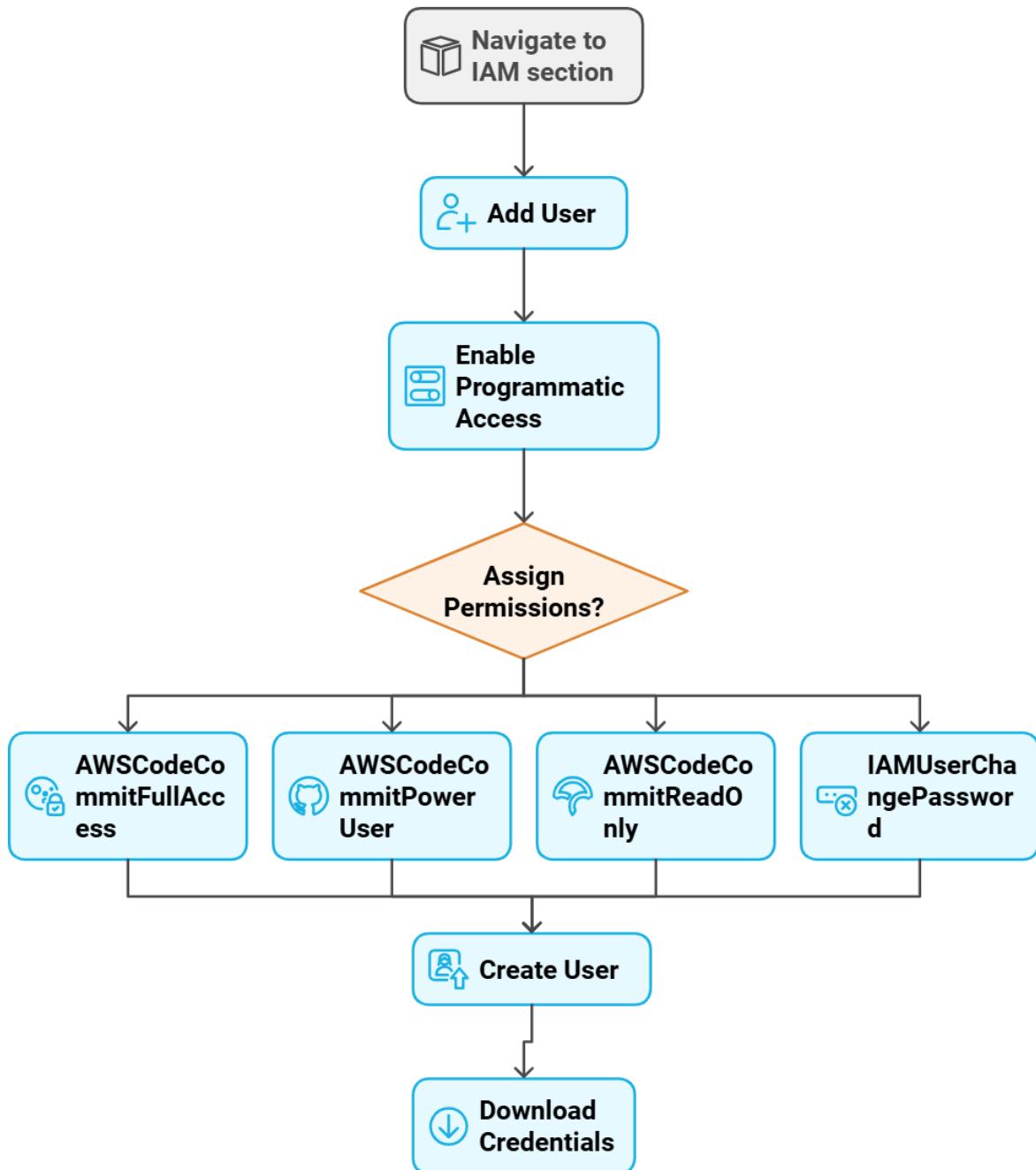


Title:- Streamlined Web Deployment: Building a CICD Pipeline with AWS CodePipeline and CodeDeploy

STEP1:-Create User



Jayshree Patil.

The screenshot shows the AWS IAM 'Users' page. The top navigation bar includes the AWS logo, search bar, and global settings. The left sidebar has 'Identity and Access Management (IAM)' selected under 'Access management'. The main content area displays 'Users (1/1) Info' with a table header: User name, Path, Group, Last activity, MFA, Password age, and Console last sign-in. A single row is shown for 'Jayshreouser' with a status of '24 hours'. Action buttons for 'Delete' and 'Create user' are at the top right, and a search bar is at the bottom left.

IAM>USER>CREATE USER>TICK(I WANT AM)>CONSOLE PASSWORD(CUSTOM)

This screenshot shows the 'Specify user details' step of the IAM User creation wizard. The left sidebar lists steps: Step 1 (Specify user details), Step 2 (Set permissions), Step 3 (Review and create), and Step 4 (Retrieve password). The main area is titled 'User details' and contains a 'User name' field with 'Jayshreouser' entered. Below it is a note about character restrictions. A checkbox 'Provide user access to the AWS Management Console - optional' is checked, with a note about using Identity Center for console access. A callout box asks if the user wants to provide console access to a person, with two options: 'Specify a user in Identity Center - Recommended' (radio button unselected) and 'I want to create an IAM user' (radio button selected). The 'Console password' section follows, with a radio button for 'Autogenerated password' (unselected) and 'Custom password' (selected), showing a masked password input field. A note specifies password requirements: at least 8 characters, mix of uppercase, lowercase, numbers, and symbols. A 'Show password' checkbox is present. A checkbox 'Users must create a new password at next sign-in - Recommended' is checked, with a note about IAMUserChangePassword policy. A callout box provides information about generating programmatic access keys. At the bottom are 'Cancel' and 'Next' buttons.

Jayshree Patil.

IAM > Users > Create user

Step 1 Specify user details

Step 2 Set permissions

Step 3 Review and create

Step 4 Retrieve password

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

i Get started with groups
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

[Create group](#)

► Set permissions boundary - optional

Cancel Previous Next

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

Step 2 Set permissions

Step 3 Review and create

Step 4 Retrieve password

User details

User name	Console password type	Require password reset
Jayshreeuser	Custom password	Yes

Permissions summary

Name	Type	Used as
IAMUserChangePassword	AWS managed	Permissions policy

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous [Create user](#)

Set permissions

Permissions policies (4)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type

All types

Policy name

▲ Type

▼ Attached via

 [AWSCodeCommitFullAccess](#)

AWS managed

Directly

 [AWSCodeCommitPowerUser](#)

AWS managed

Directly

 [AWSCodeCommitReadOnly](#)

AWS managed

Directly

 [IAMUserChangePassword](#)

AWS managed

Directly



Remove

Add permissions

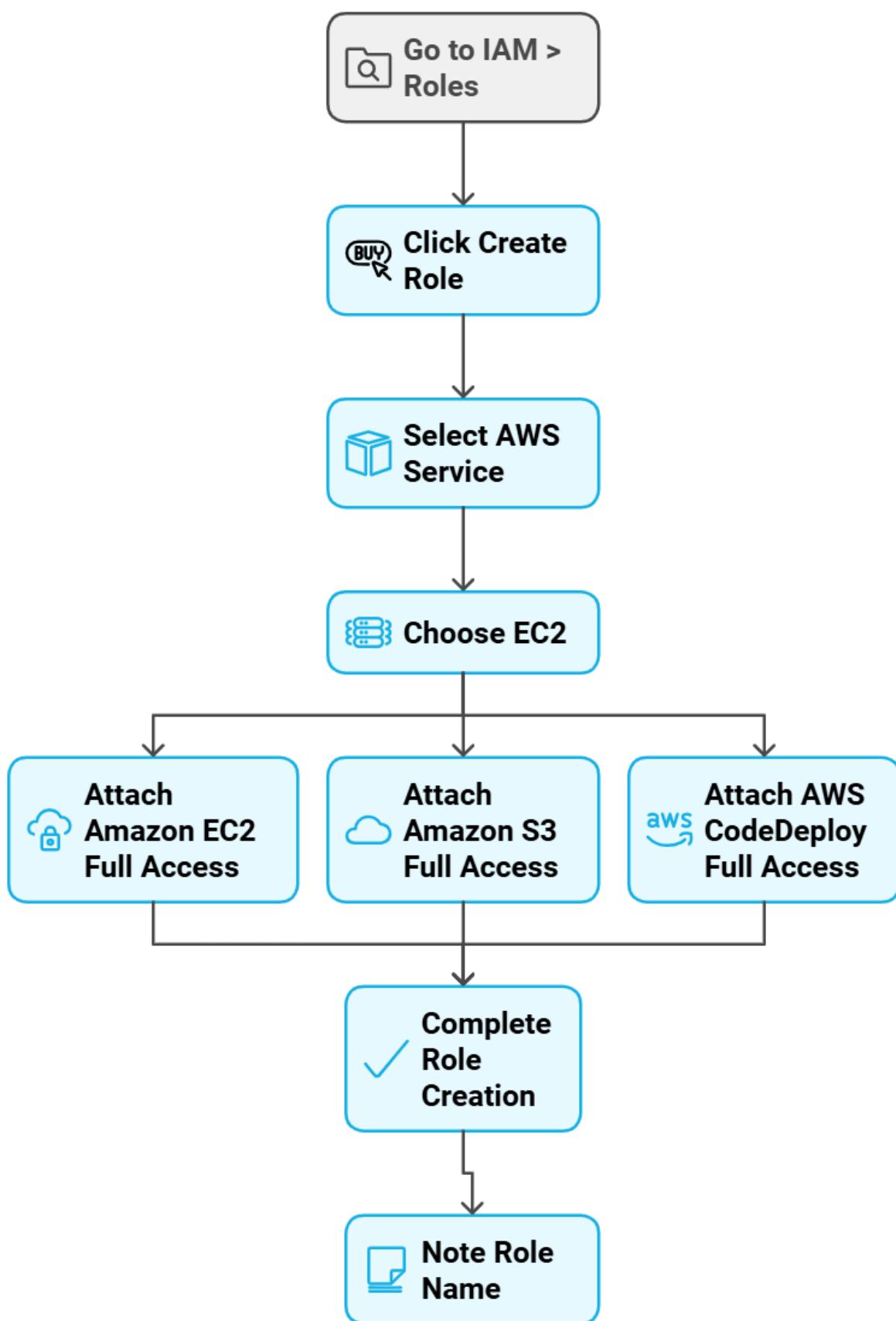
Jayshree Patil.

Once the user created >go to user security>httpgit credentials generate>download

The screenshot shows the AWS Identity and Access Management (IAM) console. On the left, there's a sidebar with options like Dashboard, Access management, User groups, and Users (which is selected). The main area is titled "HTTPS Git credentials for AWS CodeCommit (0)". It displays a message: "Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can have a maximum of 2 sets of credentials (active or inactive) at a time. [Learn more](#)". Below this is a table with columns "User name", "Created", and "Status". A button "Generate credentials" is located at the bottom right of the table area.

This screenshot shows a modal window titled "Generate credentials" with the message "Your new credentials are available." A note below it says, "Save your user name and password or download the credentials file. This is the only time you can view the password or download it. You cannot recover it later. However, you can reset your password at any time." It also mentions, "You can use these credentials when connecting from your local computer, or from tools that require a static user name and password. [Learn more](#)". The modal includes fields for "User name" (Jayshreeuser-at-762233745497) and "Password" (represented by a series of asterisks). At the bottom are buttons for "Download credentials" and "Close".

STEP2:- Create Role For EC2



Jayshree Patil.

The screenshot shows the 'Select trusted entity' step of the 'Create role' wizard. It displays a 'Trusted entity type' section with five options: 'AWS service' (selected), 'AWS account', 'Web identity', 'SAML 2.0 federation', and 'Custom trust policy'. Below this is a 'Use case' section for 'EC2'.

The screenshot shows the 'Name, review, and create' step of the 'Create role' wizard. It displays 'Role details' with a 'Role name' of 'ec2_role' and a 'Description' of 'Allows EC2 instances to call AWS services on your behalf.' Below this is the 'Step 1: Select trusted entities' section, which includes a 'Trust policy' code editor showing a JSON-based policy definition.

The screenshot shows the final 'Name, review, and create' step of the 'Create role' wizard. It lists three available policies: 'AmazonEC2FullAccess', 'AmazonS3FullAccess', and 'AWSCodeDeployFullAccess'. Each policy has a checkbox and a '+' icon.

Step 4:-Create Role For Codedepoly

The screenshots illustrate the process of creating a new IAM role:

- Step 1: Select trusted entity**
 - Shows the 'Trusted entity type' section with options: AWS service (selected), AWS account, SAML 2.0 federation, and Web identity.
 - Shows the 'Use case' section with 'Service or use case' set to 'CodeDeploy'.
 - Shows the 'Permissions policies' section with 'AWSCodeDeployRole' selected.
 - Shows the 'Set permissions boundary - optional' section with 'Create role without a permissions boundary' selected.
- Step 2: Add permissions**
 - Shows the 'Role details' section with 'Role name' set to 'codedeploy_role'.
 - Shows the 'Description' section with 'Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.'
- Step 3: Name, review, and create**
 - Shows the 'Step 1: Select trusted entities' section with a JSON trust policy defined.

Jayshree Patil.

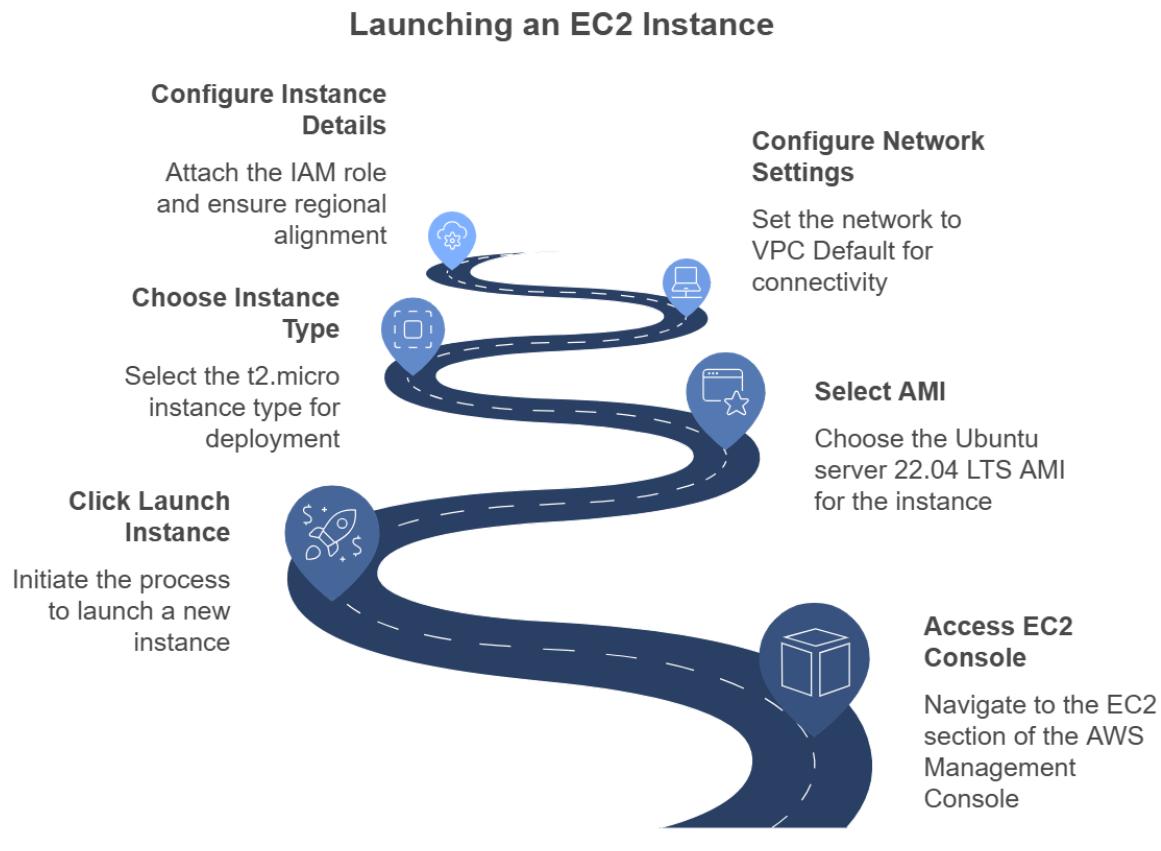
Then go to permissions and add other permissions

The screenshot shows the AWS IAM Role configuration page for a role named 'codedeploy_role'. The left sidebar shows navigation options like Dashboard, Access management, Policies, and Access reports. The main panel displays the 'Permissions policies' section, which lists one policy: 'AWSCodeDeployRole' (AWS managed). There are sections for 'Permissions boundary' and 'Generate policy based on CloudTrail events'. A 'Filter by Type' dropdown is set to 'All types'. Buttons at the top right include 'View role', 'Simulate', 'Remove', and 'Add permissions'.

The screenshot shows the 'Permissions policies' list with 5 items. The policies listed are: 'AmazonEC2FullAccess', 'AmazonEC2RoleforAWSCodeDeploy', 'AmazonS3FullAccess', 'AmazonS3ReadOnlyAccess', and 'AWSCodeDeployFullAccess'. All policies are AWS managed and have 2 attached entities. A 'Filter by Type' dropdown is set to 'All types'. Buttons at the top right include 'Simulate', 'Remove', and 'Add permissions'.

Policy name	Type	Attached entities
AmazonEC2FullAccess	AWS managed	2
AmazonEC2RoleforAWSCodeDeploy	AWS managed	2
AmazonS3FullAccess	AWS managed	2
AmazonS3ReadOnlyAccess	AWS managed	1
AWSCodeDeployFullAccess	AWS managed	2

STEP5:- Launch Instance(Ubantu)



[EC2](#) > ... > [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

mycid_ec2

[Add additional tags](#)

Recents Quick Start

Memory usage: 258 MB

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 Free tier eligible

ami-03fa85deedfcac80b (64-bit (x86)) / ami-057318957b68ae793 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Ubuntu Server 22.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2024-09-27

Architecture	AMI ID	Username	Verified provider
64-bit (x86)	ami-03fa85deedfcac80b	ubuntu	Verified provider

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t2.micro Free tier eligible

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0164 USD per Hour
On-Demand Linux base pricing: 0.0146 USD per Hour
On-Demand Windows base pricing: 0.0192 USD per Hour
On-Demand RHEL base pricing: 0.029 USD per Hour
On-Demand SUSE base pricing: 0.0146 USD per Hour

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ 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

CICD Create new key pair

Jayshree Patil.

▼ Network settings [Info](#)

VPC - required | [Info](#)

vpc-01e949dcbb2bcc4be (default) (default) ▾

172.31.0.0/16

Subnet | [Info](#)

subnet-Offa10186019e5aca 3

VPC: vpc-01e949dcbb2bcc4be Owner: 762233745497

Availability Zone: ap-southeast-1a Zone type: Availability Zone

IP addresses available: 4091 CIDR: 172.31.32.0/20

Auto-assign public IP | [Info](#)

Enable

Additional charges apply when outside of free tier allowance

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

Common security groups | [Info](#)

Select security groups

CICD-SG sg-0ded27dff73e27312 X

VPC: vpc-01e949dcbb2bcc4be

Compare security group rules

Security groups that you add or remove here will be added to or removed from all your network interfaces.

► Advanced network configuration

▼ Advanced details [Info](#)

Domain join directory | [Info](#)

Select

Create new directory

IAM instance profile | [Info](#)

ec2_role

arn:aws:iam::762233745497:instance-profile/ec2_role

Create new IAM profile

Hostname type | [Info](#)

IP name

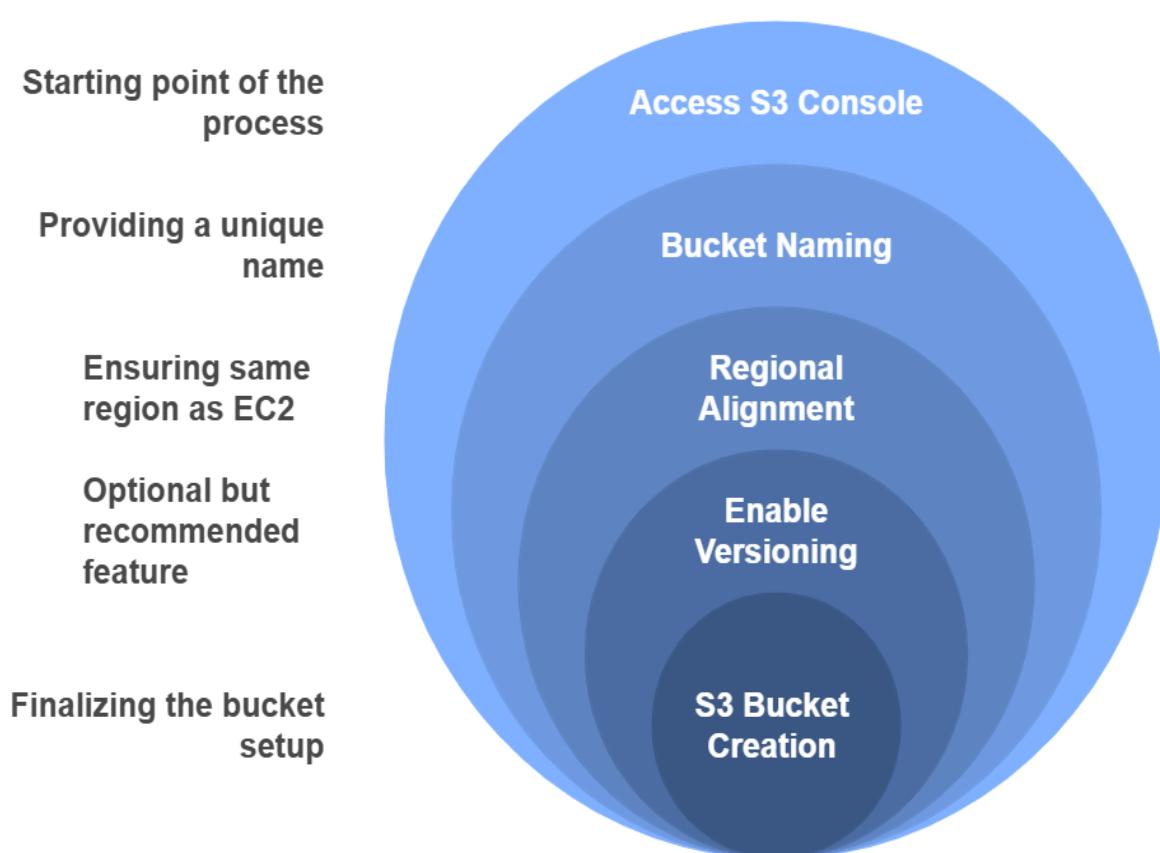
DNS Hostname | [Info](#)

(note :- attach ec2_role)

STEP4:-Create Bucket

1. Go to **S3** in the AWS Management Console.
2. Click **Create Bucket**.
3. Provide a unique name and ensure the bucket is in the same region as the EC2 instance.

S3 Bucket Creation Process



Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region
Asia Pacific (Singapore) ap-southeast-1

Bucket name Info
 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](#)

Format: s3://bucket/prefix

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)
All objects in this bucket are owned by this account.
Access to this bucket and its objects is specified using only policies.

ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership
Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.

Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point

Jayshree Patil.

Default encryption [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type | [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the **Storage** tab of the [Amazon S3 pricing page](#).

Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable

Enable

► Advanced settings

Info After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

[Cancel](#) [Create bucket](#)

Amazon S3 > Buckets

Account snapshot - updated every 24 hours [All AWS Regions](#) [View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

[General purpose buckets](#) [Directory buckets](#)

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

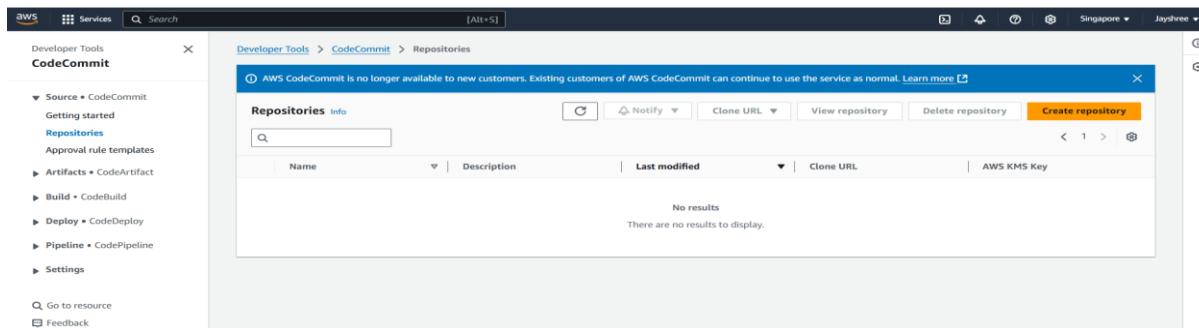
Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
mycicdbucket0311	Asia Pacific (Singapore) ap-southeast-1	View analyzer for ap-southeast-1	November 16, 2024, 10:10:23 (UTC+05:30)

STEP5:- GO TO CODECOMMIT AND CREATE REPOSITORY

Set Up CodeCommit Repository

1. Navigate to CodeCommit in the AWS Management Console.
2. Click Create Repository and provide a name for your 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 tag

► Additional configuration

AWS KMS key

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**

Jayshree Patil.

After creation:

- Copy the HTTPS or SSH URL of the repository.
- Use the credentials generated in Step 1 to interact with the repository.

Local Setup for CodeCommit

1. Open VS Code or your preferred IDE.

(CREATE FOLDER ON DESKTOP → UPLOAD ON VS CODE → CLICK LEFT AND CREATE ONE MORE FOLDER ON VS CODE GIVE NAME → THEN OPEN INTERGRADED TERMINAL AND CLONE THE REPO.-->create file index.html)

2. Clone the repository: git clone <repository-url>

3. Use the credentials generated in Step 1 to interact with the repository.(Put password and username)

4. Navigate to the cloned repository directory:- cd <repository-name>

5. git init

6. ls

7. Add your project files:

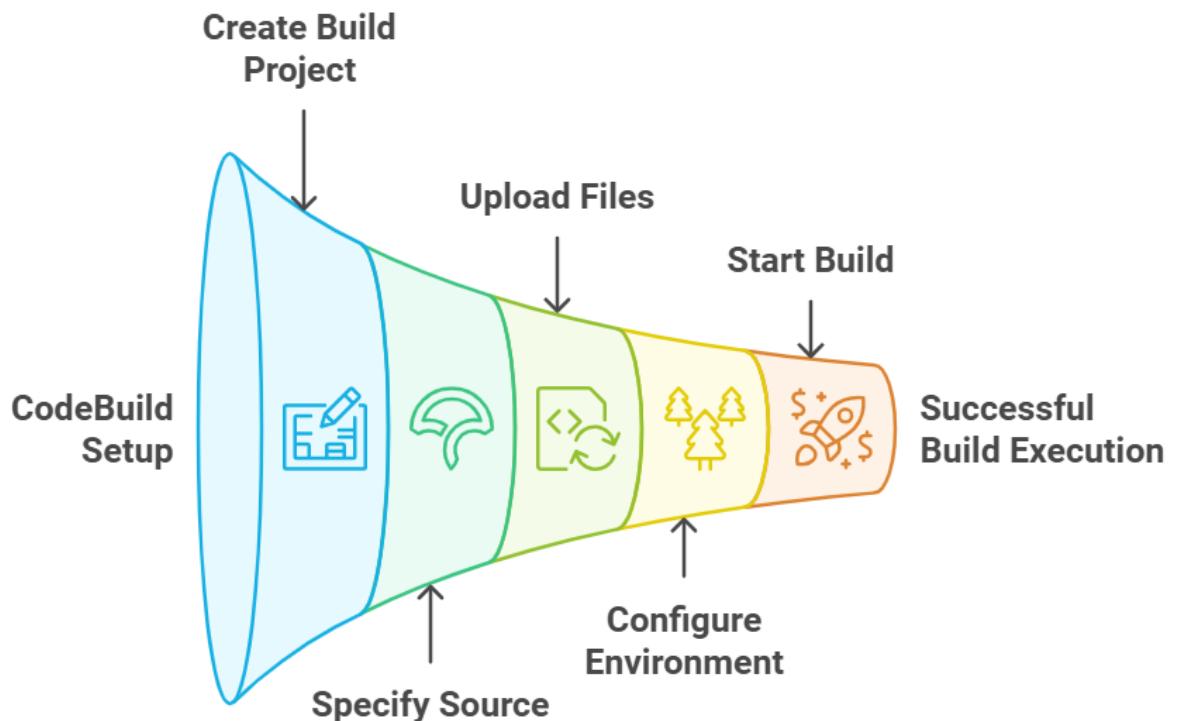
- Create a **index.html** file or other necessary files.
- **git add .**
- **git commit -m "Initial commit"**
- **git push origin master**

(STEP 6:-GO TO CODEBUILD (NOTE ;- BEFORE CREATING CODEBUILD CREATE EC2, AND IN VS CODE FOLDER UPLOAD buildspec and appspec and again pull both files and amazon s3 bucket also create)

STEP 6: Configure CodeBuild

1. Go to **CodeBuild** in the AWS Management Console.
2. Click **Create Build Project**.
3. Provide a name and specify the source:
 - Choose **CodeCommit** as the source provider.
 - Select the repository created in Step 5.
4. Upload the necessary files:
 - Include buildspec.yml in the repository root directory.
 - Add an appspec.yml file for CodeDeploy (if applicable).
5. Configure the environment:
 - Use an AWS-managed image for the build environment (e.g., Ubuntu).
6. Start the build and verify successful execution.

Setting Up a CodeBuild Project



Jayshree Patil.

Developer Tools > CodeBuild > Build projects

Name	Source provider	Repository	Latest build status	Description	Last Modified
No results There are no results to display.					

Create project

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

Create build project

Project configuration

Project name

mycodebuildproject

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 _.

Public build access - optional

Public build access allows you to make the build results, including logs and artifacts, for this project available for the general public.

Enable public build access

► Additional configuration

Description, Build badge, Concurrent build limit, tags

Source

Add source

Source 1 - Primary

Source provider

AWS CodeCommit

Repository

Codecommitrepository

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.

refs/heads/master

f7f9b15a buildspec

Git clone depth - optional

1

Git submodules - optional

Use Git submodules

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

Ubuntu

Runtime(s)

Standard

Queued timeout
Default time in build queue is 8 hours

Hours Minutes

8 0

Timeout must be between 5 minutes and 8 hours

Privileged

Enable this flag if you want to build Docker images or want your builds to get elevated privileges

Report auto-discover [Info](#) Disable report auto-discover

Auto-discover directory - optional

**/*

CodeBuild will search for supported report file types in this directory. **/* by default

Certificate
If you have a self-signed certificate or a certificate signed by a certification authority, choose the option to install it from your S3 bucket.

Do not install any certificate Install certificate from your S3 bucket

VPC
Select a VPC that your AWS CodeBuild project will access.

▼

Compute

3 GB memory, 2 vCPUs 7 GB memory, 4 vCPUs 15 GB memory, 8 vCPUs 70 GB memory, 36 vCPUs

Image

aws/codebuild/standard:7.0 ▼

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

codebuild-mycodebuildproject-service-role

Type your service role name

▼ **Additional configuration**
Timeout, privileged, certificate, VPC, compute type, environment variables, file systems, auto-retry

Auto-retry limit
CodeBuild will automatically call retry build using the project's service role up to the auto-retry limit

0

Timeout
Default timeout is 1 hour

Hours Minutes

1 0

Timeout must be between 5 minutes and 36 hours

Buildspec

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`

Batch configuration

You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.

Define batch configuration - *optional*

You can also define or override batch configuration when starting a build batch.

Artifacts packaging

None

The artifact files will be uploaded to the bucket.

Zip

AWS CodeBuild will upload artifacts into a compressed file that is put into the specified bucket.

Disable artifact encryption

Disable encryption if using the artifact to publish a static website or sharing content with others

► **Additional configuration**

Cache, encryption key

Logs

CloudWatch

CloudWatch logs - *optional*

Checking this option will upload build output logs to CloudWatch.

Group name - *optional*

`aws/codebuild/mycodebuildproject`

The group name of the logs in CloudWatch Logs. The log group name will be /aws/codebuild/<project-name> by default.

Stream name prefix - *optional*

The prefix of the stream name of the CloudWatch Logs.

S3

Jayshree Patil.

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

X

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.

▾

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.

Namespace type - *optional*

▾

Choose Build ID to insert the build ID into the path to the build output ZIP file or folder, e.g. MyPath/MyBuildID/MyArtifact.zip. Otherwise, choose None.

STEP7: Create an Application in CodeDeploy

1. Navigate to **CodeDeploy** in the AWS Management Console.
2. Click **Create Application**.
3. Choose the compute platform (e.g., **EC2/On-Premises**).
4. Create a Deployment Group:
 - o Link it to the application.
 - o Specify the EC2 instances using tags or Auto Scaling Groups.
 - o Use the service role created earlier.
5. If errors occur during deployment, ensure:

- o CodeDeploy agent is installed and active on the EC2 instance:

```
#!/bin/bash

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

sudo apt-get update

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

cd /tmp

wget https://aws-codedeploy-ap-southeast-1.s3.ap-southeast-
1.amazonaws.com/releases/codedeploy-agent_1.3.2-1902_all.deb

mkdir codedeploy-agent_1.3.2-1902_ubuntu22

dpkg-deb -R codedeploy-agent_1.3.2-1902_all.deb codedeploy-agent_1.3.2-
1902_ubuntu22

sed 's/Depends:.*/Depends:ruby3.0/' -i ./codedeploy-agent_1.3.2-
1902_ubuntu22/DEBIAN/control

dpkg-deb -b codedeploy-agent_1.3.2-1902_ubuntu22/

sudo dpkg -i codedeploy-agent_1.3.2-1902_ubuntu22.deb

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

sudo service codedeploy-agent status
```

Jayshree Patil.

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

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

Jayshree Patil.

AFTER CREATING APPLICATION CREATE DEPLACEMENT GROUP

The screenshot shows the AWS CodeDeploy application details page for 'mycodedeply1'. A green banner at the top states 'Application created' with the sub-instruction 'In order to create a new deployment, you must first create a deployment group.' Below this, the 'Deployment groups' tab is selected, showing a table with one row: 'Name' (mycodedeply1), 'Status' (No deployment groups), 'Last attempted deployment' (N/A), 'Last successful deployment' (N/A), and 'Trigger count' (0). A prominent orange 'Create deployment group' button is located at the bottom right of the table.

The screenshot shows the 'Create deployment group' wizard. Step 1: Application. It displays the selected application 'mycodedeply1' with its compute type set to 'EC2/On-premises'. Step 2: Deployment group name. It shows a text input field containing 'deploymentgroup' with a character limit of 100 characters. Step 3: Service role. It shows a search input field containing 'arn:aws:iam::762233745497:role/codeploy_role'.

Services Search [Alt+S]

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	Remove tag
<input type="text" value="Name"/> X	<input type="text" value="mycid_ec2"/> X	Remove tag

Add tag

+ Add tag group

On-premises instances

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

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

⚠ We recommend configuring your CodeDeploy Agent install and updates with AWS Systems Manager.
AWS Systems Manager provides more control over CodeDeploy Agent version updates and rollbacks than installing using other methods. [Learn more](#)

Install AWS CodeDeploy Agent

Never

Only once

Now and schedule updates

Jayshree Patil.

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

Advanced - optional

[Cancel](#) [Create deployment group](#)

ap-southeast-1.console.aws.amazon.com/codesuite/codedeploy/deployments/d-ISYMV9009?region=ap-southeast-1

Success
Deployment created

[Developer Tools](#) > [CodeDeploy](#) > [Deployments](#) > d-ISYMV9009

d-ISYMV9009

Deployment status

Installing application on your instances
100%
1 of 1 instances updated Succeeded

Deployment details

Application mycodedeply1	Deployment ID d-ISYMV9009	Status Succeeded
Deployment configuration CodeDeployDefault.AllAtOnce	Deployment group deploymentgroup	Initiated by User action
Deployment description -		

Revision details

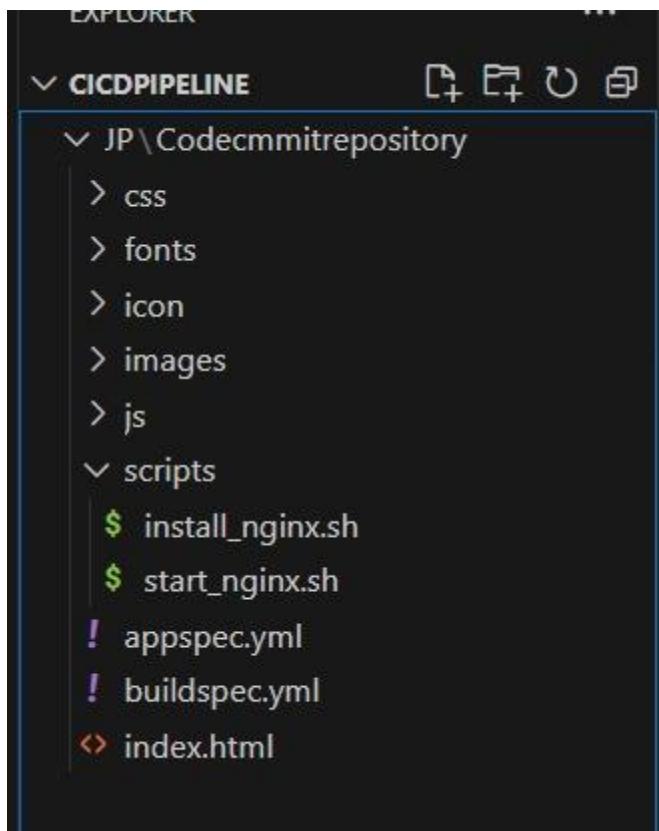
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback Search     31°C Smoke ENG IN 12:35 16-11-2024

Jayshree Patil.

1. If errors occur during deployment, ensure:

FILE FORMAT :-



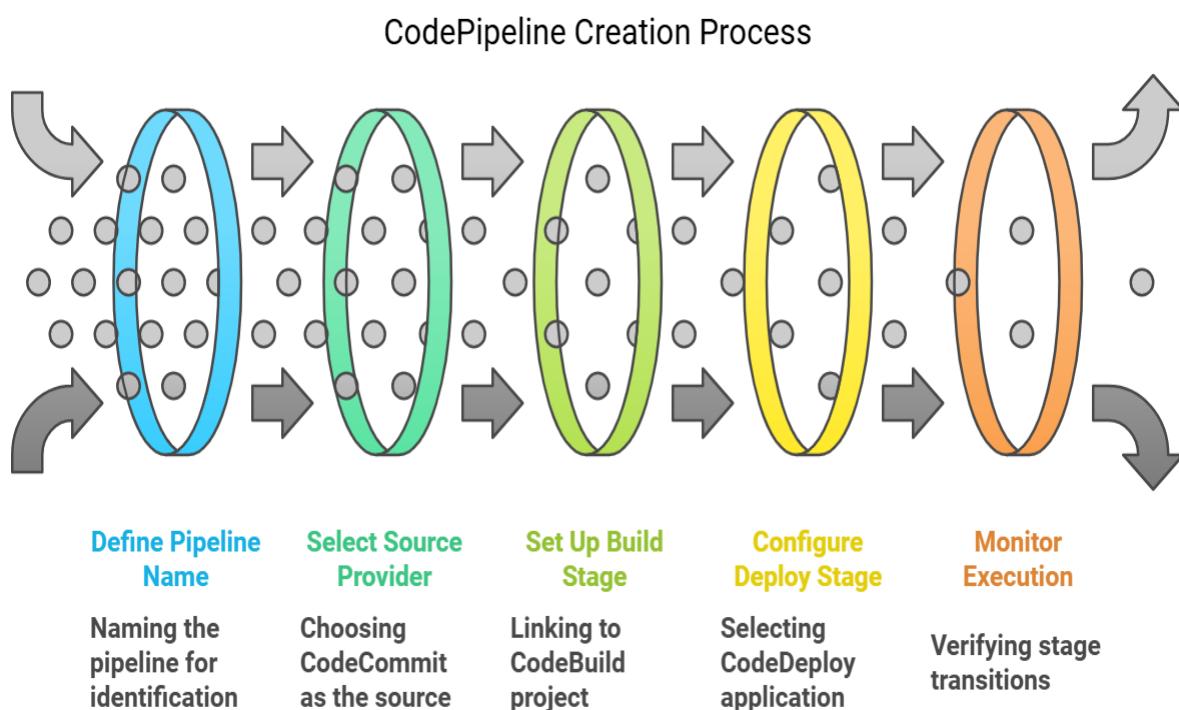
Connect ec2 and run codeagent as given earlier and check active ot not

```
Reading database ... 84761 files and directories currently installed.)  
Preparing to unpack codedeploy-agent_1.3.2-1902 ubuntu22.deb ...  
Unpacking codedeploy-agent (1.3.2-1902) over (1.3.2-1902) ...  
Setting up codedeploy-agent (1.3.2-1902) ...  
codedeploy-agent.service is not a native service, redirecting to systemd-sysv-install.  
Executing: /lib/systemd/systemd-sysv-install enable codedeploy-agent  
  codedeploy-agent.service          loaded active running LSB: AWS CodeDeploy Host Agent  
  codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent  
    Loaded: loaded (/etc/init.d/codedeploy-agent; generated)  
    Active: active (running) since Sat 2024-11-16 08:45:51 UTC; 79ms ago  
      Docs: man:systemd-sysv-generator(8)  
    Process: 7666 ExecStart=/etc/init.d/codedeploy-agent start (code=exited, status=0/SUCCESS)  
    Tasks: 2 (limit: 1130)  
   Memory: 35.1M
```

Also check nginx file upload on script folder and uploaded on vs code

STEP 8: Create a CodePipeline

1. Navigate to **CodePipeline** in the AWS Management Console.
2. Click **Create Pipeline** and provide a name.
3. Select the source provider:
 - o Choose **CodeCommit** and specify the repository and branch.
4. Set up the build stage:
 - o Link it to the CodeBuild project created in Step 6.
5. Configure the deploy stage:
 - o Select the CodeDeploy application and deployment group created in Step 7.
6. Create the pipeline and monitor its execution:
 - o Verify successful transitions between stages (Source → Build → Deploy).



Jayshree Patil.

The screenshot shows the AWS CodePipeline console interface. At the top, there's a navigation bar with 'Developer Tools' and 'CodePipeline'. Below it, a sidebar lists various pipeline components: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), Getting started, Pipelines (selected), and Settings. There are also links to 'Go to resource' and 'Feedback'.

The main content area shows the 'Pipelines' page with a search bar and a table header: Name, Latest execution status, Latest source revisions, Latest execution started, and Most recent executions. A message indicates 'No results' and 'There are no results to display.'

Below this, the URL is 'Developer Tools > CodePipeline > Pipelines > Create new pipeline'. The process is labeled 'Step 1 Choose creation option' and 'Step 1 of 6'. The 'Creation options' section contains two choices: 'Create pipeline from template' (radio button not selected) and 'Build custom pipeline' (radio button selected). A note says 'Build a pipeline from scratch to meet your specific needs.' To the right are 'Cancel' and 'Next' buttons.

In the next step, 'Step 2 Choose pipeline settings' (Step 2 of 6), the pipeline name is set to 'patilpipeline'. A note states: '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.' Under 'Execution mode', 'Superseded' is selected. Other options are 'Queued (Pipeline type V2 required)' and 'Parallel (Pipeline type V2 required)'. Under 'Service role', 'New service role' is selected, with the note 'Create a service role in your account'. Another option is 'Existing service role' with the note 'Choose an existing service role from your account'.

Jayshree Patil.

Role name

AWSCodePipelineServiceRole-ap-southeast-1-patilpipeline

Type your service role name

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

Variables

You can add variables at the pipeline level. You can choose to assign the value when you start the pipeline. Choosing this option requires pipeline type V2. [Learn more](#)

No variables defined at the pipeline level in this pipeline.

[Add variable](#)

You can add up to 50 variables.

i The first pipeline execution will fail if variables have no default values.

▼ Advanced settings

Artifact store

Default location

Create a default S3 bucket in your account.

Custom location

Choose an existing S3 location from your account in the same region and account as your pipeline

Encryption key

Default AWS Managed Key

Use the AWS managed customer master key for CodePipeline in your account to encrypt the data in the artifact store.

Customer Managed Key

To encrypt the data in the artifact store under an AWS KMS customer managed key, specify the key ID, key ARN, or alias ARN.

[Cancel](#)

[Previous](#)

[Next](#)

Add source stage Info

Step 3 of 6

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.

Codecommitrepository

Branch name

Choose a branch of the repository

master

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

Branch name

Choose a branch of the repository

master

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

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. [Learn more](#)

Enable automatic retry on stage failure

Cancel

Previous

Next

Add build stage Info

Step 4 of 6

Build - *optional*

Build provider
Choose the tool you want to use to run build commands and specify artifacts for your build action.

Commands Other build providers

AWS CodeBuild ▾

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.

mycodebuildproject 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.

Region

Asia Pacific (Singapore) ▾

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

X
Defined by: Source
No more than 100 characters

Enable automatic retry on stage failure

Cancel Previous Skip build stage Next

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose creation option

Step 2 Choose pipeline settings

Step 3 Add source stage

Step 4 Add build stage

Step 5 Add deploy stage

Step 6 Review

Add deploy stage Info

Step 5 of 6

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
Asia Pacific (Singapore)

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

BuildArtifact X
Defined by: Build

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.

mycodedeply1 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.

deploymentgroup X

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

BuildArtifact X
Defined by: Build

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.

mycodedeply1 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.

deploymentgroup X

Configure automatic rollback on stage failure

Enable automatic retry on stage failure

[Cancel](#) [Previous](#) [Skip deploy stage](#) [Next](#)

Jayshree Patil.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1
[Choose creation option](#)

Step 2
[Choose pipeline settings](#)

Step 3
[Add source stage](#)

Step 4
[Add build stage](#)

Step 5
[Add deploy stage](#)

Step 6
Review

Review Info

Step 6 of 6

Step 1: Choose pipeline settings

Pipeline settings	
Pipeline name	patilpipeline
Pipeline type	V2
Execution mode	SUPERSEDED
Artifact location	codepipeline-ap-southeast-1-389132101204
Service role name	AWSCodePipelineServiceRole-ap-southeast-1-patilpipeline

Variables

Name	Default value	Description
No variables		
No variables defined at the pipeline level in this pipeline.		

Step 2: Add source stage

Source action provider

Source action provider	AWS CodeCommit
RepositoryName	Codecmmitempty
Default branch	master
PollForSourceChanges	true
OutputArtifactFormat	CODE_ZIP
Enable automatic retry on stage failure	Enabled

Step 3: Add build stage

Build action provider

Build action provider

AWS CodeBuild

ProjectName

mycodebuildproject

Commands

-

Enable automatic retry on stage failure

Enabled

Step 4: Add deploy stage

Deploy action provider

Deploy action provider

AWS CodeDeploy

ApplicationName

mycodedeply1

DeploymentGroupName

deploymentgroup

Configure automatic rollback on stage failure

Enabled

Enable automatic retry on stage failure

Disabled

Cancel

Previous

Create pipeline

Jayshree Patil.

The screenshot shows the AWS CodePipeline console interface. At the top, there's a navigation bar with 'Developer Tools > CodePipeline > Pipelines > patilpipeline'. Below the navigation is a header for the pipeline 'patilpipeline' with options: 'Edit', 'Stop execution', 'Clone pipeline', and a prominent orange 'Release change' button. The pipeline type is listed as 'V2' and the execution mode as 'SUPERSEDED'. A note indicates that the pipeline has a source configured for polling, suggesting migration to event-based detection. The pipeline consists of three stages: Source, Build, and Deploy. The Source stage is shown as 'Succeeded' with a green checkmark. It uses 'AWS CodeCommit' as the provider, with the commit ID 'b699f77b' and a timestamp of 'Succeeded - 41 minutes ago'. A 'View details' button is available. Below the Source stage is a 'Disable transition' button. The Build stage is also 'Succeeded' with a green checkmark, using 'AWS CodeBuild' as the provider, with the same commit details. A 'Start rollback' button is located to the right of the Build stage. The Deploy stage is 'Succeeded' with a green checkmark, using 'AWS CodeDeploy' as the provider, with the same commit details. A 'Start rollback' button is also present here. On the left side, a sidebar menu for 'CodePipeline' lists various components: Source (CodeCommit), Artifacts (CodeArtifact), Build (CodeBuild), Deploy (CodeDeploy), Pipeline (CodePipeline), Getting started, Pipelines, Pipeline (selected), History, Settings, Settings, Go to resource, and Feedback. The main content area has a vertical scrollbar on the right.

Jayshree Patil.

STEP 9: Test the Deployment

1. Copy the **Public IP** of the EC2 instance.
2. Paste it into your browser to verify the deployed application.

The screenshot shows the AWS CloudWatch Metrics interface. At the top, there's a search bar and a navigation bar with tabs like 'Metrics', 'Logs', and 'CloudWatch Metrics'. Below the navigation bar, there's a table with columns: Metric Name, Namespace, Unit, and Last Value. The table contains one row for 'CPU Utilization' with values: 'CPUUtilization', 'AWS/EC2', 'Percent', and '0.00'. There are also 'View Metrics' and 'Edit Metrics' buttons.



Jayshree Patil.

The screenshot shows a web browser window with multiple tabs open at the top. The main content area displays a website for "Spicy Healthy Food Recipes". The header includes the word "Spicy" in red, and navigation links for "ABOUT", "SCREENSHOT", and "CONTACT". A phone number "Call Us : 12345677890" is also present. The main title "Healthy Food Recipes" is displayed in large, bold, dark brown letters. Below the title, there is a short paragraph of placeholder text (Lorem Ipsum) followed by a contact form with fields for Name, Email, Phone Number, and Message.

The screenshot shows a web browser window with multiple tabs open at the top. The main content area displays a testimonial section titled "Clients says" in yellow. It features a circular profile picture of a smiling man wearing sunglasses, with the name "MARK DU" below it. To the right of the testimonial is a large image of a healthy meal consisting of a slice of bread, some basil leaves, and a soft-boiled egg with tomato slices and cheese on a green plate.