Problem Statement:

Deploy a simple Nginx application using AWS code commit and deploy & access via browser

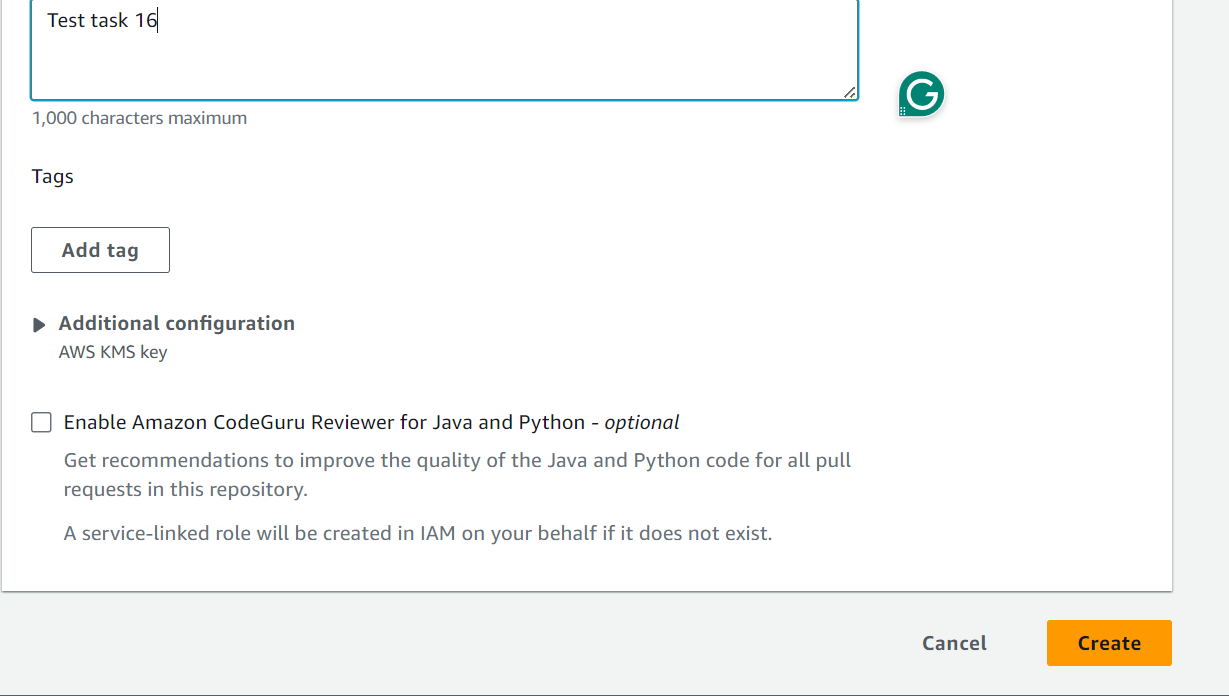
Sample index.html file:

<!DOCTYPE html>  
<html>  
<body>  
  
<h1>This is Guvi DevOps!!!</h1>  
<p>My first paragraph.</p>  
  
</body>  
</html>

The task is to deploy the above application in the EC2 machine using Code Commit, Code Build and Code Deploy.

Solution:

Create a repository in the code commit:



Add the index.html file in the repository by clicking either the Add files option or cloning the repository to the local and pushing the file to the repository

For the second method,

First, we need to have the index.html file in the local, then we must give the following commands,

git clone HTTPS\_URL,

git add . ,

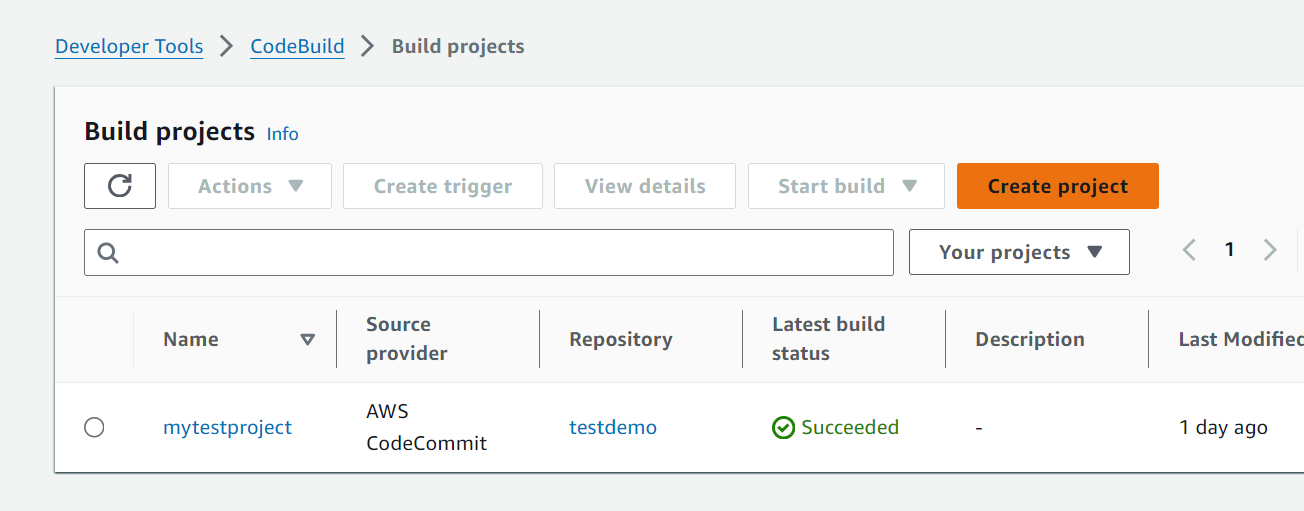
git commit -m “This is a Test Commit”,

git push,

Along with the index.html file we need to push 4 files, buildspec.yml, appspec.yml, install\_nginx.sh, start\_nginx.sh.

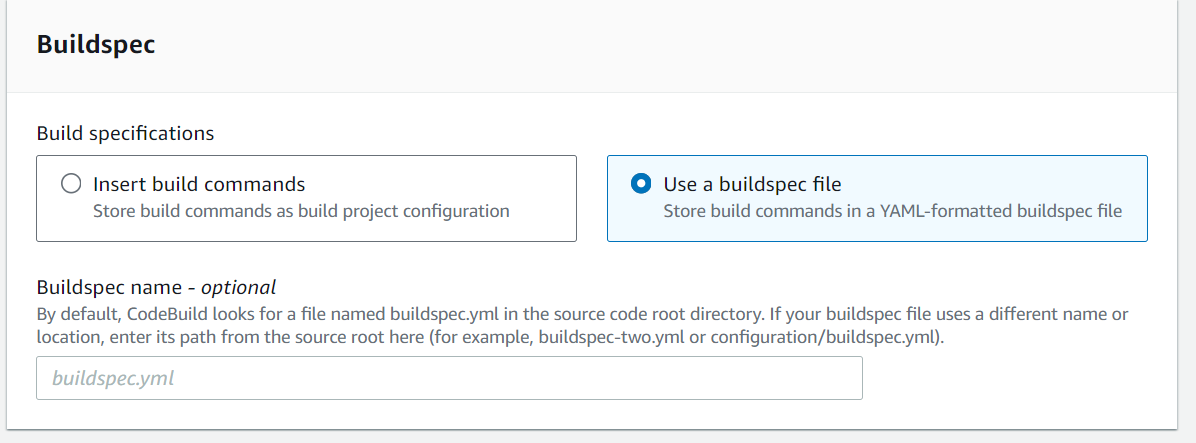
Buildspec file is used as configuration files during build phase.

Go to Code Build 🡪 Create Projects 🡪 Click on Create Project

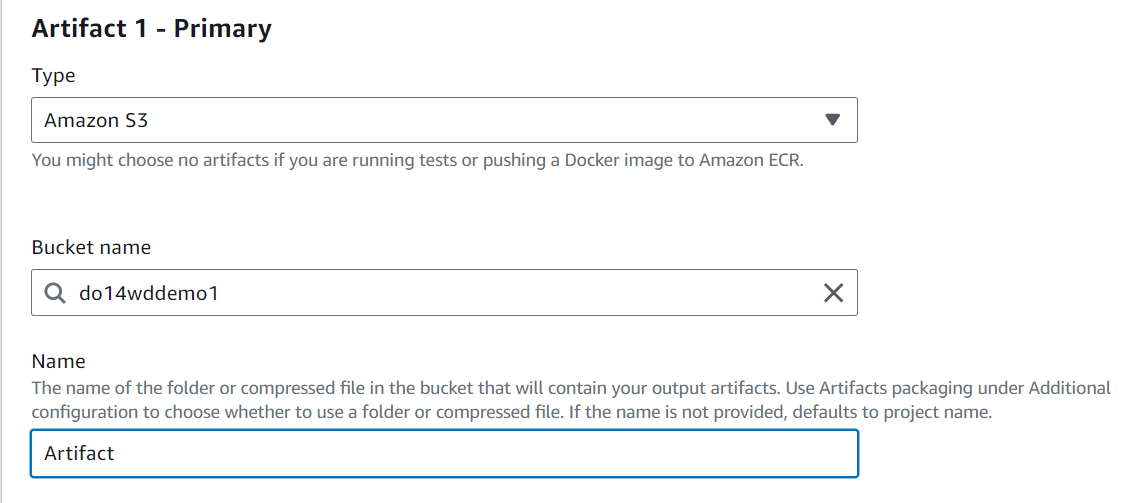


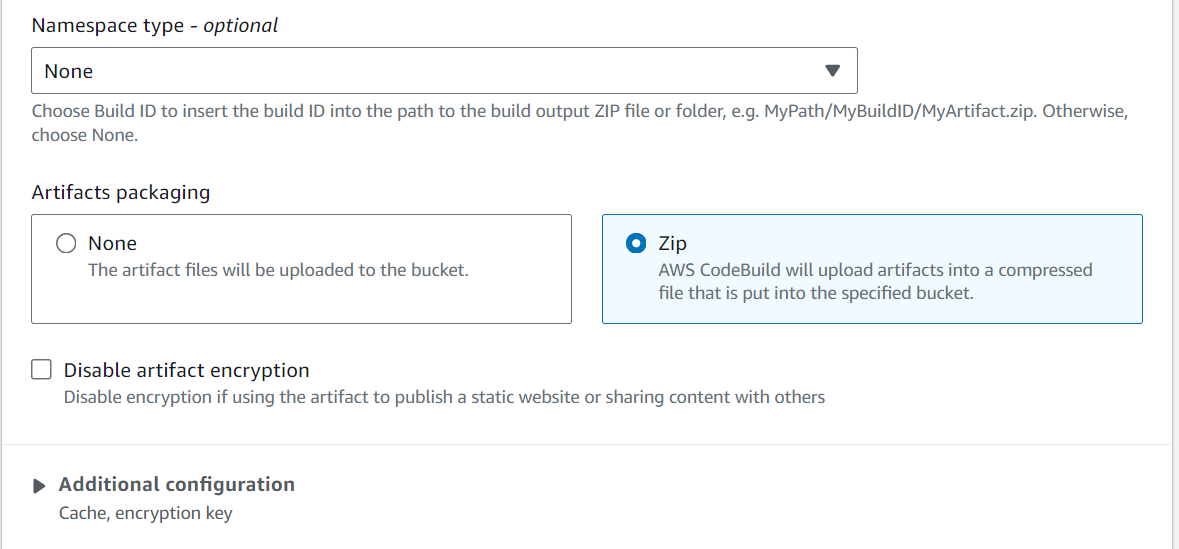
Select the Source as “Code Commit” 🡪 Provide the bucket name 🡪 Provide the branch

Select “Use a buildspec file” option on the Buildspec file

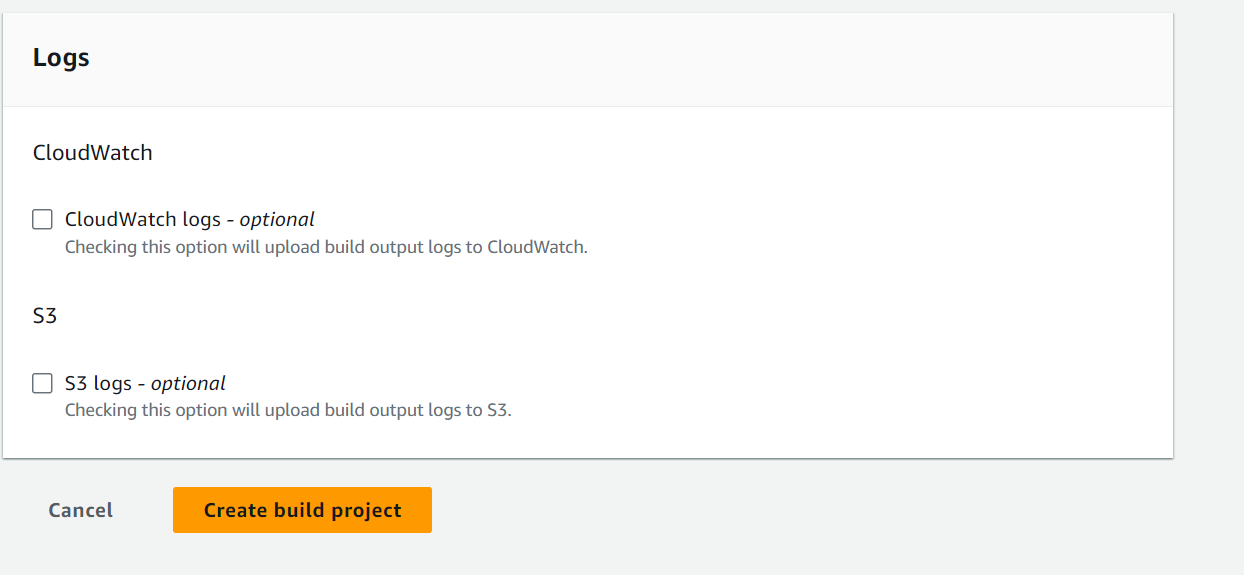


In the artifacts section, select the Amazon S3 and name it as Artifact and select the Zip option 🡪 this enables the build to store the artifact in the S3 bucket in the zip file.

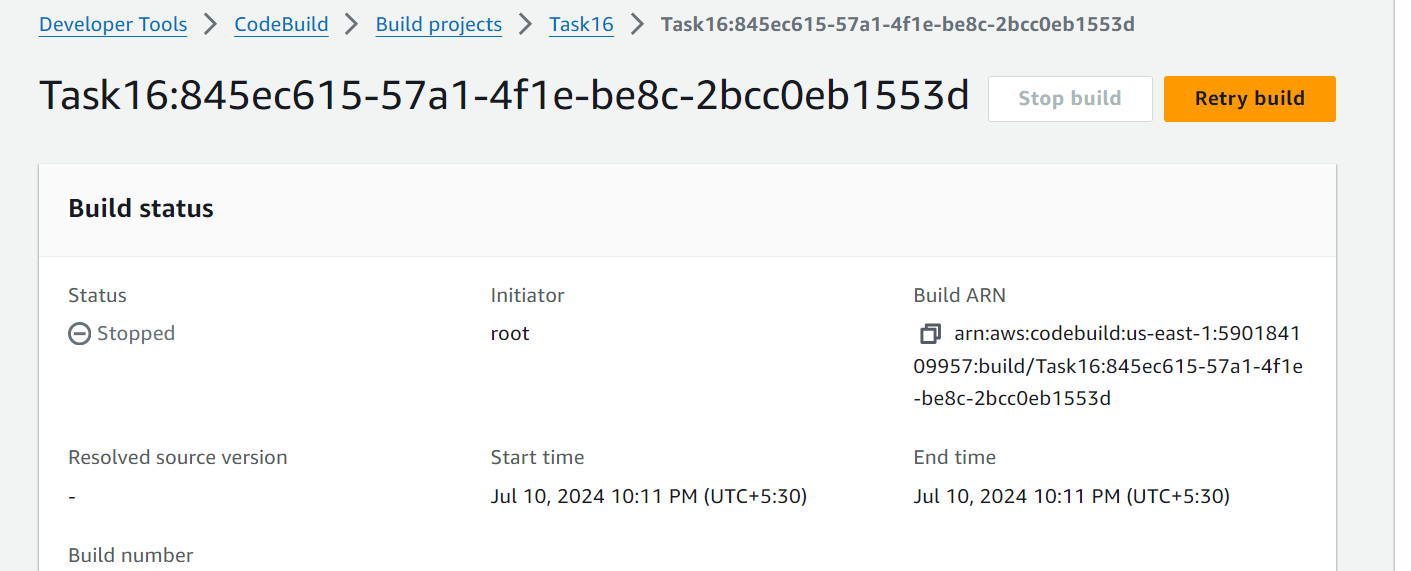




Click on the “Create Build Project”

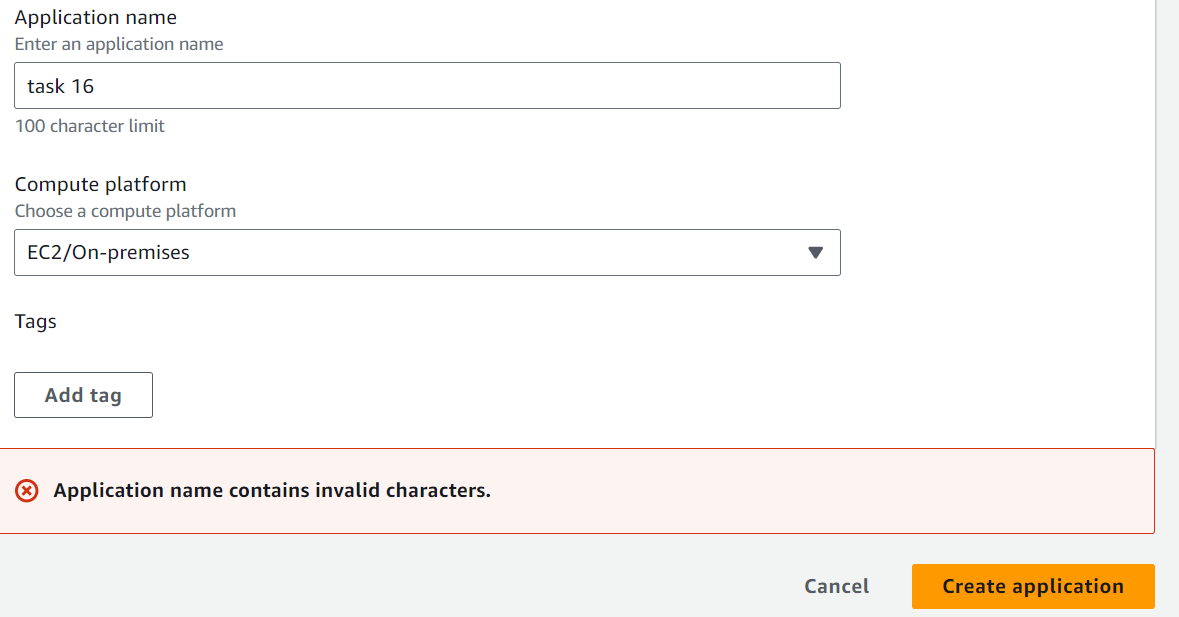


Click on Start build or Retry build to start the build process

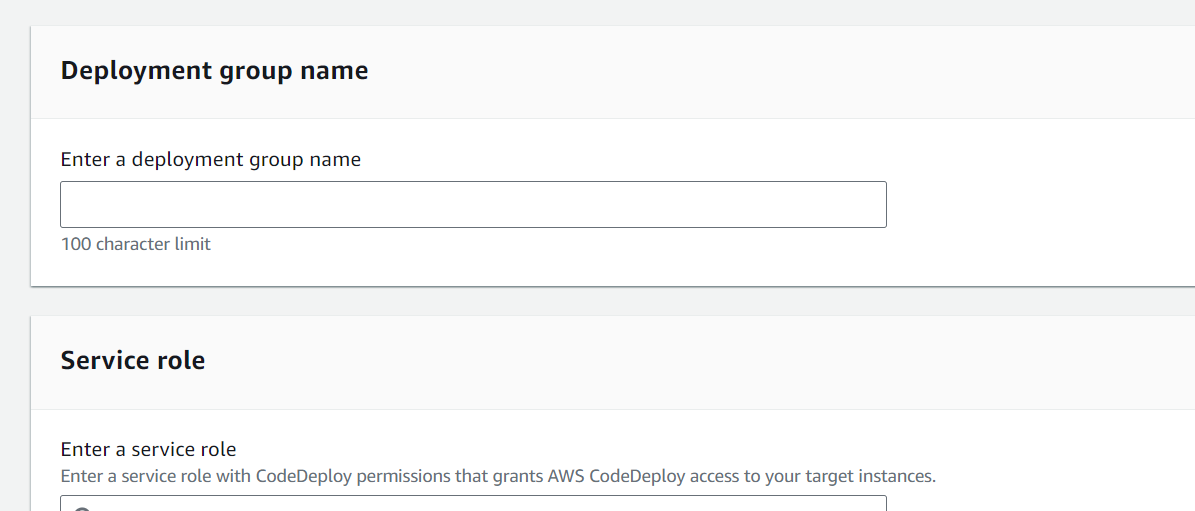


Now the build will be available in the S3 bucket to work

Go to Codedeploy 🡪 Create application



Now click on the Create Deployment group after creating the application, now name the deployment group and create a deployment group



Note that while selecting the Service role, the service role must be created with the following policies,

AmazonEC2fullaccess

AwsCodedeployfullaccess

AmazonS3fullaccess

AmazonEC2Roleforcodedeploy

AmazonEC2Roleforcodedeploylimited

Now select the EC2 instance as the service and map the ec2 machine here

Now we need to install the AWS Code Deploy agent in the selected EC2 instance with the following commands for Ubuntu 22.2 version

#!/bin/bash

# This installs the CodeDeploy agent and its prerequisites on Ubuntu 22.04.

sudo apt-get update

sudo apt-get install ruby-full ruby-webrick wget -y

cd /tmp

wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/releases/codedeploy-agent\_1.3.2-1902\_all.deb

mkdir codedeploy-agent\_1.3.2-1902\_ubuntu22

dpkg-deb -R codedeploy-agent\_1.3.2-1902\_all.deb codedeploy-agent\_1.3.2-1902\_ubuntu22

sed 's/Depends:.\*/Depends:ruby3.0/' -i ./codedeploy-agent\_1.3.2-1902\_ubuntu22/DEBIAN/control

dpkg-deb -b codedeploy-agent\_1.3.2-1902\_ubuntu22/

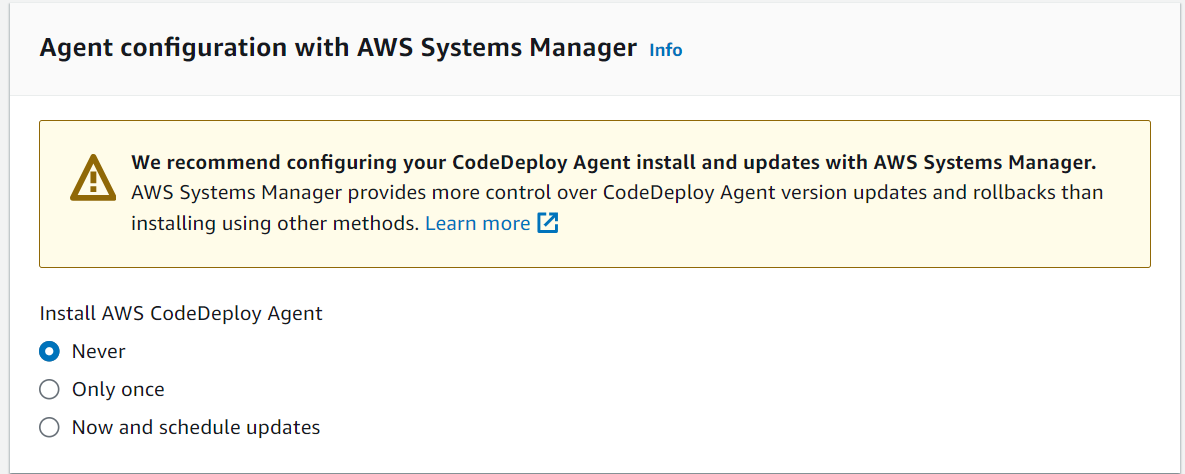
sudo dpkg -i codedeploy-agent\_1.3.2-1902\_ubuntu22.deb

systemctl list-units --type=service | grep codedeploy

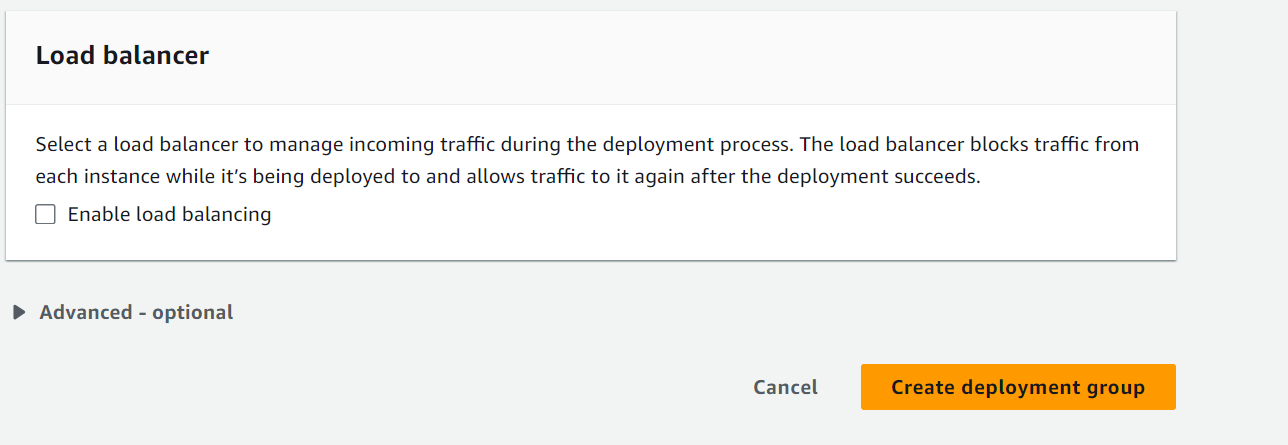
sudo service codedeploy-agent status

The above command will install the code deploy agent,

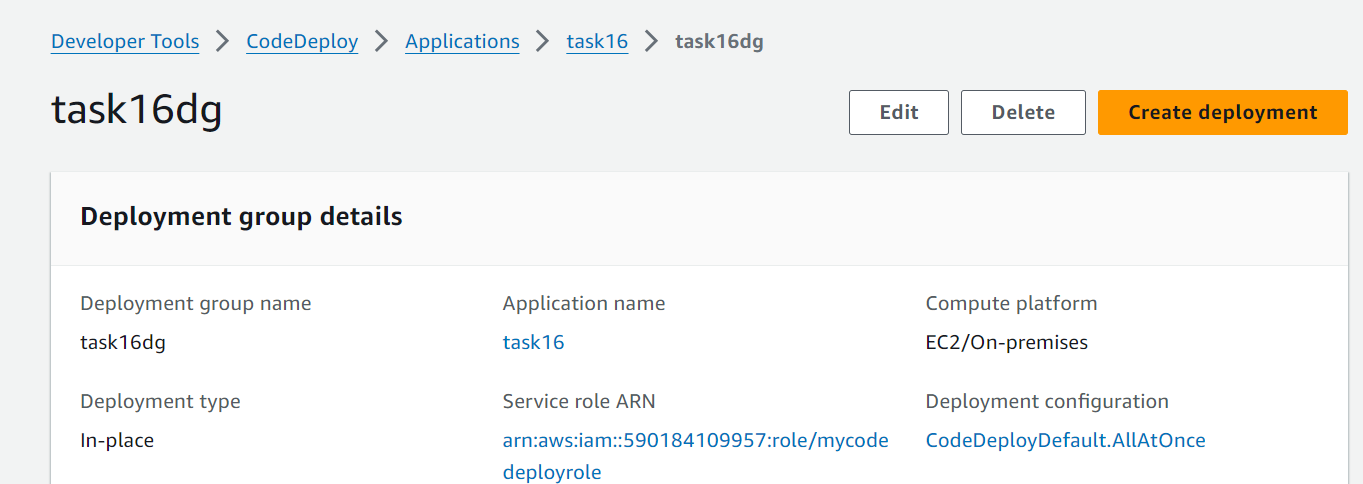
After installing the agent select the Never option while creating the Deployment group



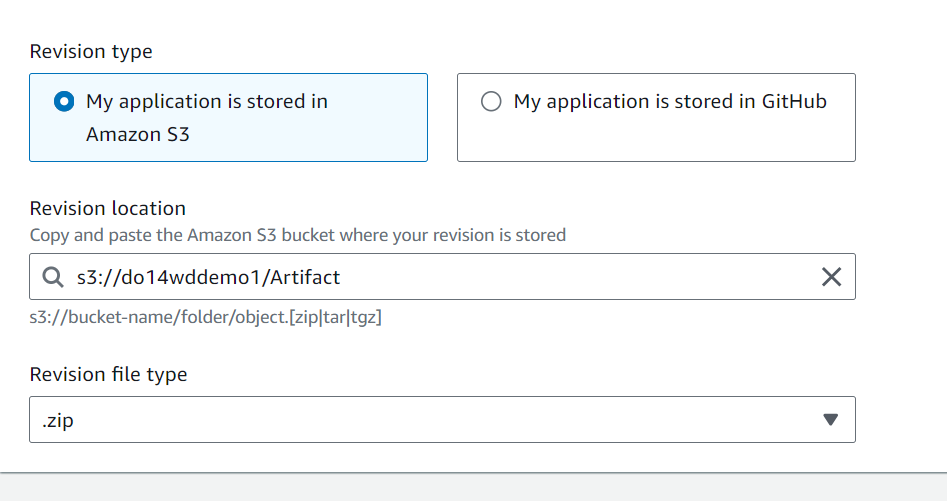
Now after completing the configuration click on Create deployment group



Now click on the Create deployment



In the Revision location, paste the S3 Url and select zip format



Now click on Create deployment