

Task Description:

Deploy a simple web application using AWS code commit, code build and deploy & access via browser and automate via codepipeline.

>> **Step 1:** Launch EC2, Install Apache + Code Deploy Agent and finally attaching IAM role to EC2

```
aws [Alt+S] Asia Pacific (Mumbai) Pranit

Verifying      : mod_http2-2.0.27-1.amzn2023.0.3.x86_64
Verifying      : mod_lua-2.4.62-1.amzn2023.x86_64

=====
WARNING:
A newer release of "Amazon Linux" is available.

Available Versions:

Version 2023.7.20250623:
Run the following command to upgrade to 2023.7.20250623:

dnf upgrade --releasever=2023.7.20250623

Release notes:
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.7.20250623.html

=====

Installed:
apr-1.7.5-1.amzn2023.0.4.x86_64          apr-util-1.6.3-1.amzn2023.0.1.x86_64          apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch  httpd-2.4.62-1.amzn2023.x86_64          httpd-core-2.4.62-1.amzn2023.x86_64
httpd-filesystem-2.4.62-1.amzn2023.noarch          httpd-tools-2.4.62-1.amzn2023.x86_64          libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch              mod_http2-2.0.27-1.amzn2023.0.3.x86_64          mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[ec2-user@ip-172-31-12-140 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-12-140 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service -> /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-12-140 ~]$
```

```
aws [Alt+S] Asia Pacific (Mumbai) Pranit

[ec2-user@ip-172-31-12-140 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-12-140 ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service -> /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-12-140 ~]$ sudo yum install ruby -y
Last metadata expiration check: 0:03:10 ago on Tue Jun 24 05:15:48 2025.
Dependencies resolved.
=====
Package                               Architecture      Version           Repository        Size
-----
Installing:
ruby3.2                               x86_64            3.2.8-184.amzn2023.0.1  amazonlinux       40 k
Installing dependencies:
ruby3.2-default-gems                 noarch            3.2.8-184.amzn2023.0.1  amazonlinux       33 k
ruby3.2-libs                          x86_64            3.2.8-184.amzn2023.0.1  amazonlinux       3.7 M
ruby3.2-rubygem-io-console            x86_64            0.6.0-184.amzn2023.0.1  amazonlinux       22 k
ruby3.2-rubygem-json                  x86_64            2.6.3-184.amzn2023.0.1  amazonlinux       49 k
ruby3.2-rubygem-psych                  x86_64            5.0.1-184.amzn2023.0.1  amazonlinux       48 k
Installing weak dependencies:
ruby3.2-rubygem-bigdecimal            x86_64            3.1.3-184.amzn2023.0.1  amazonlinux       65 k
ruby3.2-rubygem-bundler                noarch            2.4.19-184.amzn2023.0.1  amazonlinux       383 k
ruby3.2-rubygem-rdoc                   noarch            6.5.1.1-184.amzn2023.0.1  amazonlinux       459 k
ruby3.2-rubygems                       noarch            3.4.19-184.amzn2023.0.1  amazonlinux       258 k
=====
Transaction Summary
-----
Install 10 Packages

Total download size: 5.0 M
Installed size: 12 M
```

EC2 > Instances > i-03966f106206d6006 > Modify IAM role

Modify IAM role Info

Attach an IAM role to your instance.

Instance ID

i-03966f106206d6006 (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-EC2-Role

Create new IAM role

Cancel

Update IAM role

aws [Search] [Alt+S] Global Pranit

Identity and Access Management (IAM) <

Search IAM

Dashboard

▼ Access management

- User groups
- Users
- Roles**
- Policies
- Identity providers
- Account settings
- Root access management [New](#)

▼ Access reports

- Access Analyzer
 - Resource analysis [New](#)
 - Unused access
 - Analyzer settings
- Credential report
- Organization activity

CodeDeploy-EC2-Role [Info](#) [Delete](#)

Allows EC2 instances to call AWS services on your behalf.

Summary [Edit](#)

Creation date June 24, 2025, 10:53 (UTC+05:30)	ARN arn:aws:iam::501406858319:role/CodeDeploy-EC2-Role	Instance profile ARN arn:aws:iam::501406858319:instance-profile/CodeDeploy-EC2-Role
Last activity -	Maximum session duration 1 hour	

Permissions Trust relationships Tags Last Accessed 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"/>	AmazonS3ReadOnlyAccess	AWS managed	1
<input type="checkbox"/>	AWSCodeDeployFullAccess	AWS managed	1

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

10:56 24-06-2025

>> **Step 2:** Create CodeCommit Repo, set-up Git and push the code

aws Services [Search] [Alt+S] Asia Pacific (Mumbai) Pranit

Developer Tools > CodeCommit > Repositories > Create repository

[AWS CodeCommit is no longer available to new customers. Existing customers of AWS CodeCommit can continue to use the service as normal. \[Learn more\]\(#\)](#)

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

1,000 characters maximum

Tags

[► Additional configuration](#)

Wanted162 / MyWebAppRepo

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

MyWebAppRepo Public

Pin Watch 0 Fork 0 Star 0

Set up GitHub Copilot
Use GitHub's AI pair programmer to autocomplete suggestions as you code.
[Get started with GitHub Copilot](#)

Add collaborators to this repository
Search for people using their GitHub username or email address.
[Invite collaborators](#)

Quick setup — if you've done this kind of thing before

[Set up in Desktop](#) or [HTTPS](#) [SSH](#) <https://github.com/Wanted162/MyWebAppRepo.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# MyWebAppRepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Wanted162/MyWebAppRepo.git
git push -u origin main
```

```
91866@LAPTOP-I9TPEN7J MINGW64 ~ (main)
$ git config --global user.name "Pranit"
git config --global user.email "pisalpranit1@gmail.com"

91866@LAPTOP-I9TPEN7J MINGW64 ~ (main)
$ mkdir MyWebAppRepo

91866@LAPTOP-I9TPEN7J MINGW64 ~ (main)
$ cd MyWebAppRepo

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ echo "<h1>Hello from AWS CodePipeline with GitHub!</h1>" > index.html
bash: !: event not found

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ echo "<h1>Hello from AWS CodePipeline with GitHub\!</h1>" > index.html

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ mkdir scripts

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ cat <<EOF > scripts/install_dependencies.sh
#!/bin/bash
sudo yum install -y httpd
sudo systemctl enable httpd
sudo systemctl start httpd
EOF
```

```

hooks:
  BeforeInstall:
    - location: scripts/install_dependencies.sh
      timeout: 300
      runas: root
  AfterInstall:
    - location: scripts/start_server.sh
      timeout: 300
      runas: root
EOF

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ git init
git add .
git commit -m "Initial commit for CodePipeline"
Initialized empty Git repository in C:/Users/91866/MyWebAppRepo/.git/
warning: in the working copy of 'appspec.yml', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'index.html', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'scripts/install_dependencies.sh', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'scripts/start_server.sh', LF will be replaced by CRLF the next time Git touches it
[master (root-commit) f64ff3f] Initial commit for CodePipeline
4 files changed, 21 insertions(+)
create mode 100644 appspec.yml
create mode 100644 index.html
create mode 100644 scripts/install_dependencies.sh
create mode 100644 scripts/start_server.sh

```

```


91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (master)
$ git remote add origin https://github.com/Wanted162/MyWebAppRepo.git

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (master)
$ git branch -M main

91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$ git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 720 bytes | 720.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Wanted162/MyWebAppRepo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.






91866@LAPTOP-I9TPEN7J MINGW64 ~/MyWebAppRepo (main)
$

```


MyWebAppRepo
Public
Pin
Watch 0

main
1 Branch
0 Tags

Add file
Code

 Wanted162	Initial commit for CodePipeline	f64ff3f · 1 hour ago	 1 Commit
 scripts	Initial commit for CodePipeline	1 hour ago	
 appspec.yml	Initial commit for CodePipeline	1 hour ago	
 index.html	Initial commit for CodePipeline	1 hour ago	

>> **Step 3:** Setup EC2 Instance for CodeDeploy and then create CodeDeploy Application and CodeDeploy Group

aws Services Search [Alt+S]

Developer Tools > CodeDeploy > Applications > Create application

Create application

Application configuration

Application name
Enter an application name

MyWebAppApp

100 character limit

Compute platform
Choose a compute platform

EC2/On-premises

Tags

Add tag

Cancel Create application

IAM > Roles > CodeDeployServiceRole

CodeDeployServiceRole Info

Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

Summary Edit

Creation date
June 24, 2025, 13:37 (UTC+05:30)

ARN
arn:aws:iam::501406858319:role/CodeDeployServiceRole

Last activity
-

Maximum session duration
1 hour

Permissions Trust relationships Tags Last Accessed Revoke sessions

Permissions policies (1) Info Simulate Remove Add permissions

You can attach up to 10 managed policies.

Filter by Type

Search

All types

Policy name	Type	Attached entities
<input type="checkbox"/> AWSCodeDeployRole	AWS managed	1

aws

Services

Q IAM

X

AWS Console Home

Developer Tools > CodeDeploy > Applications > MyWebAppApp > Create deployment group

Create deployment group

Application

Application

MyWebAppApp

Compute type

EC2/On-premises

Deployment group name

Enter a deployment group name

MyWebAppDeploymentGroup

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::501406858319:role/CodeDeployServiceRole

X

aws

Services

Q IAM

X

Asia Pacific (Mumbai)

Pranit

Developer Tools

CodeDeploy

Source • CodeCommit

Artifacts • CodeArtifact

Build • CodeBuild

▼ Deploy • CodeDeploy

Getting started

Deployments

Applications

Application

Settings

Deployment configurations

On-premises instances

► Pipeline • CodePipeline

► Settings

Q Go to resource

Feedback

Success

Deployment group created

X

?

?

?

Developer Tools > CodeDeploy > Applications > MyWebAppApp > MyWebAppDeploymentGroup

MyWebAppDeploymentGroup

EditDeleteCreate deployment

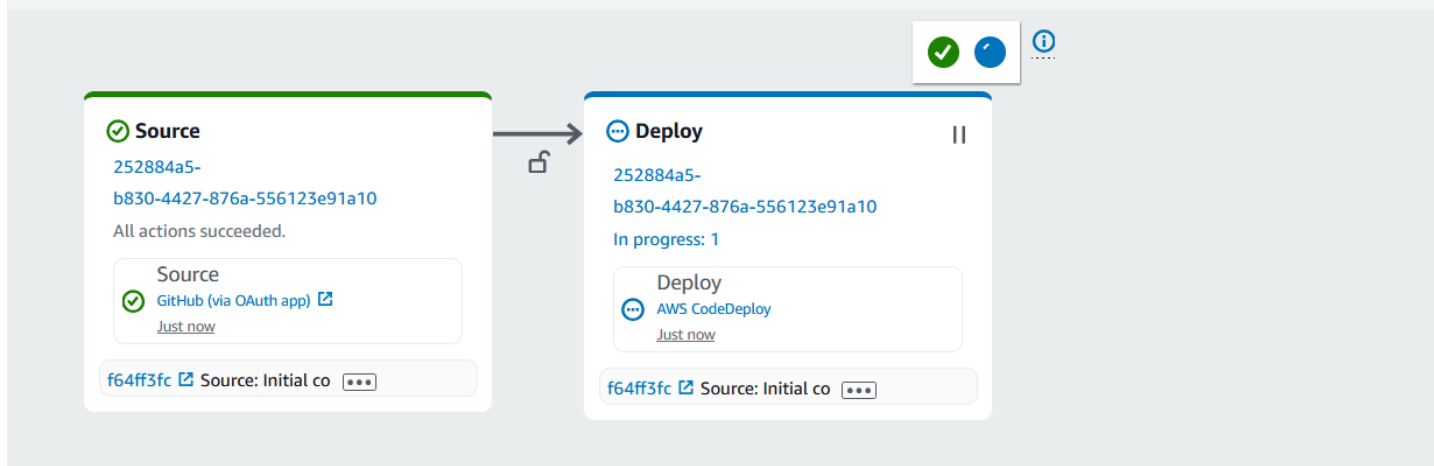
Deployment group details

Deployment group name	Application name	Compute platform
MyWebAppDeploymentGroup	MyWebAppApp	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::501406858319:role/CodeDeployServiceRole	CodeDeployDefault.AllAtOnce
Rollback enabled	Agent update scheduler	
False	Learn to schedule update in AWS Systems Manager	

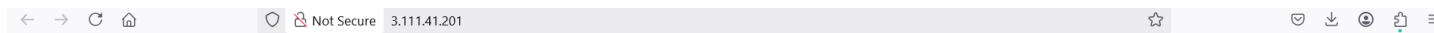
Environment configuration: Amazon EC2 instances

Key	Value
Name	code-deploy

>> **Step 4:** Create Code-Pipeline with Github source



>> **Final Step:** Access your site by clicking on your Public IP of EC2 instance which should show the message “It works!”



It works!