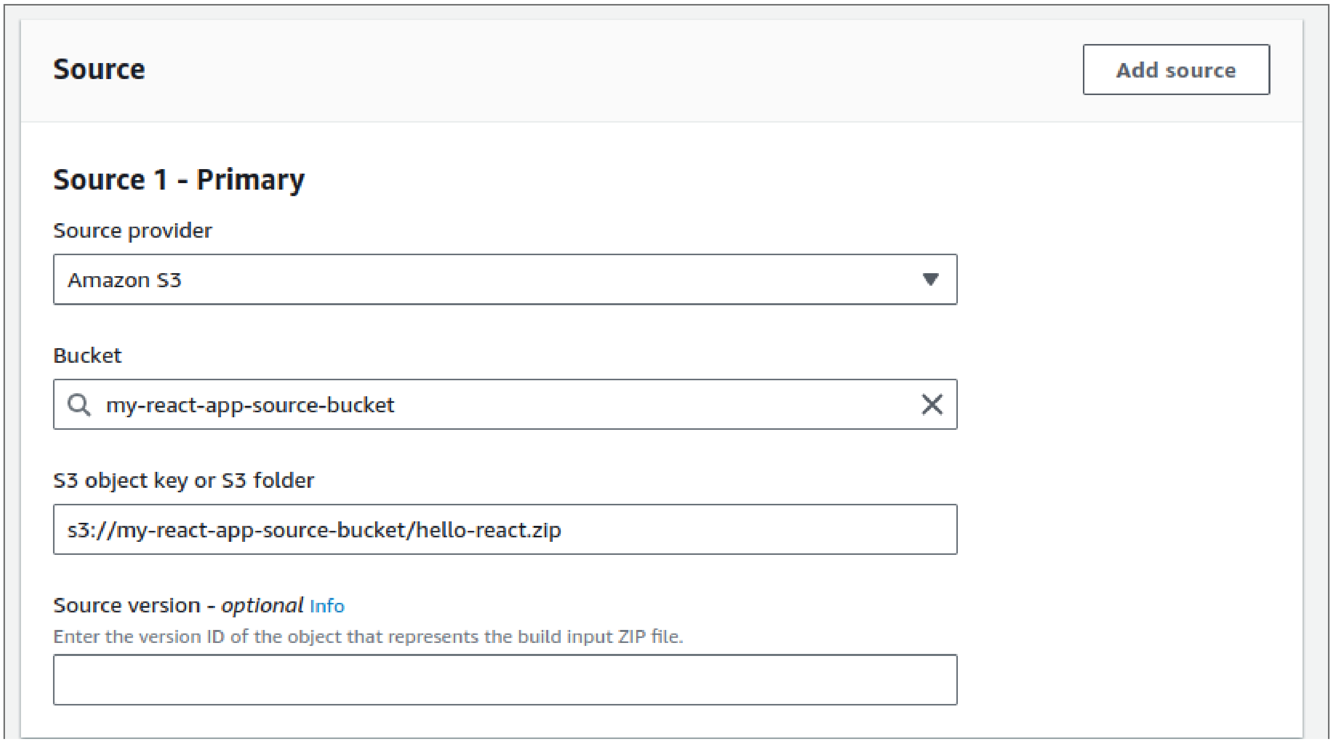
Project name

my-react-build-project

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

▶ Additional configuration

Description, Build badge, Concurrent build limit, tags



Source

Source 1 - Primary

Source provider

Amazon S3 ▼

Bucket

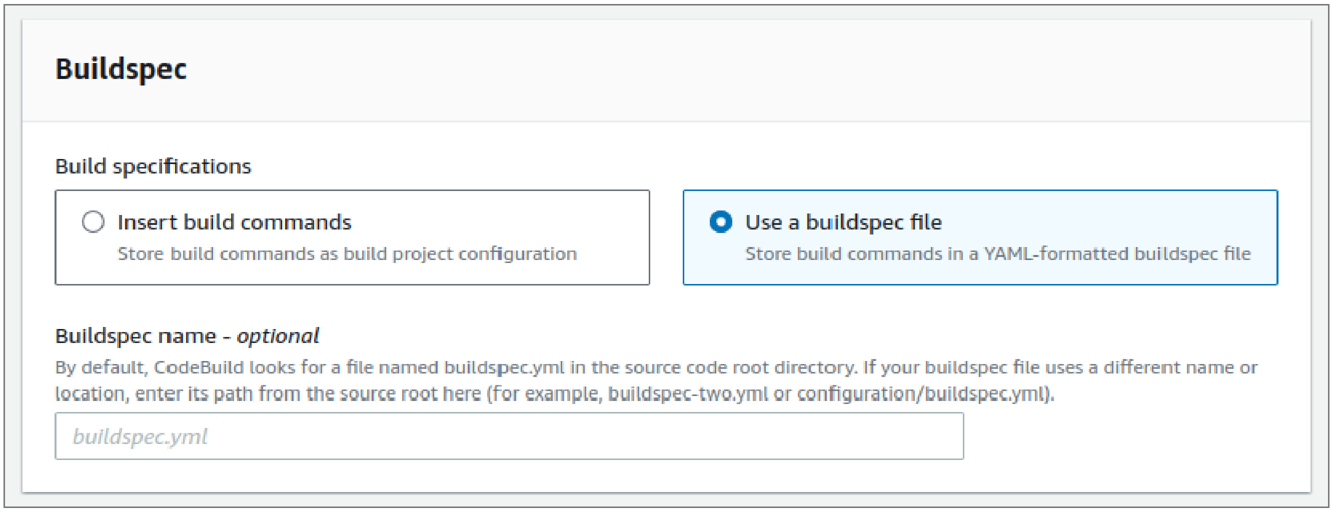
my-react-app-source-bucket

S3 object key or S3 folder

s3://my-react-app-source-bucket/hello-react.zip

Source version - *optional* Info

Enter the version ID of the object that represents the build input ZIP file.



Buildspec

Build specifications

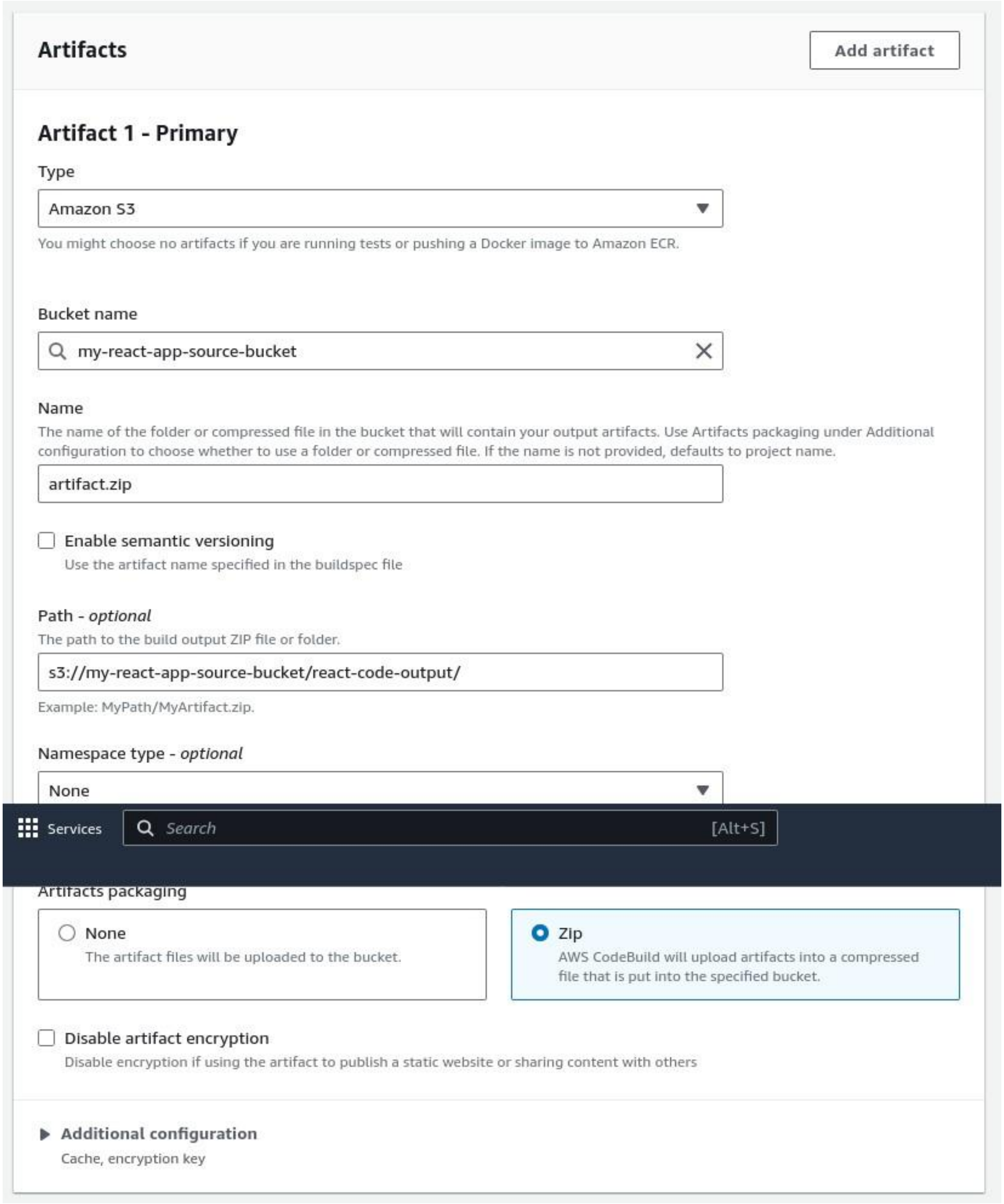
○ Insert build commands
Store build commands as build project configuration

● Use a buildspec file
Store build commands in a YAML-formatted buildspec file

Buildspec name - *optional*

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

*buildspec.yml*



## Artifacts

### Artifact 1 - Primary

Type

Amazon S3

You might choose no artifacts if you are running tests or pushing a Docker image to Amazon ECR.

Bucket name

my-react-app-source-bucket

Name

The name of the folder or compressed file in the bucket that will contain your output artifacts. Use Artifacts packaging under Additional configuration to choose whether to use a folder or compressed file. If the name is not provided, defaults to project name.

artifact.zip

☐ Enable semantic versioning

Use the artifact name specified in the buildspec file

Path - *optional*

The path to the build output ZIP file or folder.

s3://my-react-app-source-bucket/react-code-output/

Example: MyPath/MyArtifact.zip.

Namespace type - *optional*

None

Artifacts packaging

○ None

The artifact files will be uploaded to the bucket.

● Zip

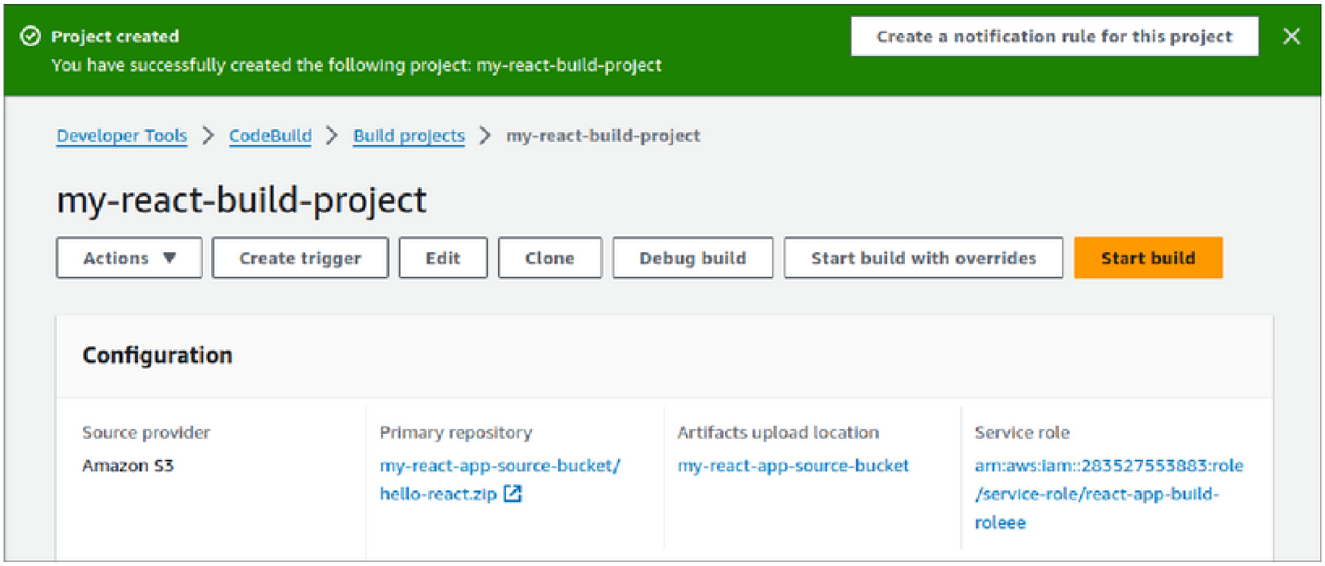
AWS CodeBuild will upload artifacts into a compressed file that is put into the specified bucket.

☐ Disable artifact encryption

Disable encryption if using the artifact to publish a static website or sharing content with others

▶ Additional configuration

Cache, encryption key



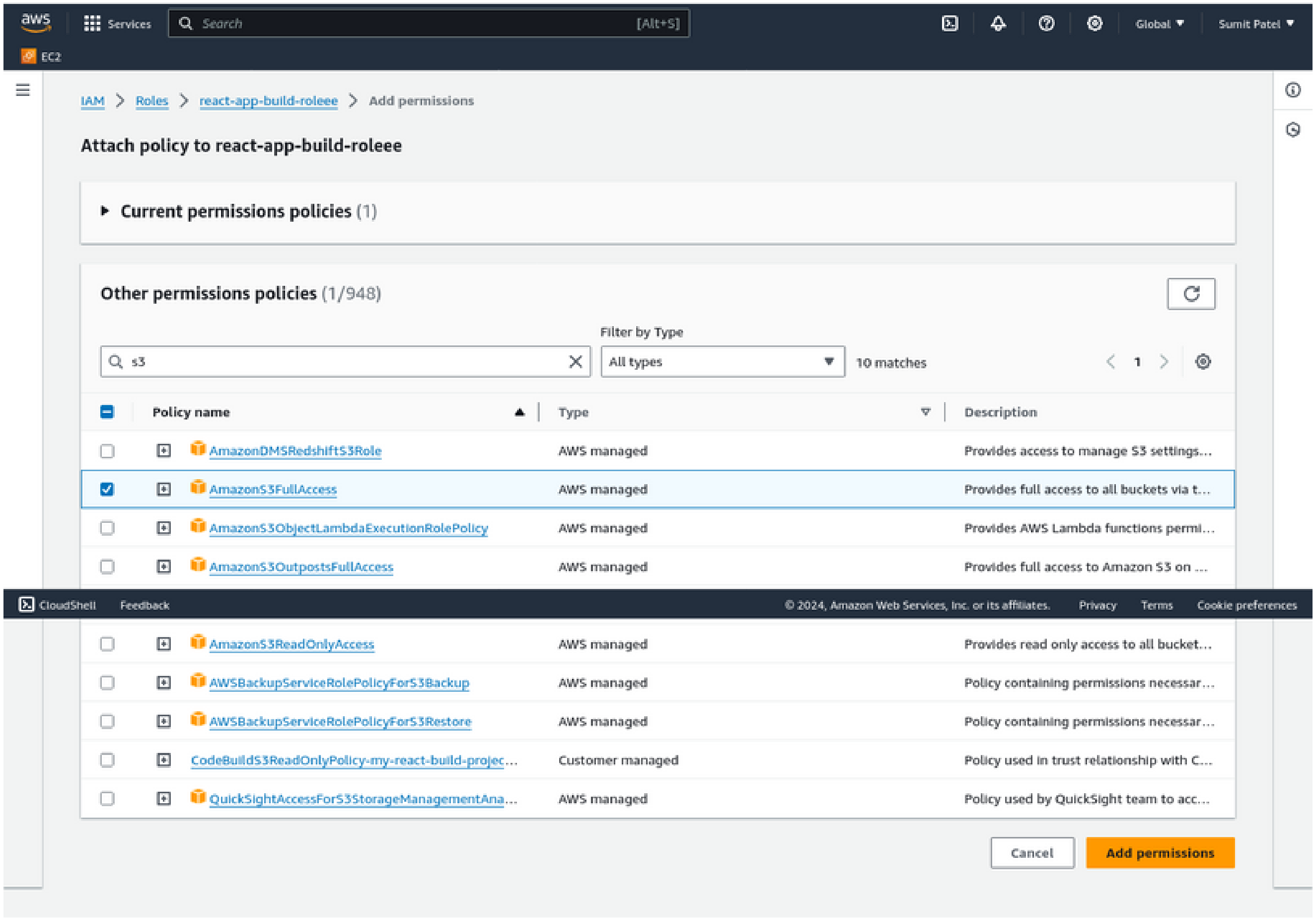
Create a notification rule for this project
Developer Tools > CodeBuild > Build projects > my-react-build-project
Actions ▼
Create trigger
Edit
Clone
Debug build
Start build with overrides
Start build

**Project created**
You have successfully created the following project: my-react-build-project

my-react-build-project

**Configuration**

| Source provider | Primary repository | Artifacts upload location | Service role |
|---|---|---|---|
| Amazon S3 | my-react-app-source-bucket/hello-react.zip | my-react-app-source-bucket | arn:aws:iam::283527553883:role/service-role/react-app-build-roleee |



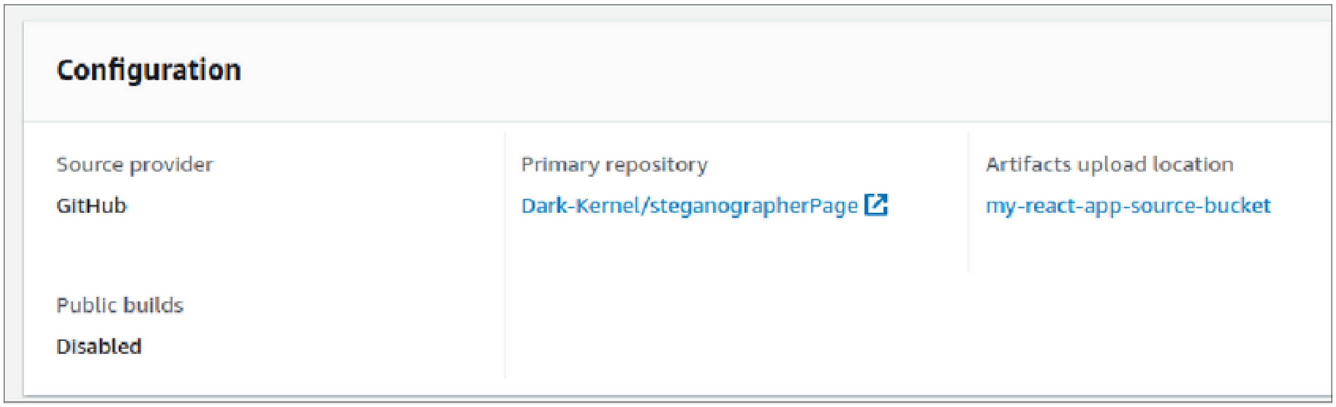
Attach policy to react-app-build-roleee

▸ Current permissions policies (1)

Other permissions policies (1/948)

s3

| | Policy name | Type | Description |
|---|---|---|---|
| □ | AmazonDMSRedshiftS3Role | AWS managed | Provides access to manage S3 settings... |
| ☑ | AmazonS3FullAccess | AWS managed | Provides full access to all buckets via t... |
| □ | AmazonS3ObjectLambdaExecutionRolePolicy | AWS managed | Provides AWS Lambda functions permi... |
| □ | AmazonS3OutpostsFullAccess | AWS managed | Provides full access to Amazon S3 on ... |
| □ | AmazonS3ReadOnlyAccess | AWS managed | Provides read only access to all bucket... |
| □ | AWSBackupServiceRolePolicyForS3Backup | AWS managed | Policy containing permissions necessar... |
| □ | AWSBackupServiceRolePolicyForS3Restore | AWS managed | Policy containing permissions necessar... |
| □ | CodeBuildS3ReadOnlyPolicy-my-react-build-projec... | Customer managed | Policy used in trust relationship with C... |
| □ | QuickSightAccessForS3StorageManagementAna... | AWS managed | Policy used by QuickSight team to acc... |



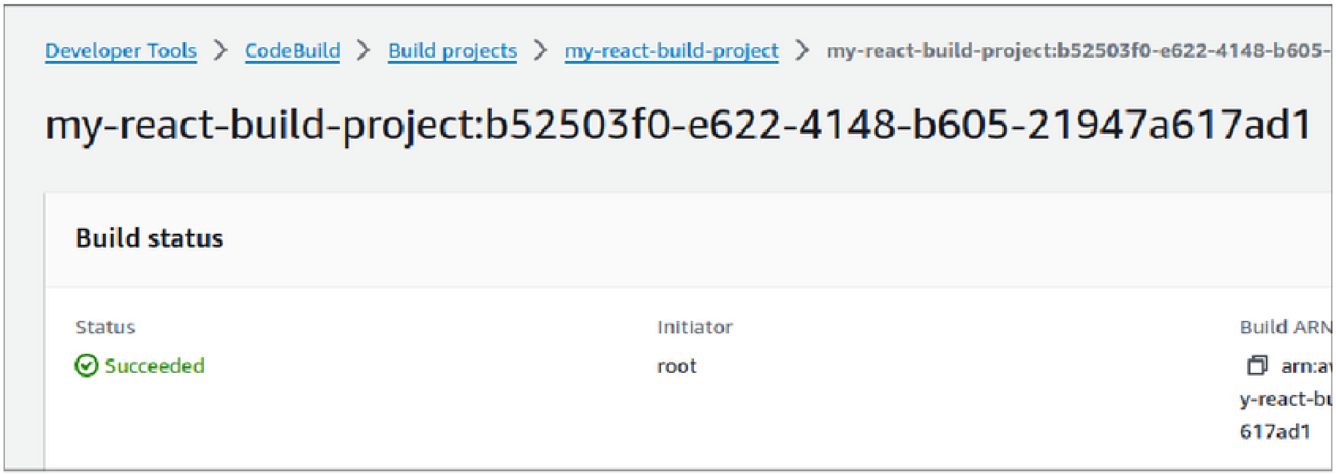
### Configuration

Source provider
GitHub

Primary repository
Dark-Kernel/steganographerPage

Artifacts upload location
my-react-app-source-bucket

Public builds
Disabled



Developer Tools > CodeBuild > Build projects > my-react-build-project > my-react-build-project:b52503f0-e622-4148-b605-

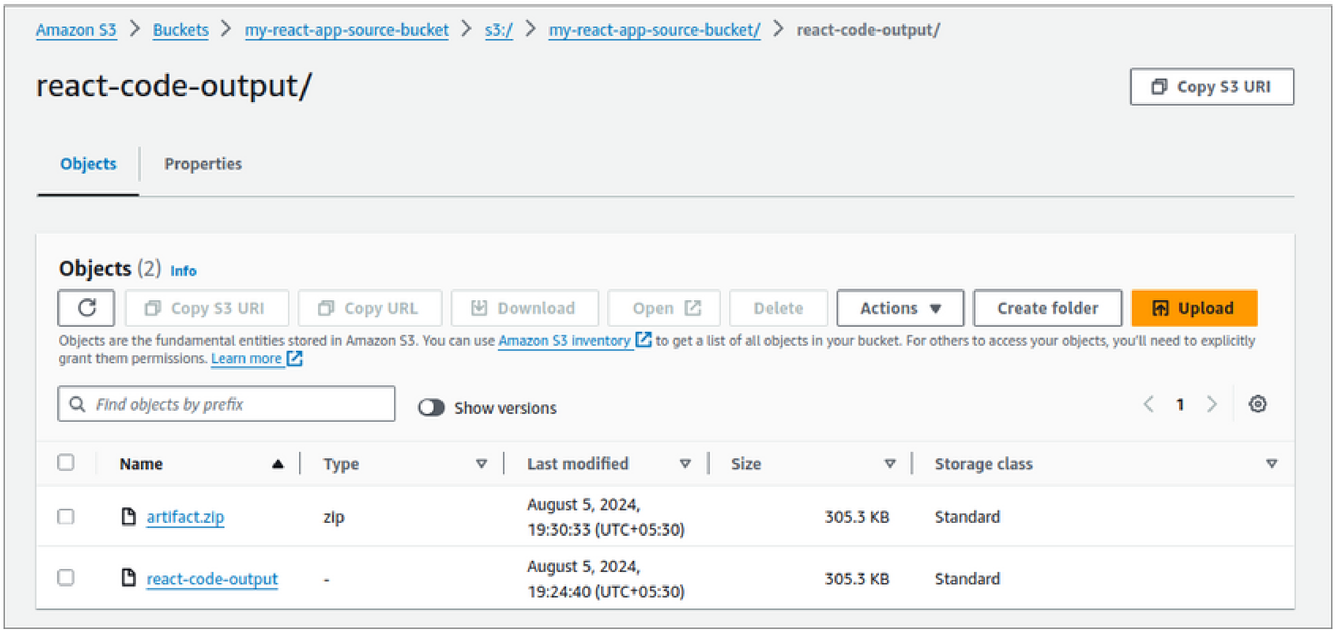
## my-react-build-project:b52503f0-e622-4148-b605-21947a617ad1

### Build status

Status
Succeeded

Initiator
root

Build ARN
arn:a
y-react-bu
617ad1

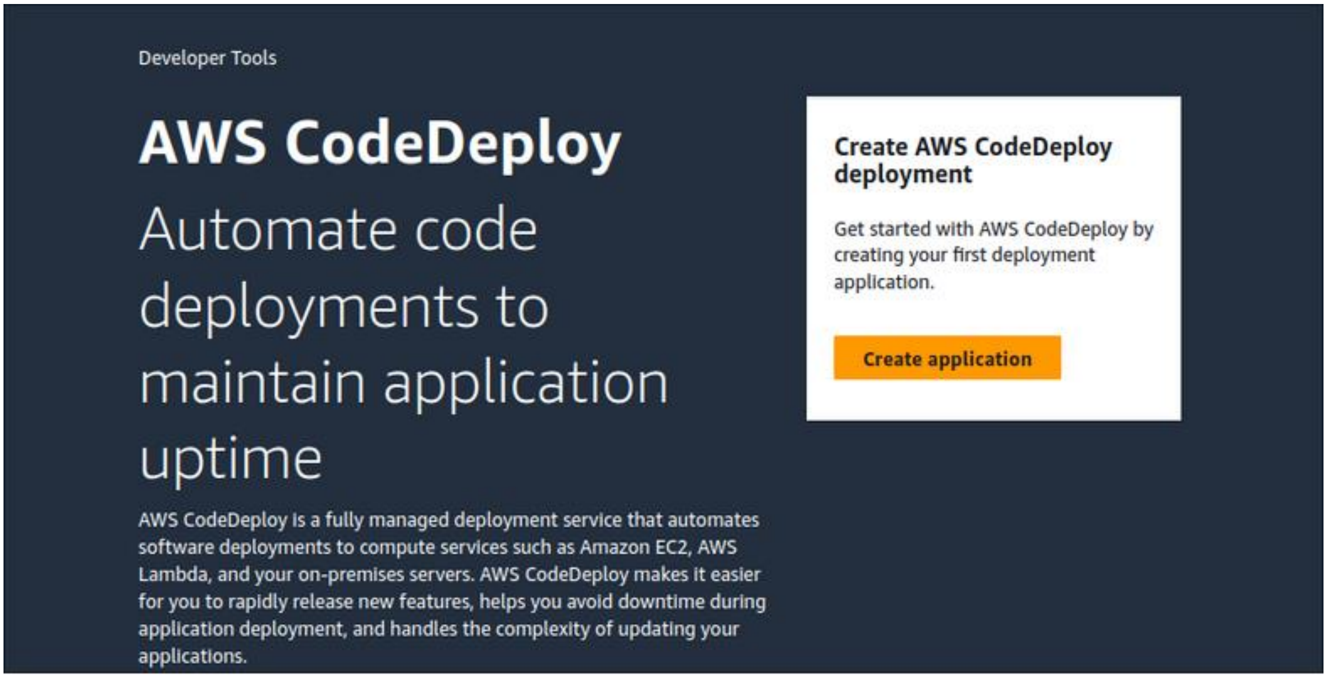


## react-code-output/

### Objects (2) Info

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

| □ | Name | Type | Last modified | Size | Storage class |
|---|---|---|---|---|---|
| □ | artifact.zip | zip | August 5, 2024, 19:30:33 (UTC+05:30) | 305.3 KB | Standard |
| □ | react-code-output | - | August 5, 2024, 19:24:40 (UTC+05:30) | 305.3 KB | Standard |



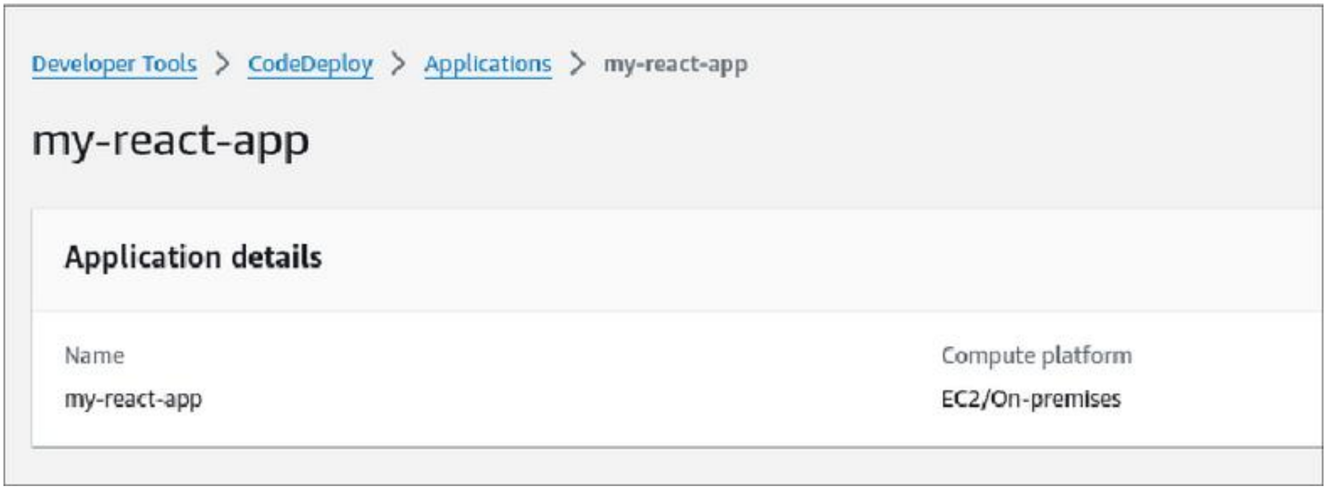
AWS CodeDeploy

Automate code deployments to maintain application uptime

AWS CodeDeploy is a fully managed deployment service that automates software deployments to compute services such as Amazon EC2, AWS Lambda, and your on-premises servers. AWS CodeDeploy makes it easier for you to rapidly release new features, helps you avoid downtime during application deployment, and handles the complexity of updating your applications.

Create AWS CodeDeploy deployment

Get started with AWS CodeDeploy by creating your first deployment application.

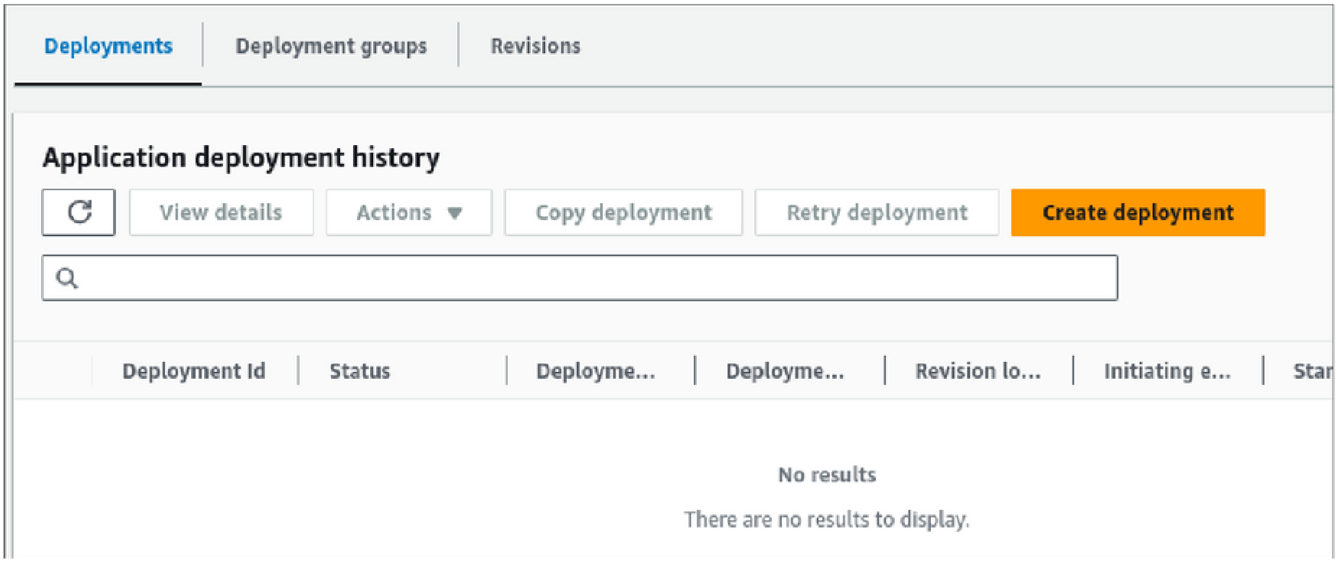


my-react-app

Application details

Name: my-react-app

Compute platform: EC2/On-premises

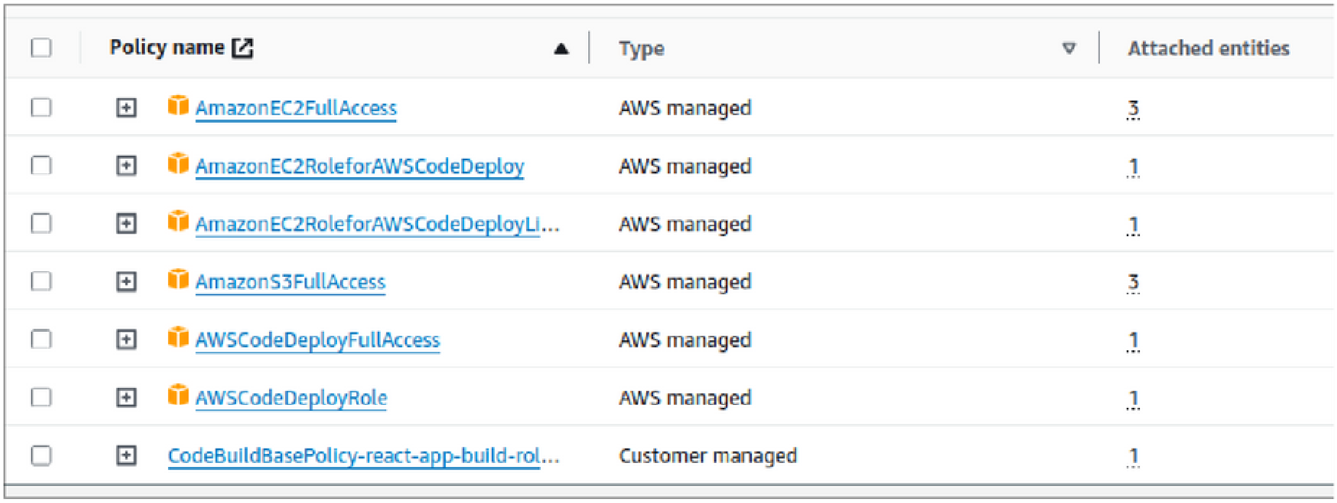


Application deployment history

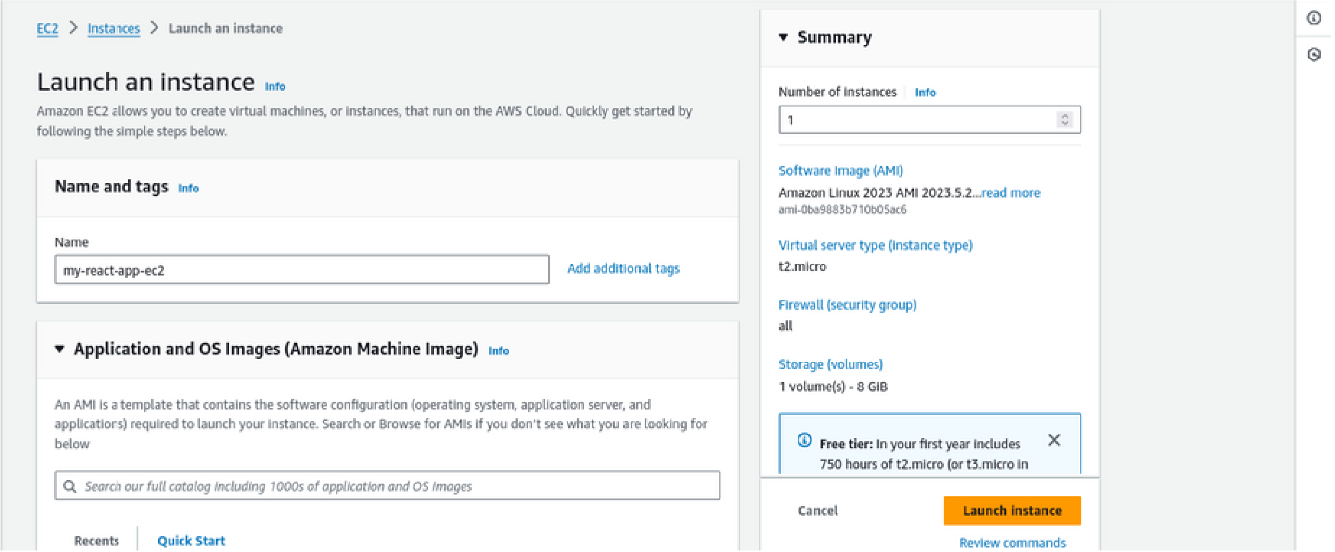
| Deployment Id | Status | Deployme... | Deployme... | Revision lo... | Initiating e... | Star |
|---|---|---|---|---|---|---|

No results

There are no results to display.



| □ | Policy name | Type | Attached entities |
|---|---|---|---|
| □ | AmazonEC2FullAccess | AWS managed | 3 |
| □ | AmazonEC2RoleforAWSCodeDeploy | AWS managed | 1 |
| □ | AmazonEC2RoleforAWSCodeDeployLi... | AWS managed | 1 |
| □ | AmazonS3FullAccess | AWS managed | 3 |
| □ | AWSCodeDeployFullAccess | AWS managed | 1 |
| □ | AWSCodeDeployRole | AWS managed | 1 |
| □ | CodeBuildBasePolicy-react-app-build-rol... | Customer managed | 1 |



Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: my-react-app-ec2

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images
Recents
Quick Start

Summary

Number of instances Info: 1

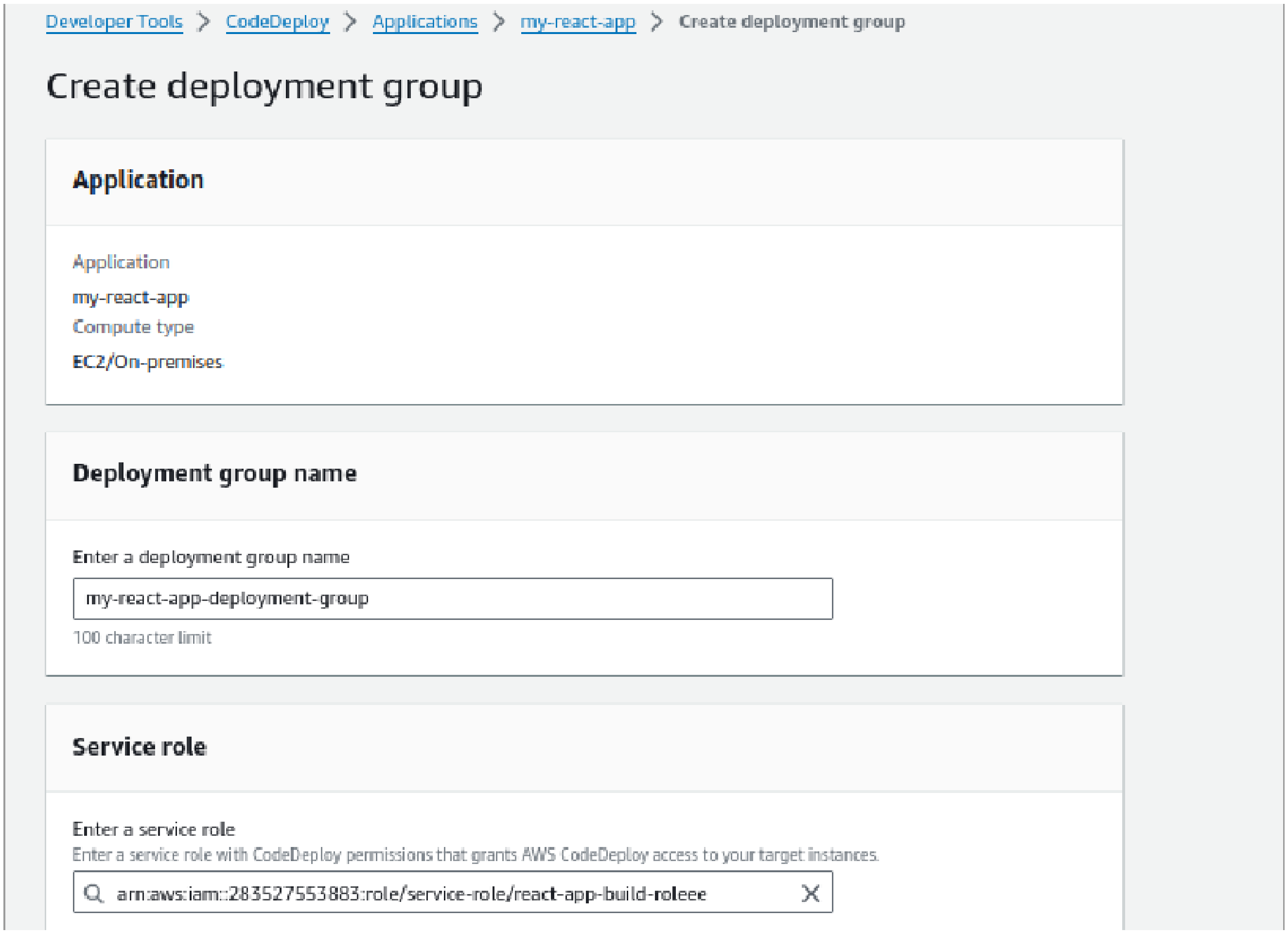
Software Image (AMI)
Amazon Linux 2023 AMI 2023.5.2...read more
ami-0ba9883b710b05ac6

Virtual server type (instance type)
t2.micro

Firewall (security group)
all

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in



Developer Tools > CodeDeploy > Applications > my-react-app > Create deployment group

Create deployment group

**Application**

Application

my-react-app

Compute type

EC2/On-premises

**Deployment group name**

Enter a deployment group name

my-react-app-deployment-group

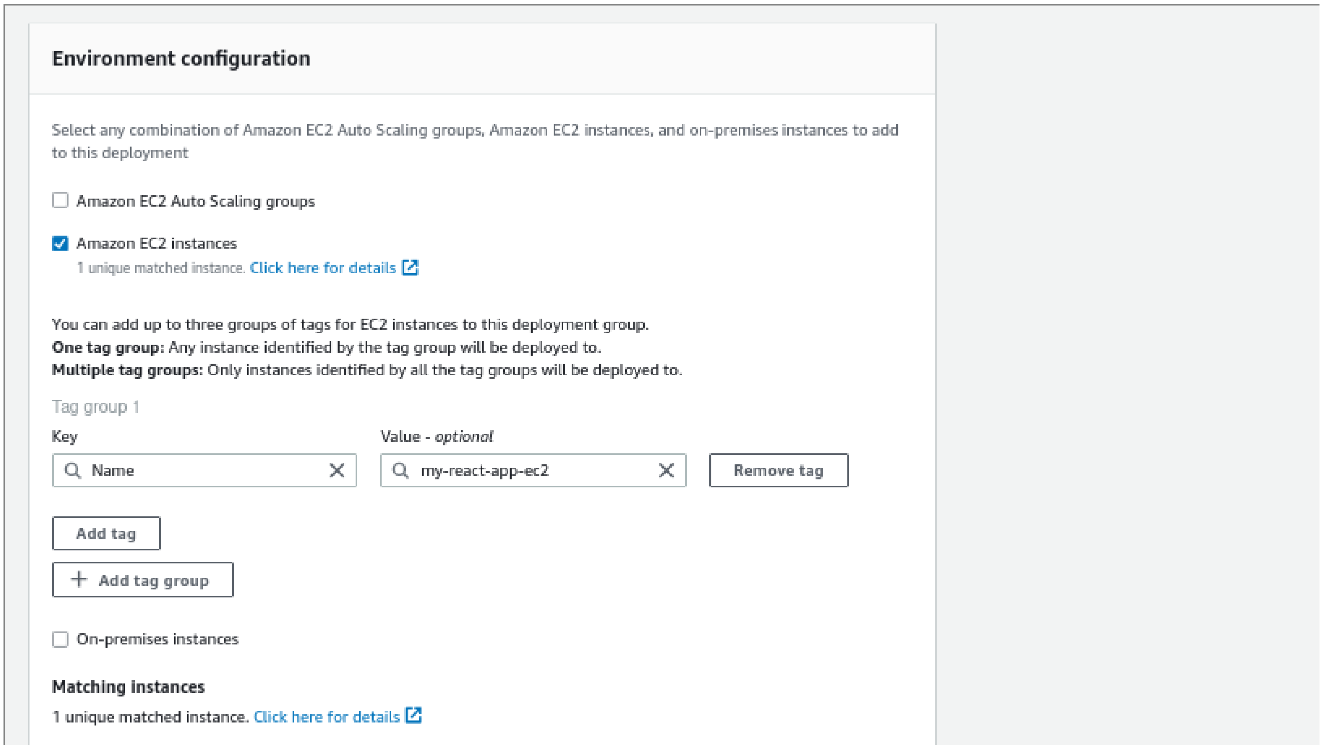
100 character limit

**Service role**

Enter a service role

Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.

arn:aws:iam::283527553883:role/service-role/react-app-build-roleee



**Environment configuration**

Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances, and on-premises instances to add to this deployment

☐ Amazon EC2 Auto Scaling groups

☑ Amazon EC2 instances

1 unique matched instance. Click here for details

You can add up to three groups of tags for EC2 instances to this deployment group.
**One tag group:** Any instance identified by the tag group will be deployed to.
**Multiple tag groups:** Only instances identified by all the tag groups will be deployed to.

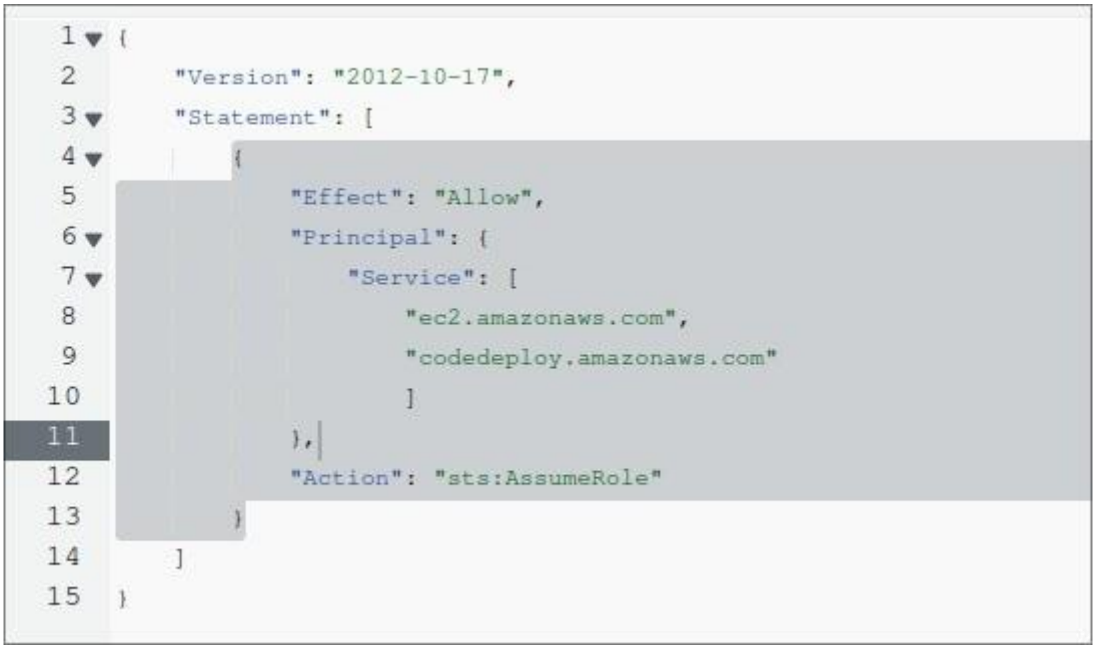
Tag group 1

| Key | Value - *optional* | |
|---|---|---|
| Name | my-react-app-ec2 | Remove tag |

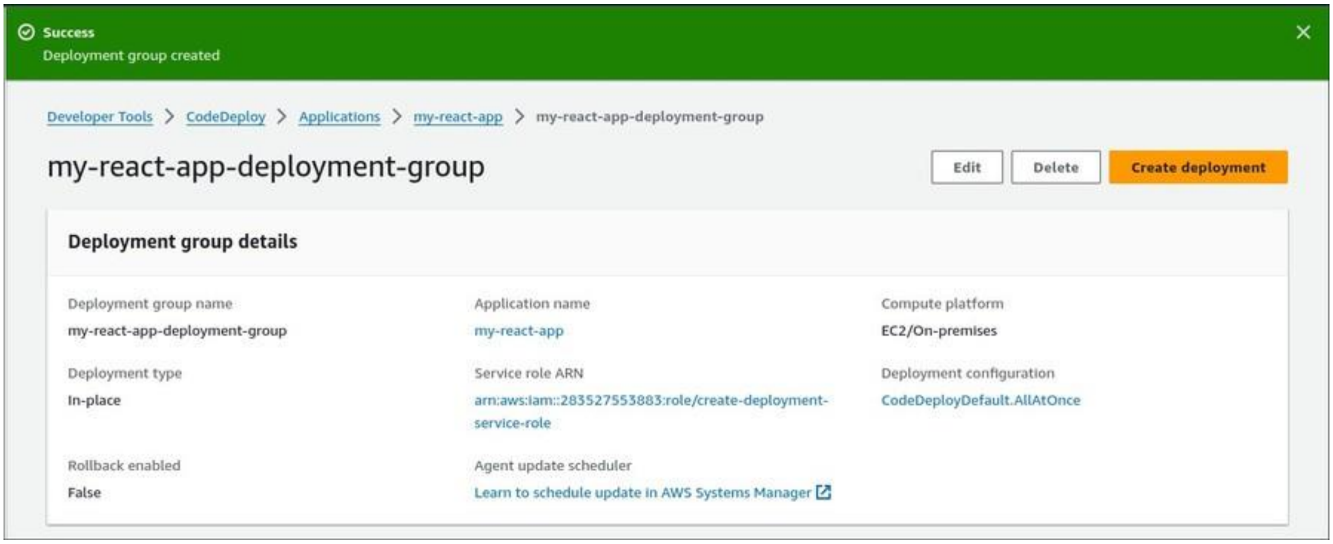
☐ On-premises instances

**Matching instances**

1 unique matched instance. Click here for details



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "ec2.amazonaws.com",
                    "codedeploy.amazonaws.com"
                    ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```



Success

Deployment group created

Developer Tools > CodeDeploy > Applications > my-react-app > my-react-app-deployment-group

my-react-app-deployment-group

Deployment group details

Deployment group name
my-react-app-deployment-group

Application name
my-react-app

Compute platform
EC2/On-premises

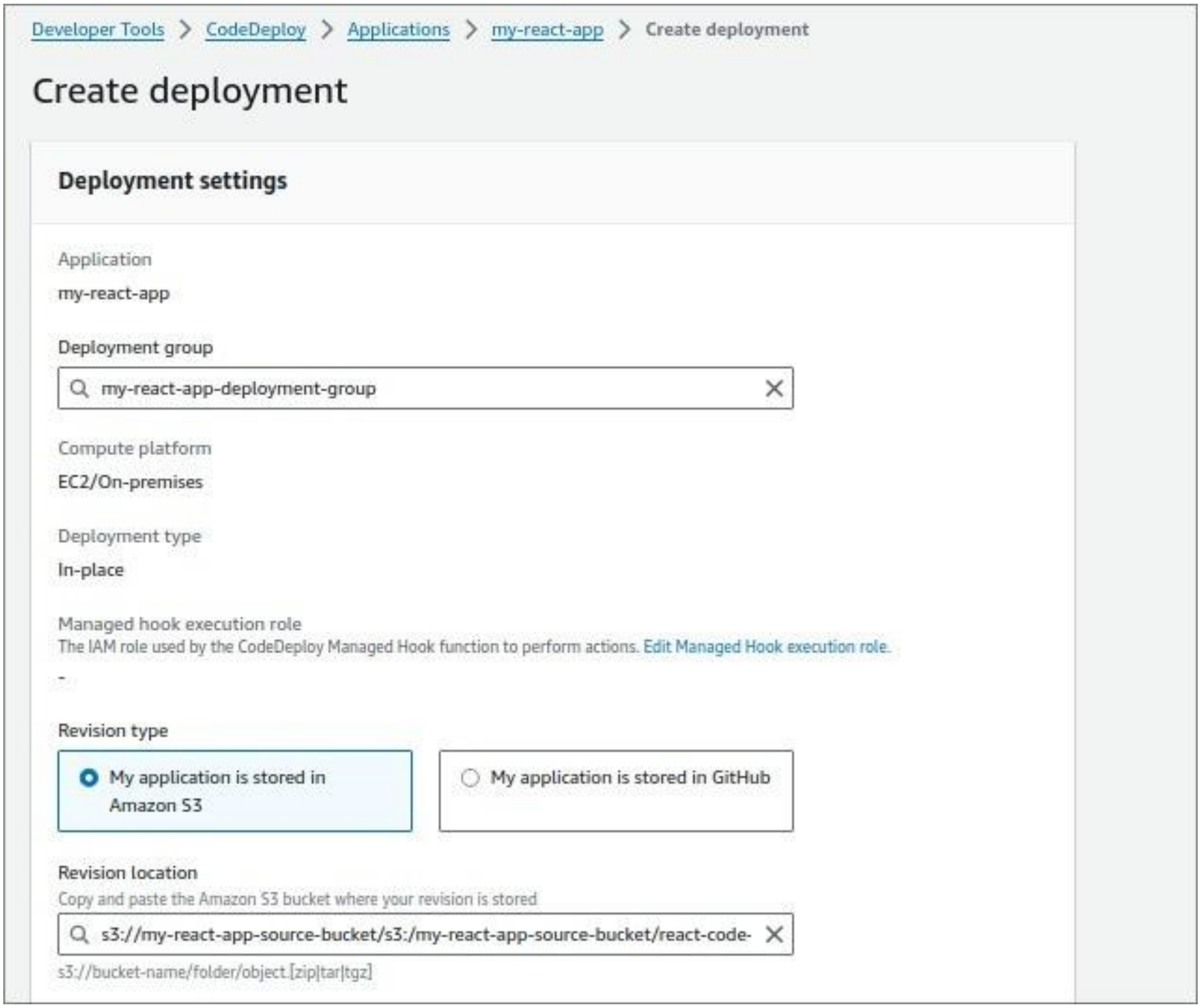
Deployment type
In-place

Service role ARN
arn:aws:iam::283527553883:role/create-deployment-service-role

Deployment configuration
CodeDeployDefault.AllAtOnce

Rollback enabled
False

Agent update scheduler
Learn to schedule update in AWS Systems Manager


Developer Tools > CodeDeploy > Applications > my-react-app > Create deployment

Create deployment

Deployment settings

Application
my-react-app

Deployment group
my-react-app-deployment-group

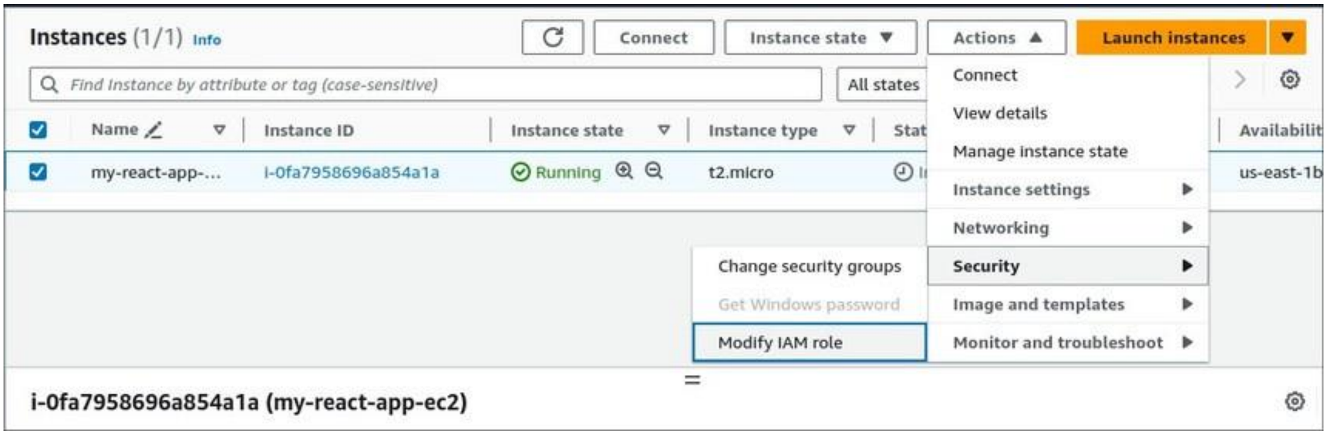
Compute platform
EC2/On-premises

Deployment type
In-place

Managed hook execution role
The IAM role used by the CodeDeploy Managed Hook function to perform actions. Edit Managed Hook execution role.
-

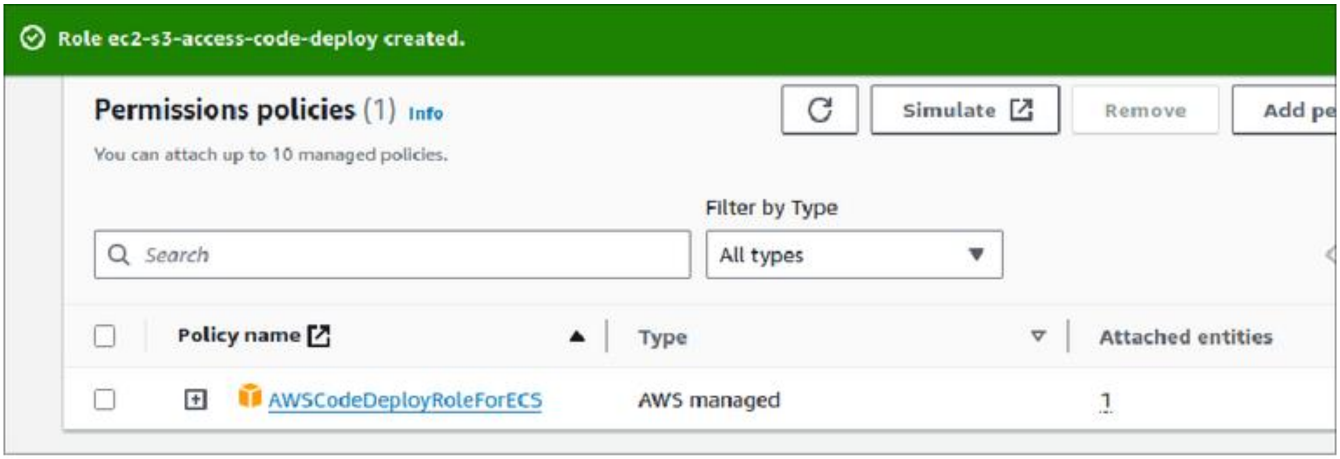
Revision type
● My application is stored in Amazon S3
○ My application is stored in GitHub

Revision location
Copy and paste the Amazon S3 bucket where your revision is stored
s3://my-react-app-source-bucket/s3:/my-react-app-source-bucket/react-code-
s3://bucket-name/folder/object.[zip|tar|tgz]



| ☑ | Name | Instance ID | Instance state | Instance type | Stat | Availabilit |
|---|---|---|---|---|---|---|
| ☑ | my-react-app-... | i-0fa7958696a854a1a | Running | t2.micro | | us-east-1b |

i-0fa7958696a854a1a (my-react-app-ec2)

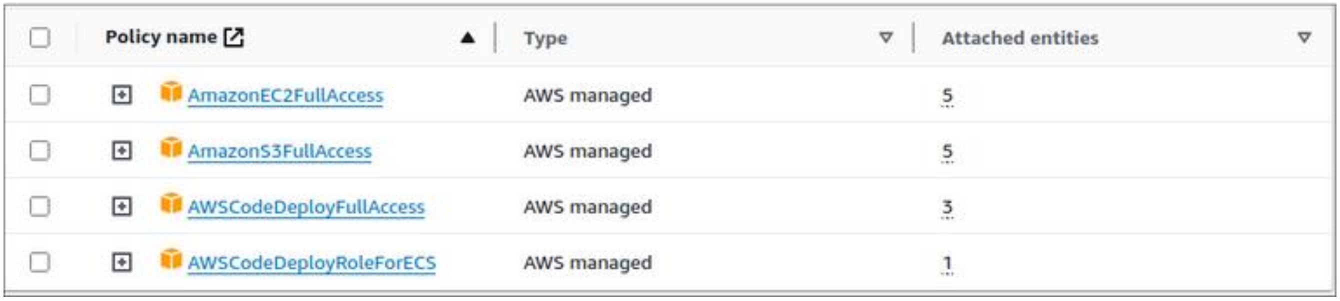


Role ec2-s3-access-code-deploy created.

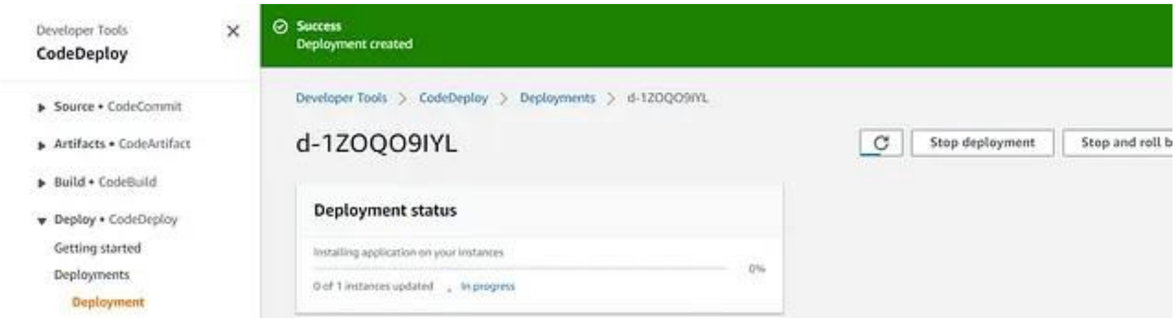
## Permissions policies (1)

You can attach up to 10 managed policies.

| □ | Policy name | Type | Attached entities |
|---|---|---|---|
| □ | AWSCodeDeployRoleForECS | AWS managed | 1 |



| □ | Policy name | Type | Attached entities |
|---|---|---|---|
| □ | AmazonEC2FullAccess | AWS managed | 5 |
| □ | AmazonS3FullAccess | AWS managed | 5 |
| □ | AWSCodeDeployFullAccess | AWS managed | 3 |
| □ | AWSCodeDeployRoleForECS | AWS managed | 1 |



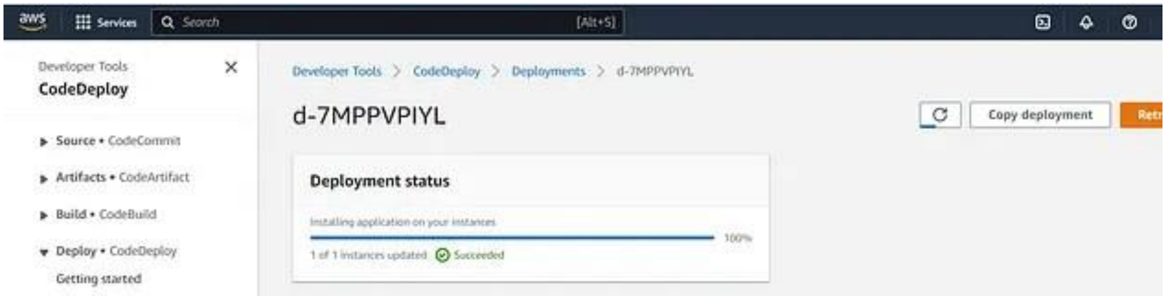
Success
Deployment created

d-1ZOQO9IYL

Deployment status

Installing application on your instances 0%

0 of 1 instances updated In progress

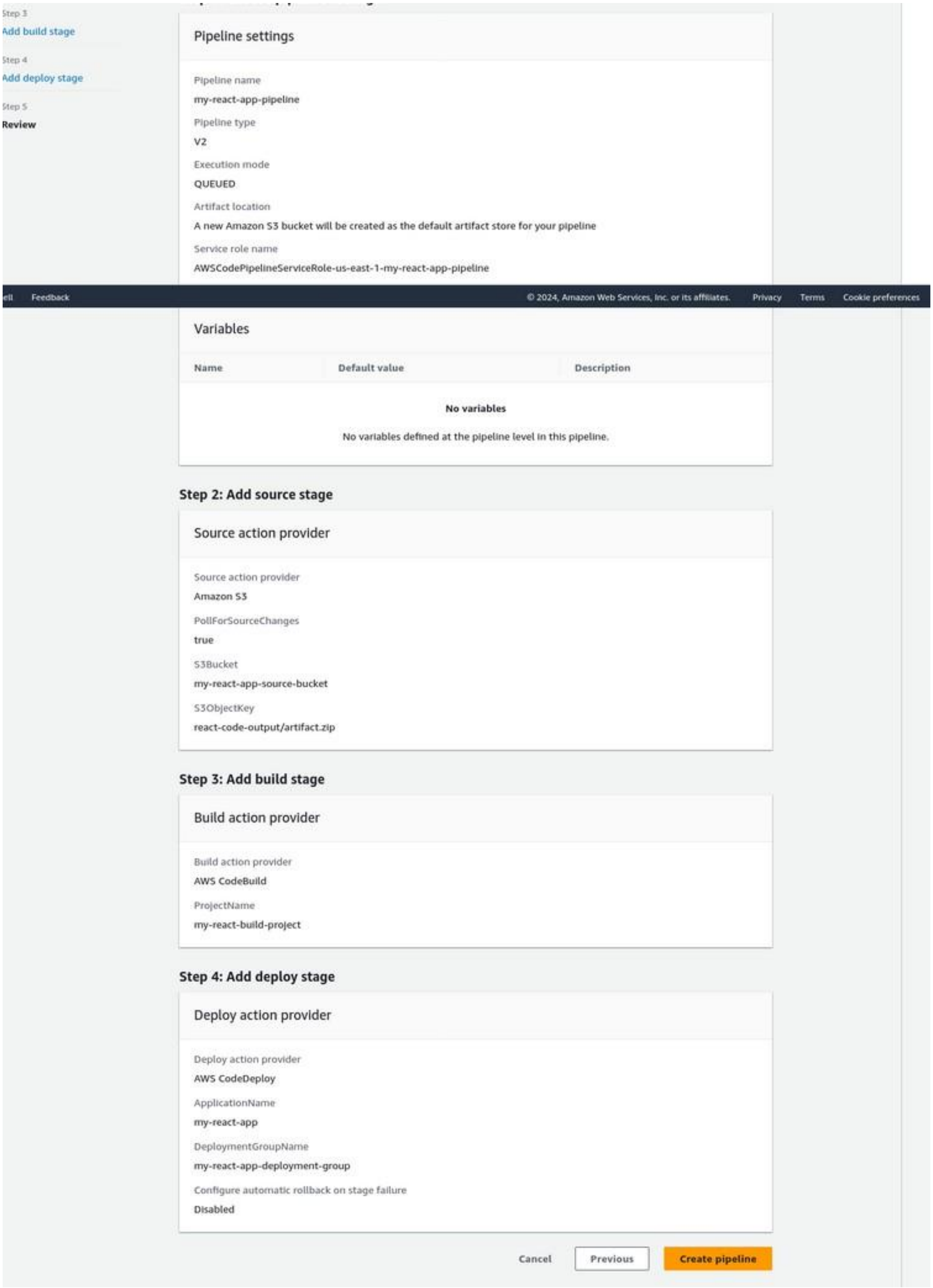


d-7MPPVPIYL

Deployment status

Installing application on your instances 100%

1 of 1 instances updated Succeeded



Step 3
Add build stage
Step 4
Add deploy stage
Step 5
Review

Pipeline settings

Pipeline name
my-react-app-pipeline

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-us-east-1-my-react-app-pipeline

ell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

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
Amazon S3

PollForSourceChanges
true

S3Bucket
my-react-app-source-bucket

S3ObjectKey
react-code-output/artifact.zip

Step 3: Add build stage

Build action provider

Build action provider
AWS CodeBuild

ProjectName
my-react-build-project

Step 4: Add deploy stage

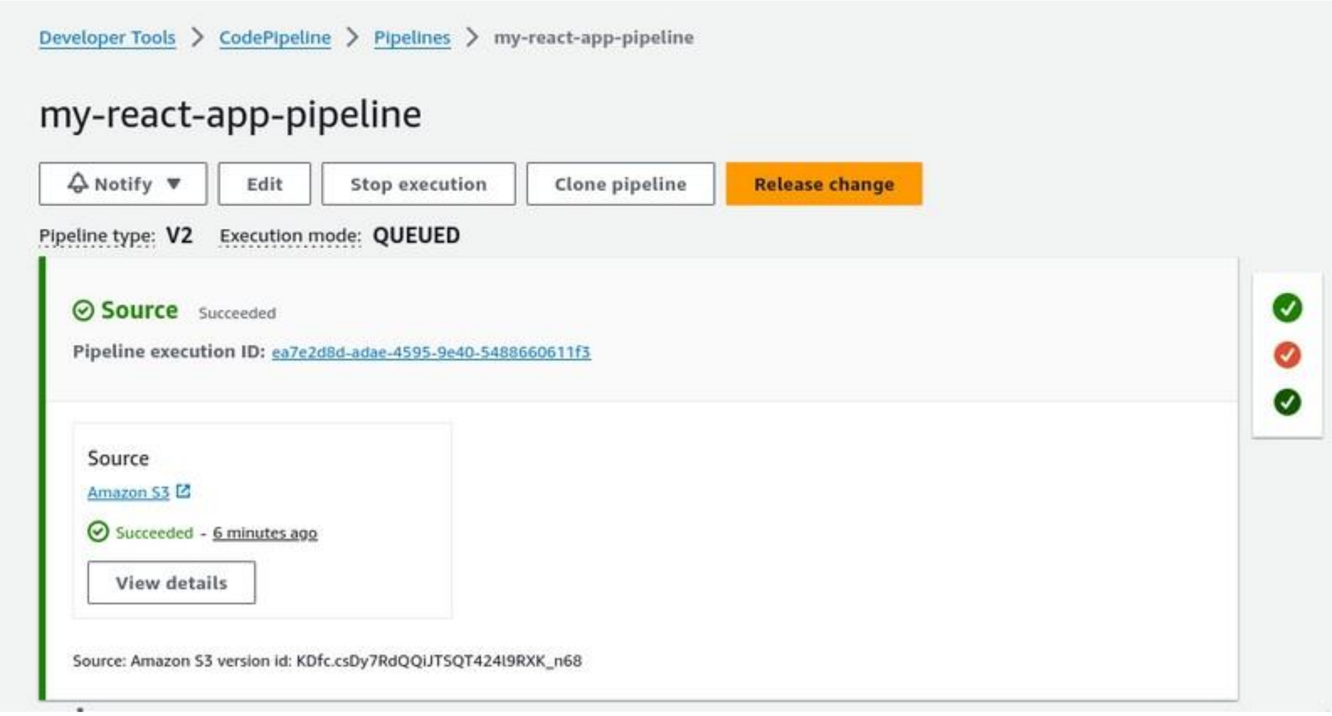
Deploy action provider

Deploy action provider
AWS CodeDeploy

ApplicationName
my-react-app

DeploymentGroupName
my-react-app-deployment-group

Configure automatic rollback on stage failure
Disabled



Developer Tools > CodePipeline > Pipelines > my-react-app-pipeline

my-react-app-pipeline

Notify ▼
Edit
Stop execution
Clone pipeline
Release change

Pipeline type: V2 Execution mode: QUEUED

✓ Source Succeeded

Pipeline execution ID: ea7e2d8d-adae-4595-9e40-5488660611f3

Source

Amazon S3

✓ Succeeded - 6 minutes ago

Source: Amazon S3 version id: KDfc.csDy7RdQQiJTSQT424l9RXK_n68

Conclusion: Thus, we have successfully Build our Application using AWS Code Build and Deploy on S3 / SEBS using AWS Code Pipeline, deployed Sample Application on EC2 instance using AWS Code Deploy.