

Project Name : Aws Cloud based Calculator App Continuous Integration Pipeline

Description : we will deploy sample calculator app on AWS cloud using code-commit & through buildspec.yml & appspec.yml file we will build our code using aws codebuild and using code deploy will deploy our calculator application using CI /CD pipeline on ec2 instance.

What is buildspec.yml file ?

It is a configuration file used by AWS CodeBuild to define the build and deployment steps for a project. It provides a set of instructions for CodeBuild to follow during the build and deployment process, including dependencies, run tests, and package the application for deployment.

What is appspec.yml ?

It is a file used by AWS CodeDeploy to define the deployment instructions for an application. It is a YAML-formatted file that specifies the resources, files, permissions, and lifecycle events required for the deployment process.

Services required : IAM (user, roles , permissions) , CodeCommit ,Artifact(S3 bucket) CodeBuild , CodeDeploy , Pipeline

Steps:

1. First of all we need a repository where we will have our application & buildspec , appspec files , for that aws > codeCommit > repository > create New repo

aws

Services

Search [Alt+S]

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

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.

Developer Tools > CodeCommit > Repositories

Repositories info

Refresh

Notify

Clone URL

View repository

Delete repository

Create repository

Search

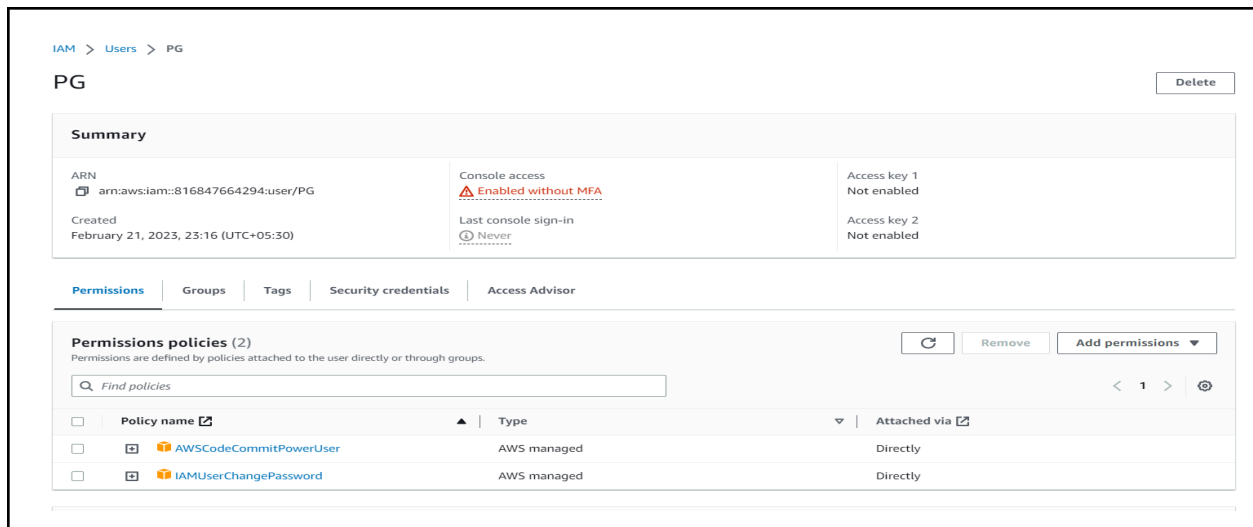
< 1 >

Settings

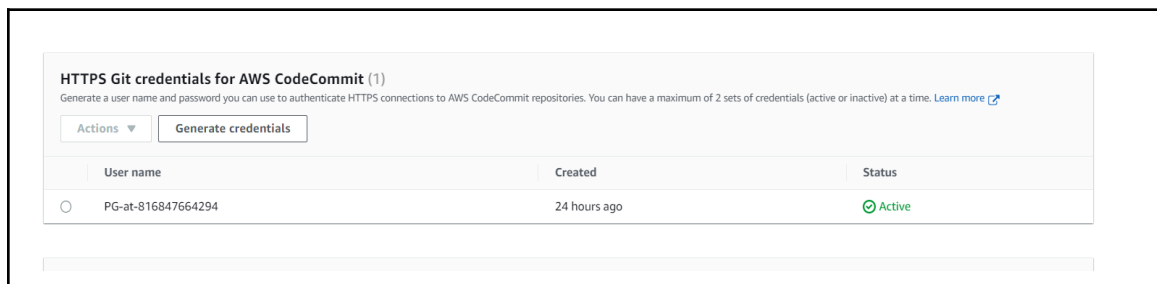
Name	Description	Last modified	Clone URL
☐ calculator-app	This is for calculator app	22 hours ago	HTTPS SSH HTTPS (GRC)

- Now a repo has been created and we need to make this repo on the local system in order to add our application and configuration file . For that let's create a user which will access this repo on local . aws > services > IAM > user > create user

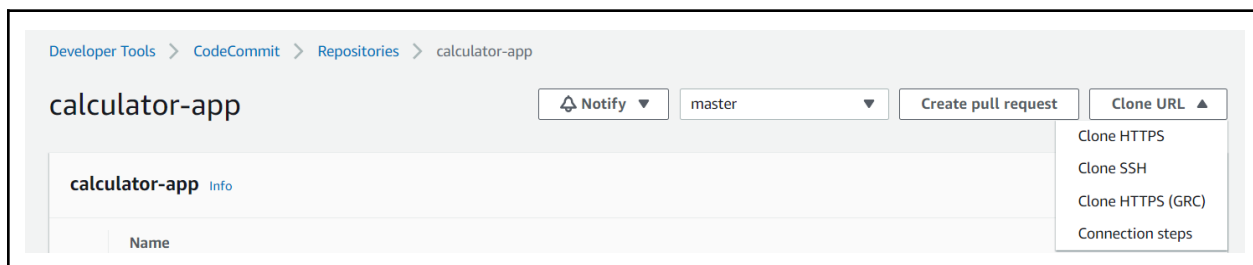
Note : Added AWSCodeCommitPowerUser permission in order to commit in this master repo from the local system.



- Let's generate codecommit credentials in order to make changes into this repo .
IAM > user > created User (in my case PG as you can see in above pic) > security cred > HTTPS Git credentials for AWS CodeCommit > generate



- Let's clone this repo on local for that codeCommit > repo > Clone Url > Clone HTTPS > git link



- Open any IDE and make a directory integrated terminal and clone this repo :
Git clone <repo link> , it will ask for username and password that we have generated on user credentials .

```
PS C:\Users\my\OneDrive\Desktop\aws_pipeline> git clone https://git-codecommit.ap-northeast-1.amazonaws.com/v1/repos/calculator-app
Cloning into 'calculator-app'...
warning: You appear to have cloned an empty repository.
```

6. Let's add our application file along with configuration file and commit into aws
Inside folder on local machine create a index.html file , buildspec.yml and appspec.yml
And some nginx script files that our application will run up .

```
✓ AWS_PIPELINE
  ✓ calculator-app
    ✓ scripts
      $ install_nginx.sh
      $ start_nginx.sh
      ! appspec.yml
      ! buildspec.yml
      <> index.html
```

```
<> index.html X ! buildspec.yml ! appspec.yml $ install_nginx.sh $ start_nginx.sh
calculator-app > <> index.html > ...
 1  <!DOCTYPE html>
 2  <html>
 3
 4  <head>
 5      <script src=
 6      "https://cdnjs.cloudflare.com/ajax/libs/mathjs/10.6.4/math.js"
 7      integrity=
 8      "sha512-BbVEDjbqdn3Eow8+empLMrJlxXRxj5nEitiCAK5A1pUr66+jLVeJo3PmJIaucRnjlB0P9R3rBUS3g5jXc8ti+fQ=="
 9      crossorigin="anonymous"
10      referrerpolicy="no-referrer"></script>
11      <script src=
12      "https://cdnjs.cloudflare.com/ajax/libs/mathjs/10.6.4/math.min.js"
13      integrity=
14      "sha512-iphNRh6dPbeuPGIrQbCdbBF/qcqadKWL335YPVfMZMHBSI6PLJh1om2xCTWhpVpmUyb4IvVS9iYnnYMkleVXLA=="
15      crossorigin="anonymous"
16      referrerpolicy="no-referrer"></script>
17      <!-- for styling -->
18      <style>
19          table {
20              border: 1px solid □black;
```

index.html

```
calculator-app > ! buildspec.yml
1  version: 0.2
2  phases:
3    install:
4      commands:
5        - echo installing NGINX
6        - sudo apt-get update
7        - sudo apt-get install nginx -y
8    build:
9      commands:
10       - echo Building started on `date`
11       - cp index.html /var/www/html/
12       - zip -r calapp.zip .
13    post_build:
14      command:
15        - echo Configuring nginx
16  artifacts:
17    files:
18      - '**/*'
```

Buildspec.yml file

```
calculator-app > ! appspec.yml
1  version: 0.0
2  os: linux
3  files:
4    - source: /
5      destination: /var/www/html
6  hooks:
7    AfterInstall:
8      - location: scripts/install_nginx.sh
9        timeout: 300
10       runas: root
11    ApplicationStart:
12      - location: scripts/start_nginx.sh
13        timeout: 300
14       runas: root
```

appspec.yml

```
calculator-app > scripts > $ install_nginx.sh
1  #!/bin/bash
2
3  sudo apt-get update
4  sudo apt-get install -y nginx
```

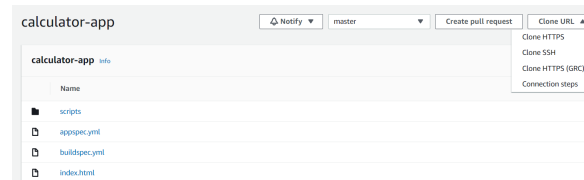
install_nginx.sh(script that will install all dependencies as per our application)

```
calculator-app > scripts > $ start_nginx.sh
1  #!/bin/bash
2
3  sudo service nginx start
```

Start_nginx.sh (script file which will start our service inorder to run up appspec.file)

7. Let's commit all files into repo using the following steps :
 - a. Git add . (adding all files into staging phase)
 - b. Git commit -m "added all files" (commit version with specified message)
 - c. Git push origin master (since our repo has master branch)

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Users\my\OneDrive\Desktop\aws_pipeline\calculator-app> git add .
PS C:\Users\my\OneDrive\Desktop\aws_pipeline\calculator-app> git commit -m "added index.html"
(master root-commit) 2d2e230 added index.html
1 file changed, 141 insertions(+)
create mode 100644 index.html
PS C:\Users\my\OneDrive\Desktop\aws_pipeline\calculator-app> git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 1.35 KiB | 460.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Validating objects: 100%
To https://git-codecommit.ap-northeast-1.amazonaws.com/v1/repos/calculator-app
 * [new branch]      master -> master
PS C:\Users\my\OneDrive\Desktop\aws_pipeline\calculator-app> git add .
PS C:\Users\my\OneDrive\Desktop\aws_pipeline\calculator-app> git commit -m "buildspec"
```



As we can see all files pushed successfully

8. It is time to build our code manually to check whether it is working or not . for that aws > codeBuild > build projects > specify your all details like code will come from aws codeCommit > build now

calculator-app-build:67f6e6db-8029-46c0-bede-094fd0f82394

Stop build

Retry build

Build status

Status

Succeeded

Initiator

codepipeline/calculator_pipeline_final

Build ARN

arn:aws:codebuild:ap-northeast-1:816847664294:build/calculator-app-build:67f6e6db-8029-46c0-bede-094fd0f82394

Resolved source version

06d9bb2024b319002f6b24154dd22f2ebf3c37e

Start time

Feb 22, 2023 1:30 AM (UTC+5:30)

End time

Feb 22, 2023 1:31 AM (UTC+5:30)

Build number

14

Build logs

Phase details

Reports

Environment variables

Build details

Resource utilization

Name	Status	Context	Duration	Start time	End time
SUBMITTED	Succeeded	-	<1 sec	Feb 22, 2023 1:30 AM (UTC+5:30)	Feb 22, 2023 1:30 AM (UTC+5:30)
QUEUED	Succeeded	-	13 secs	Feb 22, 2023 1:30 AM (UTC+5:30)	Feb 22, 2023 1:30 AM (UTC+5:30)
PROVISIONING	Succeeded	-	29 secs	Feb 22, 2023 1:30 AM (UTC+5:30)	Feb 22, 2023 1:30 AM (UTC+5:30)
DOWNLOAD_SOURCE	Succeeded	-	3 secs	Feb 22, 2023 1:30 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
INSTALL	Succeeded	-	42 secs	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
PRE_BUILD	Succeeded	-	<1 sec	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
BUILD	Succeeded	-	<1 sec	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
POST_BUILD	Succeeded	-	<1 sec	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
UPLOAD_ARTIFACTS	Succeeded	-	<1 sec	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
FINALIZING	Succeeded	-	<1 sec	Feb 22, 2023 1:31 AM (UTC+5:30)	Feb 22, 2023 1:31 AM (UTC+5:30)
COMPLETED	Succeeded	-	-	Feb 22, 2023 1:31 AM (UTC+5:30)	-

As you can see my code build status is successful.

Developer Tools > CodeBuild > Build projects > calculator-app-build

calculator-app-build

Notify

▼

Share

Edit ▼

Delete build project

Start build with overrides

Start build

Configuration

Source provider

AWS CodeCommit

Primary repository

calculator-app

Artifacts upload location

calculator-bucket-new

Build badge

Disabled

Public builds

Disabled

Build history

Batch history

Build details

Build triggers

Metrics

Here i have specified artifact location as well which is s3 bucket where my zip file will generated (basically my application image will be generated)

9. Now let’s make a deployment group and deploy our app . So a deployment group is needed in order to run up our deployment .

Here i have created deployment application

Here I have created a deployment group with code-deploy service role where all permission provided in this service.

Provided my ec2 instance as well where we will deploy our application.

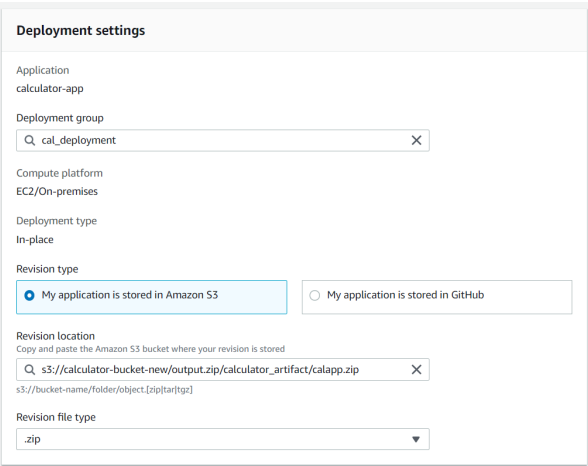
10. Now let's create ec2 application and install agent manually on instance for that :
 Aws > ec2 > launch instance & we will run a script with specified region in my case it is Tokyo
 which is ap-northeast-1

Ec2 instance running and up
(t2.micro)

This is the script for installing agent on ec2

And here is our agent running and active

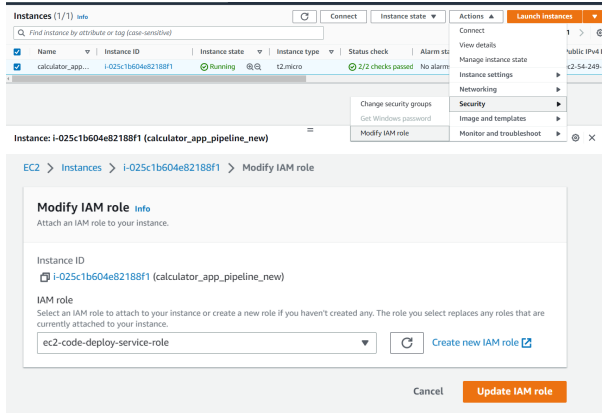
11. Let's make a deployment :
 Aws > code deploy > application > create deployment



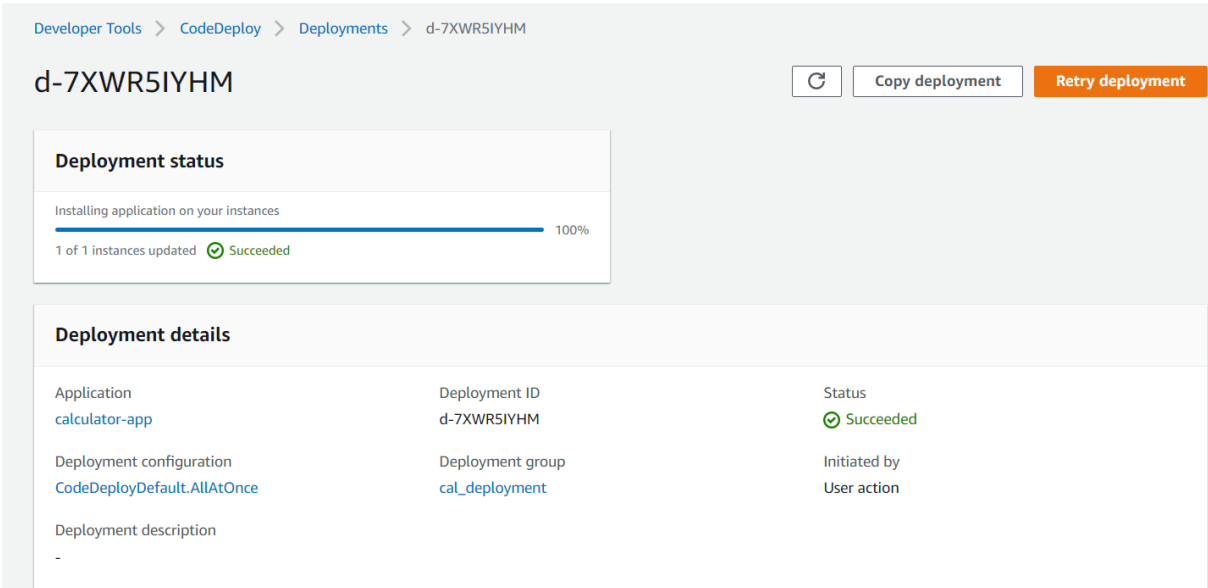
I have specified my artifact location and deployment group which we created earlier

Now we need ec2 code deploy service role in order to install all into deployment

Ec2 > select instance > security > modify iam role > chose ec2 code deploy role



Now installation completed :



12. Now let's create a CI / CD pipeline into aws pipeline : aws > developer tool > codePipeline > create pipeline which will be source -> build -> deploy

Overall steps in pipeline : source > build > deploy

Source step

Edit action

Action name

Choose a name for your action

Build

No more than 100 characters

Action provider

AWS CodeBuild

Region

Asia Pacific (Tokyo)

Input artifacts

Choose an input artifact for this action. [Learn more](#)

SourceArtifact

Add

No more than 100 characters

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.

calculator-app-build

X

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.

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](#)

BuildVariables

Output artifacts

Choose a name for the output of this action.

BuildArtifact

Add

No more than 100 characters

Cancel

Done

Build step

Edit action

Action name

Choose a name for your action

Deploy

No more than 100 characters

Action provider

AWS CodeDeploy

Region

Asia Pacific (Tokyo)

Input artifacts

Choose an input artifact for this action. [Learn more](#)

BuildArtifact

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.

calculator-app

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.

cal_deployment

X

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](#)

DeployVariables

Cancel

Done

Deploy step

13. Final step let's build our CI /CD pipeline automation :

Click save pipeline settings > start build and it will start automatically and in background it will tell us build process , steps and etc

The screenshot displays the AWS CodePipeline console for a pipeline named 'calculator_pipeline_final'. The breadcrumb navigation at the top reads 'Developer Tools > CodePipeline > Pipelines > calculator_pipeline_final'. The pipeline is shown with a green vertical bar on the left, indicating it is in a 'Succeeded' state. The stages and their details are as follows:

- Source Stage:** Succeeded. Pipeline execution ID: 54babb0e-658b-436c-ad54-2a67d7423b55. The stage box shows 'Source' using 'AWS CodeCommit', 'Succeeded - 1 day ago', and ID '06d9bb20'. Below the box, it says '06d9bb20 Source: end line revised'.
- Build Stage:** Succeeded. Pipeline execution ID: 54babb0e-658b-436c-ad54-2a67d7423b55. The stage box shows 'Build' using 'AWS CodeBuild', 'Succeeded - 1 day ago', and a 'Details' link. Below the box, it says '06d9bb20 Source: end line revised'.
- Deploy Stage:** Succeeded. Pipeline execution ID: 54babb0e-658b-436c-ad54-2a67d7423b55. The stage box shows 'Deploy' using 'AWS CodeDeploy', 'Succeeded - 23 hours ago', and a 'Details' link. Below the box, it says '06d9bb20 Source: end line revised'.

Between the stages, there are downward arrows and a 'Disable transition' button.

Our pipeline is succeeded it means our all code successfully passed .Now let's check our ec2 instance public IP our calculator app should be running :

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

Connect

Instance state ▾

Actions ▾

Launch Instances

☒	Name ▾	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 D
☒	calculator_app...	i-025c1b604e82188f1	Running	t2.micro	2/2 checks passed	No alarms +	ap-northeast-1a	ec2-54-249-8

Instance: i-025c1b604e82188f1 (calculator_app_pipeline_new)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Instance summary [Info](#)

Instance ID i-025c1b604e82188f1 (calculator_app_pipeline_new)	Public IPv4 address 54.249.86.37 open address	Private IPv4 addresses 172.31.33.152
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-54-249-86-37.ap-northeast-1.compute.amazonaws.com open address
Hostname type IP name: ip-172-31-33-152.ap-northeast-1.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-33-152.ap-northeast-1.compute.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address	VPC ID	AWS Compute Optimizer finding