

Task 16: Deploy a simple Nginx application using AWS code commit and deploy & access via browser

1. Create code commit repo:

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > Create repository

Create repository


Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name

100 characters maximum. Other limits apply.

Description - *optional*

 **Success**
Repository successfully created

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > demoapp

demoapp

Name

awspipeline

Add additional tags

▼ Application and OS Images (Amazon Machine Image) Info

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

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

Recents

Quick Start

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Li
SUSE

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Instances (1/1) Info

Refresh instances

Connect

Instance state ▼

Actions ▼

Launch instances ▼

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 Zone	Public IPv4 DNS
<input checked="" type="checkbox"/>	awspipeline	i-03b82e263033c6bef	Running	t2.micro	Initializing	View alarms +	us-east-1c	ec2-44-201-178-

Connect with that instance and install git

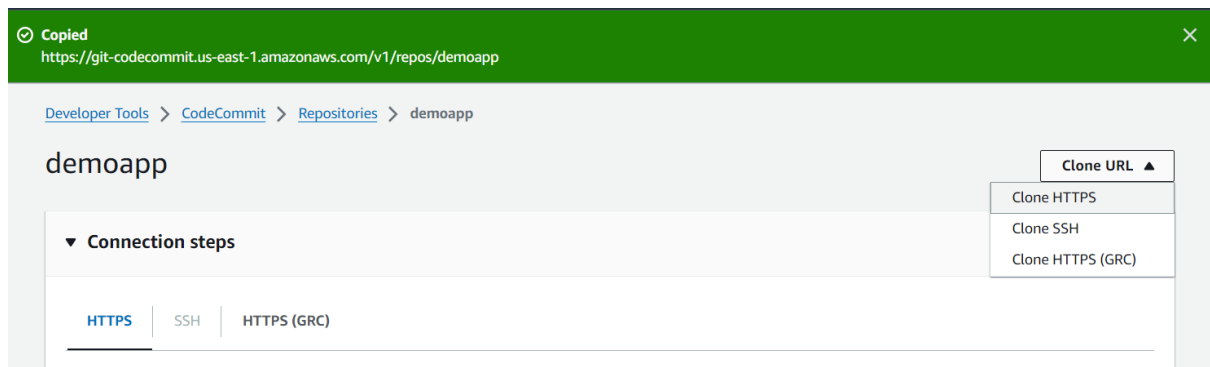
```
[ec2-user@ip-172-31-94-139 ~]$ sudo yum update -y
Last metadata expiration check: 0:02:34 ago on Thu Jul 18 06:04:46 2024.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-94-139 ~]$ sudo yum install git -y
Last metadata expiration check: 0:02:50 ago on Thu Jul 18 06:04:46 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository
Installing:			
git	x86_64	2.40.1-1.amzn2023.0.3	amazonlinux
Installing dependencies:			
git-core	x86_64	2.40.1-1.amzn2023.0.3	amazonlinux
git-core-doc	noarch	2.40.1-1.amzn2023.0.3	amazonlinux
perl-Error	noarch	1:0.17029-5.amzn2023.0.2	amazonlinux
perl-File-Find	noarch	1:37-477.amzn2023.0.6	amazonlinux
perl-git	noarch	2.40.1-1.amzn2023.0.3	amazonlinux
perl-TermReadKey	x86_64	2.38-9.amzn2023.0.2	amazonlinux
perl-lib	x86_64	0.65-477.amzn2023.0.6	amazonlinux

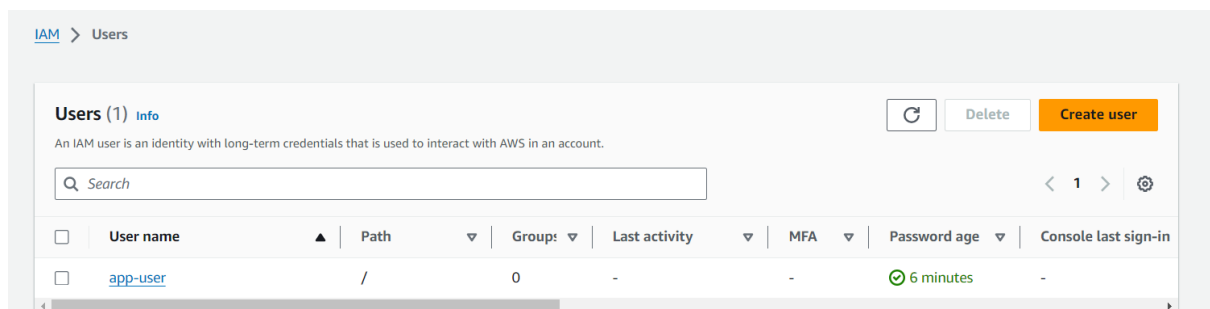
```
Transaction Summary
Installing: 1 Package
Installing dependencies: 6 Packages
Total size: 11.5 MB
Total download size: 11.5 MB
Installed:
git-2.40.1-1.amzn2023.0.3.x86_64
perl-Error-1:0.17029-5.amzn2023.0.2.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64

Complete!
[ec2-user@ip-172-31-94-139 ~]$ git --version
git version 2.40.1
```

Clone the created repo in amazon linux



IAM user can be created [for clone the repo]. During clone ask username and password so only we created IAM user



Now clone that repo.

```
[ec2-user@ip-172-31-94-139 ~]$ git clone https://git-codecommit.us-east-1.amazonaws.com/v1/repos/demoapp
Cloning into 'demoapp'...
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
warning: You appear to have cloned an empty repository.
[ec2-user@ip-172-31-94-139 ~]$
```

Inside demoapp repo we added few files [All files written in amazon linux instance]

- I. Index.html
- II. Buildspec.yaml
- III. Appspec.yml
- install_nginx.sh
- Start_nginx.sh:
- i. Index.html:

```
[ec2-user@ip-172-31-94-139 ~]$ ls
demoapp
[ec2-user@ip-172-31-94-139 ~]$ vi index.html
```

```
hai
good evening
how are you?
~
~
```

```
[ec2-user@ip-172-31-94-139 ~]$ git add index.html
[ec2-user@ip-172-31-94-139 ~]$ git commit -m "add index.html file"
[master (root-commit) 970c701] add index.html file
Committer: EC2 Default User <ec2-user@ip-172-31-94-139.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
```

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 3 insertions(+)
create mode 100644 index.html
```

```
[ec2-user@ip-172-31-94-139 ~]$
```

```
[ec2-user@ip-172-31-94-139 ~]$ git remote -v
[ec2-user@ip-172-31-94-139 ~]$ git remote add origin https://git-codecommit.us-east-1.amazonaws.com/v1/repos/demoapp
[ec2-user@ip-172-31-94-139 ~]$ git push --set-upstream origin master
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 270 bytes | 270.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/demoapp
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
[ec2-user@ip-172-31-94-139 ~]$ git push
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
Everything up-to-date
[ec2-user@ip-172-31-94-139 ~]$
```

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > demoapp

demoapp

Reference

Notify

master

Create pull request

Clone URL

demoapp [Info](#)

Add file

Name



index.html

II) Buildspec.yaml

```
version: 0.2
phases:
  install:
    commands:
      - echo "Installing Nginx server"
      - sudo apt-get update
      - sudo apt-get install nginx -y

  build:
    commands:
      - echo "Build started on 18 july"
      - sudo cp index.html /var/www/html/
  post_build:
    commands:
      - echo "Configuring Nginx"
artifacts:
  files:
    - '**/*'
```

```
[ec2-user@ip-172-31-94-139 ~]$ vi buildspec.yaml
[ec2-user@ip-172-31-94-139 ~]$ git add buildspec.yaml
[ec2-user@ip-172-31-94-139 ~]$ git commit -m "add buildspec.yaml"
[master 5765ef5] add buildspec.yaml
Committer: EC2 Default User <ec2-user@ip-172-31-94-139.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:
```

```
git config --global --edit
```

After doing this, you may fix the identity used for this commit with:

```
git commit --amend --reset-author
```

```
1 file changed, 19 insertions(+)
create mode 100644 buildspec.yaml
```

```
[ec2-user@ip-172-31-94-139 ~]$
```

```
[ec2-user@ip-172-31-94-139 ~]$ git push
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 498 bytes | 498.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/demoapp
 970c701..5765ef5 master -> master
[ec2-user@ip-172-31-94-139 ~]$
```

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > demoapp

demoapp

🔔 Notify ▼

Reference

master ▼

Create pull request

Clone URL ▼

demoapp [Info](#)

Add file ▼

Name



buildspec.yaml



index.html

2. Code build:

Goto code build → click create project

[Developer Tools](#) > [CodeBuild](#) > [Build projects](#) > Create build project

Create build project

Project configuration

Project name

awscodepipeline

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

► Additional configuration

Description, Build badge, Concurrent build limit, tags

Source

Add source

Source 1 - Primary

Source provider

AWS CodeCommit ▼

Repository

Source

[Add source](#)

Source 1 - Primary

Source provider

AWS CodeCommit

Repository

demoapp

Reference type

Choose the source version reference type that contains your source code.

- ☒ Branch
- ☐ Git tag
- ☐ Commit ID

Branch

Choose a branch that contains the code to build.

master

Commit ID - *optional*

Choose a commit ID. This can shorten the duration of your build.

Q

Source version [Info](#)

Environment

Provisioning model [Info](#)

☒ On-demand

Automatically provision build infrastructure in response to new builds.

☐ Reserved capacity

Use a dedicated fleet of instances for builds. A fleet's compute and environment type will be used for the project.

Environment image

☒ Managed image

Use an image managed by AWS CodeBuild

☐ Custom image

Specify a Docker image

Compute

☒ EC2

Optimized for flexibility during action runs

☐ Lambda

Optimized for speed and minimizes the start up time of workflow actions

Operating system

Amazon Linux

Runtime(s)

Standard

Image version

Always use the latest image for this runtime version ▼

☐ Use GPU-enhanced compute

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

codepipeline-servicerole

Type your service role name

► Additional configuration

Timeout, privileged, certificate, VPC, compute type, environment variables, file systems

Build specifications

☐ **Insert build commands**
Store build commands as build project configuration

☒ **Use a buildspec file**
Store build commands in a YAML-formatted buildspec file

Buildspec name - *optional*

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

buildspec.yml

Artifact asks in which s3 bucket stores this.so we create s3 bucket

Create s3 bucket [awspipeline-s3]

Amazon S3 > Buckets > Create bucket

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region

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

Bucket type

[Info](#)

☒ General purpose

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

☐ Directory - New

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

Bucket name

[Info](#)

awspipeline-s3

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.

Choose bucket

General purpose buckets (1) [Info](#) All AWS Regions

↺

Copy ARN

Empty

Delete

Create bucket

Find buckets by name

Name

▲

AWS Region

▼

IAM Access Analyzer

Creation date

▼

☒

[awspipeline-s3](#)

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

[View analyzer for us-east-1](#)

July 18, 2024, 13:03:10 (UTC+05:30)

Click create folder.

[Amazon S3](#) > [Buckets](#) > [awspipeline-s3](#) > Create folder

Create folder [Info](#)

Use folders to group objects in buckets. When you create a folder, S3 creates an object using the name that you specify followed by a slash (/). This object then appears as folder on the console. [Learn more](#)



Your bucket policy might block folder creation

If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.

Folder

Folder name

Folder names can't contain "/". [See rules for naming](#)

Successfully created folder "artifacts".

[Amazon S3](#) > [Buckets](#) > awspipeline-s3

awspipeline-s3 [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

Objects (1) [Info](#)



Copy S3 URI

Copy URL

Download

Open

Delete

Actions

Create folder

Upload

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

< 1 >

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	artifacts/	Folder	-	-	-

Artifacts

Add artifact

Artifact 1 - Primary

Type

Amazon S3

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

Bucket name

awspipeline-s3

Name

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

artifact

☐ Enable semantic versioning

Use the artifact name specified in the buildspec file

Path - *optional*

The path to the build output ZIP file or folder.

Example: MyPath/MyArtifact.zip.

Click create build project. Project created

Project created

You have successfully created the following project: awscodpipeline

Create a notification rule for this project

Developer Tools > CodeBuild > Build projects > awscodpipeline

awscodpipeline

Actions

Create trigger

Edit

Clone

Debug build

Start build with overrides

Start build

Configuration

Source provider AWS CodeCommit	Primary repository demoapp	Artifacts upload location awspipeline-s3	Service role arn:aws:iam::381492100082:role/service-role/codepipeline-servicerole
Public builds Disabled			

Click start build

Build started

You have successfully started the following build: awscodpipeline:2818cb1a-fd5d-454d-855b-8b5026c2449f

Developer Tools > CodeBuild > Build projects > awscodpipeline > awscodpipeline:2818cb1a-fd5d-454d-855b-8b5026c2449f

awscodpipeline:2818cb1a-fd5d-454d-855b-8b5026c2449f

Stop buildRetry build

Build status

Status

✔ Succeeded

Initiator

root

Build ARN

arn:aws:codebuild:us-east-1:381492100082:build/awscodpipeline:2818cb1a-fd5d-454d-855b-8b5026c2449f

Resolved source version

5765ef59183e3c2385a85027243f817c71f6f5bc

Start time

Jul 18, 2024 1:18 PM (UTC+5:30)

End time

Jul 18, 2024 1:19 PM (UTC+5:30)

Build number

1

Build logs	Phase details	Reports	Environment variables	Build details	Resource utilization
Name	Status	Context	Duration	Start time	End time
SUBMITTED	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:18 PM (UTC+5:30)	Jul 18, 2024 1:18 PM (UTC+5:30)
QUEUED	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:18 PM (UTC+5:30)	Jul 18, 2024 1:18 PM (UTC+5:30)
PROVISIONING	✔ Succeeded	-	3 secs	Jul 18, 2024 1:18 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
DOWNLOAD_SOURCE	✔ Succeeded	-	8 secs	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
INSTALL	✔ Succeeded	-	32 secs	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
PRE_BUILD	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
BUILD	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
POST_BUILD	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
UPLOAD_ARTIFACTS	✔ Succeeded	-	2 secs	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
FINALIZING	✔ Succeeded	-	<1 sec	Jul 18, 2024 1:19 PM (UTC+5:30)	Jul 18, 2024 1:19 PM (UTC+5:30)
COMPLETED	✔ Succeeded	-	-	Jul 18, 2024 1:19 PM (UTC+5:30)	-

Once the application build – file stored in artifacts folder

Amazon S3 > Buckets > awspipeline-s3 > artifacts/

artifacts/

Copy S3 URI

ObjectsProperties

Objects (3) Info

Copy S3 URICopy URLDownloadOpenDeleteActionsCreate folderUpload

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 prefixShow versions

Name

Type

Last modified

Size

Storage class

.git/

Folder

-

-

-

buildspec.yaml

yaml

July 18, 2024, 13:19:44 (UTC+05:30)

378.0 B

Standard

index.html

html

July 18, 2024, 13:19:45 (UTC+05:30)

30.0 B

Standard

3. Code deploy:

Click code deploy → create an application

[Developer Tools](#) > [CodeDeploy](#) > [Applications](#) > Create application

Create application

Application configuration

Application name
Enter an application name

100 character limit

Compute platform
Choose a compute platform

Tags

✓ 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](#) > codepipeline-application

codepipeline-application

Application details

Name	Compute platform
codepipeline-application	EC2/On-premises

Then create deployment group. In d.g ask about

1. 2 service role has to create.so we create this role in IAM
2. ec2 instance

Code deploy service role -1:

Goto IAM .click roles →create roles

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

CodeDeploy

Click next

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

codedeploy-servicerole1

Maximum 64 characters. Use alphanumeric and '+=, @-/\[\]\!#\$%^&*()';:~`' characters.

Description

Add a short explanation for this role.

Allows CodeDeploy to call AWS services such as Auto Scaling 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 `codedeploy-servicerole1` created.

View role

IAM > Roles

Roles (5) Info

Refresh

Delete

Create role

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.

Search

< 1 > ⚙

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>	AWSServiceRoleForElasticLoadBalancing	AWS Service: elasticloadbalancing (S	7 hours ago
<input type="checkbox"/>	AWSServiceRoleForSupport	AWS Service: support (Service-Linker	-
<input type="checkbox"/>	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
<input type="checkbox"/>	codedeploy-servicerole1	AWS Service: codedeploy	-

Attach the permissions[administration access] to this roles

Policy was successfully attached to role.

×

Summary

Edit

Creation date

July 18, 2024, 15:02 (UTC+05:30)

ARN

`arn:aws:iam::381492100082:role/codedeploy-servicerole1`

Last activity

-

Maximum session duration

1 hour

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (2) Info

Refresh

Simulate

Remove

Add permissions

You can attach up to 10 managed policies.

Search

Filter by Type

All types

< 1 > ⚙

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	2
<input type="checkbox"/>	AWSCodeDeployRole	AWS managed	1

Create new instance for code deploy:

EC2 > Instances > 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

code deploy

Add additional tags

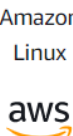
▼ Application and OS Images (Amazon Machine Image) [Info](#)

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


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

Recents

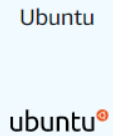
Quick Start




Amazon Linux




macOS




Ubuntu




Windows



Red Hat



SUSE Li



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

ami-0a0e5d9c7acc336f1 (64-bit (x86)) / ami-070f589e4b4a3fece (64-bit (Arm))

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

Free tier eligible

Instances (1/2) Info

Connect

Instance state ▾

Actions ▾

Launch instances ▾

Find Instance by attribute or tag (case-sensitive)

All states ▾

< 1 >

<input checked="" type="checkbox"/>	Name <div></div>	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 DNS
<input checked="" type="checkbox"/>	code deploy	i-06615d4c96bc7791f	<div><div></div>Running<div></div></div>	t2.micro	-	<div>View alarms +</div>	us-east-1c	ec2-3-82-103-2

create deployment group:

Application

Application
codepipeline-application
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name

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.

Attached the service role we already created

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

☒ 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

☐ On-premises instances

Matching instances

1 unique matched instance. [Click here for details](#)

Click create deployment group

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.

or

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

Success
 Deployment group created

Developer Tools > CodeDeploy > Applications > codepipeline-application > demoapplication-dg

demoapplication-dg

Edit

Delete

Create deployment

Deployment group details

Deployment group name demoapplication-dg	Application name codepipeline-application	Compute platform EC2/On-premises
Deployment type In-place	Service role ARN arn:aws:iam::381492100082:role/codedeploy-servicerole1	Deployment configuration CodeDeployDefault.AllAtOnce
Rollback enabled False	Agent update scheduler Learn to schedule update in AWS Systems Manager	

Install code deploy agent. For this we write shell script.connect code deploy ec2[ubuntu server]

Install.sh :

```
#!/bin/bash

# Download and install CodeDeploy agent
sudo yum update -y
sudo yum install -y ruby wget
wget https://aws-codedeploy-us-east-1.s3.amazonaws.com/latest/install
chmod +x ./install
sudo ./install auto

# Start CodeDeploy agent
sudo service codedeploy-agent start

# Optional: Verify CodeDeploy agent status
sudo service codedeploy-agent status
```

```
[ec2-user@ip-172-31-94-139 ~]$ vi install.sh
[ec2-user@ip-172-31-94-139 ~]$ chmod +x install.sh
[ec2-user@ip-172-31-94-139 ~]$ ./install.sh
Last metadata expiration check: 4:28:03 ago on Thu Jul 18 06:04:46 2024.
Dependencies resolved.
Nothing to do.
Complete!
Last metadata expiration check: 4:28:04 ago on Thu Jul 18 06:04:46 2024.
Package wget-1.21.3-1.amzn2023.0.3.x86_64 is already installed.
Dependencies resolved.
```

Package	Architecture	Version	Repository
Installing:			
ruby3.2	x86_64	3.2.2-180.amzn2023.0.2	amazonlinux
Installing dependencies:			
ruby3.2-default-gems	noarch	3.2.2-180.amzn2023.0.2	amazonlinux
ruby3.2-libs	x86_64	3.2.2-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygem-io-console	x86_64	0.6.0-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygem-json	x86_64	2.6.3-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygem-psych	x86_64	5.0.1-180.amzn2023.0.2	amazonlinux
Installing weak dependencies:			
ruby3.2-rubygem-bigdecimal	x86_64	3.1.3-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygem-bundler	noarch	2.4.10-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygem-rdoc	noarch	6.5.0-180.amzn2023.0.2	amazonlinux
ruby3.2-rubygems	noarch	3.4.10-180.amzn2023.0.2	amazonlinux

```
ubuntu@ip-172-31-80-155:~$ vi install.sh
ubuntu@ip-172-31-80-155:~$ chmod +x install.sh
ubuntu@ip-172-31-80-155:~$ ./install.sh
```

```
● codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
   Loaded: loaded (/etc/init.d/codedeploy-agent; generated)
   Active: active (running) since Thu 2024-07-18 09:58:00 UTC; 1s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 2511 ExecStart=/etc/init.d/codedeploy-agent start (code=exited, status=0/SUCCESS)
    Tasks: 3 (limit: 1120)
   Memory: 57.0M
      CPU: 997ms
   CGroup: /system.slice/codedeploy-agent.service
```

Code deploy service role -2: [for ec2]

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

Click next. Later on attach permissions .click next

Role details

Role name

Enter a meaningful name to identify this role.

codedeploy-servicerole2

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

Click create role.role created.

Role **codedeploy-servicerole2** created. [View role](#)

Roles (7) [Info](#) [Refresh](#) [Delete](#) [Create role](#)

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.

<input type="checkbox"/>	Role name	Trusted entities	Last activity
<input type="checkbox"/>	aws-servicerole-for-elasticbeanstalk	AWS Service: elasticbeanstalk	-
<input type="checkbox"/>	codedeploy-servicerole1	AWS Service: codedeploy	-
<input type="checkbox"/>	codedeploy-servicerole2	AWS Service: ec2	-
<input type="checkbox"/>	codepipeline-servicerole	AWS Service: codebuild	2 hours ago

Attach permissions for this role [give full access to ec2, s3, deploy full access, code deploy role]

Get into that role click attach policies.

Policies have been successfully attached to role. [X](#)

Last activity: - Maximum session duration: 1 hour

[Permissions](#) [Trust relationships](#) [Tags](#) [Access Advisor](#) [Revoke sessions](#)

Permissions policies (5) [Info](#) [Refresh](#) [Simulate](#) [Remove](#) [Add permissions](#)

You can attach up to 10 managed policies.

Filter by Type: [All types](#)

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	AdministratorAccess	AWS managed - job function	3
<input type="checkbox"/>	AmazonEC2FullAccess	AWS managed	1
<input type="checkbox"/>	AmazonS3FullAccess	AWS managed	1
<input type="checkbox"/>	AWSCodeDeployFullAccess	AWS managed	1
<input type="checkbox"/>	AWSCodeDeployRole	AWS managed	2

Goto instance give modify iam role

Instances (1/2) [Info](#) [Refresh](#) [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

[All states](#)

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm	Public IP
<input checked="" type="checkbox"/>	code deploy	i-06615d4c96bc7791f	Running	t2.micro	2/2 checks passed	View	ec2-3-81
<input type="checkbox"/>	awspipeline	i-03b82e263033c6bef	Running	t2.micro			ec2-3-81

[Connect](#)
[View details](#)
[Manage instance state](#)
[Instance settings](#)
[Networking](#)
[Security](#)
[Image and templates](#)
[Monitor and troubleshoot](#)

[Change security groups](#)
[Get Windows password](#)
[Modify IAM role](#)

EC2 > Instances > i-06615d4c96bc7791f > Modify IAM role

Modify IAM role [Info](#)

Attach an IAM role to your instance.

Instance ID

 i-06615d4c96bc7791f (code deploy)

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.

codedeploy-servicerole2 ▼



[Create new IAM role](#) 

Cancel

Update IAM role

✓ Successfully attached codedeploy-servicerole2 to instance i-06615d4c96bc7791f

Instances (1/2) [Info](#)



Connect

Instance state ▼
















Actions ▼

Launch instances ▼

Find Instance by attribute or tag (case-sensitive)

All states ▼

< 1 > 

	Name 	Instance ID	Instance state 	Instance type 	Status check	Alarm status	Availability Zone 	Public IP
<input checked="" type="checkbox"/>	code deploy	i-06615d4c96bc7791f	 Running  	t2.micro	 2/2 checks passed	View alarms 	us-east-1c	ec2-3-81
<input type="checkbox"/>	awspipeline	i-03b82e263033c6bef	 Running  	t2.micro	 2/2 checks passed	View alarms 	us-east-1c	ec2-3-81

After that restart code deploy agent

```
ubuntu@ip-172-31-80-155:~$ sudo service codedeploy-agent restart
ubuntu@ip-172-31-80-155:~$
```

iii) Appspec.yml

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html
hooks:
  AfterInstall:
    - location: install_nginx.sh
      timeout: 300
      runas: root
  ApplicationStart:
    - location: start_nginx.sh
      timeout: 300
      runas: root
```

- install_nginx.sh

```
#!/bin/bash

sudo apt-get update
sudo apt-get install nginx -y
```

- Start_nginx.sh:

```
#!/bin/bash

sudo service nginx start
```

```
[ec2-user@ip-172-31-94-139 ~]$ vi install_nginx.sh
[ec2-user@ip-172-31-94-139 ~]$ vi start_nginx.sh
```

```
[ec2-user@ip-172-31-94-139 ~]$ git add appspec.yml
[ec2-user@ip-172-31-94-139 ~]$ git add install_nginx.sh
[ec2-user@ip-172-31-94-139 ~]$ git add start_nginx.sh
```

```
[ec2-user@ip-172-31-94-139 ~]$ git commit -m "add all files"
[master 12bf0b8] add all files
Committer: EC2 Default User <ec2-user@ip-172-31-94-139.ec2.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

3 files changed, 21 insertions(+)
create mode 100644 appspec.yml
create mode 100644 install_nginx.sh
create mode 100644 start_nginx.sh
[ec2-user@ip-172-31-94-139 ~]$ git push
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (4/4), done.

[ec2-user@ip-172-31-94-139 ~]$ git push
Username for 'https://git-codecommit.us-east-1.amazonaws.com': app-user-at-381492100082
Password for 'https://app-user-at-381492100082@git-codecommit.us-east-1.amazonaws.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 641 bytes | 641.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.us-east-1.amazonaws.com/v1/repos/demoapp
    5765ef5..12bf0b8  master -> master
[ec2-user@ip-172-31-94-139 ~]$
```

All files written in amazon linux instance

Developer Tools > CodeCommit > Repositories > demoapp

demoapp

Reference: master

[Notify](#) [Create pull request](#) [Clone URL](#)

demoapp [Info](#) [Add file](#)

Name
appspec.yml
buildspec.yml
index.html
install_nginx.sh
start_nginx.sh

Open the port no: 80 for codedeploy ec2 instance

4. Code pipeline:

Click create pipeline

Choose pipeline settings [Info](#)

Step 1 of 5

Pipeline settings

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

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.

Remaining are same .click next. pipeliene role also assigned automatically

Select aws code commit.All names comes under [already we created]

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 demoapp

×

Branch name
Choose a branch of the repository

Q master

×

master

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

Output artifact format
Choose the output artifact format.

☒ **CodePipeline default**
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include Git metadata about the repository.

☐ **Full clone**
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full Git clone. Only supported for AWS CodeBuild actions.

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.

AWS CodeBuild ▼

Region

US East (N. Virginia) ▼

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.



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.

Add deploy stage [Info](#)

Step 4 of 5

Deploy - optional

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.

codepipeline-application

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.

demoapplication-dg

☐ Configure automatic rollback on stage failure

Cancel

Previous

Skip deploy stage

Next

Click next .click create pipeline

Success

Stage Source successfully retried

Developer Tools > CodePipeline > Pipelines > taskpipeline

taskpipeline

Notify Edit Stop execution Clone pipeline Release change

Pipeline type: V2 Execution mode: QUEUED

Source Succeeded

Pipeline execution ID: [4d8be7ba-f9ff-4364-a876-0e047a4f7244](#)

Source

[AWS CodeCommit](#)

Succeeded - Just now

[12bf0b84](#)

View details

[12bf0b84](#) Source: add all files

✓ **Build** ⓘ Succeeded

Pipeline execution ID: [4d8be7ba-f9ff-4364-a876-0e047a4f7244](#)

Build

[AWS CodeBuild](#)

✓ Succeeded - [Just now](#)

[View details](#)

[12bf0b84](#) Source: add all files

✓ **Deploy** ⓘ Succeeded

Pipeline execution ID: [4d8be7ba-f9ff-4364-a876-0e047a4f7244](#)

Deploy

[AWS CodeDeploy](#)

✓ Succeeded - [Just now](#)

[View details](#)

[12bf0b84](#) Source: add all files

[Developer Tools](#) > [CodePipeline](#) > Pipelines

📘 Introducing the new V2 pipeline type with improved release safety, pipeline triggers, parameterized pipelines, and a new billing model. [Learn more](#) ✕

Pipelines [Info](#)



🔔 Notify ▼

[View history](#)

[Release change](#)

[Delete pipeline](#)

[Create pipeline](#)

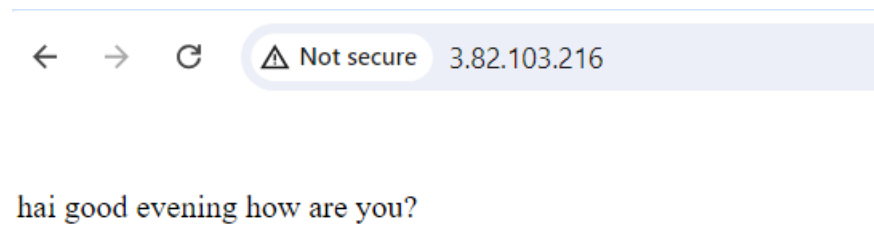


< 1 > ⚙️

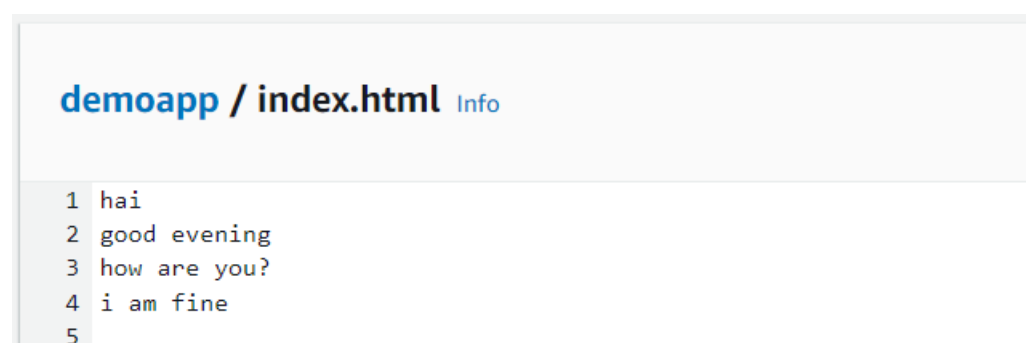
	Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
○	taskpipeline (Type: V2 Execution mode: QUEUED)	✓ Succeeded	Source – 12bf0b84 : add all files	9 minutes ago	✓ View details

Goto code deploy ec2.open the port no:80..copy the address run in browser.

This is the content I gave in index.html



I gave a change in index.html file.



That chance automatically build

Most recent executions ×

Trigger

CloudWatchEvent - [codepipeline-demoap-master-171292-rule](#) ↗

Pipeline execution ID	Status	Last updated
ecb48eba ↗	✔ Succeeded	4 minutes ago

Trigger

CreatePipeline - [root](#) ↗

Pipeline execution ID	Status	Last updated
4d8be7ba ↗	✔ Succeeded	11 minutes ago

Done



Not secure

3.82.103.216

hai good evening how are you? i am fine

