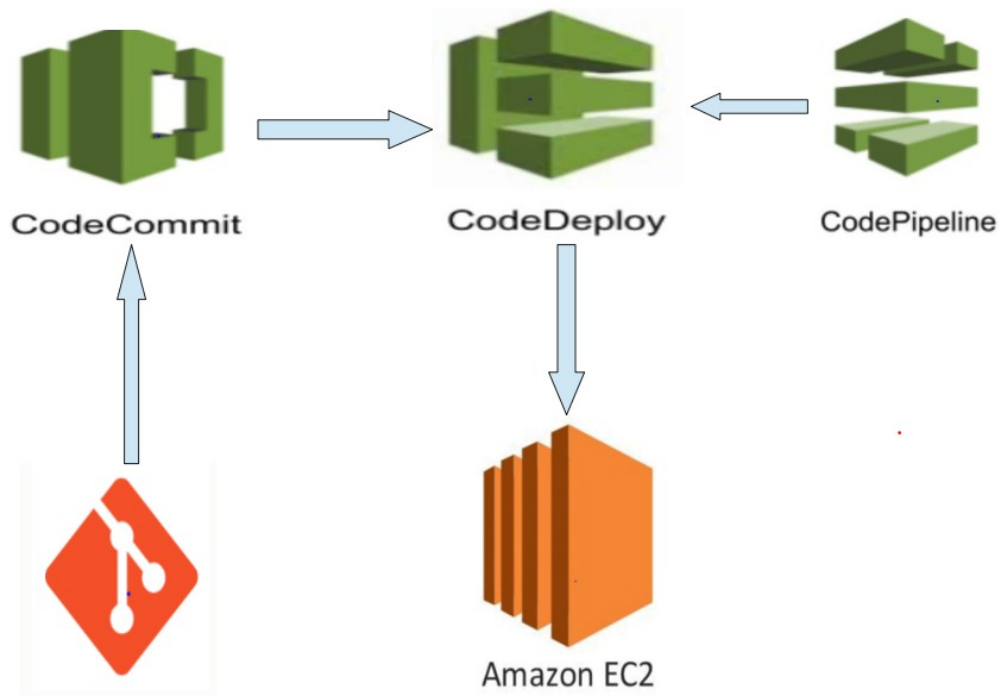
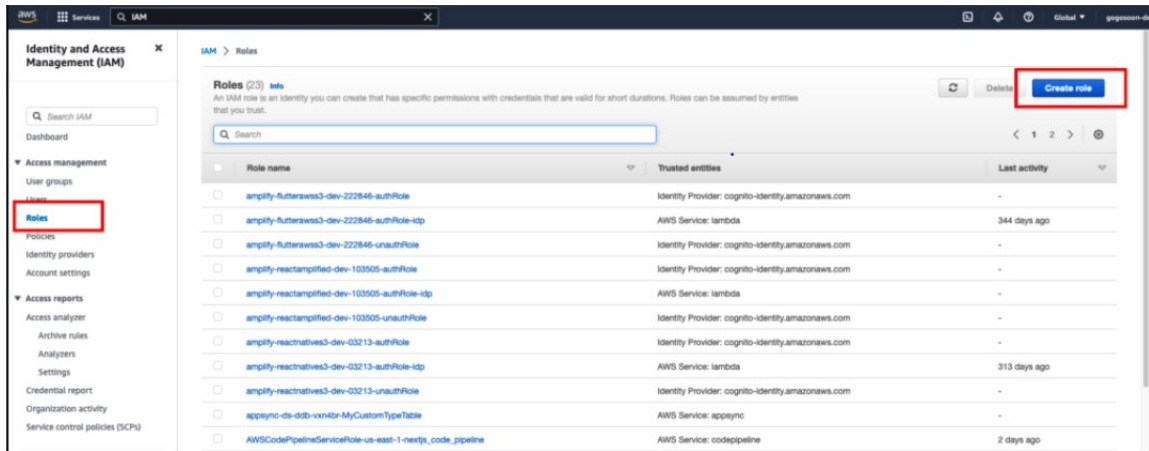


How to Setup Auto-Deployment using CodePipeline and CodeDeploy



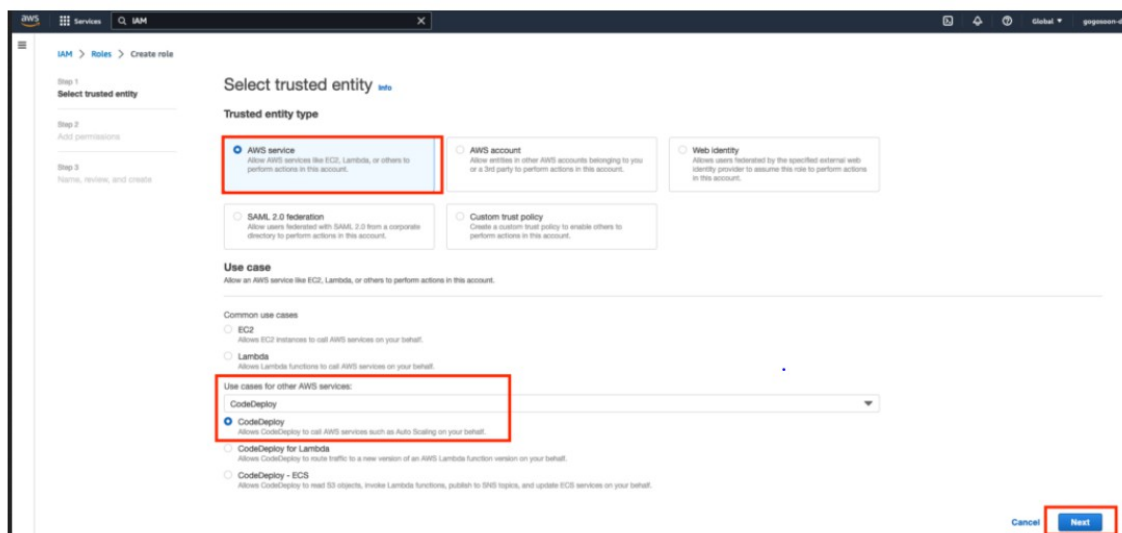
Workflow:-

Create an IAM Role for CodeDeploy:



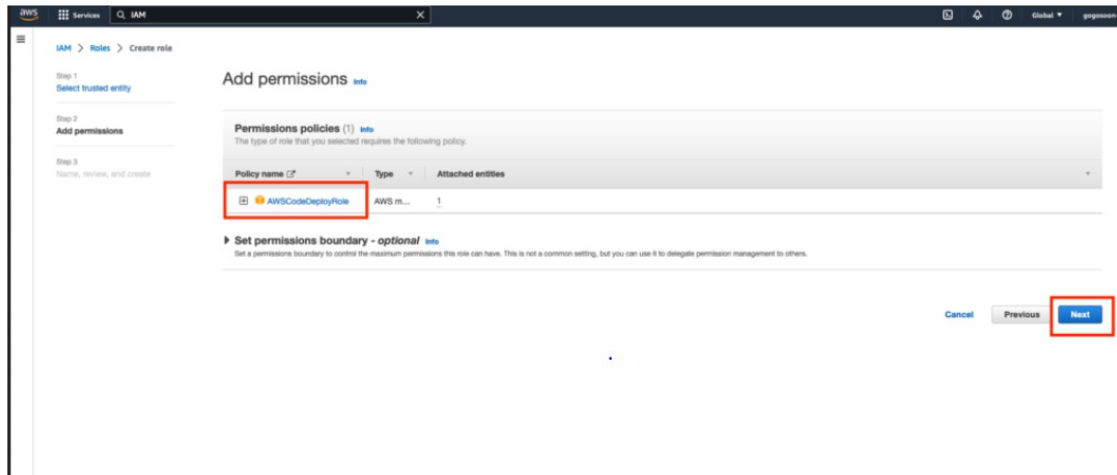
Create IAM role

Choose AWS service in Trusted entity types and choose CodeDeploy in the Use cases section and proceed to the next step.



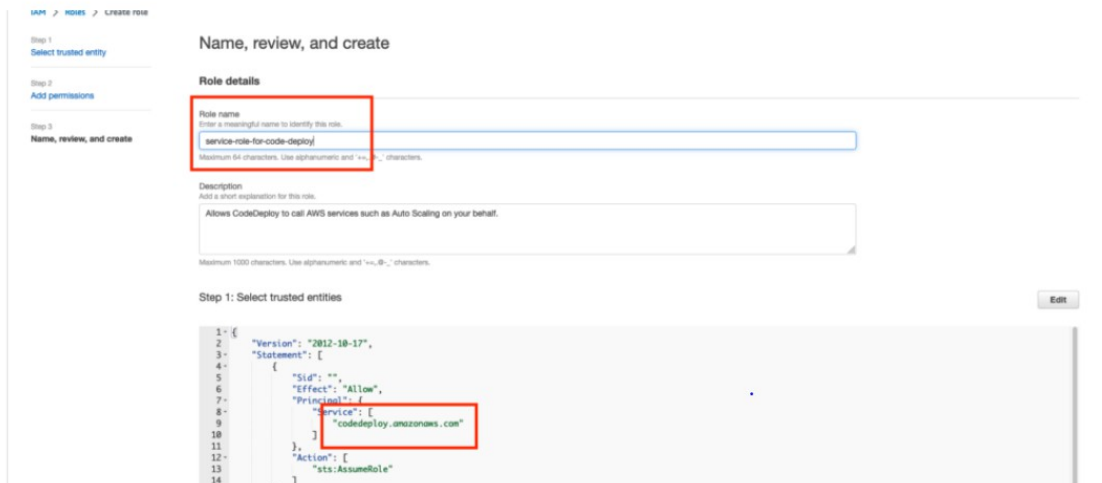
IAM role for CodeDeploy

Now, you can see that the AWSCodeDeployRole policy is the only policy available, and it'll be chosen by default in this (Permissions) step.



AWSCodeDeploy Permission

Enter a name for your IAM role. You should choose a meaningful name to identify this in the future. I'm calling it service-role-for-code-deploy.



AWSCodeDeploy Permission Review

Create an IAM role for EC2

Let's create the next role. This role is for EC2. Choose AWS service in the Trusted entity type, EC2 in the Common use cases section, and choose CodeDeploy in Use cases for

other AWS services.

Step 1: Select trusted entity

Select trusted entity

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: Allow 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.

Common use cases

- ☒ EC2: Allow EC2 instances to call AWS services on your behalf.
- ☐ Lambda: Allow Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:

CodeDeploy

Cancel Next

IAM role for EC2

search for codedeploy
select "AmazonEC2RoleForCodeDeploy"

Step 2: Add permissions

Add permissions

Permissions policies (Selected 1/826)

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

Filter policies by property or policy name and press enter: 11 matches

Search: codedeploy X Clear filters

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonEC2RoleForCodeDeploy	AWS m...	Provides EC2 access to S3 bucket to download revision. This role is needed by the CodeDeploy agent on EC2 instances.
<input type="checkbox"/> AWSCodeDeployRoleForECS	AWS m...	Provides CodeDeploy service wide access to perform an ECS blue/green deployment on your behalf. Grants full access to sup...
<input type="checkbox"/> AWSCodeDeployReadOnlyAccess	AWS m...	Provides read only access to CodeDeploy resources.
<input type="checkbox"/> AWSCodeDeployFullAccess	AWS m...	Provides full access to CodeDeploy resources.
<input type="checkbox"/> AWSCodeDeployRole	AWS m...	Provides CodeDeploy service access to expand tags and interact with Auto Scaling on your behalf.
<input type="checkbox"/> AWSCodeDeployRoleForECSLimited	AWS m...	Provides CodeDeploy service limited access to perform an ECS blue/green deployment on your behalf.
<input type="checkbox"/> AWSCodeDeployRoleForLambda	AWS m...	Provides CodeDeploy service access to perform a Lambda deployment on your behalf.
<input type="checkbox"/> AWSCodeDeployDeployerAccess	AWS m...	Provides access to register and deploy a revision.
<input type="checkbox"/> AWSCodeDeployRoleForLambdaLimited	AWS m...	Provides CodeDeploy service limited access to perform a Lambda deployment on your behalf.
<input type="checkbox"/> AWSCodeDeployRoleForCloudFormation	AWS m...	Provides CodeDeploy service access to invoke Lambda function on your behalf to perform blue/green deployment through Clou...
<input type="checkbox"/> AmazonEC2RoleForAWSCodeDeployLimited	AWS m...	Provides EC2 limited access to S3 bucket to download revision. This role is needed by the CodeDeploy agent on EC2 instances.

Adding AmazonEC2RoleForCodeDeploy permission

Launch Ec2 Instance

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Q Filter resources by tag(s)

Instances (running)	0	Auto Scaling Groups	0
Dedicated Hosts	0	Elastic IPs	0
Instances	0	Key pairs	4
Load balancers	0	Placement groups	0
Security groups	8	Snapshots	0
Volumes	0		

EC2 > Instances > Launch an instance

Launch an instance

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

Name

mywebserver

Add additional tags

▼ Application and OS Images (Amazon Machine Image)

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

▼ Summary

Number of instances

2

When launching more than 1 instance, consider EC2 Auto Scaling.

Software Image (AMI)

Amazon Linux 2023 AMI 2023.0.2...read more

ami-06e46074ae430fba6

Virtual server type (instance type)

t2.micro

Firewall (security group)

Cancel

Launch instance

Review commands

▼ 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

Quick Start



[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

Free tier eligible ▼

ami-06e46074ae430fba6 (64-bit (x86), uefi-preferred) / ami-085a3abb84068d568 (64-bit (Arm), uefi)

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 AMI 2023.0.20230329.0 x86_64 HVM kernel-6.1

Architecture

64-bit (x86) ▼

Boot mode

uefi-preferred

AMI ID

ami-06e46074ae430fba6

Verified provider

Instance type [Info](#)

Instance type

t2.micro

Free tier eligible

Family: t2 1 vCPU 1 GiB Memory

On-Demand Windows pricing: 0.0162 USD per Hour ▼

On-Demand SUSE pricing: 0.0116 USD per Hour

On-Demand RHEL pricing: 0.0716 USD per Hour

On-Demand Linux pricing: 0.0116 USD per Hour

☐ All generations

[Compare instance types](#)

▼ 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*

key_pem



Create new key pair

▼ Network settings [Info](#)

Edit

Network [Info](#)

vpc-0d3d739e277994ee8

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

Network [Info](#)

vpc-0d3d739e277994ee8

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

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

Security groups [Info](#)

Select security groups

default sg-0557104f995061b47 ✕

VPC: vpc-0d3d739e277994ee8



Compare security group rules

Instances (2) Info							
				Refresh	Connect	Instance state ▼	Actions ▼
<input type="text" value="Find instance by attribute or tag (case-sensitive)"/>				< 1 > ⚙			
<input type="checkbox"/>	Name ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability
<input type="checkbox"/>	mywebserver03	i-0e09bc01f7a007bad	✔ Running 🔍 🔍	t2.micro	🕒 Initializing	No alarms +	us-eas
<input type="checkbox"/>	node_server	i-051b6b382c1dfa5b0	✔ Running 🔍 🔍	t2.micro	🕒 Initializing	No alarms +	us-eas

Connect Ec2 instance

Instances (1/2) Info							
				Refresh	Connect	Instance state ▼	Actions ▼
<input type="text" value="Find instance by attribute or tag (case-sensitive)"/>				< 1 > ⚙			
<input checked="" type="checkbox"/>	Name ▼	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability
<input checked="" type="checkbox"/>	mywebserver03	i-0e09bc01f7a007bad	✔ Running 🔍 🔍	t2.micro	🕒 Initializing	No alarms +	us-ea
<input type="checkbox"/>	node_server	i-051b6b382c1dfa5b0	✔ Running 🔍 🔍	t2.micro	🕒 Initializing	No alarms +	us-ea

copy ssh key

[EC2 Instance Connect](#)
[Session Manager](#)
[SSH client](#)
[EC2 serial console](#)

Instance ID

i-0e09bc01f7a007bad (mywebserver03)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is key_pem.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.


```
 chmod 400 key_pem.pem
```
4. Connect to your instance using its Public DNS:


```
 ec2-44-212-0-192.compute-1.amazonaws.com
```

Example:

```
 ssh -i "key_pem.pem" ec2-user@ec2-44-212-0-192.compute-1.amazonaws.com
```

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

open git bash , putty

[illegible]

set hostname

```
$sudo yum install hostnamectl set-hostname localmachine
```

```
$sudo -i
```

```
kali.radhika@AHM-LL-Radhaga1 MINGW64 ~  
$ cd Downloads/  
  
kali.radhika@AHM-LL-Radhaga1 MINGW64 ~/Downloads  
$ ssh -i "key.pem" ec2-user@ec2-44-212-0-192.compute-1.amazonaws.com  
The authenticity of host 'ec2-44-212-0-192.compute-1.amazonaws.com (44.212.0.192)' can't be established.  
ED25519 key fingerprint is SHA256:IRu952h8z+oym4iTih8f6aLnLdxpPXXwYH4IRE00kK0.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added 'ec2-44-212-0-192.compute-1.amazonaws.com' (ED25519) to the list of known hosts.  
  
#_  
##### Amazon Linux 2023  
~ ~ ~ #####  
~ ~ ~ \####/  
~ ~ ~ #/  
~ ~ ~ V ~ ~ ~>  
~ ~ ~ . _ . /  
~ ~ ~ /m/_/_/_/_____/
```

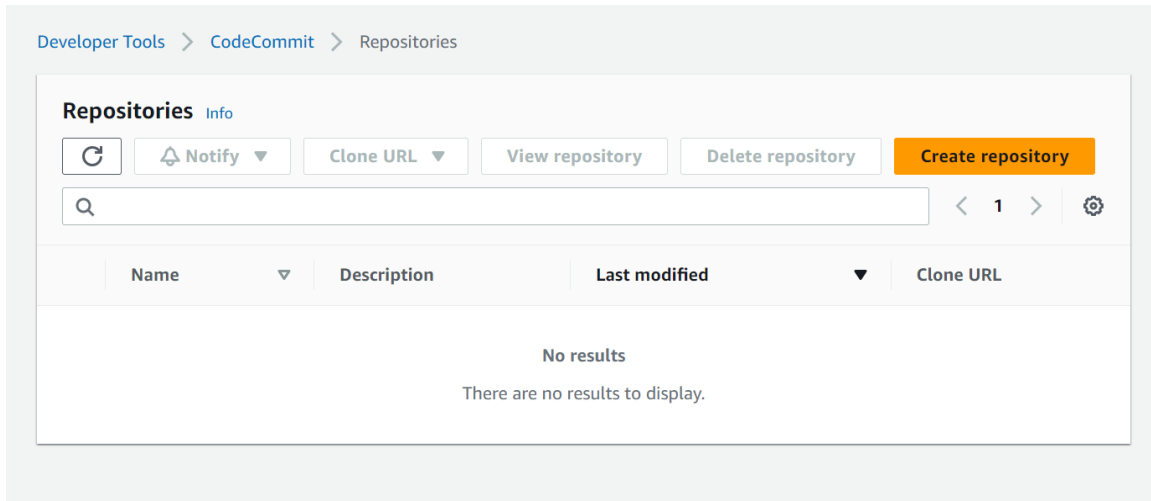
```
[ec2-user@ip-172-31-95-194 ~]$ sudo hostnamectl set-hostname localmachine  
[ec2-user@ip-172-31-95-194 ~]$ sudo -i  
[root@localmachine ~]# |
```

```
install git
```

```
$yum install git -y
```

How to create Repository:

Open Aws CodeCommit



select create repository

Repository name

myrepo23

100 characters maximum. Other limits apply.

Description - *optional*

myrepo23

1,000 characters maximum

Tags

Add

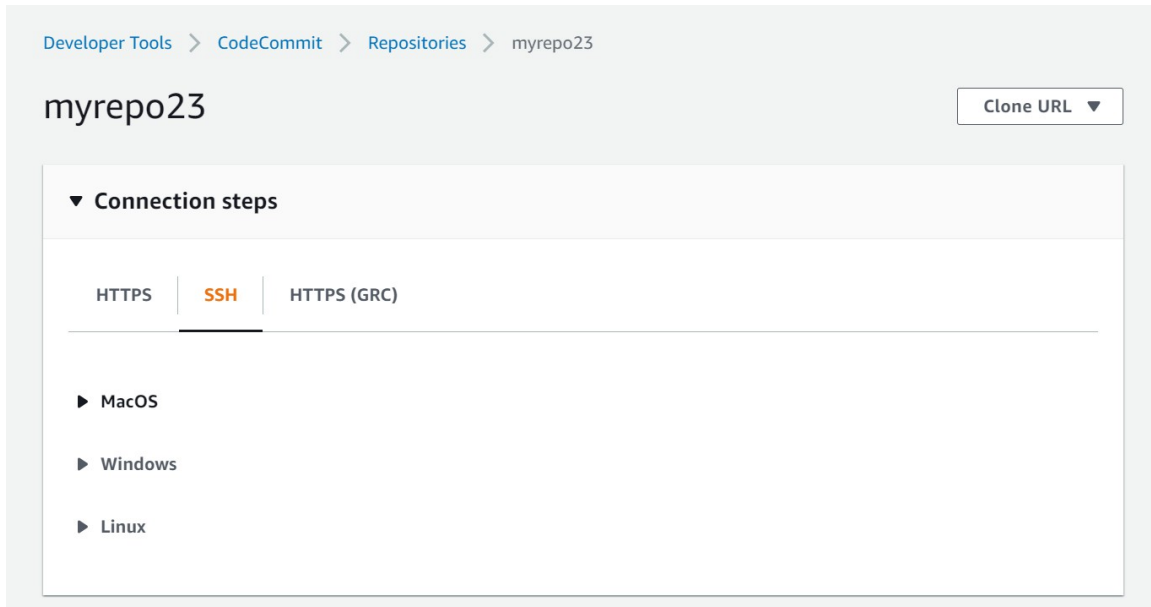
☐ Enable Amazon CodeGuru Reviewer for Java and Python - *optional*

Get recommendations to improve the quality of the Java and Python code for all pull requests in this repository.

A service-linked role will be created in IAM on your behalf if it does not exist.

Cancel **Create**

Next connction steps



open Linux

Step 2: Register SSH Public Key

Upload your SSH public key to your IAM user. [Learn how to upload your SSH public key](#)

Once you have uploaded your SSH public key, copy the SSH Key ID. You will need it in the next step.

Step 3: Edit Local SSH Configuration

Edit your SSH configuration file named "config" in your local ~/.ssh directory. Add the following lines to the file, where the value for User is the SSH Key ID you copied in Step 2.

```
Host git-codecommit.*.amazonaws.com
User Your-IAM-SSH-Key-ID-Here
IdentityFile ~/.ssh/Your-Private-Key-File-Name-Here
```

Once you have saved the file, make sure it has the right permissions by running the following command in the ~/.ssh directory.

```
chmod 600 config
```

Step 4: Clone the repository

Clone your repository to your local computer and start working on code. Run the following command:

```
git clone ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/myrepo23
```

Copy 

open git bash check id_rsa.pub key

```
[root@localmachine ~]# cd .ssh
[root@localmachine .ssh]# ls -al
total 4
drwx-----. 2 root root 29 Apr 12 11:34 .
dr-xr-x---. 4 root root 118 Apr 12 11:54 ..
-rw-----. 1 root root 555 Apr 12 11:34 authorized_keys
[root@localmachine .ssh]# |
```

generate ssh-key

Step 2: Register SSH Public Key

\$ssh-keygen

```
[root@localmachine .ssh]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:7Lk+FI0hIHeiPIJ8nDcooNa7ZeHvxOKIiC9Z0mdDU9o root@localmachine
The key's randomart image is:
+---[RSA 3072]-----+
|
|o + . o
|*+o+o= o
|=* *=+E.o
|o.++.o So
|. o.++ ...
|++o=.o.o
|*. o . +..
| o. +oo.
+----[SHA256]-----+
[root@localmachine .ssh]# ls -al
total 12
drwx-----. 2 root root 61 Apr 12 11:59 .
dr-xr-x---. 4 root root 118 Apr 12 11:54 ..
-rw-----. 1 root root 555 Apr 12 11:34 authorized_keys
-rw-----. 1 root root 2602 Apr 12 11:59 id_rsa
-rw-r--r--. 1 root root 571 Apr 12 11:59 id_rsa.pub
[root@localmachine .ssh]# |
```

copy id_rsa.pub key

IAM > Users

Users (3) [Info](#)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

< 1 >

<input type="checkbox"/>	User name	Groups	Last activity	MFA	Password a...
<input type="checkbox"/>	XXXXXXXXXX	None	✓ 38 minutes ago	None	✓ 5 days ago
<input type="checkbox"/>	XXXXXXXXXX	None	Never	None	None
<input type="checkbox"/>	XXXXXXXXXX	None	✓ 12 days ago	None	✓ 13 days ago

Go to IAM select user

select security credentials

SSH public keys for AWS CodeCommit (0)

User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. [Learn more](#)

Actions

SSH Key ID	Uploaded	Status
No SSH public keys		
<input type="button" value="Upload SSH public key"/>		

upload ssh public key(id_rsa.pub key)

SSH public keys for AWS CodeCommit (1)

User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. [Learn more](#)

Actions

SSH Key ID	Uploaded	Status
<input type="radio"/> XXXXXXXXXX	Now	✓ Active

Step 3: Edit Local SSH Configuration

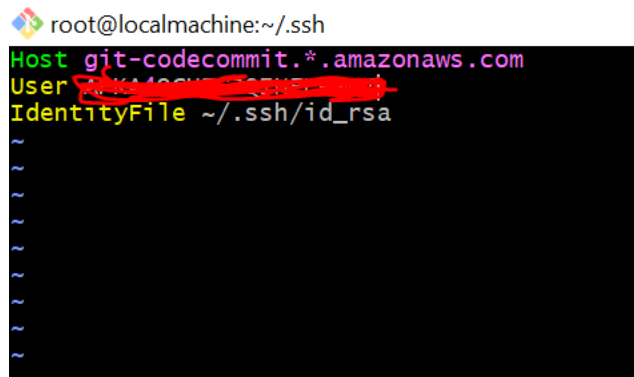
Edit your SSH configuration file named "config" in your local ~/.ssh directory. Add the following lines to the file, where the value for User is the SSH Key ID you copied in Step 2.

```
$vi ~/.ssh/config
```

```
Host git-codecommit.*.amazonaws.com
```

```
User Your-IAM-SSH-Key-ID-Here
```

```
IdentityFile ~/.ssh/Your-Private-Key-File-Name-Here
```

A terminal window screenshot showing the SSH configuration file content. The prompt is root@localmachine:~/.ssh. The file content is: Host git-codecommit.*.amazonaws.com, User [redacted], IdentityFile ~/.ssh/id_rsa. There are several tilde (~) characters below the IdentityFile line.

```
root@localmachine:~/.ssh
Host git-codecommit.*.amazonaws.com
User [redacted]
IdentityFile ~/.ssh/id_rsa
~
~
~
~
~
~
~
```

Step 4: Clone the repository

```
$mkdir project
```

```
$cd project
```

```
$mkdir codecommit
```

```
$cd Codecommit
```

```
drwxr-xr-x. 3 root root 24 Apr 12 11:54 project
[root@localmachine ~]# cd project
[root@localmachine project]# cd codecommit
[root@localmachine codecommit]# ls -al
total 0
drwxr-xr-x. 2 root root 6 Apr 12 11:54 .
drwxr-xr-x. 3 root root 24 Apr 12 11:54 ..
[root@localmachine codecommit]# |
```

\$git clone ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/myrepo23

```
[root@localmachine codecommit]# git clone ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/myrepo23
Cloning into 'myrepo23'...
The authenticity of host 'git-codecommit.us-east-1.amazonaws.com (52.94.229.29)' can't be established.
RSA key fingerprint is SHA256:eLMYlj0DKA4uvDZcl/KgtIayZANwX6t8+8isPtotBoY.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'git-codecommit.us-east-1.amazonaws.com' (RSA) to the list of known hosts.
warning: You appear to have cloned an empty repository.
[root@localmachine codecommit]# |
```

\$ls -al

\$cd myrepo23

```
[root@localmachine codecommit]# ls -al
total 0
drwxr-xr-x. 3 root root 22 Apr 12 12:15 .
drwxr-xr-x. 3 root root 24 Apr 12 11:54 ..
drwxr-xr-x. 3 root root 18 Apr 12 12:15 myrepo23
[root@localmachine codecommit]# cd myrepo23
[root@localmachine myrepo23]# ls -al
total 0
drwxr-xr-x. 3 root root 18 Apr 12 12:15 .
drwxr-xr-x. 3 root root 22 Apr 12 12:15 ..
drwxr-xr-x. 7 root root 119 Apr 12 12:15 .git
[root@localmachine myrepo23]# |
```

\$vi appspec.yml

root@localmachine:~/project/codecommit/myrepo23

```
version: 0.0
os: linux
files:
  - source: /index.html
    destination: /var/www/html/
hooks:
  BeforeInstall:
    - location: scripts/install_dependencies
      timeout: 300
      runas: root
    - location: scripts/start_server
      timeout: 300
      runas: root
  ApplicationStop:
    - location: scripts/stop_server
      timeout: 300
      runas: root
```

\$vi index.html

```
welcome to web application
```

\$mkdir scripts

\$vi scripts/install_dependencies


```
#!/bin/bash
yum install -y httpd
```

```
~
~
~
~
~
~
~
~
~
~
```

\$vi scripts/start_server

```
#!/bin/bash
service httpd start
```

```
~
~
~
~
~
~
~
~
~
~
```

\$vi scripts/stop_server

```
#!/bin/bash
isExistApp = 'pgrep httpd'
if [[ -n $isExistApp ]]; then
    service httpd stop
fi
```

```
~
~
~
~
~
~
~
~
```

\$ls -al

```

root@localmachine myrepo23]# ls -al
total 8
-rwxr-xr-x. 4 root root 70 Apr 12 12:25 .
-rwxr-xr-x. 3 root root 22 Apr 12 12:15 ..
-rwxr-xr-x. 7 root root 119 Apr 12 12:15 .git
-rw-r--r--. 1 root root 358 Apr 12 12:19 appsepc.yml
-rw-r--r--. 1 root root 27 Apr 12 12:22 index.html
-rwxr-xr-x. 2 root root 73 Apr 12 12:29 scripts
root@localmachine myrepo23]# |

```

\$git add .

\$git config --global user.email " @gmail.com"

\$git config --global user.name "your name"

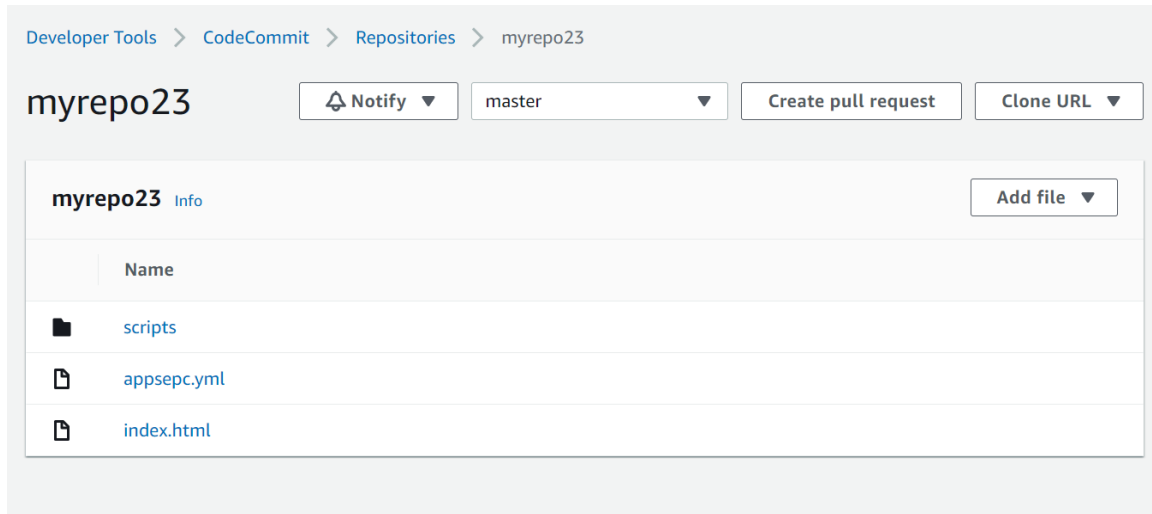
\$git commit -m "first commit" -a

\$git push origin master

```

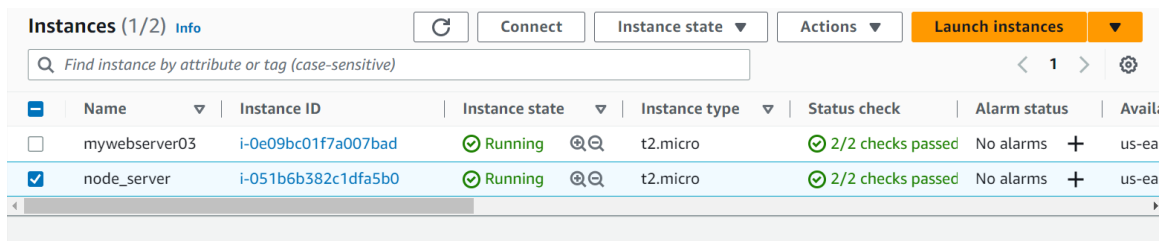
root@localmachine myrepo23]# git add .
root@localmachine myrepo23]# git config --global user.email " "
root@localmachine myrepo23]# git config --global user.name " "
root@localmachine myrepo23]# git commit -m "first commit" -a
[master (root-commit) b4be07d] first commit
5 files changed, 27 insertions(+)
create mode 100644 appsepc.yml
create mode 100644 index.html
create mode 100644 scripts/install_dependencies
create mode 100644 scripts/start_server
create mode 100644 scripts/stop_server
root@localmachine myrepo23]# git push origin master
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (8/8), 785 bytes | 785.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/myrepo23
* [new branch]      master -> master
root@localmachine myrepo23]# |

```

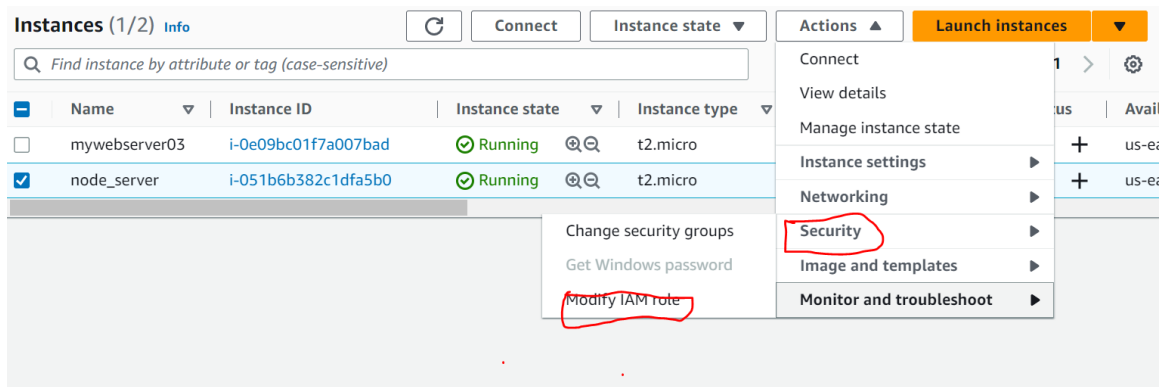


Login into another aws Ec2 instance (for application deploy):-

consider as a node



Attach the IAM Role to EC2



Update IAM role

Install codedeploy-agent:-

\$ sudo yum update

\$sudo yum install ruby

\$sudo yum install wget

\$cd /home/ec2-user

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

\$chmod +x ./install

\$sudo ./install auto

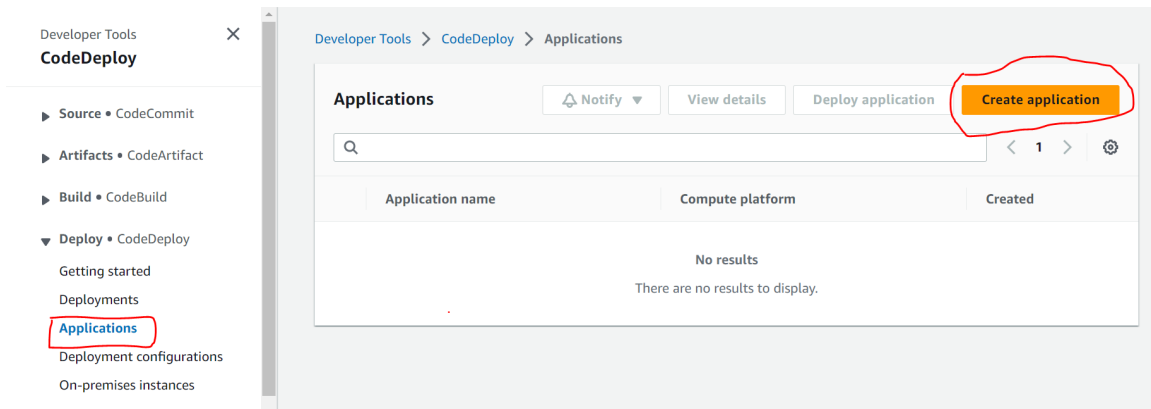
\$sudo service codedeploy-agent start

\$sudo service codedeploy-gent status

```
[root@node ec2-user]# service codedeploy-agent status
/opt/codedeploy-agent/vendor/gems/gli-2.11.0/lib/gli/commands/help_modules/global_help_format.rb:37: warning: Passed. Do not use it, and specify other arguments as keyword arguments.
/opt/codedeploy-agent/vendor/gems/gli-2.11.0/lib/gli/commands/help_modules/global_help_format.rb:37: warning: Passed. Use keyword argument like ERB.new(str, trim_mode: ...) instead.
/opt/codedeploy-agent/vendor/gems/gli-2.11.0/lib/gli/commands/help_modules/command_help_format.rb:27: warning: Passed. Do not use it, and specify other arguments as keyword arguments.
/opt/codedeploy-agent/vendor/gems/gli-2.11.0/lib/gli/commands/help_modules/command_help_format.rb:27: warning: Passed. Use keyword argument like ERB.new(str, trim_mode: ...) instead.
The AWS CodeDeploy agent is running as PID 27436
[root@node ec2-user]#
```

How to Create the CodeDeploy Application:-

In the AWS Console, search "CodeDeploy" in the search bar at the top. Select "Applications" in the left pane. Click on the "Create application" button on the top right.



Create application

Application configuration

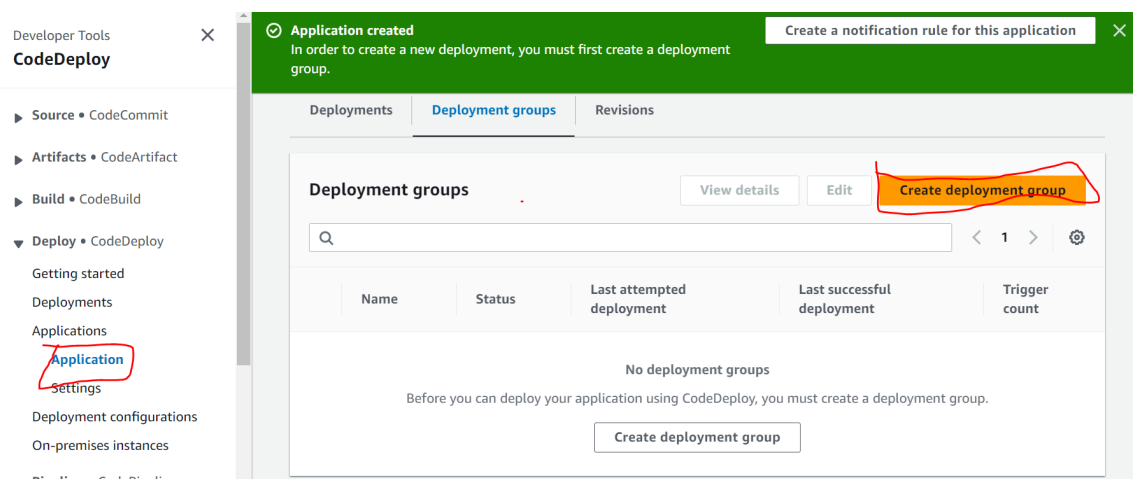
Application name
Enter an application name
todayapp11
100 character limit

Compute platform
Choose a compute platform
EC2/On-premises

Tags
Add tag

Cancel **Create application**

create deployment group



Application
todayapp11
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name

todayapp11group

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::855117476448:role/aws_codedeploy

Deployment type

Choose how to deploy your application

☒ In-place

Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

☐ Blue/green

Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

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

In the Environment configuration section, select "Amazon EC2 instances" and select the key as Name. Enter your EC2 instance name in the value.

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

Remove tag

Agent configuration with AWS Systems Manager [Info](#)



Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent.

Make sure the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. [Learn more](#)

Install AWS CodeDeploy Agent

☐ Never

☐ Only once

☒ Now and schedule updates

Basic scheduler

Cron expression

Days



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

select Create deployment group

[How to Create the CodePipeline:-](#)

Developer Tools

CodePipeline

▶ Source • CodeCommit

▶ Artifacts • CodeArtifact

▶ Build • CodeBuild

▶ Deploy • CodeDeploy

▼ Pipeline • CodePipeline

Getting started

Pipelines

▶ Settings

Go to resource

Developer Tools > CodePipeline > Pipelines

Pipelines info

↺

🔔 Notify ▼

View history

Release change

Delete pipeline

Create pipeline

< 1 > ⚙️

Name	Most recent execution	Latest source revisions	Last executed
No results			
There are no results to display.			

Enter the Pipeline name, and Role name. Remember, we created roles for EC2 and CodeDeploy, but not for CodePipeline. AWS by default creates it from here.

Pipeline settings

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

testpip31

No more than 100 characters

Service role

☒ **New service role**
Create a service role in your account

☐ **Existing service role**
Choose an existing service role from your account

Role name

AWSCodePipelineServiceRole-us-east-1-testpip31

Type your service role name

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

Select Next

Source

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

AWS CodeCommit

Repository name
Choose a repository that you have already created where you have pushed your source code.

Q myrepo23 X

Branch name
Choose a branch of the repository

Q master X

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

Select Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add build stage [Info](#)

Build - optional

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

Cancel Previous **Skip build stage** Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
Choose pipeline settings

Step 2
Add source stage

Step 3
Add build stage

Step 4
Add deploy stage

Step 5
Review

Add deploy stage [Info](#)

You cannot skip this stage
Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

Deploy

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

AWS CodeDeploy

Region
US East (N. Virginia)

Application name

Deploy

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

AWS CodeDeploy

Region
US East (N. Virginia)

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.

Q todayapp11 X

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.

Q todayapp11group X

select next

Step 4: Add deploy stage

Deploy action provider

Deploy action provider

AWS CodeDeploy

ApplicationName

todayapp11

DeploymentGroupName

todayapp11group

Cancel

Previous

Create pipeline

Developer Tools > CodePipeline > Pipelines > testpip31

testpip31

Notify ▼

Edit

Stop execution

Clone pipeline

Release change

 **Source** In progress

Pipeline execution ID: 024d9c97-743c-4f20-b9fe-96f6c68ec7f5

Source



[AWS CodeCommit](#)

 In progress - Just now

Disable transition

✔ Source Succeeded

Pipeline execution ID: 024d9c97-743c-4f20-b9fe-96f6c68ec7f5

Source



AWS CodeCommit

✔ Succeeded - Just now

b4be07d1

b4be07d1 Source: first commit



Disable transition

⌚ Deploy In progress

↓

Disable transition

✔ Deploy Succeeded

Pipeline execution ID: [3885a52b-689f-4353-ba90-dc81fa51c666](#)

Deploy ⓘ

[AWS CodeDeploy](#)

✔ Succeeded - 1 minute ago

[Details](#)

[cf5ae0a5](#) Source: first commit