# AdvDevops Case Study 1: Cloud Deployment with Automation

● **Concepts Used:** EC2, S3, CodeBuild, CodePipeline, CodeDeploy.
● **Problem Statement:** "Build a simple web application using AWS CodeBuild and deploy it to an S3 bucket. Then, automate the deployment process using AWS CodePipeline, ensuring the application is deployed on an EC2 instance. Use a sample index.html page for demonstration."
● **Tasks:**

○ Set up AWS CodeBuild for the web app.
○ Create a pipeline that deploys to an S3 bucket.
○ Use AWS CodeDeploy to push updates to an EC2 instance.

## 1. Introduction

**Case study overview:**

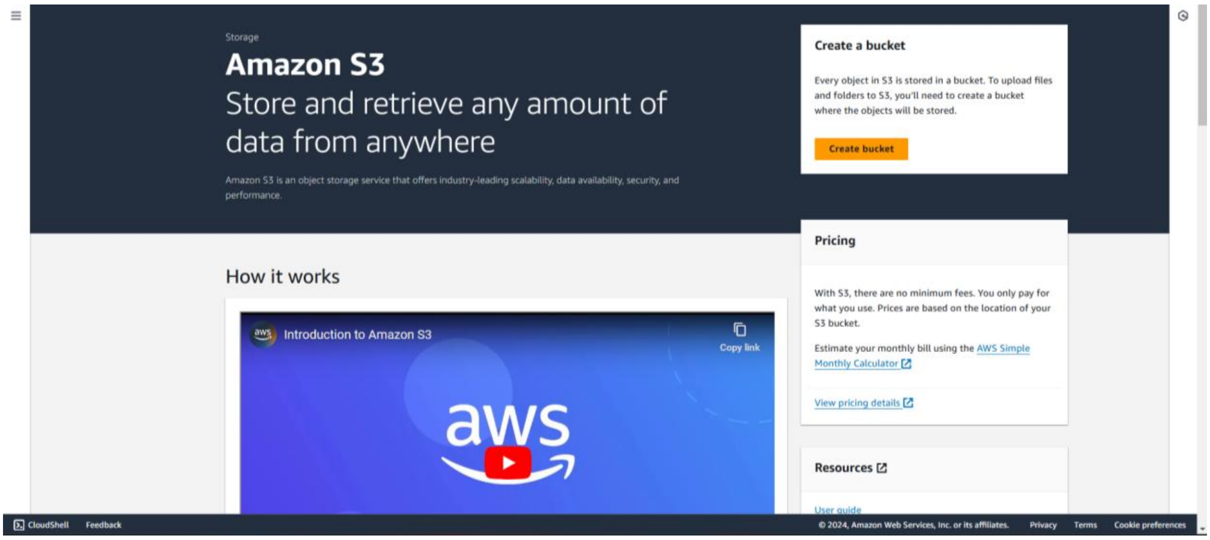
This case study involves storing a web application in an S3 bucket, building it through AWS CodeBuild, automation using AWS CodePipeline to deploy the web app to an EC2 instance via CodeDeploy.

**Key features and Applications:**

Automated Deployment: AWS CodePipeline automates code deployment to S3 and EC2.
CI/CD Integration: Continuous integration and deployment with CodeBuild and CodeDeploy streamlines updates.
Scalability: EC2 provides scalable compute power and S3 offers scalable storage.
Version Control: CodeDeploy enables easy rollback to previous versions in case of failure
Web App Hosting: Ideal for hosting dynamic and static web apps.

## Step 1: Create an S3 Bucket

In the AWS Management Console, navigate to **S3** by searching for it in the search bar at the top.



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.

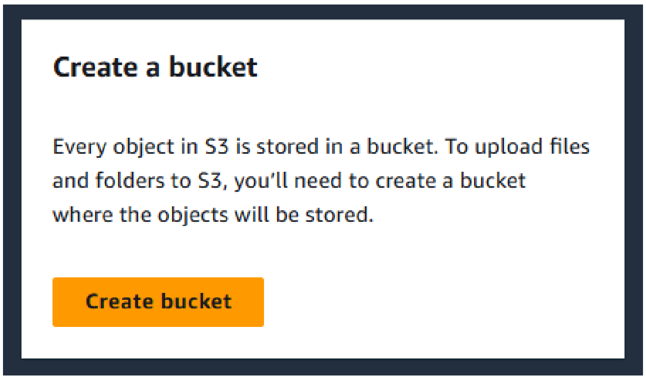
How it works
Introduction to Amazon S3
Copy link
aws
Create a bucket

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.

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the AWS Simple Monthly Calculator

Click Create bucket.

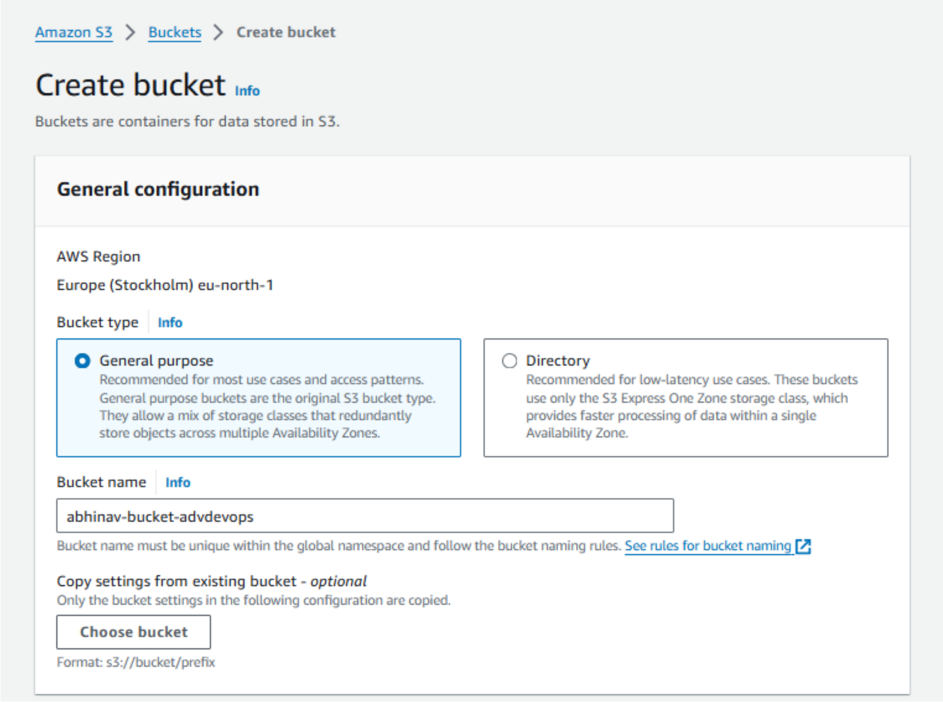


Create a bucket

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.

Give your bucket a unique name (e.g., my-app-bucket). If you want the content of your web app (like the index.html file) to be publicly accessible, uncheck Block all public access.

For testing/deployment, you can allow public access



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

Europe (Stockholm) eu-north-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
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.

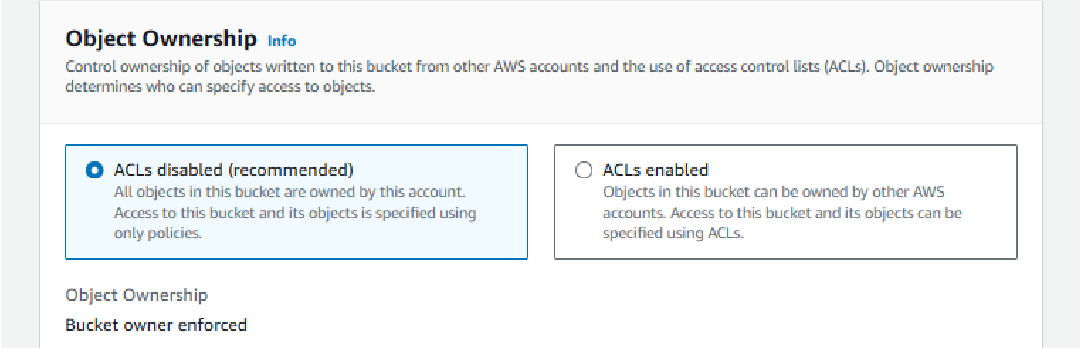
Bucket name Info

abhinav-bucket-advdevops

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

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix



Object Ownership Info

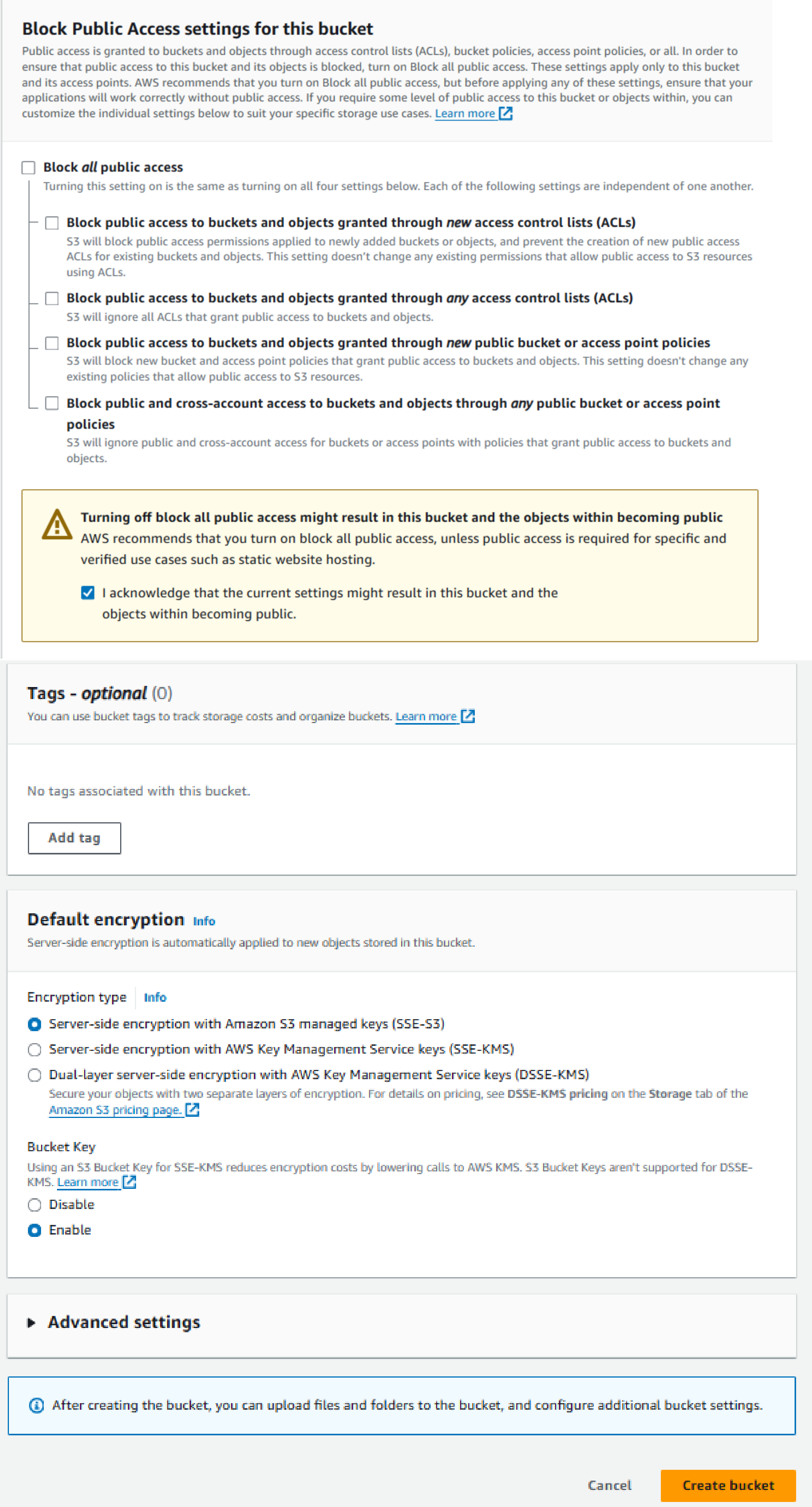
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced



### 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 ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- □ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  S3 will ignore all ACLs that grant public access to buckets and objects.
- □ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- □ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

⚠ **Turning off block all public access might result in this bucket and the objects within becoming public**
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

### Tags - *optional* (0)

You can use bucket tags to track storage costs and organize buckets. Learn more

No tags associated with this bucket.

### Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type Info

● Server-side encryption with Amazon S3 managed keys (SSE-S3)
○ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
○ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

○ Disable
● Enable

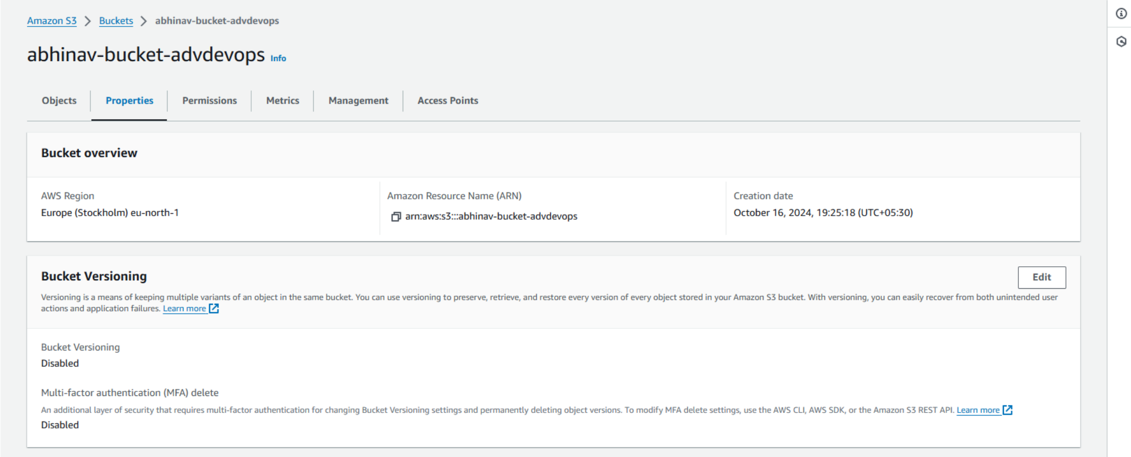
ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Enable Versioning on Your S3 Bucket

To do this

Select Your Bucket: Click on your bucket named (abhinav-bucket-advdevops.)

Go to Properties: In the bucket settings, click on the "Properties" tab.



abhinav-bucket-advdevops

Bucket overview

AWS Region
Europe (Stockholm) eu-north-1

Amazon Resource Name (ARN)
arn:aws:s3:::abhinav-bucket-advdevops

Creation date
October 16, 2024, 19:25:18 (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.

Bucket Versioning
Disabled

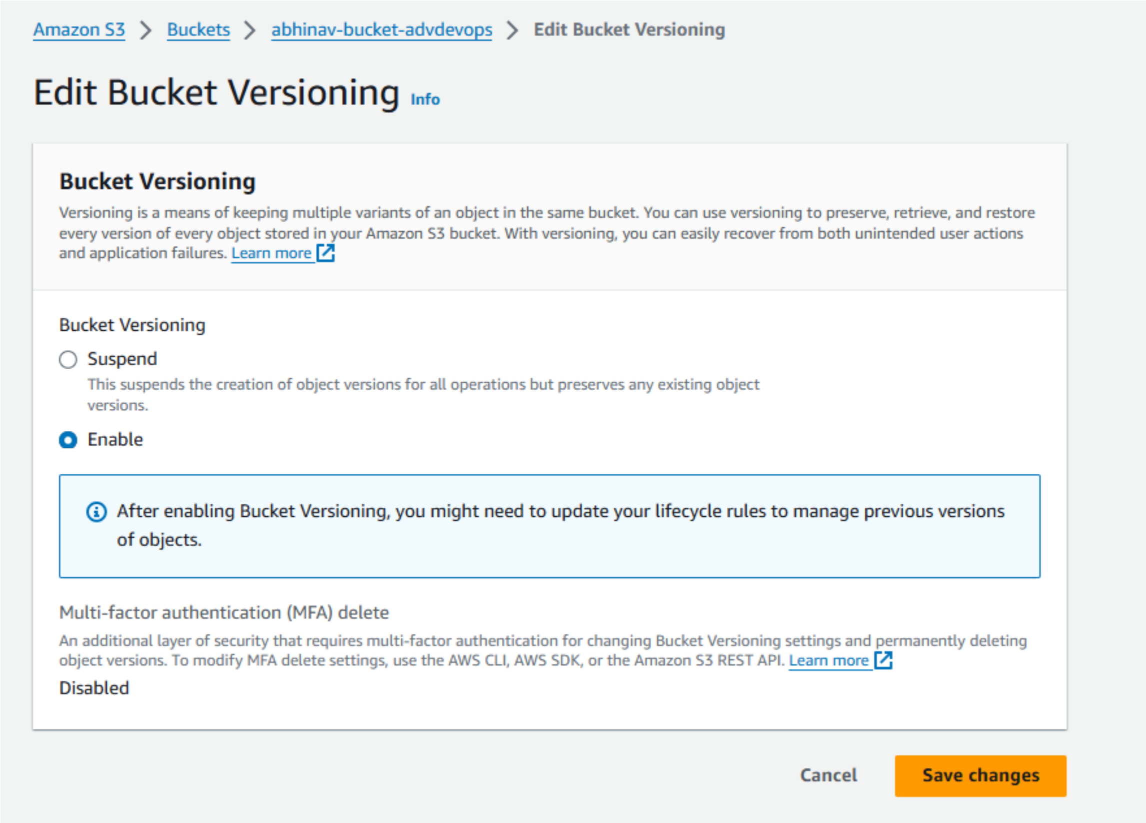
Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API.
Disabled

Scroll down to the "Bucket Versioning" section.

Click on "Edit".

Select "Enable" for versioning.

Click "Save changes".


Amazon S3 > Buckets > abhinav-bucket-advdevops > Edit Bucket Versioning
Info
Learn more
Learn more
Cancel
Save changes

Edit Bucket Versioning

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.

Bucket Versioning

○ Suspend
This suspends the creation of object versions for all operations but preserves any existing object versions.

● Enable

After enabling Bucket Versioning, you might need to update your lifecycle rules to manage previous versions of objects.

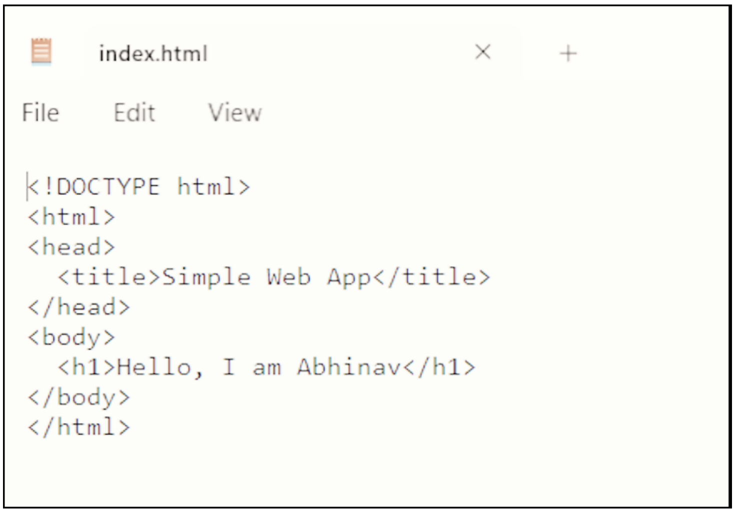
Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API.
Disabled

## Step 2: Create a Simple Web Application (index.html)

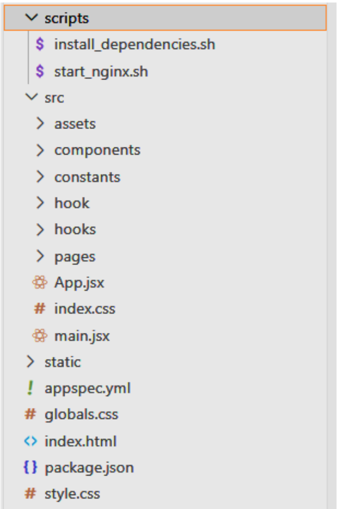
On your local machine, create a new file called index.html.

Add the following simple HTML code into the file: index.html

This is necessary for a basic web app. I implemented my Project Integration (shop.co)



```html
<!DOCTYPE html>
<html>
<head>
  <title>Simple Web App</title>
</head>
<body>
  <h1>Hello, I am Abhinav</h1>
</body>
</html>
```



```
∨ scripts
   $ install_dependencies.sh
   $ start_nginx.sh
∨ src
   > assets
   > components
   > constants
   > hook
   > hooks
   > pages
     App.jsx
   # index.css
     main.jsx
> static
! appspec.yml
# globals.css
<> index.html
{} package.json
# style.css
```

Create the following

appspec.yml (configuration file listing instructions to execute when the application is deployed)

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /usr/share/nginx/html
    overwrite: true
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: ec2-user
  AfterInstall:
    - location: scripts/start_nginx.sh
      timeout: 300
      runas: ec2-user
```

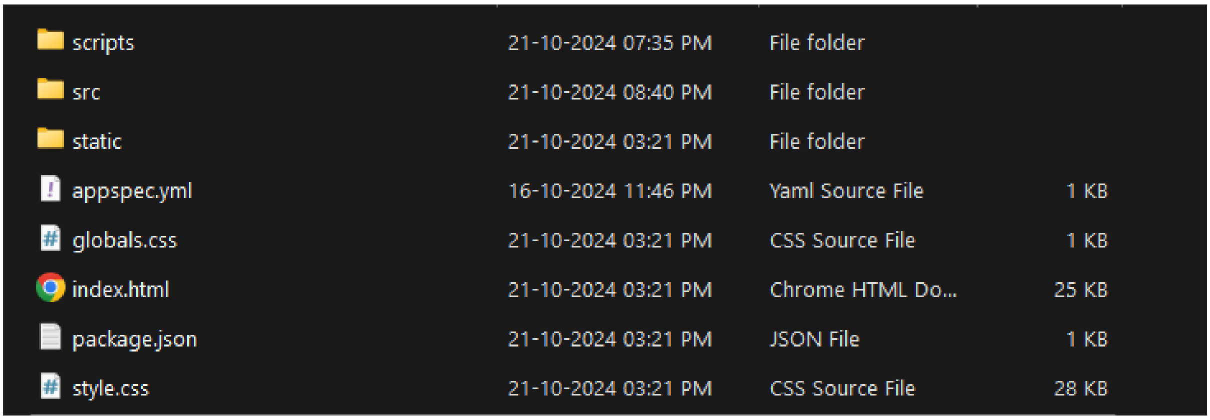
scripts\install_dependencies.sh (update your system & install nginx)

```bash
#!/bin/bash
sudo yum update -y
sudo yum install -y nginx
```

scripts\start_nginx.sh (restart nginx service)

```bash
#!/bin/bash
sudo service nginx restart
```

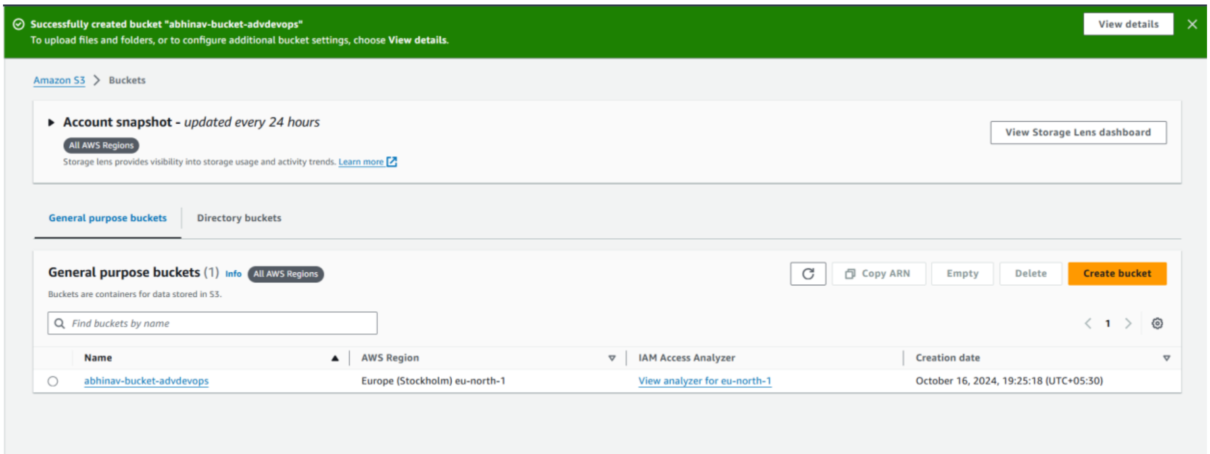
Zip all into webapp.zip (project files for deployment).



| | | | |
|---|---|---|---|
| scripts | 21-10-2024 07:35 PM | File folder | |
| src | 21-10-2024 08:40 PM | File folder | |
| static | 21-10-2024 03:21 PM | File folder | |
| appspec.yml | 16-10-2024 11:46 PM | Yaml Source File | 1 KB |
| globals.css | 21-10-2024 03:21 PM | CSS Source File | 1 KB |
| index.html | 21-10-2024 03:21 PM | Chrome HTML Do... | 25 KB |
| package.json | 21-10-2024 03:21 PM | JSON File | 1 KB |
| style.css | 21-10-2024 03:21 PM | CSS Source File | 28 KB |

## Step 3: **Upload the Web App (webapp.zip) to S3**

Go back to the S3 Console. Click on the bucket you created in Step 1



Successfully created bucket "abhinav-bucket-advdevops"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

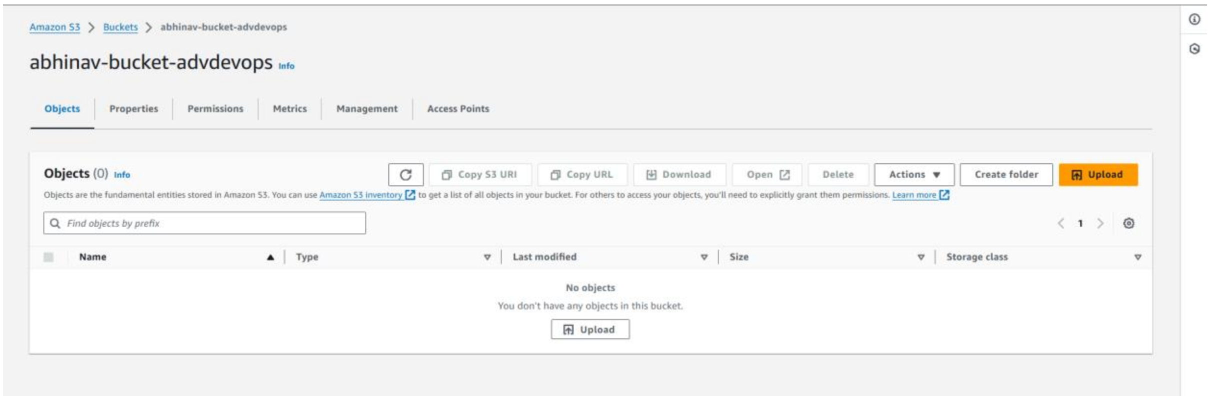
Amazon S3 > Buckets

**Account snapshot** - *updated every 24 hours*
All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more

**General purpose buckets** (1) Info All AWS Regions
Buckets are containers for data stored in S3.

| Name | AWS Region | IAM Access Analyzer | Creation date |
|---|---|---|---|
| abhinav-bucket-advdevops | Europe (Stockholm) eu-north-1 | View analyzer for eu-north-1 | October 16, 2024, 19:25:18 (UTC+05:30) |

In the bucket, click **Upload**.



Amazon S3 > Buckets > abhinav-bucket-advdevops

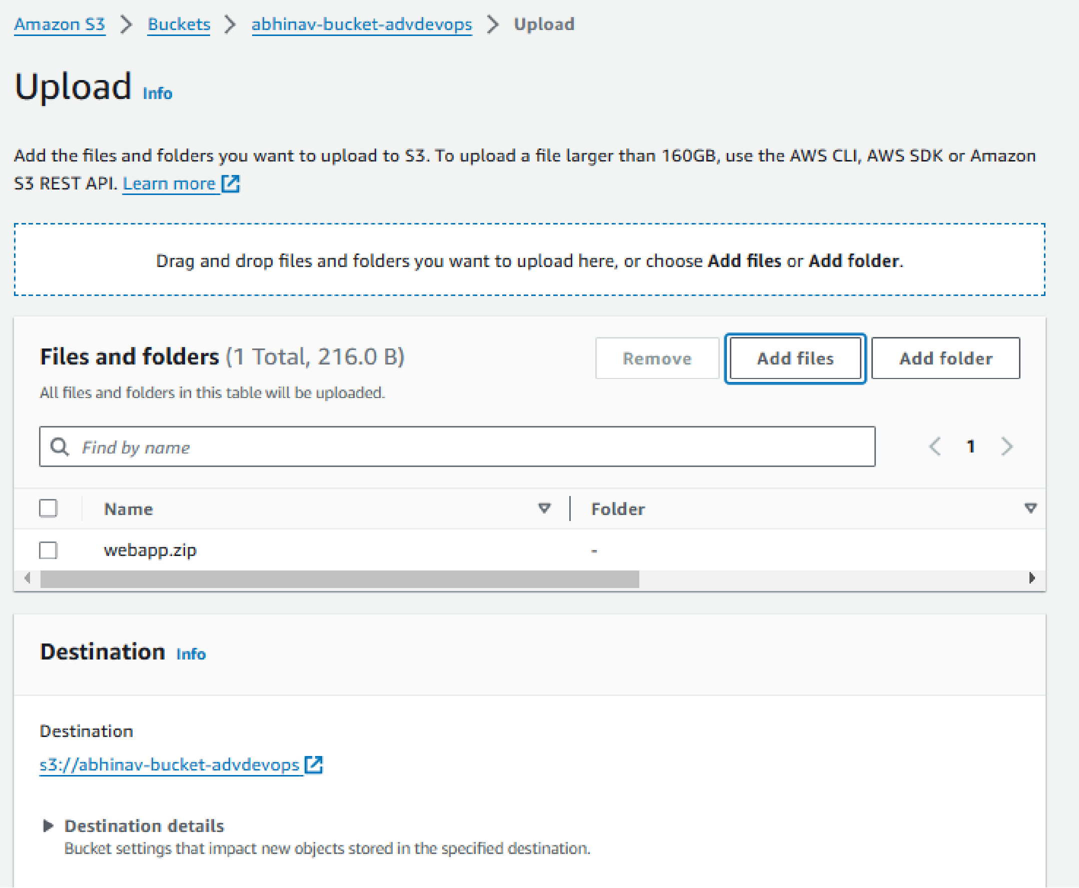
abhinav-bucket-advdevops Info

**Objects** (0) 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 |
|---|---|---|---|---|

No objects
You don't have any objects in this bucket.

Click **Add files**, then select the webapp.zip file from your local machine. Leave all other options as default and click **Upload**.



Amazon S3 > Buckets > abhinav-bucket-advdevops > Upload

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** (1 Total, 216.0 B)

All files and folders in this table will be uploaded.

| Name | Folder |
|---|---|
| webapp.zip | - |

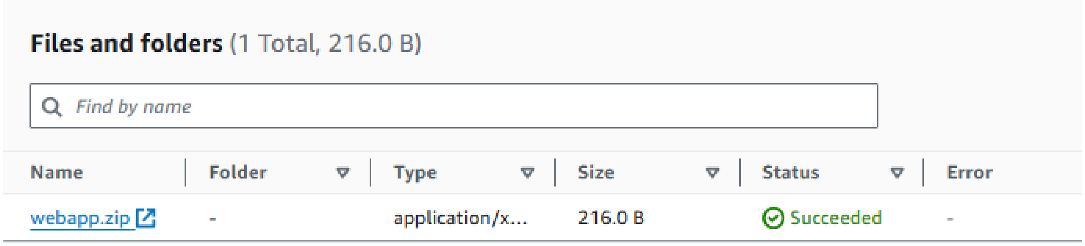
**Destination** Info

Destination

s3://abhinav-bucket-advdevops

**Destination details**

Bucket settings that impact new objects stored in the specified destination.

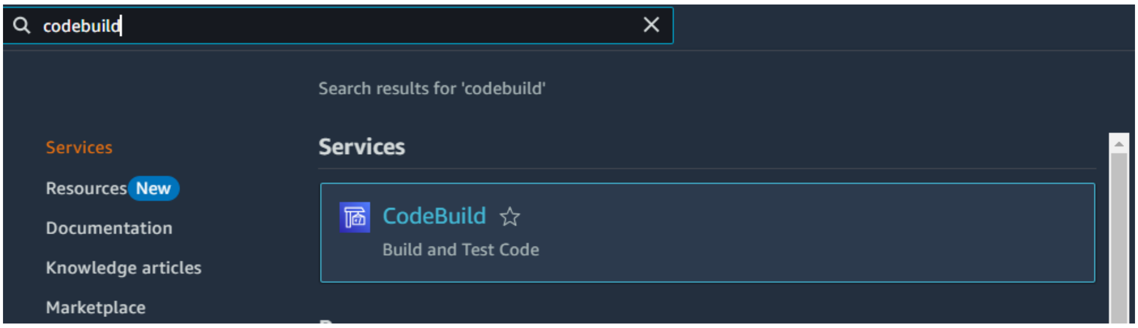


**Files and folders** (1 Total, 216.0 B)

| Name | Folder | Type | Size | Status | Error |
|---|---|---|---|---|---|
| webapp.zip | - | application/x... | 216.0 B | Succeeded | - |

## Step 4: Set Up AWS CodeBuild for the Web App

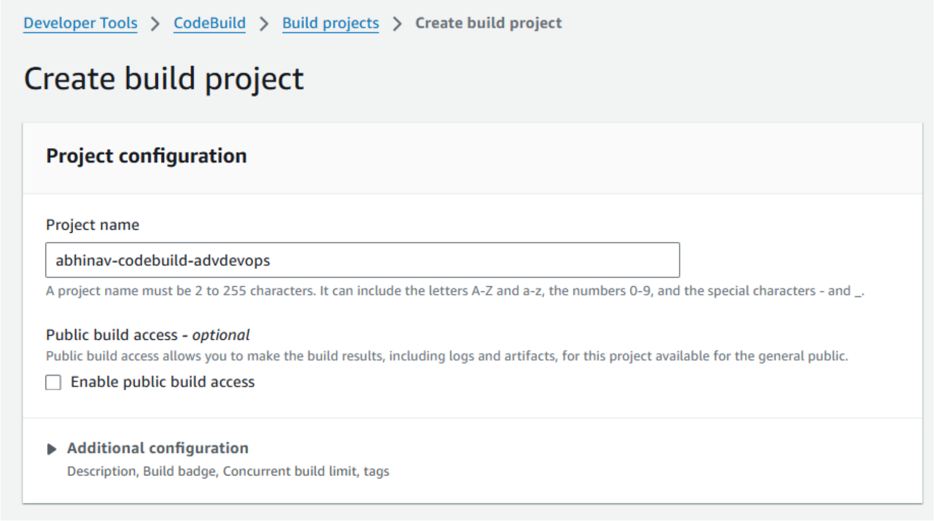
In the AWS Management Console, navigate to **CodeBuild** by searching for it in the top search bar.



Click Create build project.

Project Configuration:

- Project Name: Enter a name



Developer Tools > CodeBuild > Build projects > Create build project

Create build project

**Project configuration**

Project name

abhinav-codebuild-advdevops

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 _.

Public build access - *optional*

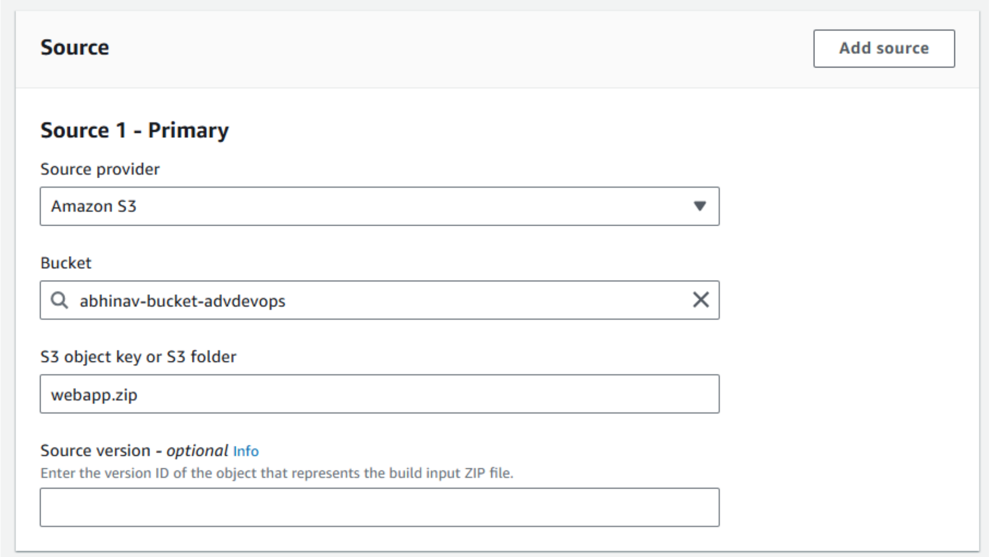
Public build access allows you to make the build results, including logs and artifacts, for this project available for the general public.

□ Enable public build access

▶ **Additional configuration**

Description, Build badge, Concurrent build limit, tags

- Source Provider: Select Amazon S3**.**
- Bucket**:** Select the bucket (your bucket name) you created earlier.
- Object Key**:** Select the webapp.zip file you uploaded.



**Source**

**Source 1 - Primary**

Source provider

Amazon S3

Bucket

abhinav-bucket-advdevops

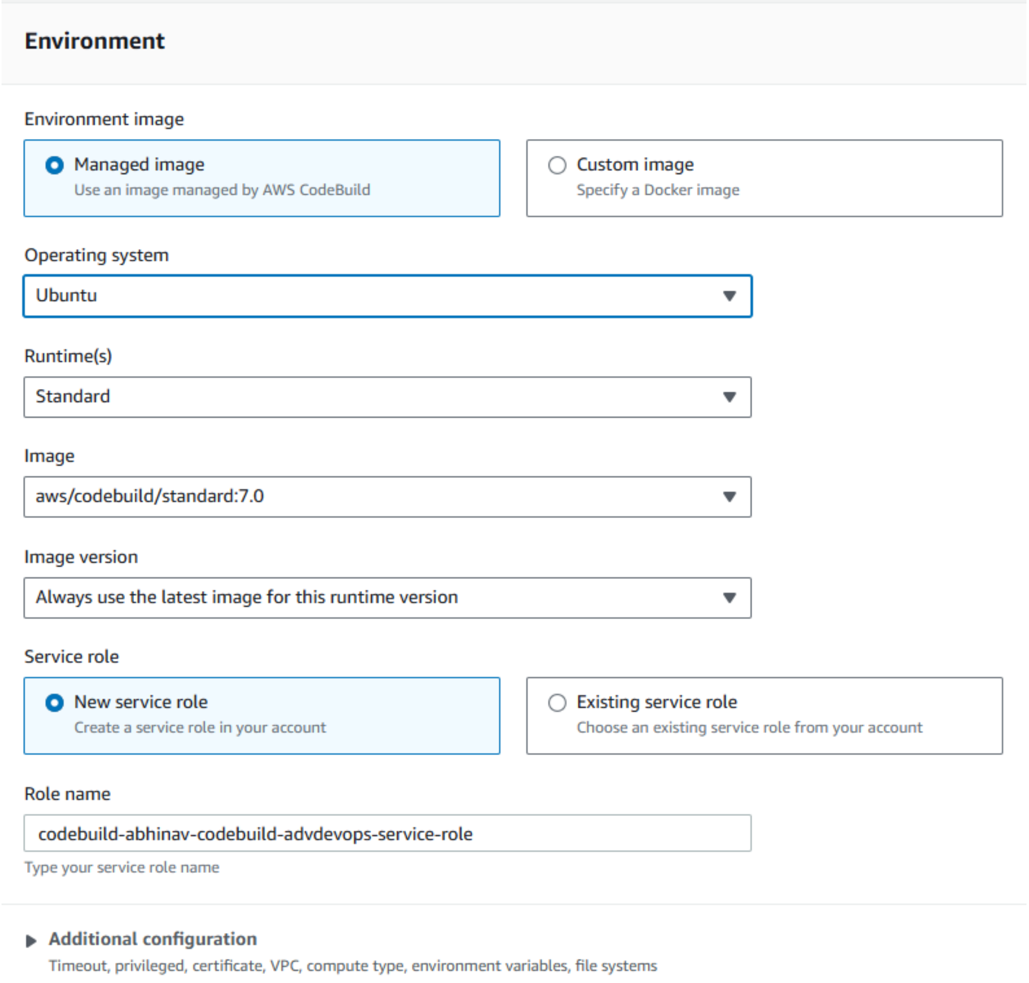
S3 object key or S3 folder

webapp.zip

Source version - *optional* Info

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

- Operating System: Select Ubuntu.



**Environment**

Environment image

● Managed image
Use an image managed by AWS CodeBuild

○ Custom image
Specify a Docker image

Operating system: Ubuntu

Runtime(s): Standard

Image: aws/codebuild/standard:7.0

Image version: Always use the latest image for this runtime version

Service role

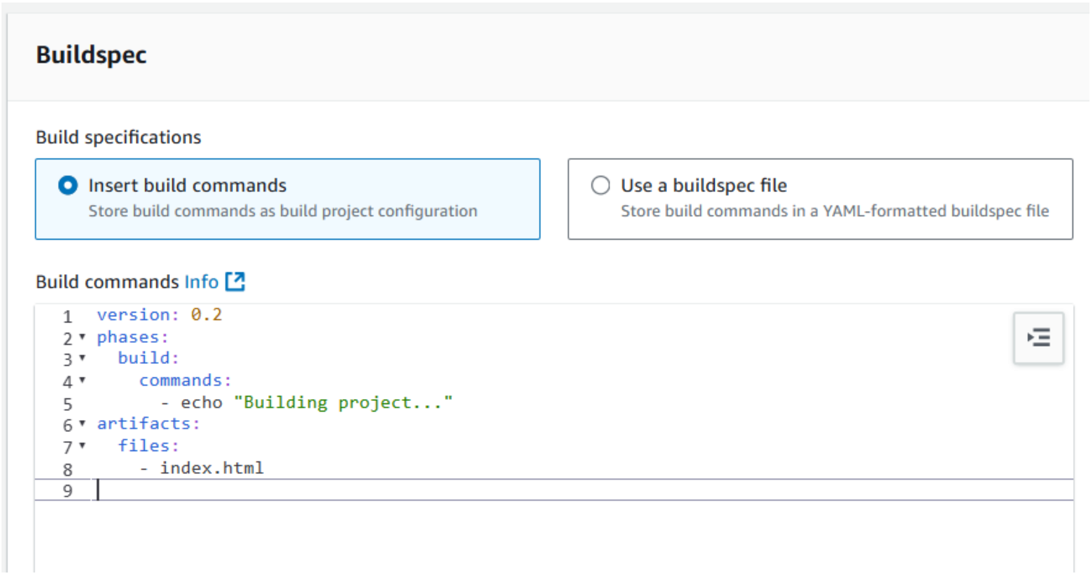
● New service role
Create a service role in your account

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

Role name: codebuild-abhinav-codebuild-advdevops-service-role
Type your service role name

▶ Additional configuration
Timeout, privileged, certificate, VPC, compute type, environment variables, file systems

Buildspec: **Select "Insert build commands" and enter the following YAML script: (Switch to editor)**



**Buildspec**

Build specifications

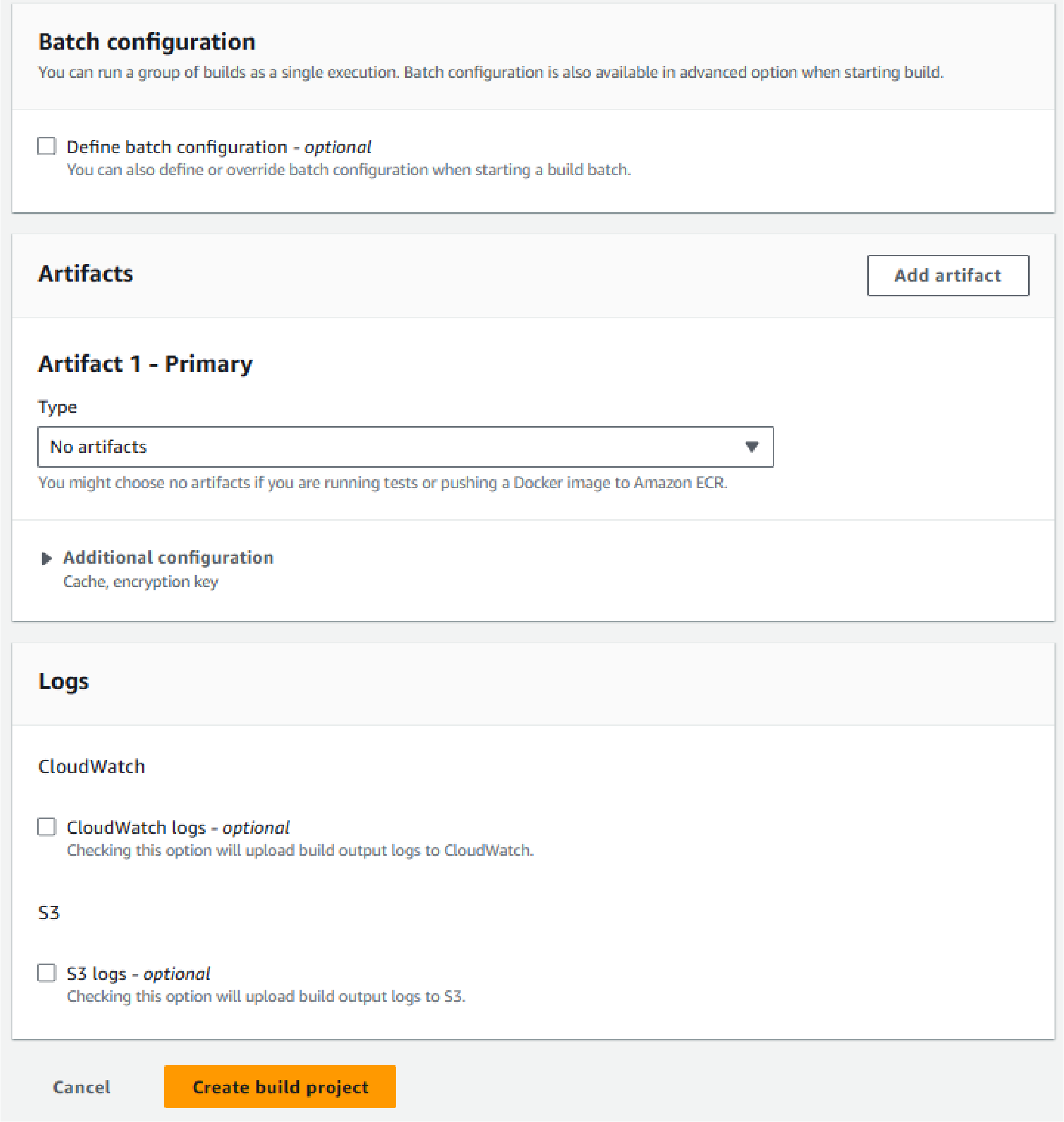
● Insert build commands
Store build commands as build project configuration

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

Build commands Info

```
version: 0.2
phases:
  build:
    commands:
      - echo "Building project..."
artifacts:
  files:
    - index.html

```

- Uncheck uploading logs to CloudWatch as it unnecessary



**Batch configuration**

You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.

☐ Define batch configuration - *optional*
You can also define or override batch configuration when starting a build batch.

**Artifacts**

**Artifact 1 - Primary**

Type: No artifacts

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

▶ Additional configuration
Cache, encryption key

**Logs**

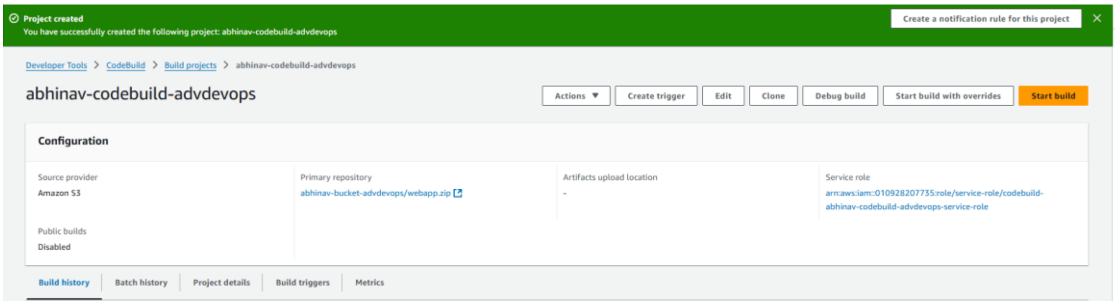
CloudWatch

☐ CloudWatch logs - *optional*
Checking this option will upload build output logs to CloudWatch.

S3

☐ S3 logs - *optional*
Checking this option will upload build output logs to S3.

Cancel
Create build project



Project created
You have successfully created the following project: abhinav-codebuild-advdevops

Create a notification rule for this project
Developer Tools > CodeBuild > Build projects > abhinav-codebuild-advdevops

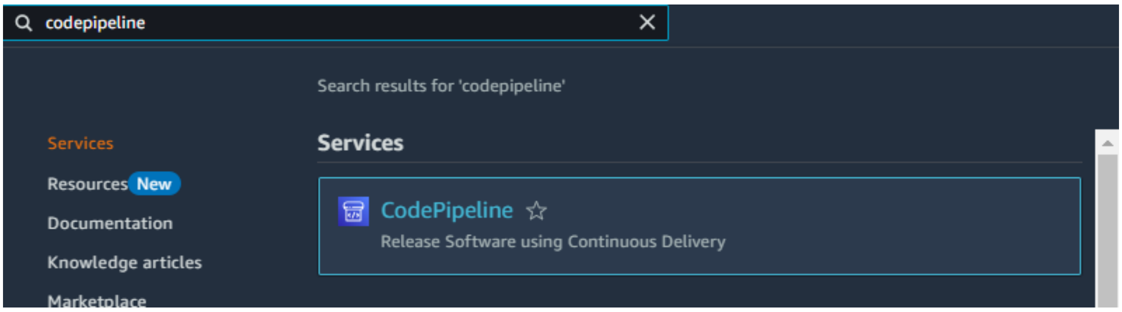
abhinav-codebuild-advdevops

Actions ▼
Create trigger
Edit
Clone
Debug build
Start build with overrides
Start build

**Configuration**

| Source provider | Primary repository | Artifacts upload location | Service role |
|---|---|---|---|
| Amazon S3 | abhinav-bucket-advdevops/webapp.zip | - | arn:aws:iam::010928207735:role/service-role/codebuild-abhinav-codebuild-advdevops-service-role |

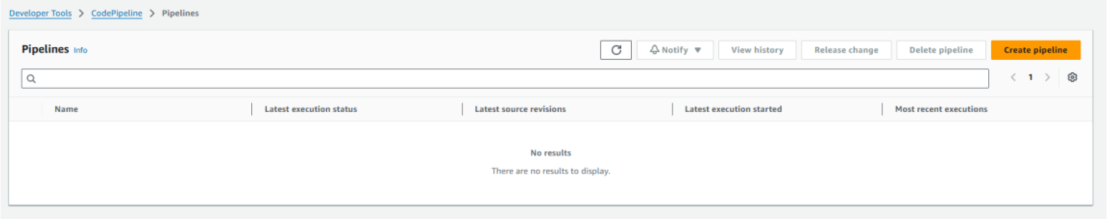
Public builds: Disabled



Search results for 'codepipeline'

CodePipeline ☆
Release Software using Continuous Delivery

Click Create Pipeline.

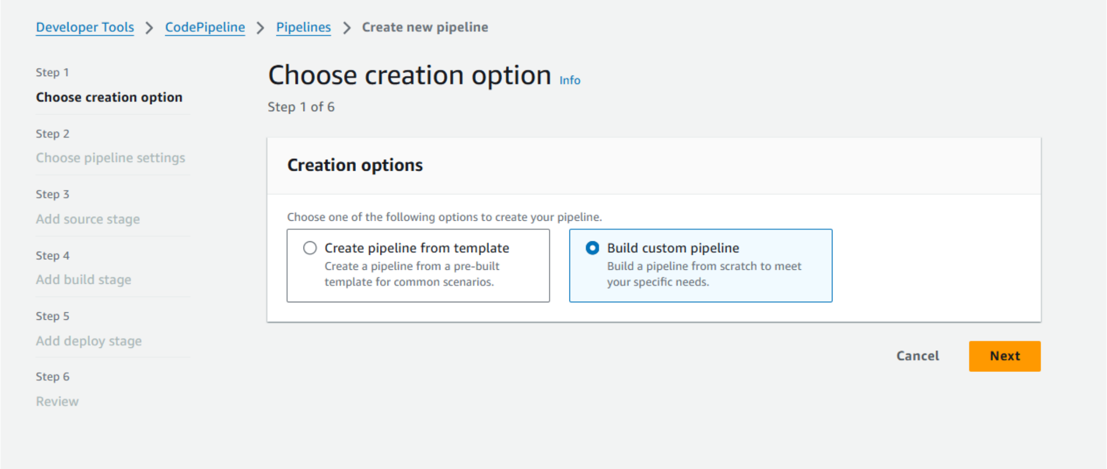


Pipelines Info

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

No results

There are no results to display.



Choose creation option Info

Step 1 of 6

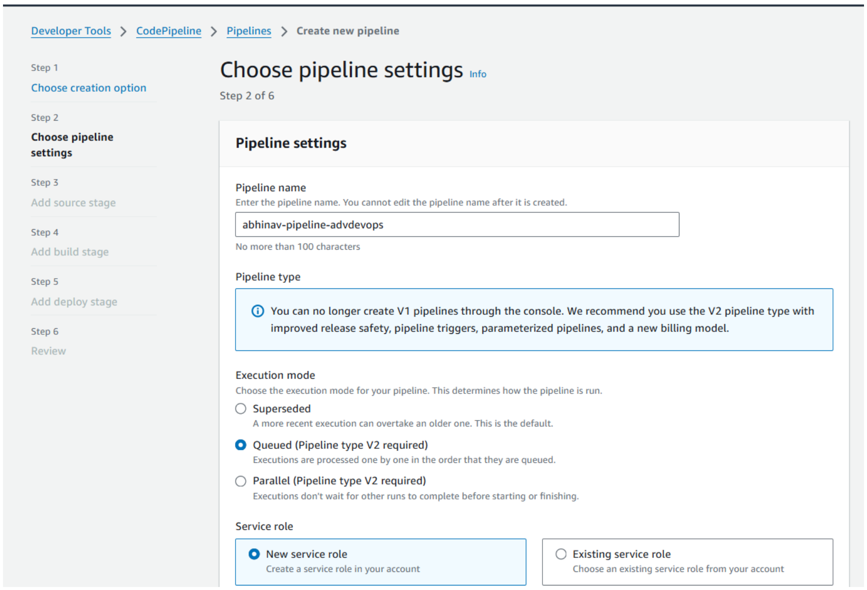
Creation options

Choose one of the following options to create your pipeline.

○ Create pipeline from template
Create a pipeline from a pre-built template for common scenarios.

● Build custom pipeline
Build a pipeline from scratch to meet your specific needs.

- Pipeline Name: Enter a name



Choose pipeline settings Info

Step 2 of 6

Pipeline settings

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

abhinav-pipeline-advdevops

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.

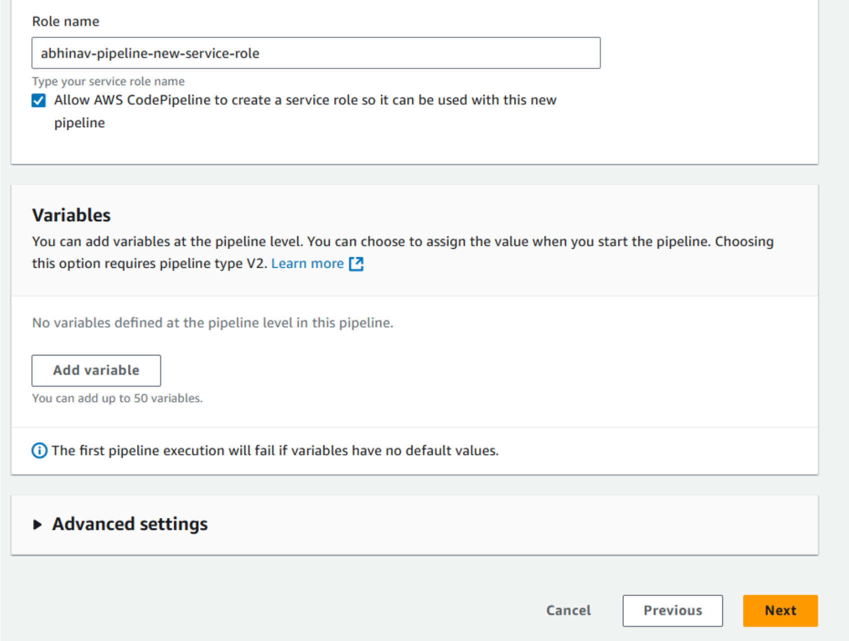
○ Parallel (Pipeline type V2 required)
Executions don't wait for other runs to complete before starting or finishing.

Service role

● New service role
Create a service role in your account

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

- Service Role: Select New service role to create a new role for this pipeline.



Role name

abhinav-pipeline-new-service-role

Type your service role name

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

**Variables**

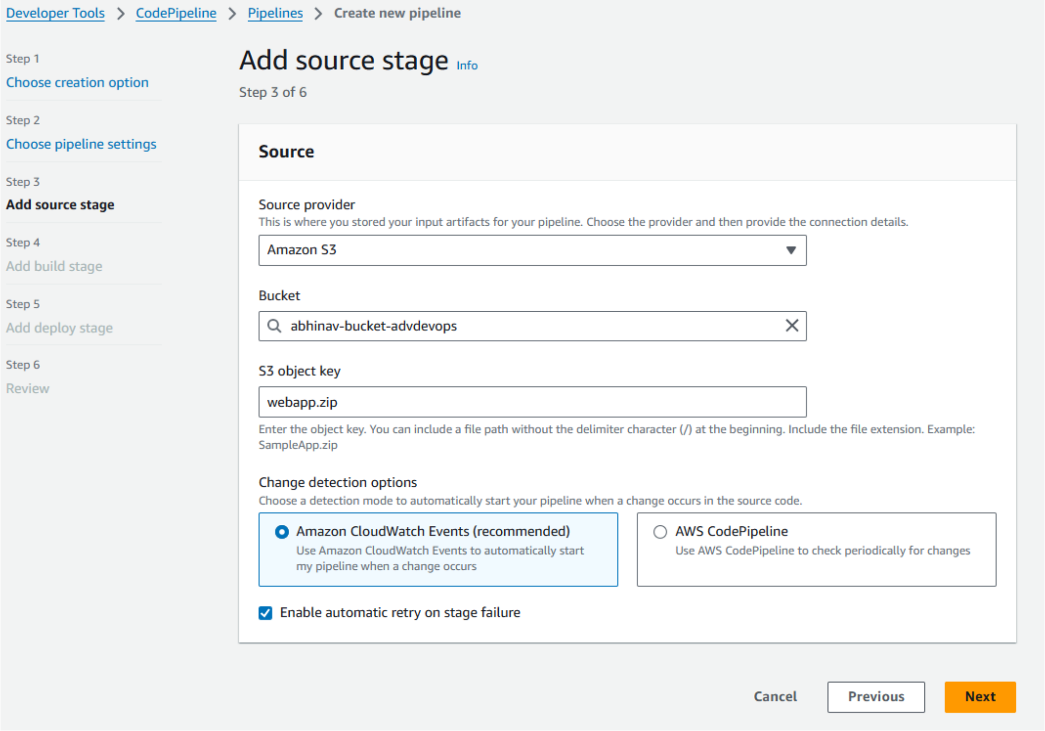
You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. Choosing this option requires pipeline type V2. Learn more

No variables defined at the pipeline level in this pipeline.

You can add up to 50 variables.

The first pipeline execution will fail if variables have no default values.

**Source Provider**: Select Amazon S3.
**Bucket:** Select the same S3 bucket you used earlier.
**S3 object key**: Enter webapp.zip as the source file.



Add source stage Info

Step 3 of 6

**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

abhinav-bucket-advdevops

S3 object key

webapp.zip

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

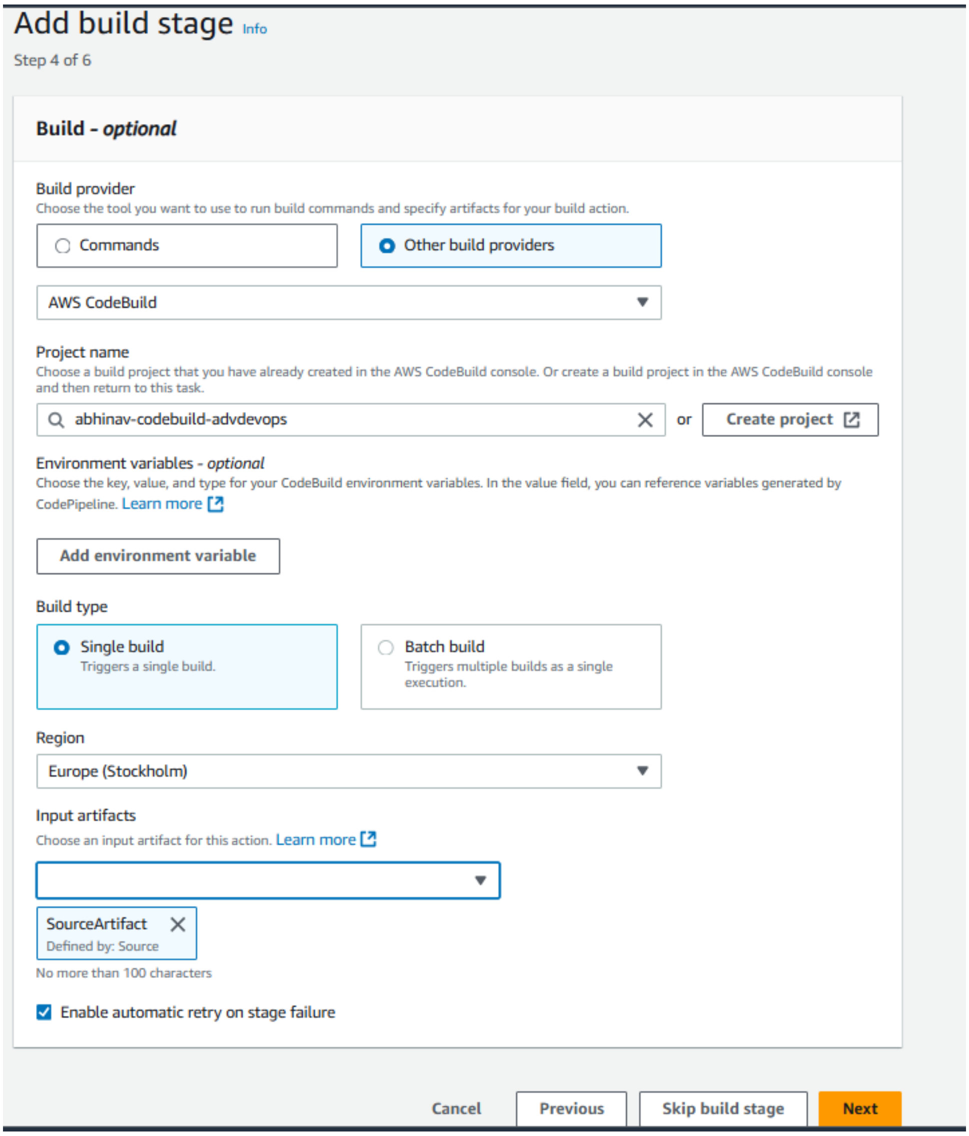
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

● 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

☑ Enable automatic retry on stage failure

**Build Provider:** Select Other Build Providers Then select AWS Codebuild
**Project Name:** Select the build project you created in Step 4



Add build stage Info

Step 4 of 6

Build - *optional*

Build provider

Choose the tool you want to use to run build commands and specify artifacts for your build action.

○ Commands ● Other build providers

AWS CodeBuild ▼

Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

abhinav-codebuild-advdevops ✕ or Create project

Environment variables - *optional*

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. Learn more

Add environment variable

Build type

● Single build
Triggers a single build.

○ Batch build
Triggers multiple builds as a single execution.

Region

Europe (Stockholm) ▼

Input artifacts

Choose an input artifact for this action. Learn more

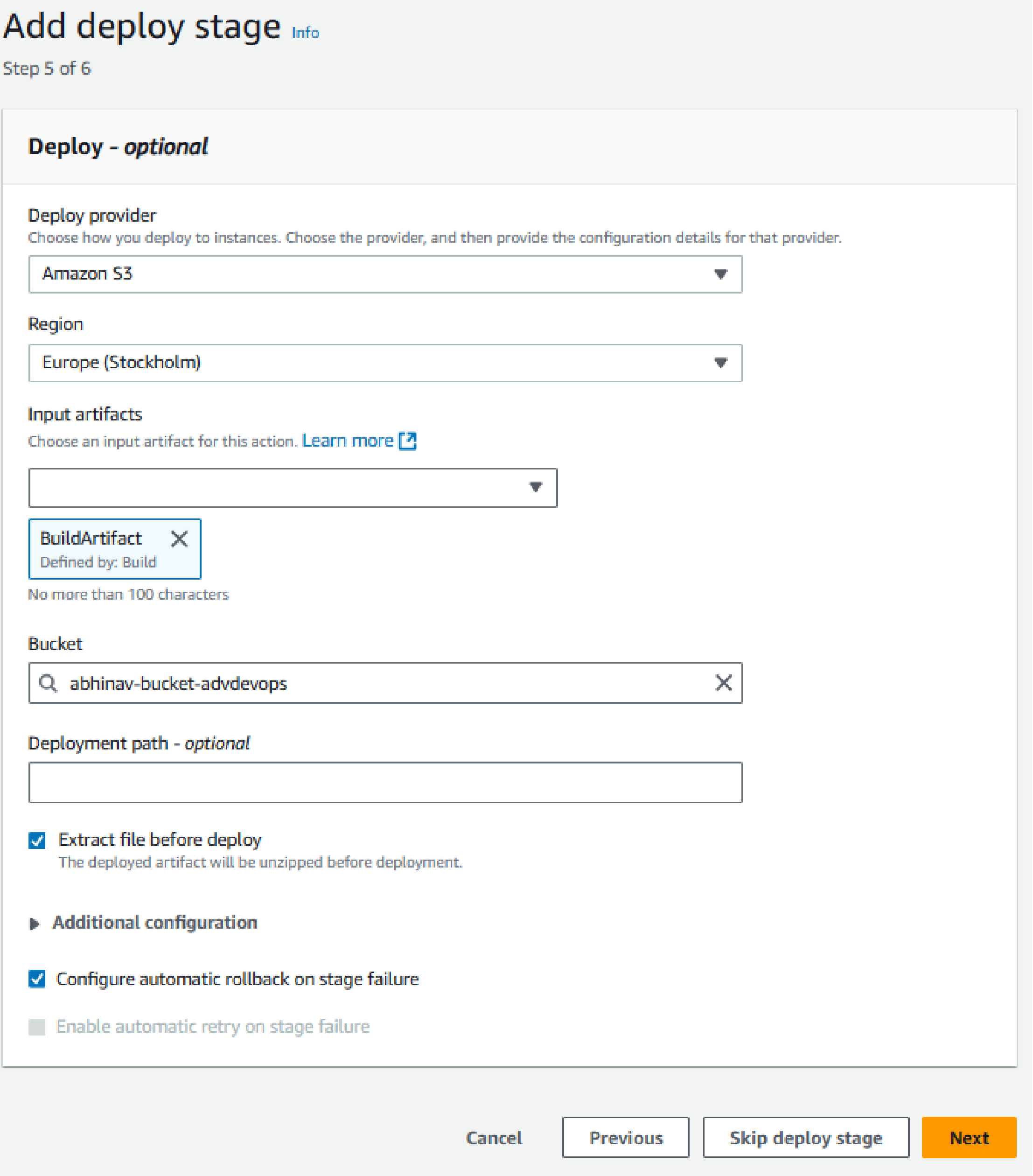
▼

SourceArtifact ✕
Defined by: Source

No more than 100 characters

☑ Enable automatic retry on stage failure

**Deploy Provider**: Select Amazon S3.
**Bucket:** Choose the same S3 bucket.
For Extract file before deploy, select Yes.



Add deploy stage Info

Step 5 of 6

**Deploy - *optional***

Deploy provider

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

Amazon S3

Region

Europe (Stockholm)

Input artifacts

Choose an input artifact for this action. Learn more

BuildArtifact
Defined by: Build

No more than 100 characters

Bucket

abhinav-bucket-advdevops

Deployment path - *optional*

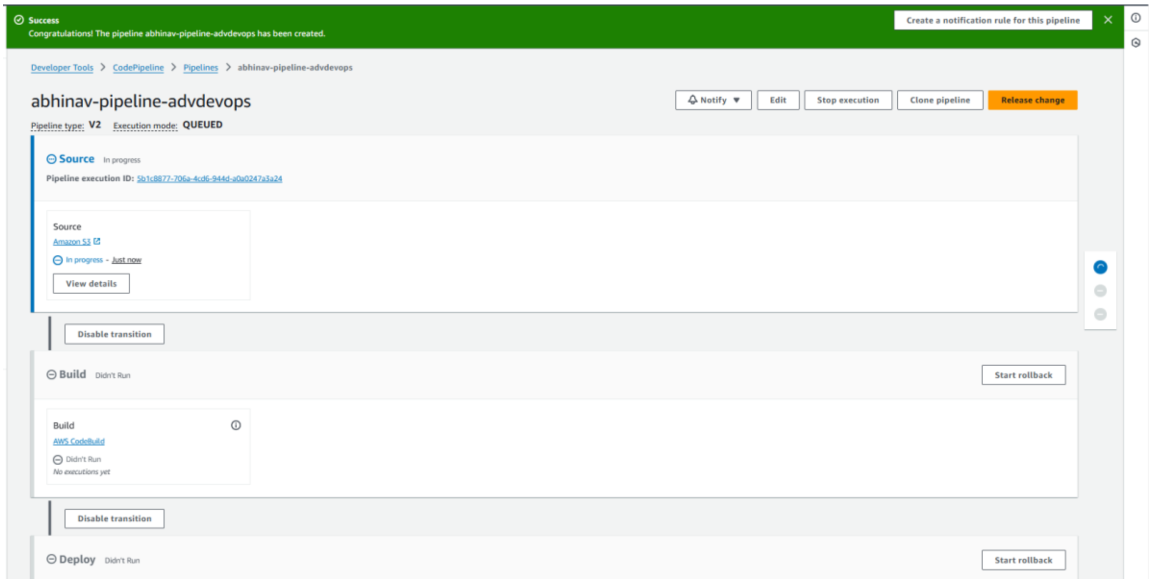
☑ Extract file before deploy
The deployed artifact will be unzipped before deployment.

Additional configuration

☑ Configure automatic rollback on stage failure

□ Enable automatic retry on stage failure

Review your bucket and create the pipeline



Success
Congratulations! The pipeline abhinav-pipeline-advdevops has been created.

Developer Tools > CodePipeline > Pipelines > abhinav-pipeline-advdevops

abhinav-pipeline-advdevops

Pipeline type: V2 Execution mode: QUEUED

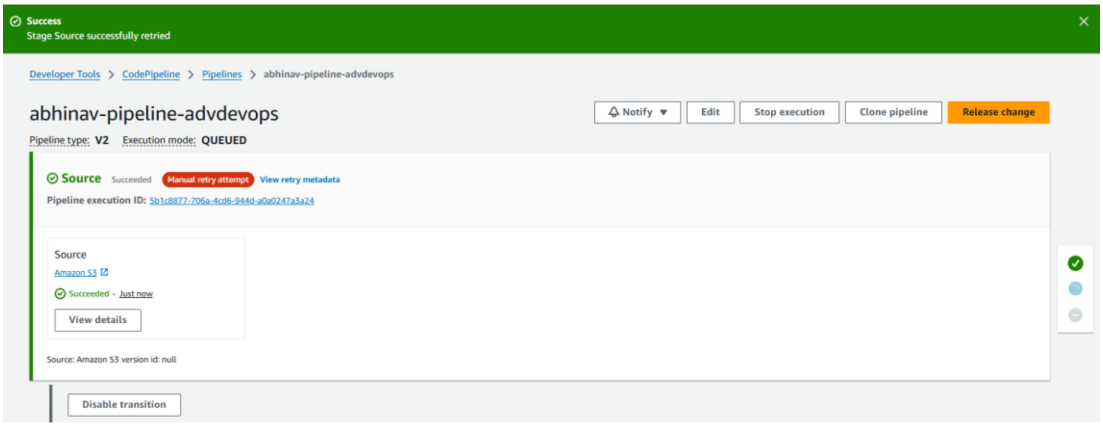
Source In progress
Pipeline execution ID: 5b1c8877-706a-4cd6-944d-a0a0247a3a24

Source
Amazon S3
In progress - Just now

Build Didn't Run

Build
AWS CodeBuild
Didn't Run
*No executions yet*

Deploy Didn't Run



Success
Stage Source successfully retried

Developer Tools > CodePipeline > Pipelines > abhinav-pipeline-advdevops

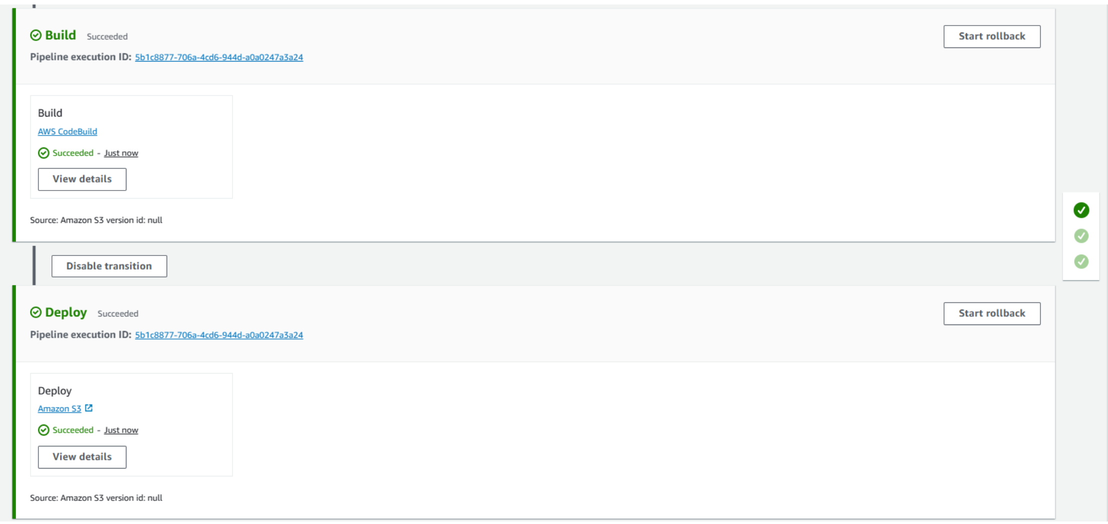
abhinav-pipeline-advdevops

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded Manual retry attempt View retry metadata
Pipeline execution ID: 5b1c8877-706a-4cd6-944d-a0a0247a3a24

Source
Amazon S3
Succeeded - Just now

Source: Amazon S3 version id: null



Build Succeeded
Pipeline execution ID: 5b1c8877-706a-4cd6-944d-a0a0247a3a24

Build
AWS CodeBuild
Succeeded - Just now

Source: Amazon S3 version id: null

Deploy Succeeded
Pipeline execution ID: 5b1c8877-706a-4cd6-944d-a0a0247a3a24

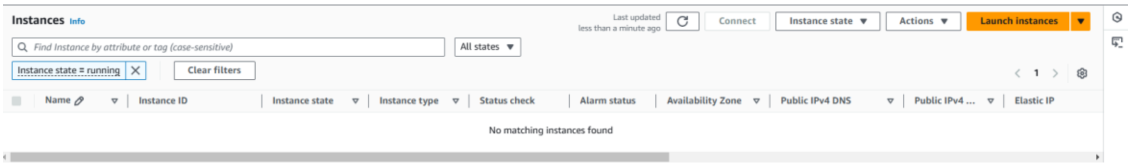
Deploy
Amazon S3
Succeeded - Just now

Source: Amazon S3 version id: null

## Step 6: Create an EC2 Instance

Click on **Launch Instance**.

Select an instance type (e.g., t3.micro for free tier eligibility).



Instances Info

Last updated less than a minute ago

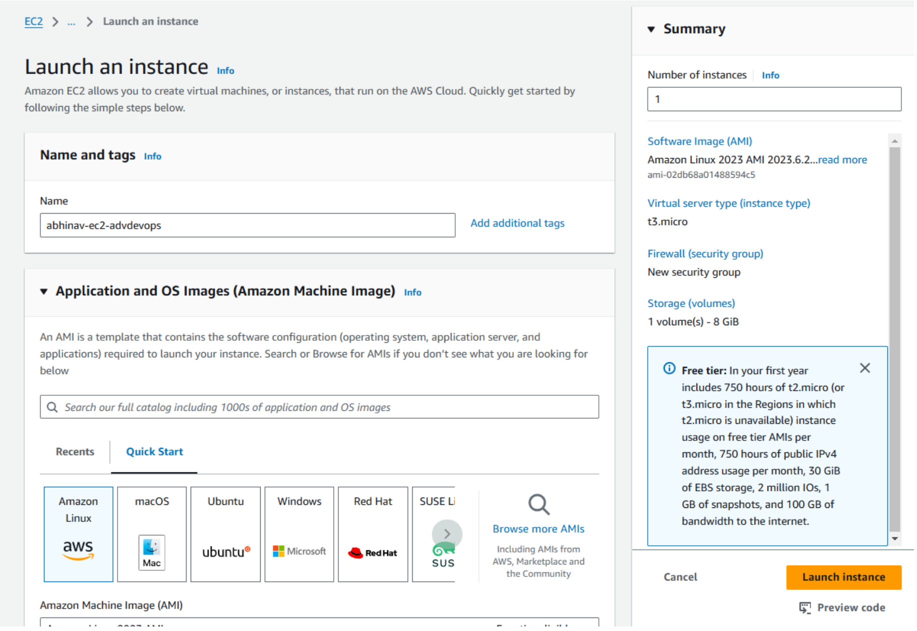
Find Instance by attribute or tag (case-sensitive)

All states ▼

Instance state = running

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... | Elastic IP |
|---|---|---|---|---|---|---|---|---|---|

No matching instances found



EC2 > ... > Launch an instance

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: abhinav-ec2-advdevops

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

Amazon Machine Image (AMI)

Summary

Number of instances Info: 1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-02db68a01488594c5

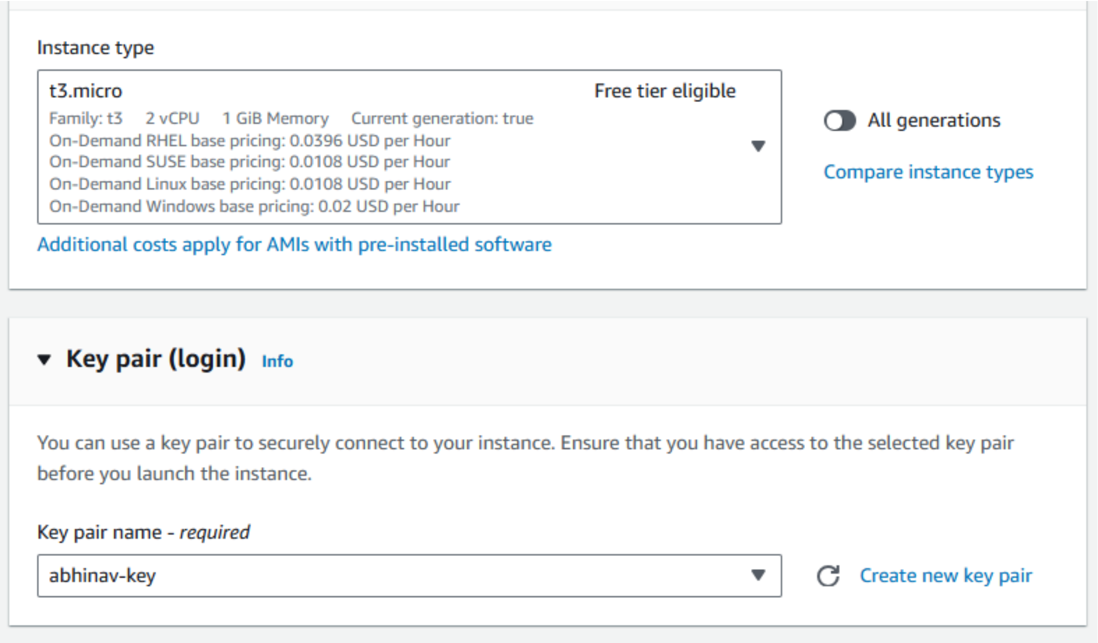
Virtual server type (instance type)
t3.micro

Firewall (security group)
New security group

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

**Free tier:** In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Create a new key-pair of RSA type. (Will be used to SSH into our instance later)



Instance type

t3.micro Free tier eligible

Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand RHEL base pricing: 0.0396 USD per Hour
On-Demand SUSE base pricing: 0.0108 USD per Hour
On-Demand Linux base pricing: 0.0108 USD per Hour
On-Demand Windows base pricing: 0.02 USD per Hour

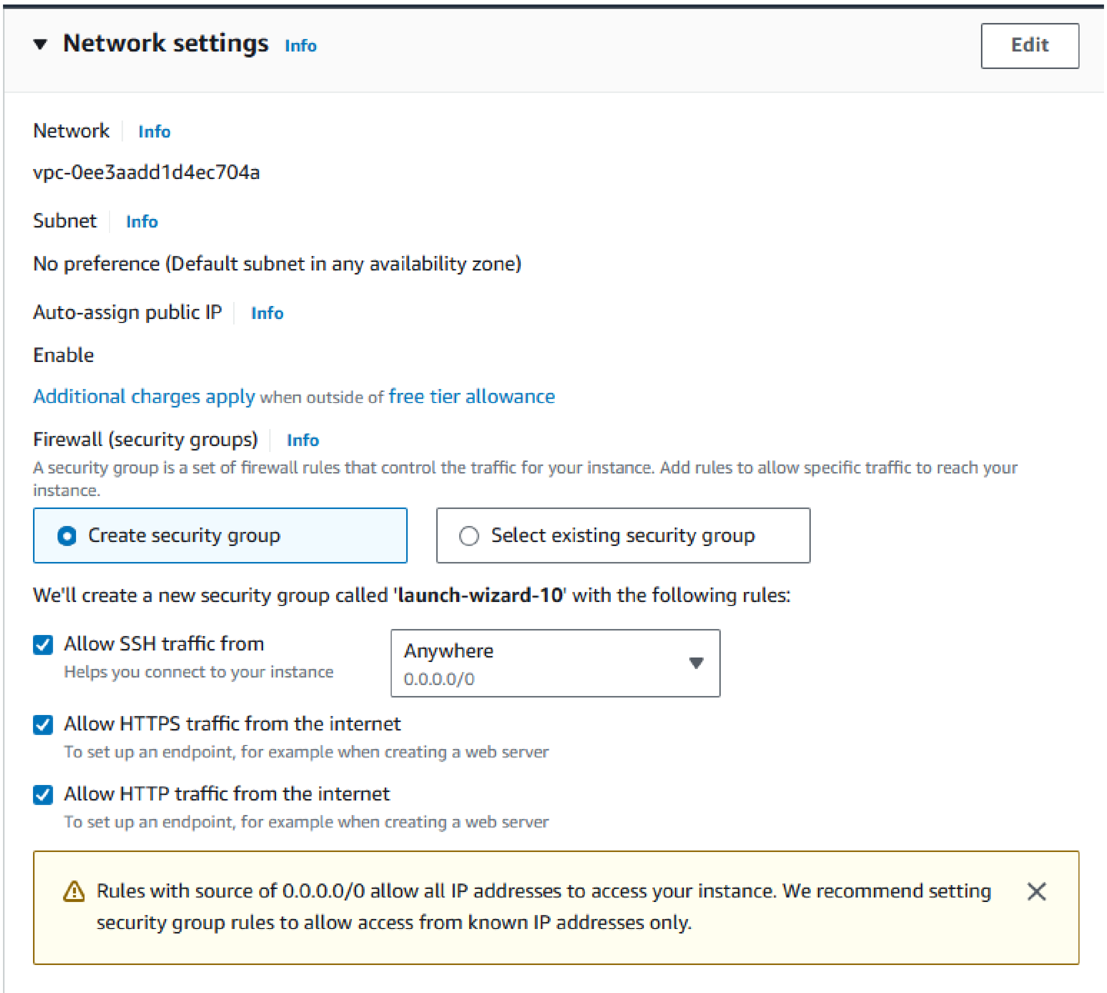
Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

abhinav-key

Allow HTTP/HTTPS traffic from anywhere



Network settings Info

Network Info

vpc-0ee3aadd1d4ec704a

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

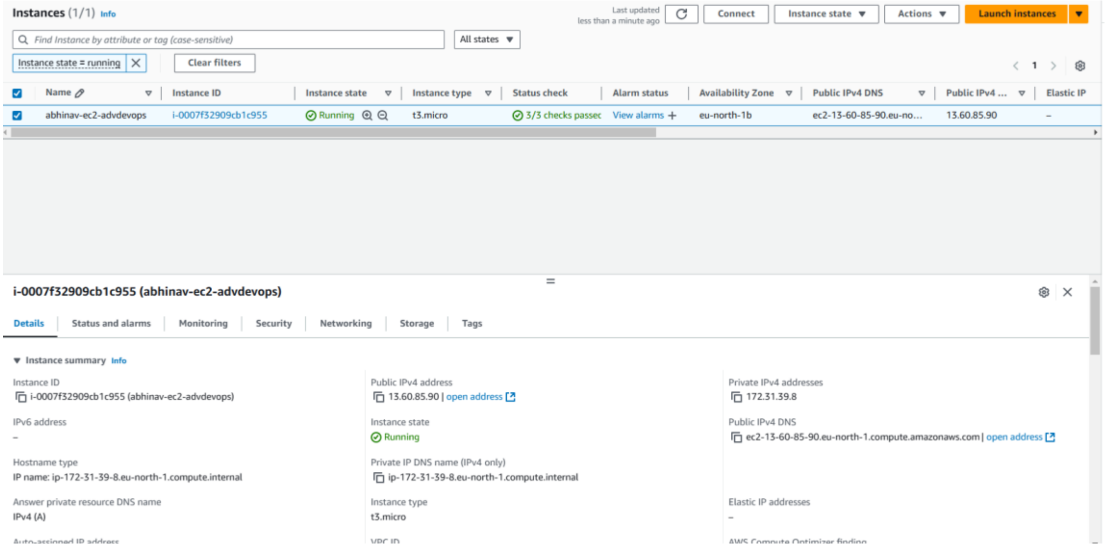
We'll create a new security group called '**launch-wizard-10**' with the following rules:

☑ Allow SSH traffic from — Anywhere 0.0.0.0/0
Helps you connect to your instance

☑ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☑ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.



Instances (1/1) Info

Last updated less than a minute ago

Find Instance by attribute or tag (case-sensitive) | All states ▼

Instance state = running ✕ | Clear filters

| ☑ | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... | Elastic IP |
|---|---|---|---|---|---|---|---|---|---|---|
| ☑ | abhinav-ec2-advdevops | i-0007f32909cb1c955 | ✓ Running | t3.micro | ✓ 3/3 checks passec | View alarms + | eu-north-1b | ec2-13-60-85-90.eu-no... | 13.60.85.90 | – |

i-0007f32909cb1c955 (abhinav-ec2-advdevops)

▼ Instance summary Info

Instance ID: i-0007f32909cb1c955 (abhinav-ec2-advdevops)

Public IPv4 address: 13.60.85.90 | open address

Private IPv4 addresses: 172.31.39.8

IPv6 address: –

Instance state: Running

Public IPv4 DNS: ec2-13-60-85-90.eu-north-1.compute.amazonaws.com | open address

Hostname type: IP name: ip-172-31-39-8.eu-north-1.compute.internal

Private IP DNS name (IPv4 only): ip-172-31-39-8.eu-north-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Instance type: t3.micro

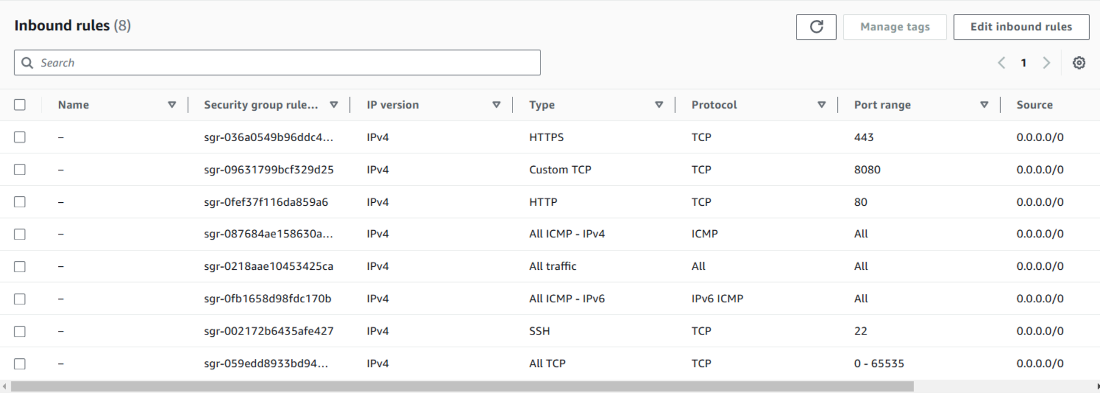
Elastic IP addresses: –

Auto-assigned IP address

VPC ID

AWS Compute Optimizer finding

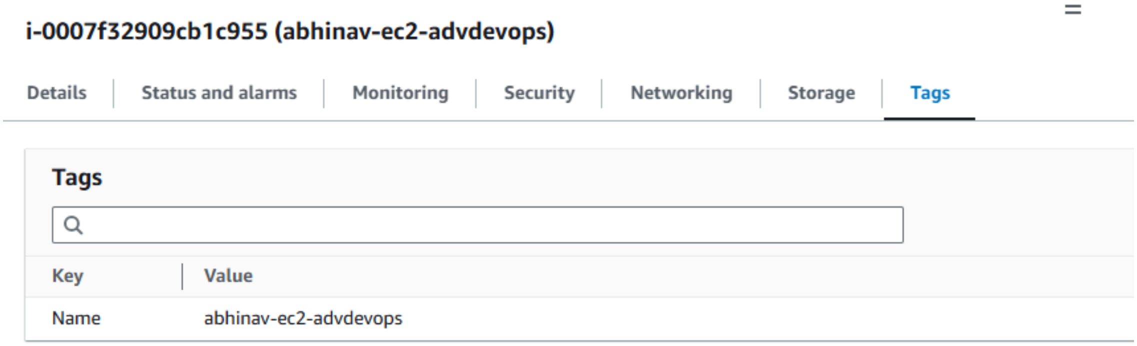
Edit inbound rules in security groups



Inbound rules (8)

| □ | Name | Security group rule... | IP version | Type | Protocol | Port range | Source |
|---|---|---|---|---|---|---|---|
| □ | – | sgr-036a0549b96ddc4... | IPv4 | HTTPS | TCP | 443 | 0.0.0.0/0 |
| □ | – | sgr-09631799bcf329d25 | IPv4 | Custom TCP | TCP | 8080 | 0.0.0.0/0 |
| □ | – | sgr-0fef37f116da859a6 | IPv4 | HTTP | TCP | 80 | 0.0.0.0/0 |
| □ | – | sgr-087684ae158630a... | IPv4 | All ICMP - IPv4 | ICMP | All | 0.0.0.0/0 |
| □ | – | sgr-0218aae10453425ca | IPv4 | All traffic | All | All | 0.0.0.0/0 |
| □ | – | sgr-0fb1658d98fdc170b | IPv4 | All ICMP - IPv6 | IPv6 ICMP | All | 0.0.0.0/0 |
| □ | – | sgr-002172b6435afe427 | IPv4 | SSH | TCP | 22 | 0.0.0.0/0 |
| □ | – | sgr-059edd8933bd94... | IPv4 | All TCP | TCP | 0 - 65535 | 0.0.0.0/0 |

Take note of tag of instance



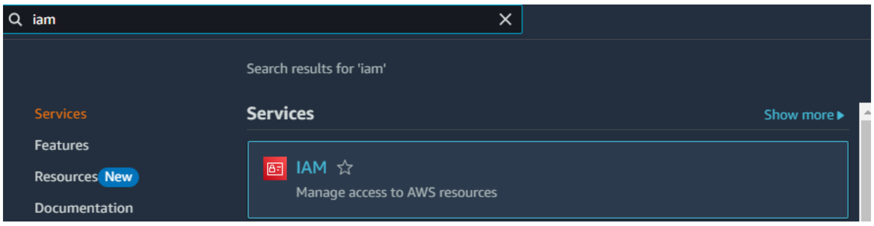
i-0007f32909cb1c955 (abhinav-ec2-advdevops)

Tags

| Key | Value |
|---|---|
| Name | abhinav-ec2-advdevops |

## Step 7: Create an IAM Role for EC2 with CodeDeploy Permissions

In the AWS Management Console, go to **Services** and select **IAM**.



iam

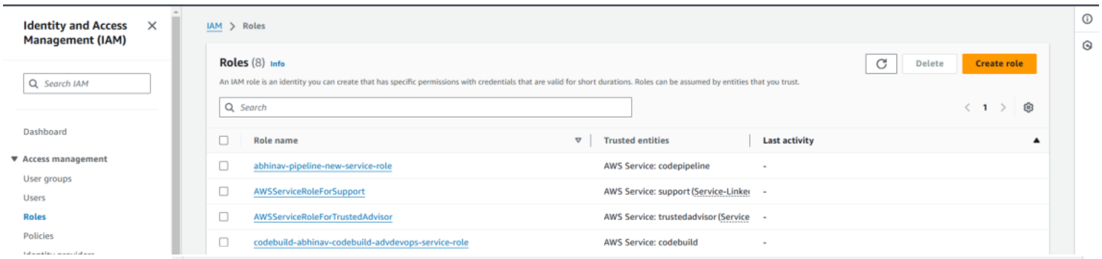
Search results for 'iam'

Services

IAM ☆

Manage access to AWS resources

Click on **Roles** in the left sidebar and then click **Create role**.



Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

Users

Roles

Policies

IAM > Roles

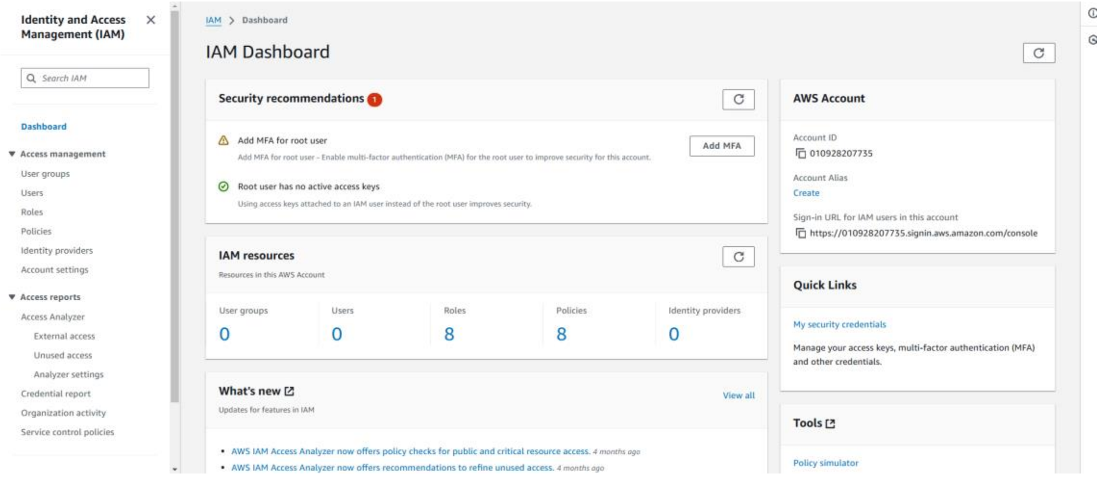
Roles (8) Info

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.

Delete | Create role

Search

| □ | Role name | Trusted entities | Last activity |
|---|---|---|---|
| □ | abhinav-pipeline-new-service-role | AWS Service: codepipeline | - |
| □ | AWSServiceRoleForSupport | AWS Service: support (Service-Linke | - |
| □ | AWSServiceRoleForTrustedAdvisor | AWS Service: trustedadvisor (Service | - |
| □ | codebuild-abhinav-codebuild-advdevops-service-role | AWS Service: codebuild | - |



Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

▼ Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Organization activity

Service control policies

IAM > Dashboard

IAM Dashboard

Security recommendations 1

Add MFA for root user

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

Add MFA

Root user has no active access keys

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

IAM resources

Resources in this AWS Account

| User groups | Users | Roles | Policies | Identity providers |
|---|---|---|---|---|
| 0 | 0 | 8 | 8 | 0 |

What's new

Updates for features in IAM

View all

- AWS IAM Access Analyzer now offers policy checks for public and critical resource access. 4 months ago
- AWS IAM Access Analyzer now offers recommendations to refine unused access. 4 months ago

AWS Account

Account ID

010928207735

Account Alias

Create

Sign-in URL for IAM users in this account

https://010928207735.signin.aws.amazon.com/console

Quick Links

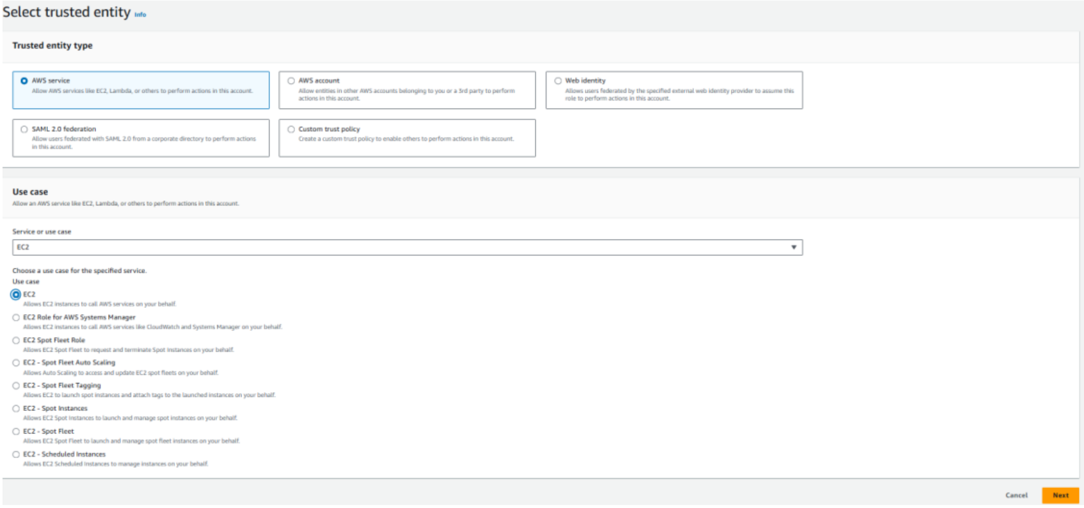
My security credentials

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

Tools

Policy simulator

Choose **AWS service** and then select **EC2**.



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.

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.

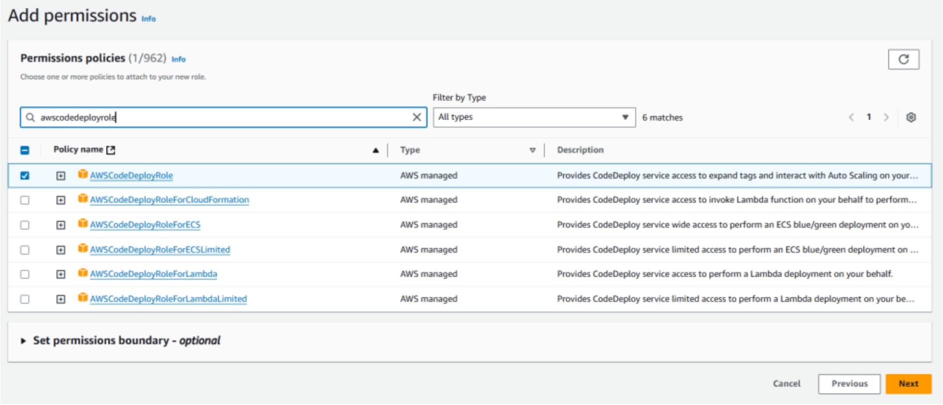
○ EC2 - Spot Instances
Allows EC2 Spot Instances to launch and manage spot instances on your behalf.

○ 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.

Cancel | Next

Search for and attach the AWSCodeDeployRole, AmazonEC2RoleforCodeDeploy policy to grant the necessary permissions.



Add permissions Info

Permissions policies (1/962) Info

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

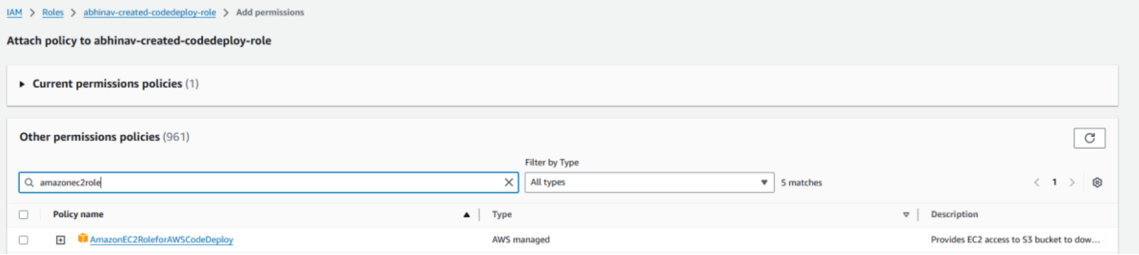
Filter by Type

awscodedeployrole | All types | 6 matches

| | Policy name | Type | Description |
|---|---|---|---|
| ☑ | AWSCodeDeployRole | AWS managed | Provides CodeDeploy service access to expand tags and interact with Auto Scaling on your... |
| □ | AWSCodeDeployRoleForCloudFormation | AWS managed | Provides CodeDeploy service access to invoke Lambda function on your behalf to perform... |
| □ | AWSCodeDeployRoleForECS | AWS managed | Provides CodeDeploy service wide access to perform an ECS blue/green deployment on yo... |
| □ | AWSCodeDeployRoleForECSLimited | AWS managed | Provides CodeDeploy service limited access to perform an ECS blue/green deployment on ... |
| □ | AWSCodeDeployRoleForLambda | AWS managed | Provides CodeDeploy service access to perform a Lambda deployment on your behalf. |
| □ | AWSCodeDeployRoleForLambdaLimited | AWS managed | Provides CodeDeploy service limited access to perform a Lambda deployment on your be... |

▸ Set permissions boundary - *optional*

Cancel | Previous | Next



IAM > Roles > abhinav-created-codedeploy-role > Add permissions

Attach policy to abhinav-created-codedeploy-role

▸ Current permissions policies (1)

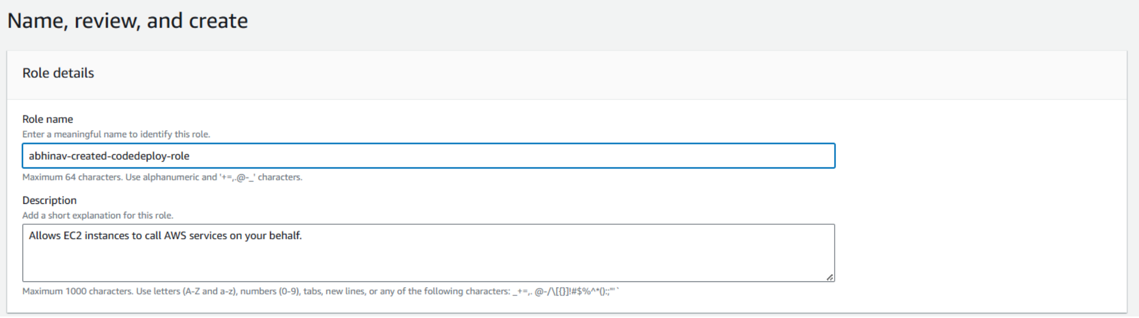
Other permissions policies (961)

Filter by Type

amazonec2role | All types | 5 matches

| □ | Policy name | Type | Description |
|---|---|---|---|
| □ | AmazonEC2RoleforAWSCodeDeploy | AWS managed | Provides EC2 access to S3 bucket to dow... |

Name your role and click **Create role**.



Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

abhinav-created-codedeploy-role

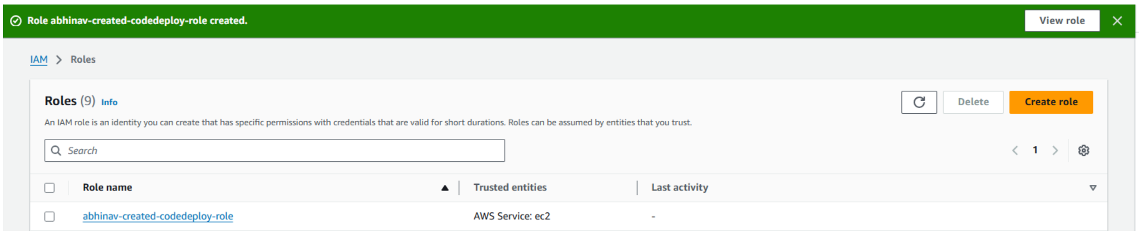
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

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: _+=,. @-/\[{}]!#$%^*():;"' `



Role abhinav-created-codedeploy-role created.

IAM > Roles

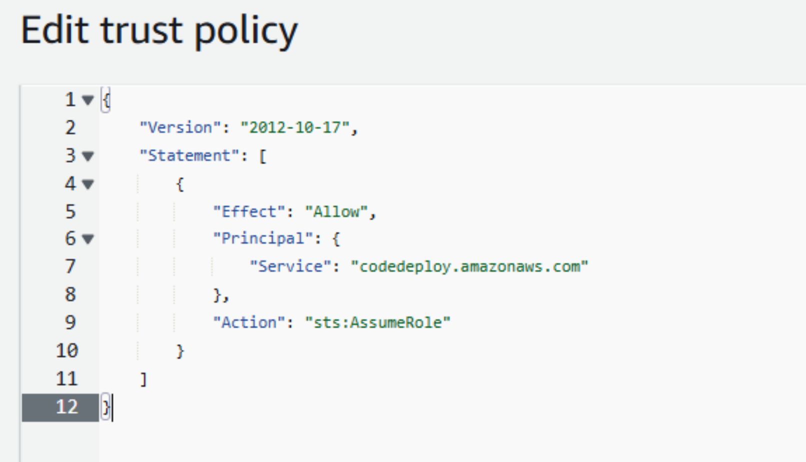
Roles (9) Info

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 |
|---|---|---|---|
| □ | abhinav-created-codedeploy-role | AWS Service: ec2 | - |

**Goto trusted relationships and edit trust policy with code**

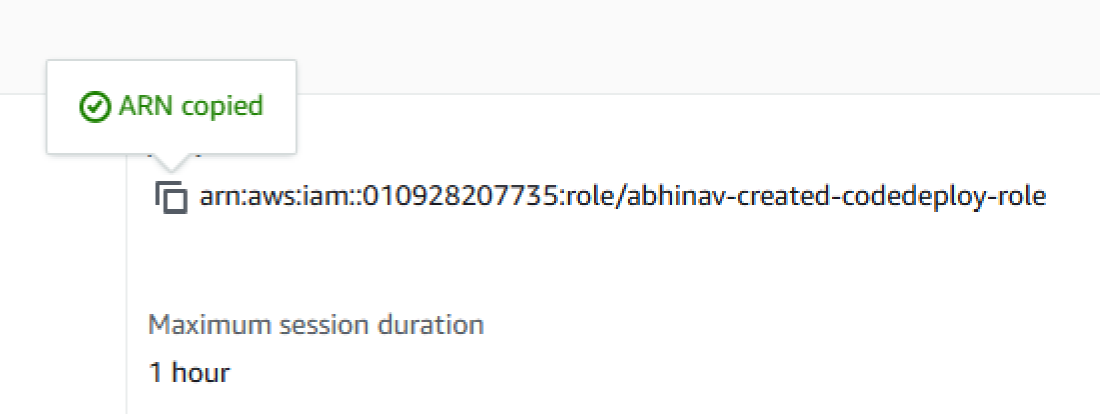
```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Principal": {
    "Service": "codedeploy.amazonaws.com"
   },
   "Action": "sts:AssumeRole"
  }
 ]
}
```



Edit trust policy

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

Copy the arn as it will be needed later



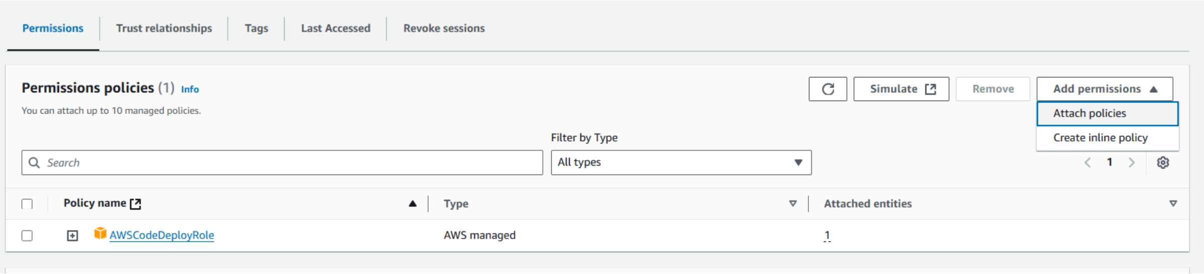
ARN copied

arn:aws:iam::010928207735:role/abhinav-created-codedeploy-role

Maximum session duration

1 hour

Add inline policy



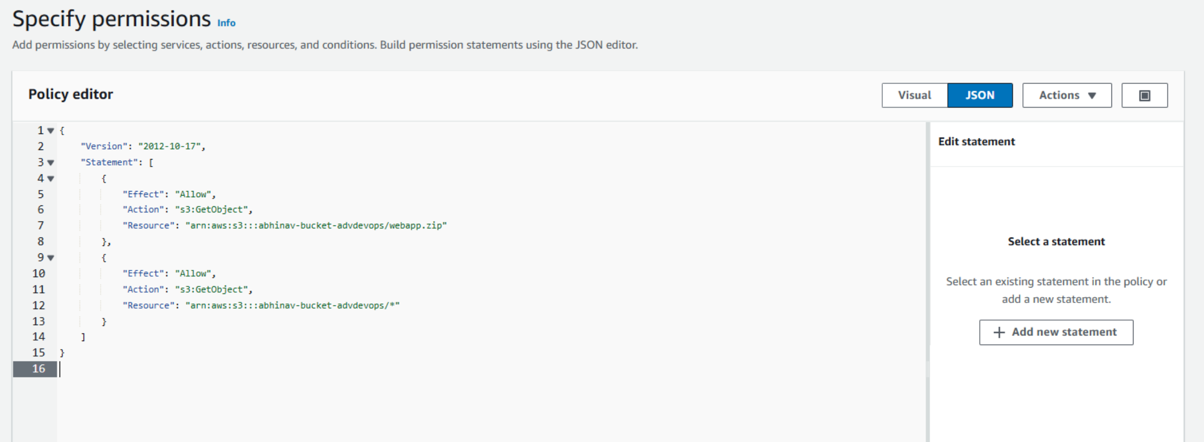
Permissions policies (1) Info

You can attach up to 10 managed policies.

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

**In Json editor**

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::abhinav-bucket-advdevops/webapp.zip"
    },
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::abhinav-bucket-advdevops/*"
    }
  ]
}
```



Specify permissions Info

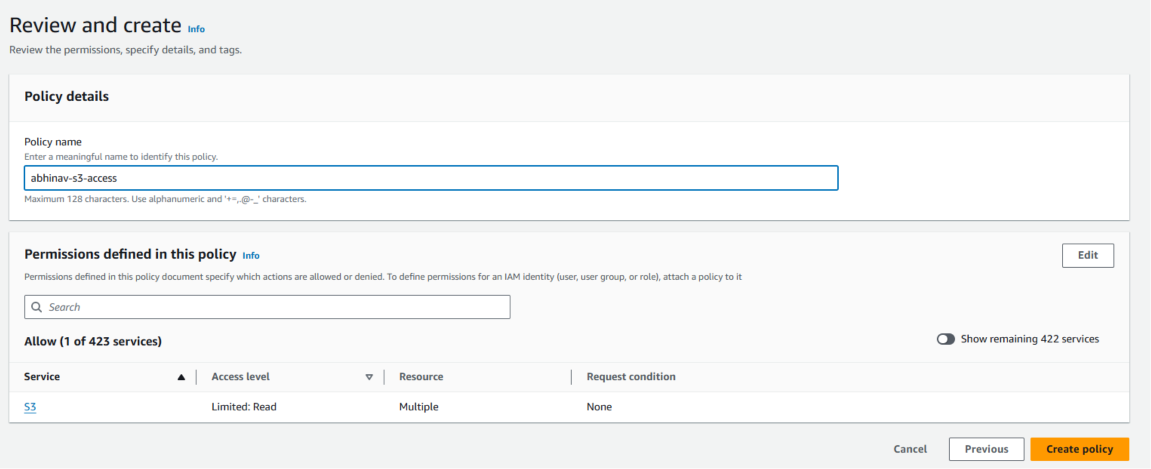
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::abhinav-bucket-advdevops/webapp.zip"
        },
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::abhinav-bucket-advdevops/*"
        }
    ]
}
```

Select an existing statement in the policy or add a new statement.

Review and create



Review and create Info

Review the permissions, specify details, and tags.

**Policy details**

Policy name

Enter a meaningful name to identify this policy.

abhinav-s3-access

Maximum 128 characters. Use alphanumeric and '+=,.@-_' characters.

**Permissions defined in this policy** Info

Permissions defined in this policy document specify which actions are allowed or denied. To define permissions for an IAM identity (user, user group, or role), attach a policy to it

**Allow (1 of 423 services)**

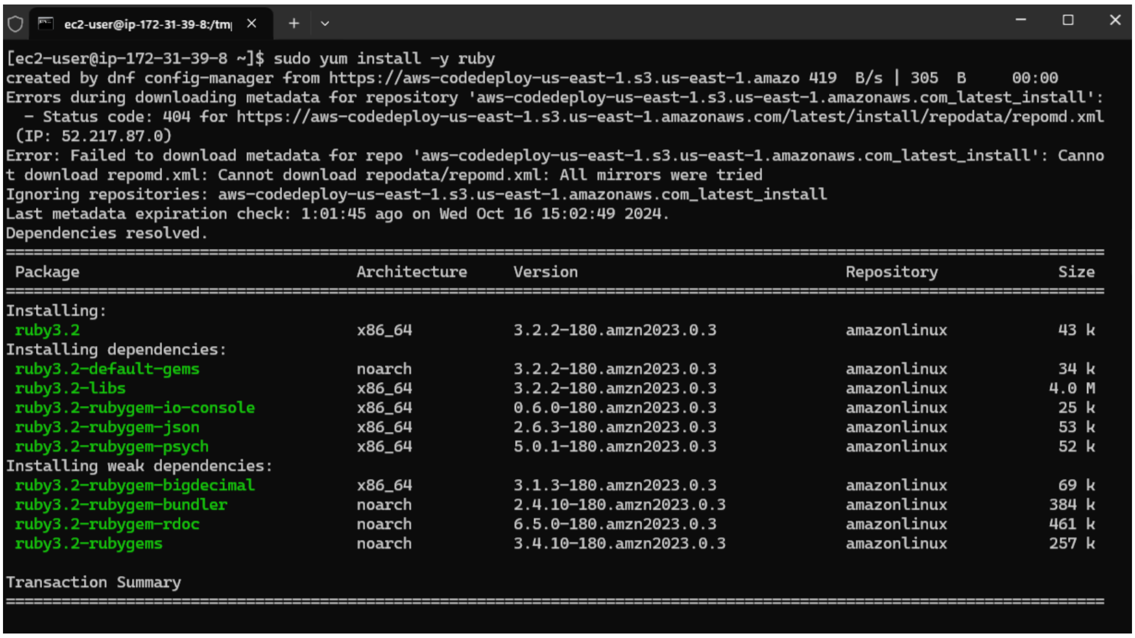
Show remaining 422 services

| Service | Access level | Resource | Request condition |
|---|---|---|---|
| S3 | Limited: Read | Multiple | None |

## Step 8: Install codedeploy-agent and nginx on ec2

Refer to the documentation to install https://docs.aws.amazon.com/codedeploy/latest/userguide/codedeploy-agent-operations-install-linux.html

sudo yum install -y ruby

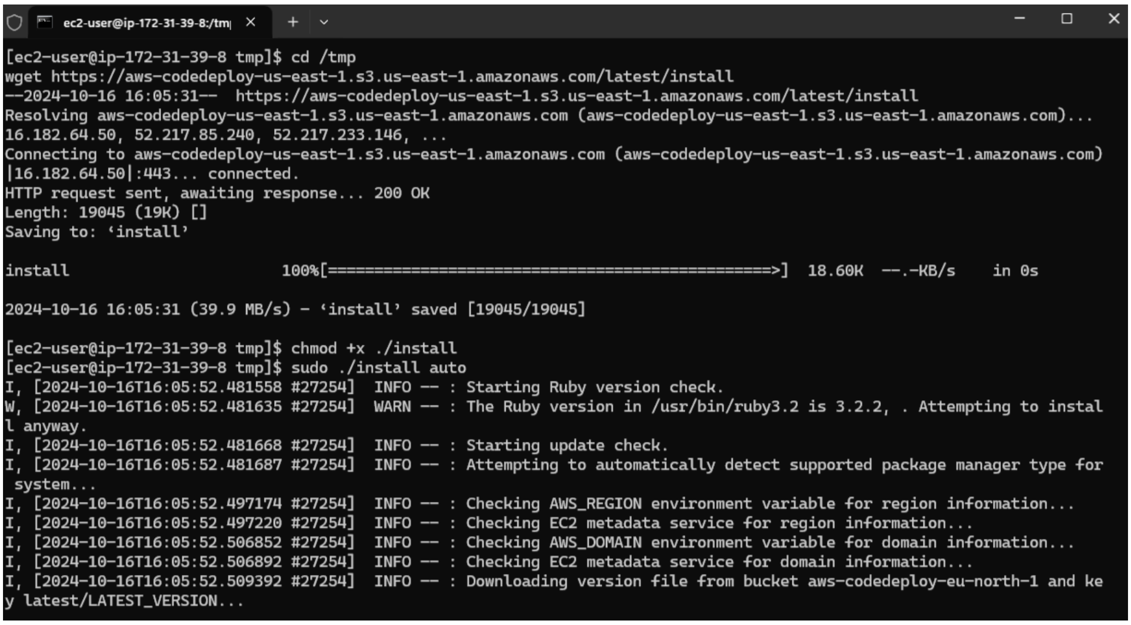


```
[ec2-user@ip-172-31-39-8 ~]$ sudo yum install -y ruby
created by dnf config-manager from https://aws-codedeploy-us-east-1.s3.us-east-1.amazo 419  B/s | 305  B     00:00
Errors during downloading metadata for repository 'aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install':
  - Status code: 404 for https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install/repodata/repomd.xml
 (IP: 52.217.87.0)
Error: Failed to download metadata for repo 'aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install': Canno
t download repomd.xml: Cannot download repodata/repomd.xml: All mirrors were tried
Ignoring repositories: aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install
Last metadata expiration check: 1:01:45 ago on Wed Oct 16 15:02:49 2024.
Dependencies resolved.
========================================================================================================================
 Package                              Architecture     Version                              Repository             Size
========================================================================================================================
Installing:
 ruby3.2                              x86_64           3.2.2-180.amzn2023.0.3               amazonlinux            43 k
Installing dependencies:
 ruby3.2-default-gems                 noarch           3.2.2-180.amzn2023.0.3               amazonlinux            34 k
 ruby3.2-libs                         x86_64           3.2.2-180.amzn2023.0.3               amazonlinux           4.0 M
 ruby3.2-rubygem-io-console           x86_64           0.6.0-180.amzn2023.0.3               amazonlinux            25 k
 ruby3.2-rubygem-json                 x86_64           2.6.3-180.amzn2023.0.3               amazonlinux            53 k
 ruby3.2-rubygem-psych                x86_64           5.0.1-180.amzn2023.0.3               amazonlinux            52 k
Installing weak dependencies:
 ruby3.2-rubygem-bigdecimal           x86_64           3.1.3-180.amzn2023.0.3               amazonlinux            69 k
 ruby3.2-rubygem-bundler              noarch           2.4.10-180.amzn2023.0.3              amazonlinux           384 k
 ruby3.2-rubygem-rdoc                 noarch           6.5.0-180.amzn2023.0.3               amazonlinux           461 k
 ruby3.2-rubygems                     noarch           3.4.10-180.amzn2023.0.3              amazonlinux           257 k

Transaction Summary
========================================================================================================================
```

cd /tmp

(wget https://**bucket-name**.s3.**region-identifier**.amazonaws.com/latest/install)

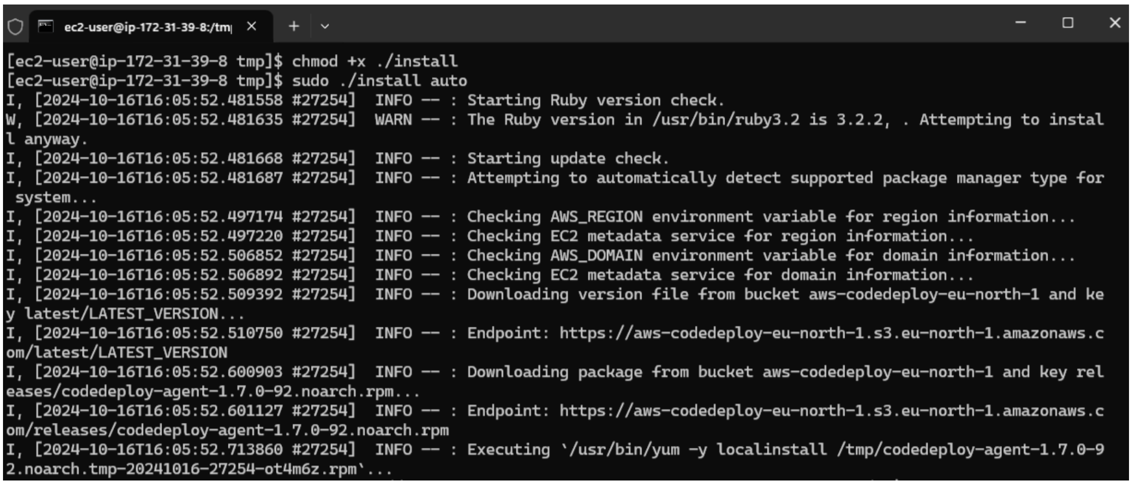
wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install



```
[ec2-user@ip-172-31-39-8 tmp]$ cd /tmp
wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
--2024-10-16 16:05:31--  https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
Resolving aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com)...
16.182.64.50, 52.217.85.240, 52.217.233.146, ...
Connecting to aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com)
|16.182.64.50|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 19045 (19K) []
Saving to: ‘install’

install                       100%[===============================================>]  18.60K  --.-KB/s    in 0s

2024-10-16 16:05:31 (39.9 MB/s) - ‘install’ saved [19045/19045]

[ec2-user@ip-172-31-39-8 tmp]$ chmod +x ./install
[ec2-user@ip-172-31-39-8 tmp]$ sudo ./install auto
I, [2024-10-16T16:05:52.481558 #27254]  INFO -- : Starting Ruby version check.
W, [2024-10-16T16:05:52.481635 #27254]  WARN -- : The Ruby version in /usr/bin/ruby3.2 is 3.2.2, . Attempting to instal
l anyway.
I, [2024-10-16T16:05:52.481668 #27254]  INFO -- : Starting update check.
I, [2024-10-16T16:05:52.481687 #27254]  INFO -- : Attempting to automatically detect supported package manager type for
 system...
I, [2024-10-16T16:05:52.497174 #27254]  INFO -- : Checking AWS_REGION environment variable for region information...
I, [2024-10-16T16:05:52.497220 #27254]  INFO -- : Checking EC2 metadata service for region information...
I, [2024-10-16T16:05:52.506852 #27254]  INFO -- : Checking AWS_DOMAIN environment variable for domain information...
I, [2024-10-16T16:05:52.506892 #27254]  INFO -- : Checking EC2 metadata service for domain information...
I, [2024-10-16T16:05:52.509392 #27254]  INFO -- : Downloading version file from bucket aws-codedeploy-eu-north-1 and ke
y latest/LATEST_VERSION...
```

chmod +x ./install

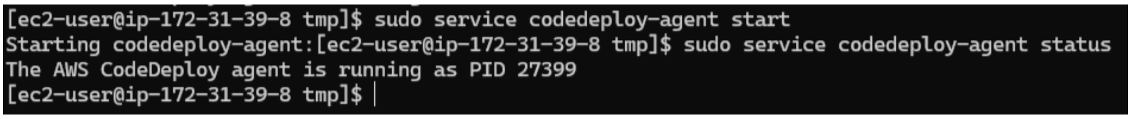
sudo ./install auto



```
[ec2-user@ip-172-31-39-8 tmp]$ chmod +x ./install
[ec2-user@ip-172-31-39-8 tmp]$ sudo ./install auto
I, [2024-10-16T16:05:52.481558 #27254]  INFO -- : Starting Ruby version check.
W, [2024-10-16T16:05:52.481635 #27254]  WARN -- : The Ruby version in /usr/bin/ruby3.2 is 3.2.2, . Attempting to instal
l anyway.
I, [2024-10-16T16:05:52.481668 #27254]  INFO -- : Starting update check.
I, [2024-10-16T16:05:52.481687 #27254]  INFO -- : Attempting to automatically detect supported package manager type for
 system...
I, [2024-10-16T16:05:52.497174 #27254]  INFO -- : Checking AWS_REGION environment variable for region information...
I, [2024-10-16T16:05:52.497220 #27254]  INFO -- : Checking EC2 metadata service for region information...
I, [2024-10-16T16:05:52.506852 #27254]  INFO -- : Checking AWS_DOMAIN environment variable for domain information...
I, [2024-10-16T16:05:52.506892 #27254]  INFO -- : Checking EC2 metadata service for domain information...
I, [2024-10-16T16:05:52.509392 #27254]  INFO -- : Downloading version file from bucket aws-codedeploy-eu-north-1 and ke
y latest/LATEST_VERSION...
I, [2024-10-16T16:05:52.510750 #27254]  INFO -- : Endpoint: https://aws-codedeploy-eu-north-1.s3.eu-north-1.amazonaws.c
om/latest/LATEST_VERSION
I, [2024-10-16T16:05:52.600903 #27254]  INFO -- : Downloading package from bucket aws-codedeploy-eu-north-1 and key rel
eases/codedeploy-agent-1.7.0-92.noarch.rpm...
I, [2024-10-16T16:05:52.601127 #27254]  INFO -- : Endpoint: https://aws-codedeploy-eu-north-1.s3.eu-north-1.amazonaws.c
om/releases/codedeploy-agent-1.7.0-92.noarch.rpm
I, [2024-10-16T16:05:52.713860 #27254]  INFO -- : Executing `/usr/bin/yum -y localinstall /tmp/codedeploy-agent-1.7.0-9
2.noarch.tmp-20241016-27254-ot4m6z.rpm`...
```

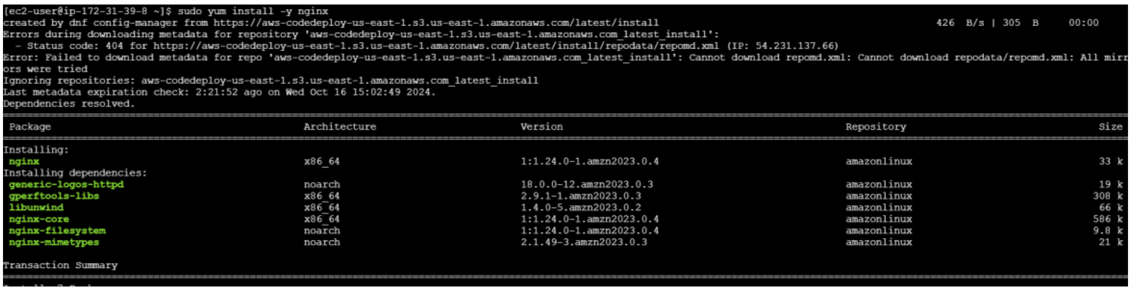
sudo service codedeploy-agent start

sudo service codedeploy-agent status



```
[ec2-user@ip-172-31-39-8 tmp]$ sudo service codedeploy-agent start
Starting codedeploy-agent:[ec2-user@ip-172-31-39-8 tmp]$ sudo service codedeploy-agent status
The AWS CodeDeploy agent is running as PID 27399
[ec2-user@ip-172-31-39-8 tmp]$ |
```

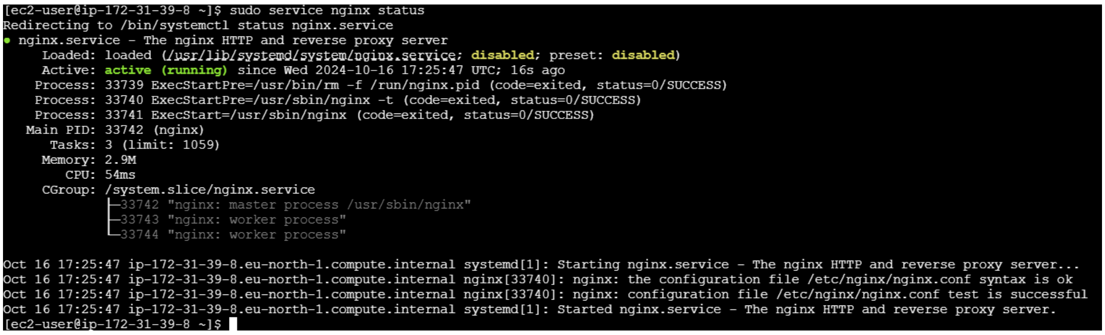
sudo yum install -y nginx



```
[ec2-user@ip-172-31-39-8 ~]$ sudo yum install -y nginx
created by dnf config-manager from https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install                   426  B/s | 305  B     00:00
Errors during downloading metadata for repository 'aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install':
  - Status code: 404 for https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install/repodata/repomd.xml (IP: 54.231.137.66)
Error: Failed to download metadata for repo 'aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install': Cannot download repomd.xml: Cannot download repodata/repomd.xml: All mirr
ors were tried
Ignoring repositories: aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com_latest_install
Last metadata expiration check: 2:21:52 ago on Wed Oct 16 15:02:49 2024.
Dependencies resolved.
==============================================================================================================================================================================================
 Package                                         Architecture                         Version                                                  Repository                                Size
==============================================================================================================================================================================================
Installing:
 nginx                                           x86_64                               1:1.24.0-1.amzn2023.0.4                                  amazonlinux                               33 k
Installing dependencies:
 generic-logos-httpd                             noarch                               18.0.0-12.amzn2023.0.3                                   amazonlinux                               19 k
 gperftools-libs                                 x86_64                               2.9.1-1.amzn2023.0.3                                     amazonlinux                              308 k
 libunwind                                       x86_64                               1.4.0-5.amzn2023.0.2                                     amazonlinux                               66 k
 nginx-core                                      x86_64                               1:1.24.0-1.amzn2023.0.4                                  amazonlinux                              586 k
 nginx-filesystem                                noarch                               1:1.24.0-1.amzn2023.0.4                                  amazonlinux                              9.8 k
 nginx-mimetypes                                 noarch                               2.1.49-3.amzn2023.0.3                                    amazonlinux                               21 k

Transaction Summary
==============================================================================================================================================================================================
```

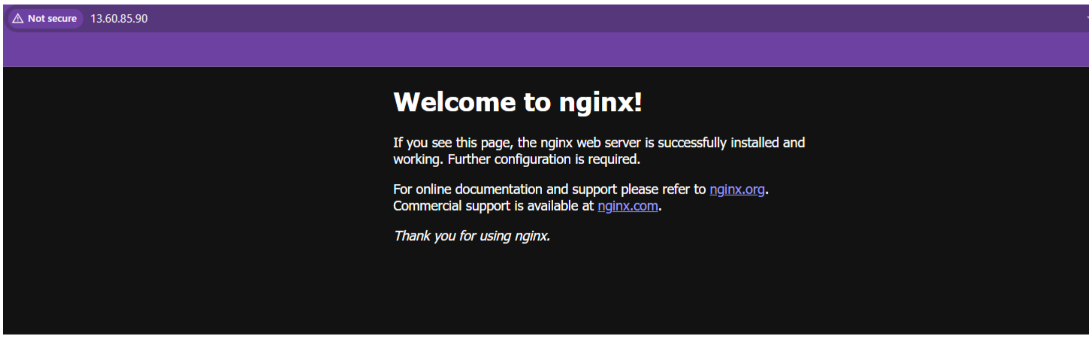
sudo service nginx start

sudo service nginx status



```
[ec2-user@ip-172-31-39-8 ~]$ sudo service nginx status
Redirecting to /bin/systemctl status nginx.service
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; disabled; preset: disabled)
     Active: active (running) since Wed 2024-10-16 17:25:47 UTC; 16s ago
    Process: 33739 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 33740 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 33741 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 33742 (nginx)
      Tasks: 3 (limit: 1059)
     Memory: 2.9M
        CPU: 54ms
     CGroup: /system.slice/nginx.service
             ├─33742 "nginx: master process /usr/sbin/nginx"
             ├─33743 "nginx: worker process"
             └─33744 "nginx: worker process"

Oct 16 17:25:47 ip-172-31-39-8.eu-north-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Oct 16 17:25:47 ip-172-31-39-8.eu-north-1.compute.internal nginx[33740]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Oct 16 17:25:47 ip-172-31-39-8.eu-north-1.compute.internal nginx[33740]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Oct 16 17:25:47 ip-172-31-39-8.eu-north-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[ec2-user@ip-172-31-39-8 ~]$
```

Check if nginx installed by opening the instance's public ipv4 ip



**Welcome to nginx!**

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

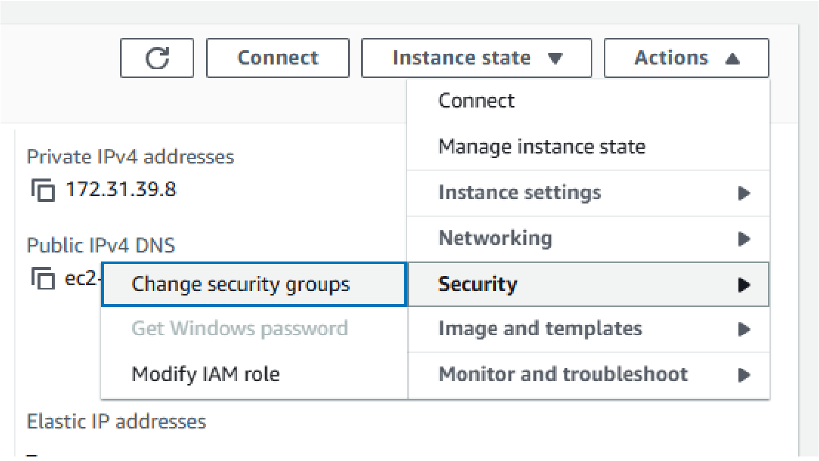
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

*Thank you for using nginx.*

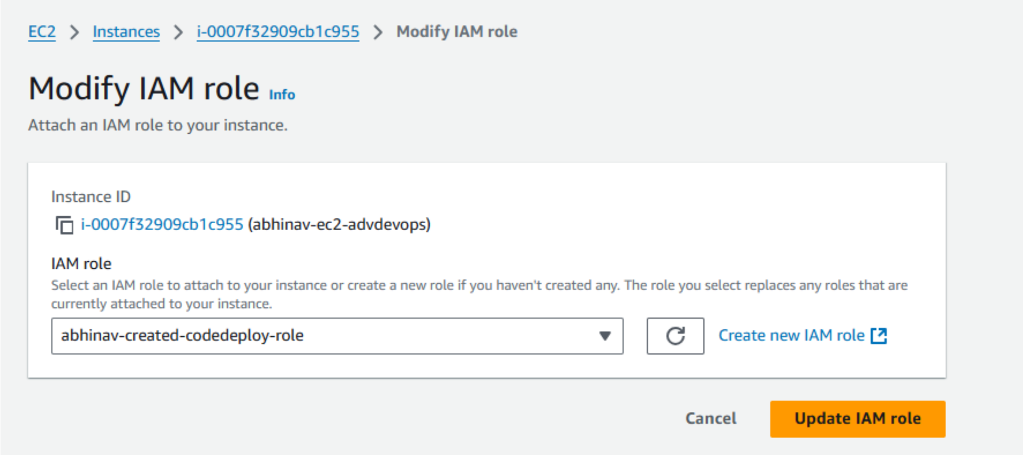
## Step 9: Attach the IAM Role to the EC2 Instance

Go back to the EC2 Dashboard.

Select your Instance and click on Actions > Security > Modify IAM Role.



**Attach the IAM Role** created earlier.



Modify IAM role Info

Attach an IAM role to your instance.

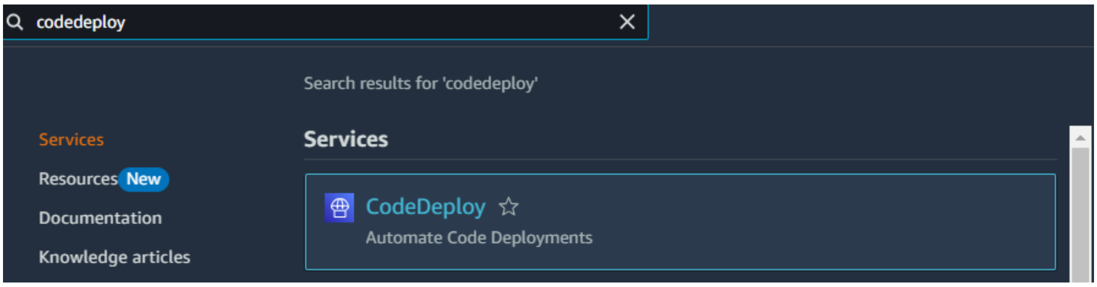
Instance ID
i-0007f32909cb1c955 (abhinav-ec2-advdevops)

IAM role
Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.

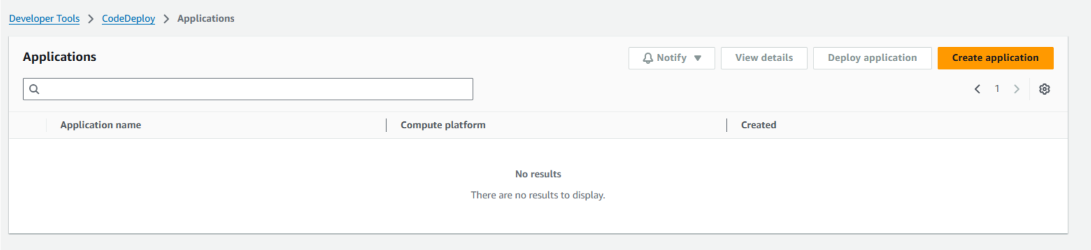
abhinav-created-codedeploy-role

## Step 10: Set Up AWS CodeDeploy

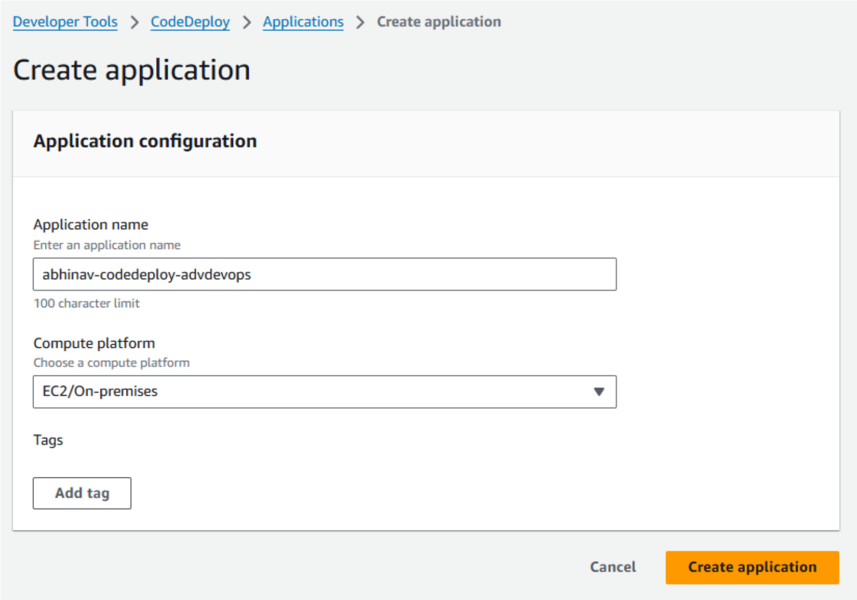
**Open the CodeDeploy Console**: In the AWS Management Console, navigate to the **CodeDeploy** service.



Click on **"Create application"**.



Enter an application name , For Compute platform, select EC2/On-premises. Click "Create application"


Developer Tools > CodeDeploy > Applications > Create application

Create application

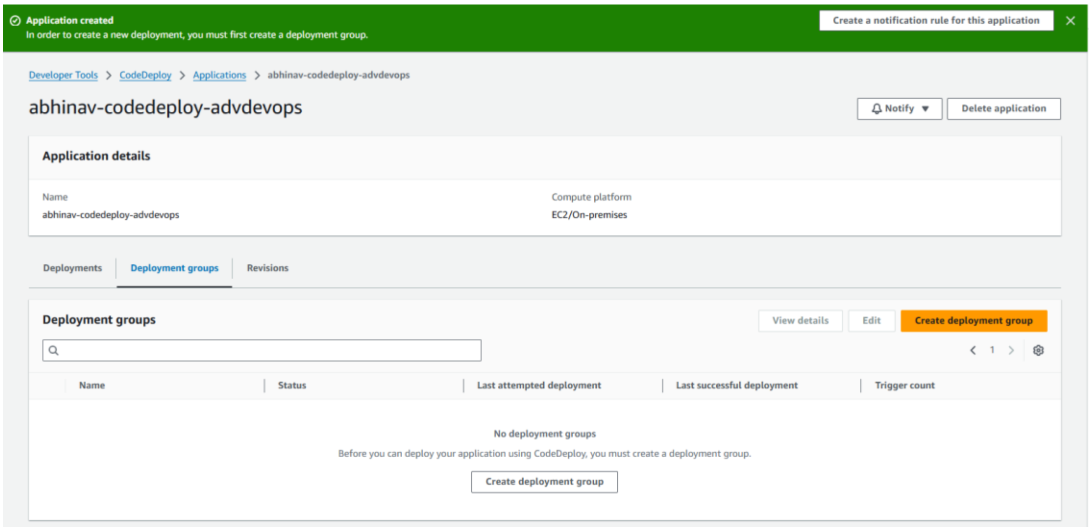
Application configuration

Application name
Enter an application name
abhinav-codedeploy-advdevops
100 character limit

Compute platform
Choose a compute platform
EC2/On-premises

Tags
Add tag
Cancel
Create application

After creating the application, you will be prompted to create a deployment group. Click on "Create deployment group".

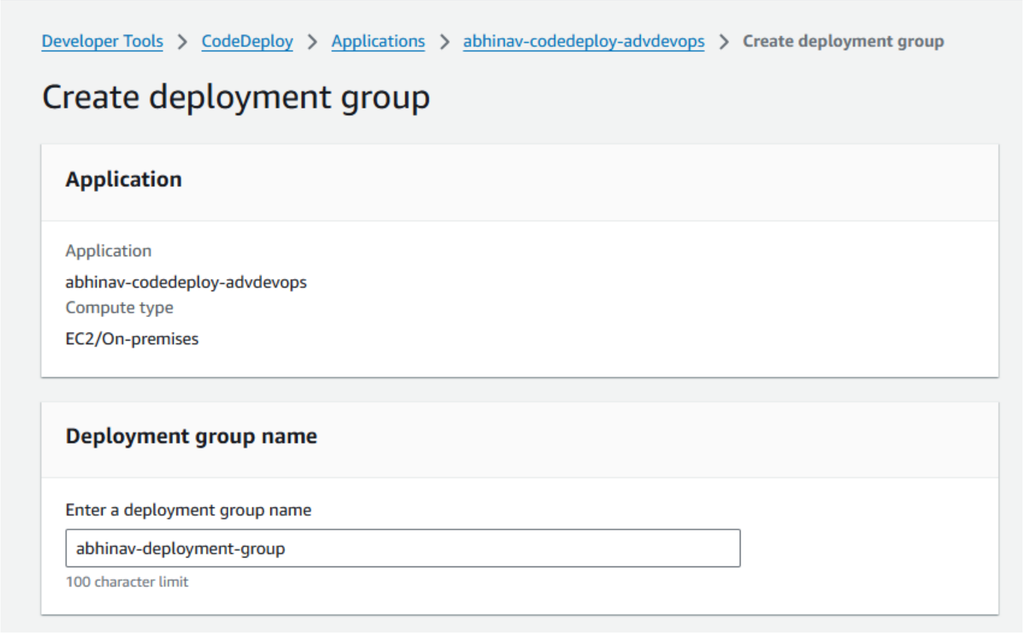

Application created
In order to create a new deployment, you must first create a deployment group.
Create a notification rule for this application
Developer Tools > CodeDeploy > Applications > abhinav-codedeploy-advdevops
abhinav-codedeploy-advdevops
Notify
Delete application

Application details

Name
abhinav-codedeploy-advdevops

Compute platform
EC2/On-premises

Deployments
Deployment groups
Revisions
Deployment groups
View details
Edit
Create deployment group
Name
Status
Last attempted deployment
Last successful deployment
Trigger count
No deployment groups
Before you can deploy your application using CodeDeploy, you must create a deployment group.
Create deployment group



Developer Tools > CodeDeploy > Applications > abhinav-codedeploy-advdevops > Create deployment group

Create deployment group

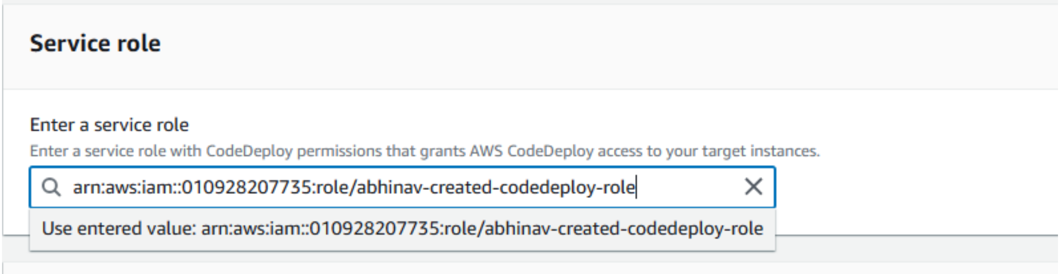
Application

Application
abhinav-codedeploy-advdevops
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
abhinav-deployment-group
100 character limit

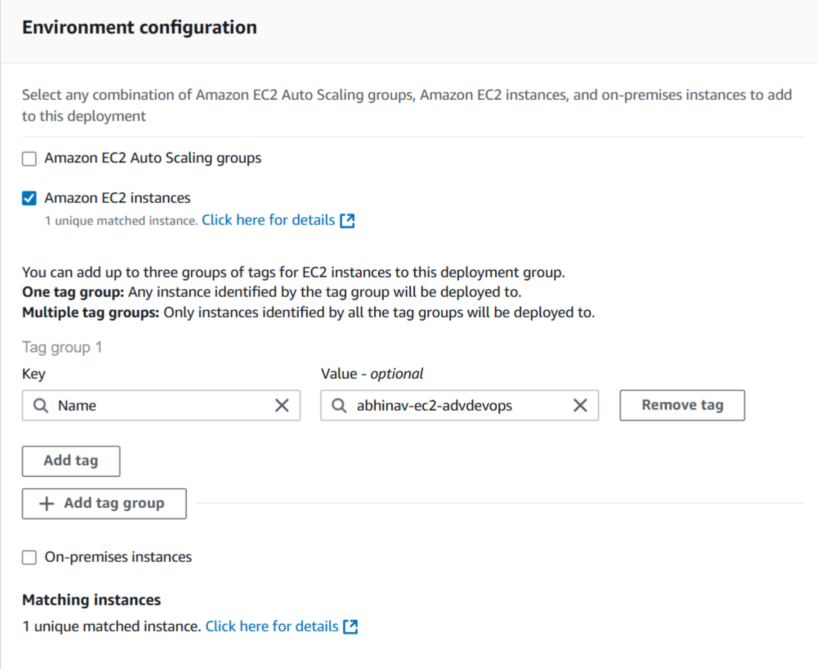
Enter the **ARN** of the service role you created for CodeDeploy (the one with permissions for CodeDeploy).
**arn:aws:iam::010928207735:role/abhinav-created-codedeploy-role**



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::010928207735:role/abhinav-created-codedeploy-role
Use entered value: arn:aws:iam::010928207735:role/abhinav-created-codedeploy-role

For **Environment configuration**, choose **EC2 instances**. Enter your key value pair which was noted from under tags



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.

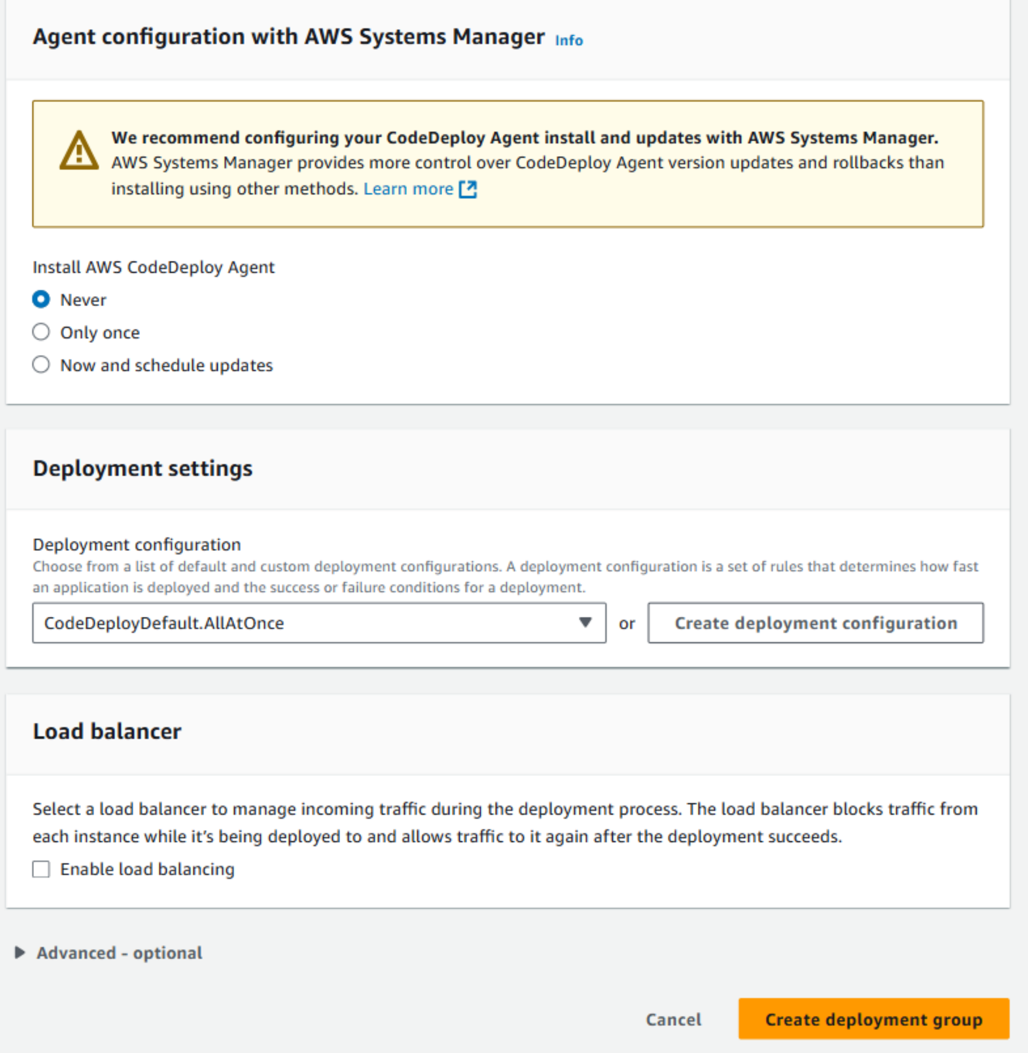
Tag group 1

| Key | Value - *optional* | |
|---|---|---|
| Name | abhinav-ec2-advdevops | Remove tag |

☐ On-premises instances

Matching instances
1 unique matched instance. Click here for details

Disable load balancing.



**Agent configuration with AWS Systems Manager** Info

**We recommend configuring your CodeDeploy Agent install and updates with AWS Systems Manager.**
AWS Systems Manager provides more control over CodeDeploy Agent version updates and rollbacks than installing using other methods. Learn more

Install AWS CodeDeploy Agent
- ● Never
- ○ Only once
- ○ Now and schedule updates

**Deployment settings**

Deployment configuration
Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast an application is deployed and the success or failure conditions for a deployment.

CodeDeployDefault.AllAtOnce ▼ or Create deployment configuration

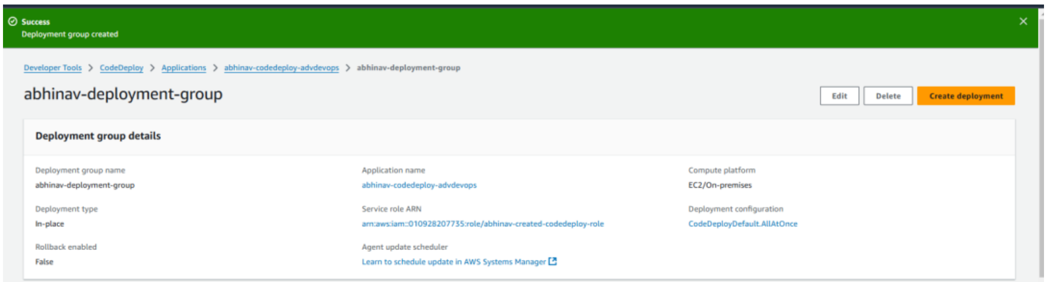
**Load balancer**

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from each instance while it's being deployed to and allows traffic to it again after the deployment succeeds.

□ Enable load balancing

▶ Advanced - optional

Create deployment group



**Success**
Deployment group created

Developer Tools > CodeDeploy > Applications > abhinav-codedeploy-advdevops > abhinav-deployment-group

abhinav-deployment-group

**Deployment group details**

Deployment group name
abhinav-deployment-group

Application name
abhinav-codedeploy-advdevops

Compute platform
EC2/On-premises

Deployment type
In-place

Service role ARN
arn:aws:iam::010928207735:role/abhinav-created-codedeploy-role

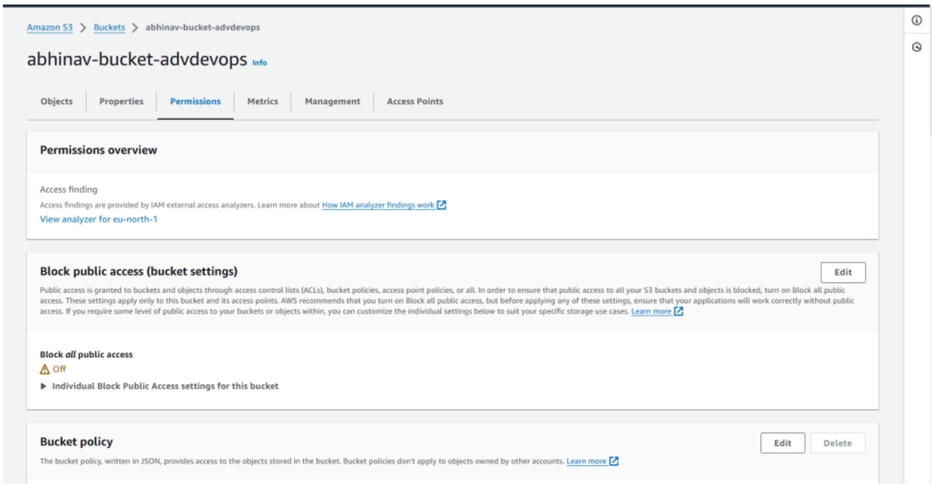
Deployment configuration
CodeDeployDefault.AllAtOnce

Rollback enabled
False

Agent update scheduler
Learn to schedule update in AWS Systems Manager

## Step 11: Give access of S3 bucket to codedeploy

Navigate to the **S3 service** in the AWS Management Console.
Find and select your bucket (e.g., abhinav-bucket-advdevops).
Go to the **Permissions** tab and check the **Bucket Policy**.



Amazon S3 > Buckets > abhinav-bucket-advdevops
abhinav-bucket-advdevops Info
Objects
Properties
Permissions
Metrics
Management
Access Points

**Permissions overview**

Access finding

Access findings are provided by IAM external access analyzers. Learn more about How IAM analyzer findings work

View analyzer for eu-north-1

**Block public access (bucket settings)**

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 all your S3 buckets and 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 your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

**Block *all* public access**

⚠ Off

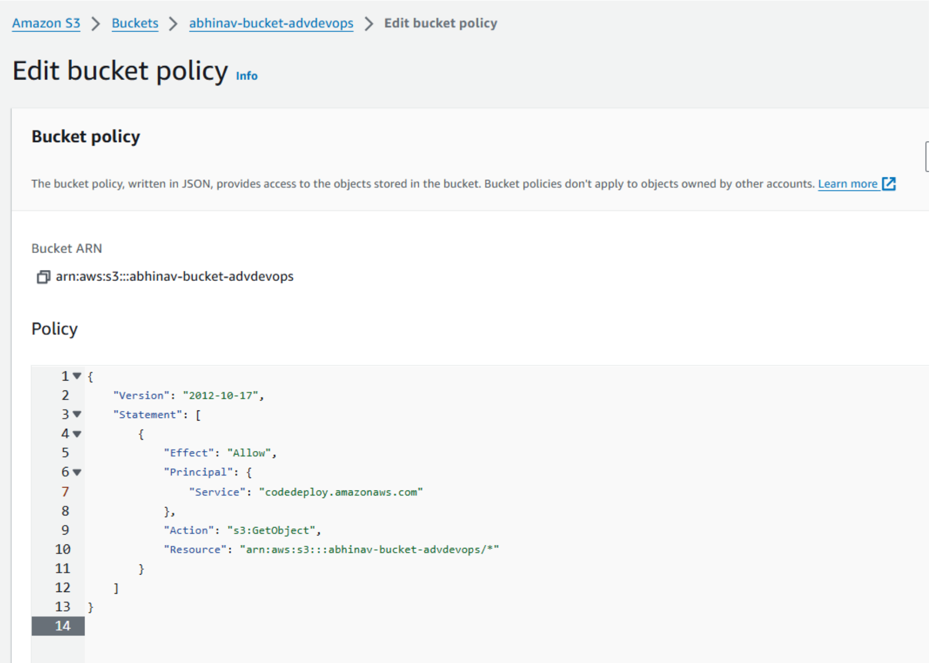
▸ Individual Block Public Access settings for this bucket

**Bucket policy**

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

**Go to the Permissions tab and check the Bucket Policy.**
**Ensure that the policy allows CodeDeploy to access the bucket**

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": {
            "Service": "codedeploy.amazonaws.com"
         },
         "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::{your_bucket_name}/*"
      }
   ]
}
```



Amazon S3 > Buckets > abhinav-bucket-advdevops > Edit bucket policy

Edit bucket policy Info

**Bucket policy**

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

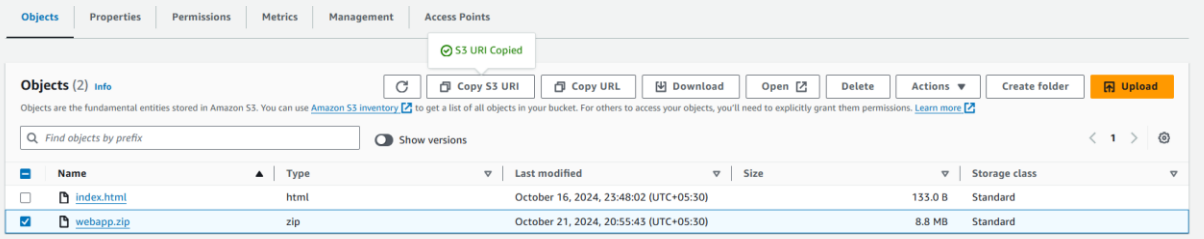
Bucket ARN

arn:aws:s3:::abhinav-bucket-advdevops

Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "codedeploy.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::abhinav-bucket-advdevops/*"
        }
    ]
}
```

**Copy the S3 bucket’s URI**

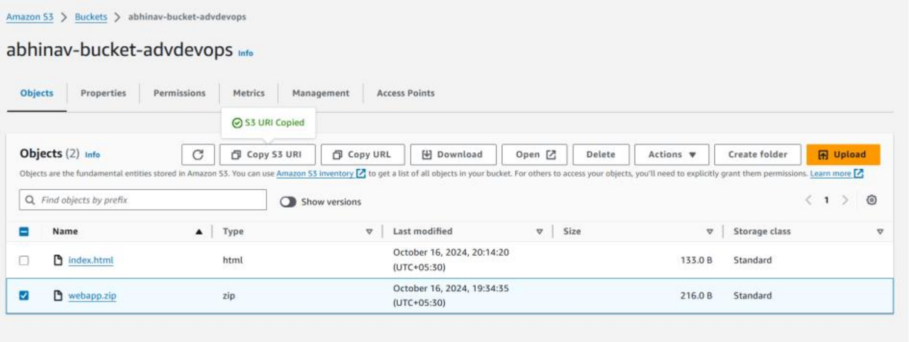


Objects (2) Info

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

Find objects by prefix · Show versions

| | Name | Type | Last modified | Size | Storage class |
|---|---|---|---|---|---|
| □ | index.html | html | October 16, 2024, 23:48:02 (UTC+05:30) | 133.0 B | Standard |
| ☑ | webapp.zip | zip | October 21, 2024, 20:55:43 (UTC+05:30) | 8.8 MB | Standard |



Amazon S3 > Buckets > abhinav-bucket-advdevops

abhinav-bucket-advdevops Info

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

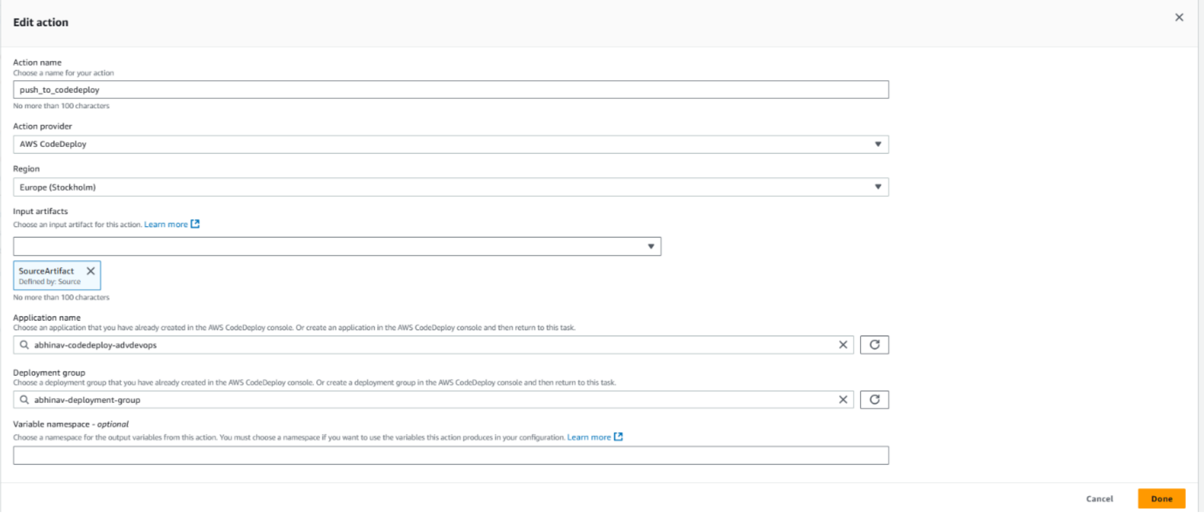
Find objects by prefix · Show versions

| | Name | Type | Last modified | Size | Storage class |
|---|---|---|---|---|---|
| □ | index.html | html | October 16, 2024, 20:14:20 (UTC+05:30) | 133.0 B | Standard |
| ☑ | webapp.zip | zip | October 16, 2024, 19:34:35 (UTC+05:30) | 216.0 B | Standard |

## Step 12: Automate CodeDeploy using CodePipeline

.

Under your CodePipeline > Deploy > Edit > Add an Action > Set Action Provider as "**AWS Codedeploy**" > configure our created application and deployment groups > **Save**



Edit action

Action name
Choose a name for your action
push_to_codedeploy
No more than 100 characters

Action provider
AWS CodeDeploy

Region
Europe (Stockholm)

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

Application name
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.
abhinav-codedeploy-advdevops

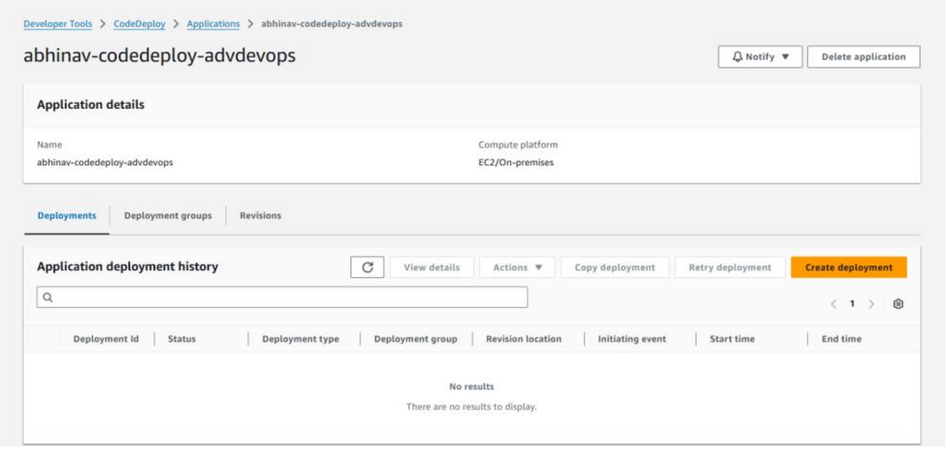
Deployment group
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.
abhinav-deployment-group

Variable namespace - *optional*
Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more

## Step 13: Create a Deployment for Your Application

Click on the Deployments tab.

Click on Create deployment.



Developer Tools > CodeDeploy > Applications > abhinav-codedeploy-advdevops

abhinav-codedeploy-advdevops

Application details

Name
abhinav-codedeploy-advdevops

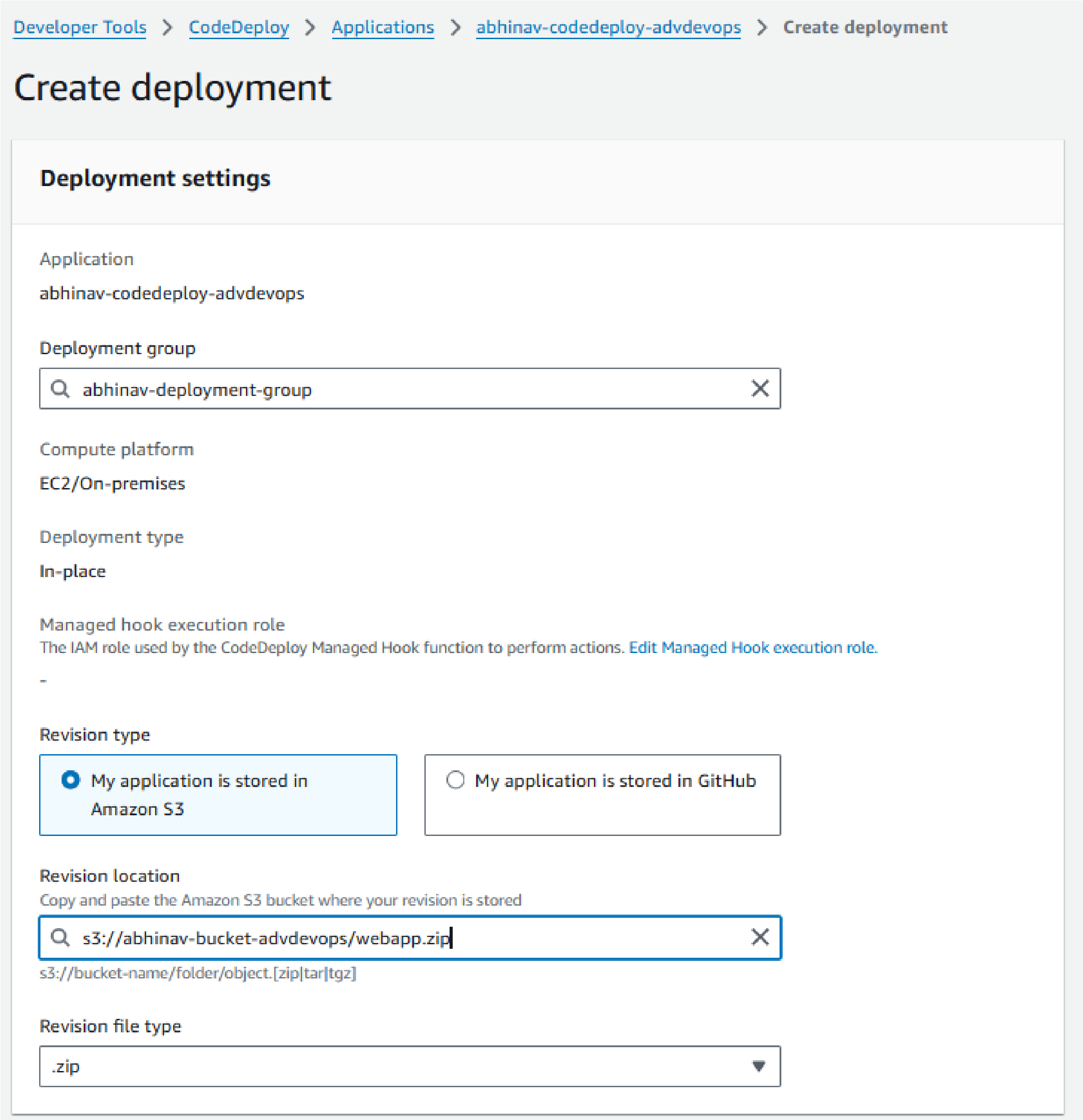
Compute platform
EC2/On-premises

Application deployment history

| Deployment Id | Status | Deployment type | Deployment group | Revision location | Initiating event | Start time | End time |
|---|---|---|---|---|---|---|---|

No results
There are no results to display.

Selet your deployment group and paste the URI you copied



Developer Tools > CodeDeploy > Applications > abhinav-codedeploy-advdevops > Create deployment

## Create deployment

### Deployment settings

Application
abhinav-codedeploy-advdevops

Deployment group
abhinav-deployment-group

Compute platform
EC2/On-premises

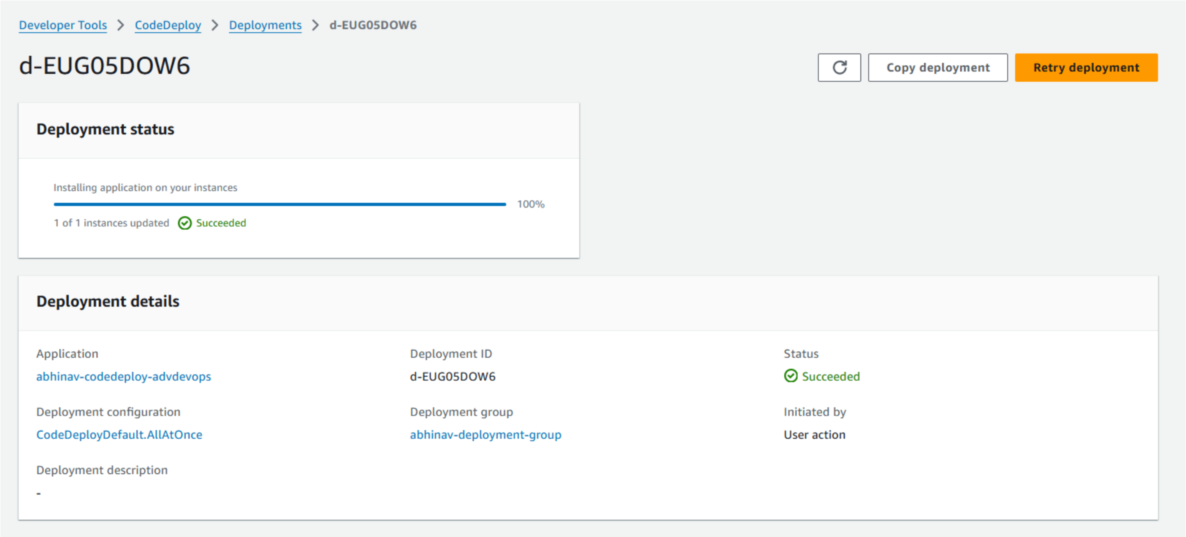
Deployment type
In-place

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://abhinav-bucket-advdevops/webapp.zip
s3://bucket-name/folder/object.[zip|tar|tgz]

Revision file type
.zip



Developer Tools > CodeDeploy > Deployments > d-EUG05DOW6

d-EUG05DOW6

**Deployment status**

Installing application on your instances
100%
1 of 1 instances updated ✓ Succeeded

**Deployment details**

Application
abhinav-codedeploy-advdevops

Deployment ID
d-EUG05DOW6

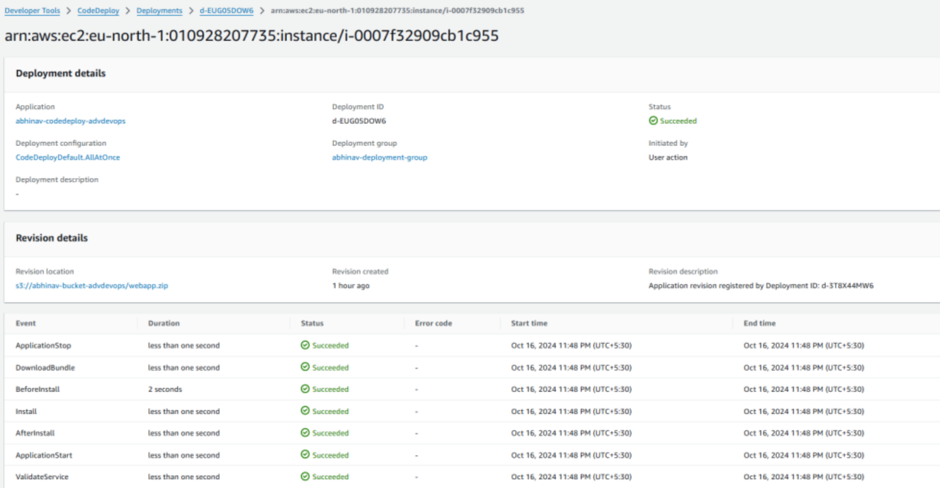
Status
✓ Succeeded

Deployment configuration
CodeDeployDefault.AllAtOnce

Deployment group
abhinav-deployment-group

Initiated by
User action

Deployment description
-



Developer Tools > CodeDeploy > Deployments > d-EUG05DOW6 > arn:aws:ec2:eu-north-1:010928207735:instance/i-0007f32909cb1c955

arn:aws:ec2:eu-north-1:010928207735:instance/i-0007f32909cb1c955

**Deployment details**

Application: abhinav-codedeploy-advdevops

Deployment ID: d-EUG05DOW6

Status: ✓ Succeeded

Deployment configuration: CodeDeployDefault.AllAtOnce

Deployment group: abhinav-deployment-group

Initiated by: User action

Deployment description: -

**Revision details**

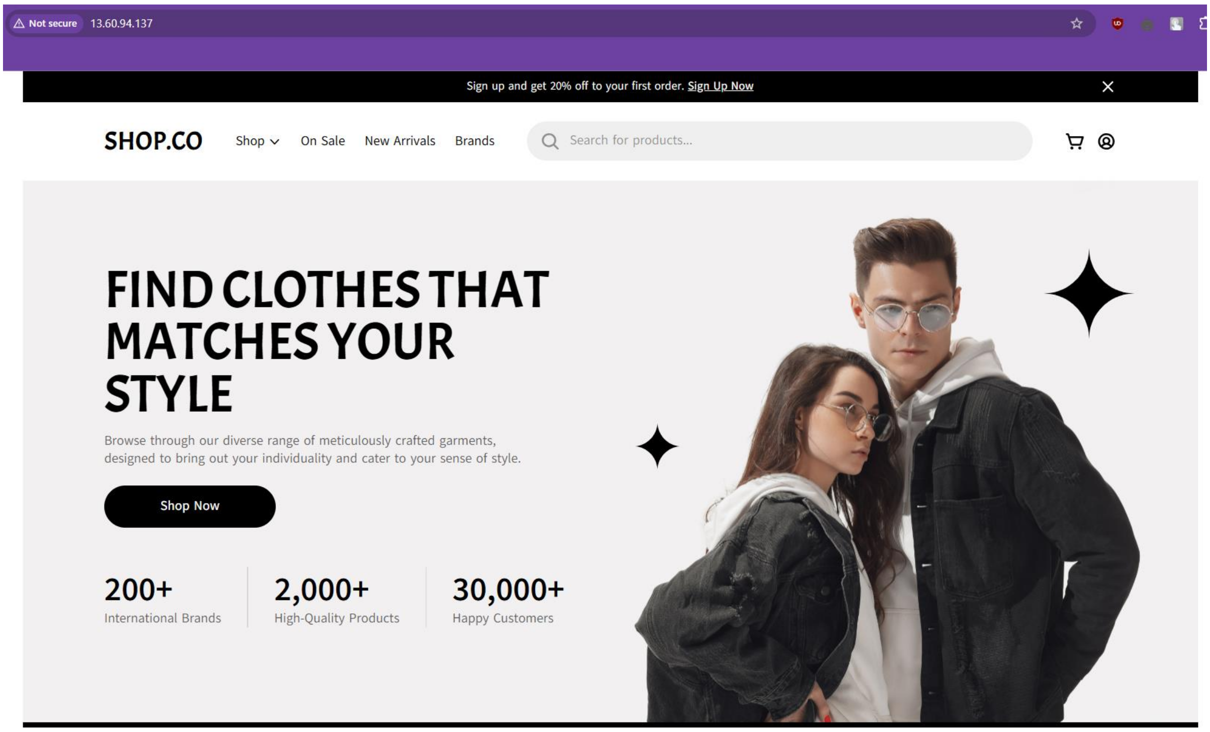
Revision location: s3://abhinav-bucket-advdevops/webapp.zip

Revision created: 1 hour ago

Revision description: Application revision registered by Deployment ID: d-3T8X44MW6

| Event | Duration | Status | Error code | Start time | End time |
|---|---|---|---|---|---|
| ApplicationStop | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| DownloadBundle | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| BeforeInstall | 2 seconds | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| Install | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| AfterInstall | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| ApplicationStart | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |
| ValidateService | less than one second | ✓ Succeeded | - | Oct 16, 2024 11:48 PM (UTC+5:30) | Oct 16, 2024 11:48 PM (UTC+5:30) |

Check if your page is hosted on your public ipv4 address of your EC2 bucket



Not secure 13.60.94.137

Sign up and get 20% off to your first order. Sign Up Now

FIND CLOTHES THAT MATCHES YOUR STYLE

Browse through our diverse range of meticulously crafted garments, designed to bring out your individuality and cater to your sense of style.

200+ International Brands

2,000+ High-Quality Products

30,000+ Happy Customers

## Conclusion

In this project, we successfully automated the deployment of a web application. The application was hosted on an EC2 instance and managed through S3 for storage. Codebuild created the Build Artifact of the project. Code Pipeline stored the source & build artifacts on an S3 bucket. It then handled automation of the project linking with AWS CodeDeploy which deployed the project to our EC2 instance. The instance had nginx setup as a web server which enabled rendering of the web application over the public IPV4 address.