

DevOps(Day-50): Your CI/CD Pipeline on AWS -Part-1

Maninder Singh

Dear Learners, After discuss the AWS interview questions in previous article, In today article we have discuss the AWS CodePipeline and its deployment.

TABLE OF CONTENTS:

What is CodeCommit?

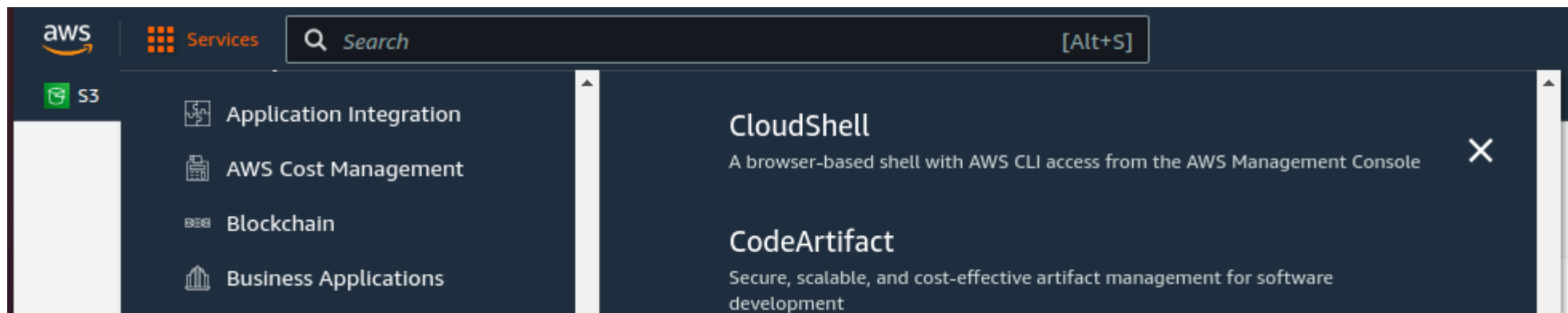
Task-01 Create and Configure a CodeCommit repository


















Task-02 Integrate the repository with server

What if tell you, in the next 4 days, you will be making a CI/CD pipeline on AWS with these tools:-

What is CodeCommit?

CodeCommit is a managed source control service by AWS that allows users to storage,manage and version their source code and artifacts securely and at a scale. It supports the Git, integrates with other AWS services,enables collaboration through branch and merge workflows and provides audit logs and compliance reports to meet regulatory and track changes.



-  Compute
-  Containers
-  Customer Enablement
-  Database
-  **Developer Tools**
-  End User Computing
-  Front-end Web & Mobile
-  Game Development
-  Internet of Things
-  Machine Learning
-  Management & Governance
-  Media Services
-  Migration & Transfer
-  Networking & Content Delivery
-  Quantum Technologies
-  Robotics
-  Satellite
-  Security, Identity, & Compliance
-  Storage

CodeBuild

Build and Test Code

Amazon CodeCatalyst

Integrated DevOps Service

★ CodeCommit

Store Code in Private Git Repositories

CodeDeploy

Automate Code Deployments

CodePipeline

Release Software using Continuous Delivery

CodeStar

Quickly develop, build, and deploy applications

Amazon CodeWhisperer

Build applications faster with the ML-powered coding companion.

AWS FIS

Improve resiliency and performance with controlled experiments.

X-Ray

Analyze and Debug Your Applications

Code Commit in AWS Diagram.

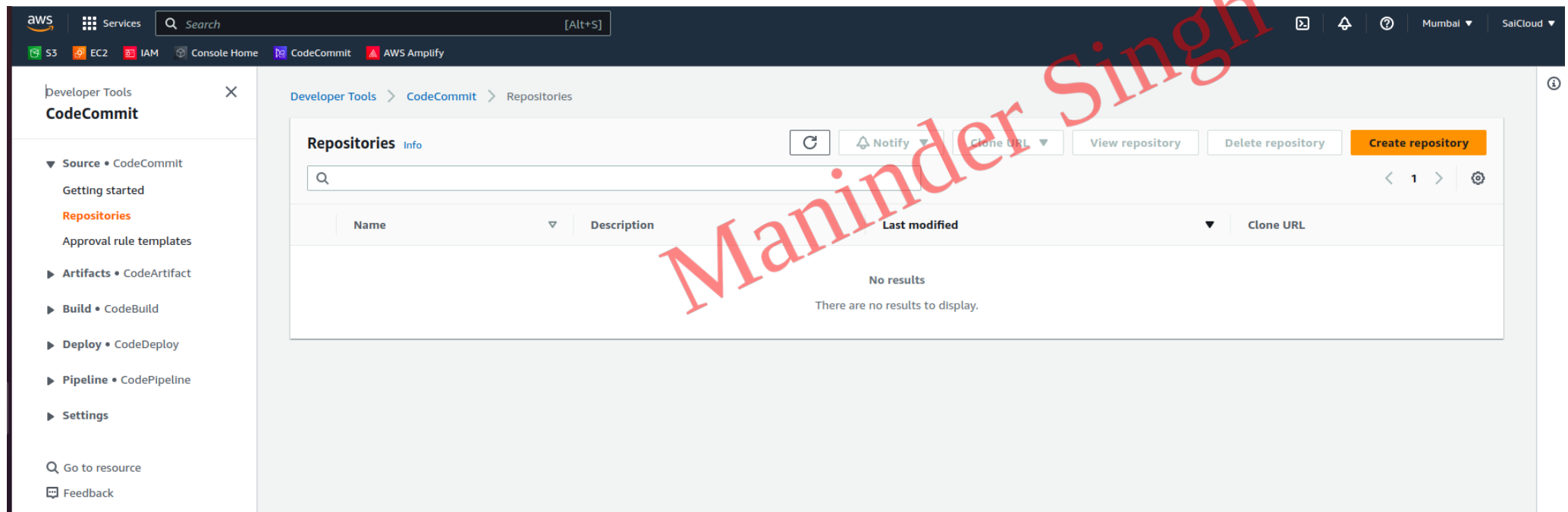
Task-01

Create and Configure a CodeCommit repository

Login to the AWS Management console with user name and password.

Set up a code repository on CodeCommit and clone it on your local.

Navigate to the CodeCommit section of AWS console.

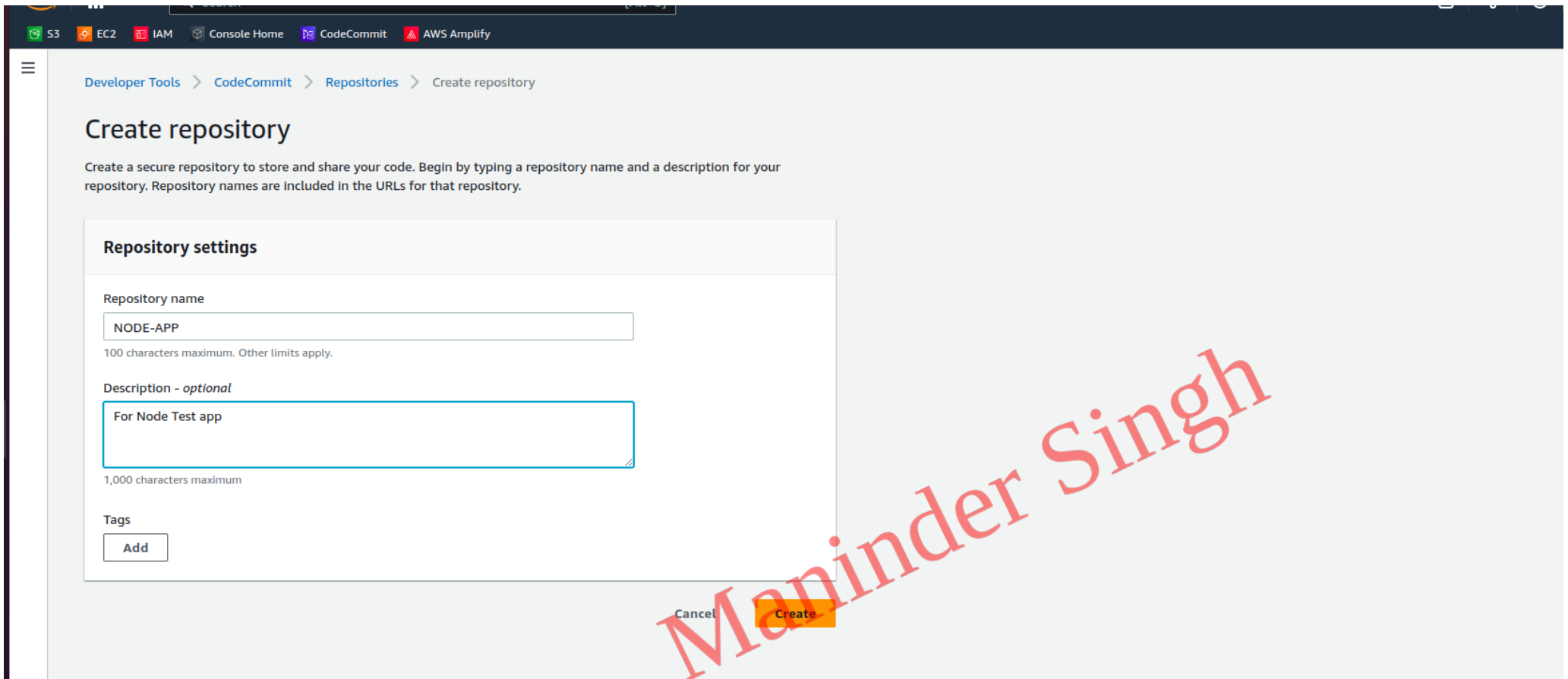


Navigate to the CodeCommit section of AWS console Diagram.

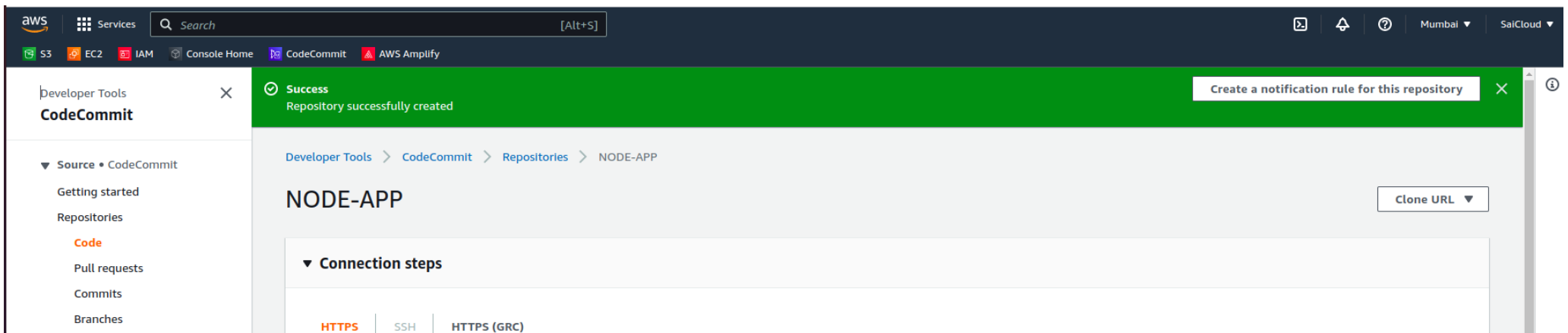
Create a repository by giving a name and description

Click on the create option.





Create a repository by giving a name and description Click on the create option. Diagram.



Git tags

Settings

Approval rule templates

▶ Artifacts • CodeArtifact

▶ Build • CodeBuild

▶ Deploy • CodeDeploy

▶ Pipeline • CodePipeline

▶ Settings

Go to resource

Feedback

⚠ You are signed in using a root account. You cannot configure SSH connections for a root account, and HTTPS connections for a root account are not recommended. Consider signing in as an IAM user and then setting up your connection.

Step 1: Prerequisites

You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. If you do not have a Git client, you can install one from [Git downloads page](#).

You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project team, or have the equivalent permissions. [Learn how to create and configure an IAM user for accessing AWS CodeCommit.](#) | [Learn how to add team members to an AWS CodeStar Project.](#)

Step 2: Set up the AWS CLI Credential Helper

Set up your connection to AWS CodeCommit repositories using the credential helper included in the AWS CLI. This is the only connection method for AWS CodeCommit repositories that does not require an IAM user, so it is the only method that supports root access, federated access, and temporary credentials. [Learn more](#)

Additional details

You can find more detailed instructions in the documentation. [View documentation](#)

A repository redidt-app is now created.

How would you rate your experience with this service console? ☆ ☆ ☆ ☆ ☆

aws

Services

Search

[Alt+S]

📧

🔔

?

Mumbai

SalCloud

S3

EC2

IAM

Console Home

CodeCommit

AWS Amplify

Developer Tools

CodeCommit

▼ Source • CodeCommit

Getting started

Repositories

Approval rule templates

▶ Artifacts • CodeArtifact

▶ Build • CodeBuild

▶ Deploy • CodeDeploy

▶ Pipeline • CodePipeline

▶ Settings

Go to resource

Feedback

Developer Tools > CodeCommit > Repositories

Repositories Info

🔄

🔔 Notify

Clone URL

View repository

Delete repository

Create repository

🔍

< 1 > ⚙

	Name	Description	Last modified	Clone URL
<input type="radio"/>	NODE-APP	For Node Test app	14 minutes ago	HTTPS SSH HTTPS (GRC)

A repository redit-app is now created Diagram.

You need to setup Git Credentials in your AWS IAM.

1 Navigate to the IAM Section and create a user and group as we have seen in our previous blog this week as a part of AWS tasks.

2 Go to the IAM console

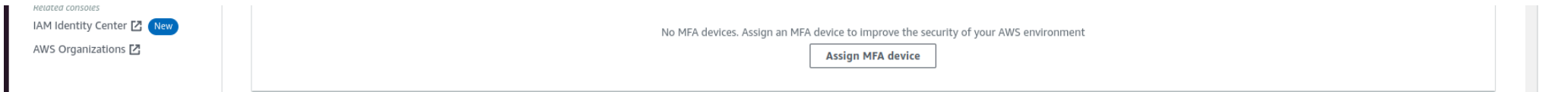
Click on Users in the left-hand menu and then click on your username.

Click on Users in the left-hand menu and then click on your username Diagram.

Scroll down to the security credentials section.

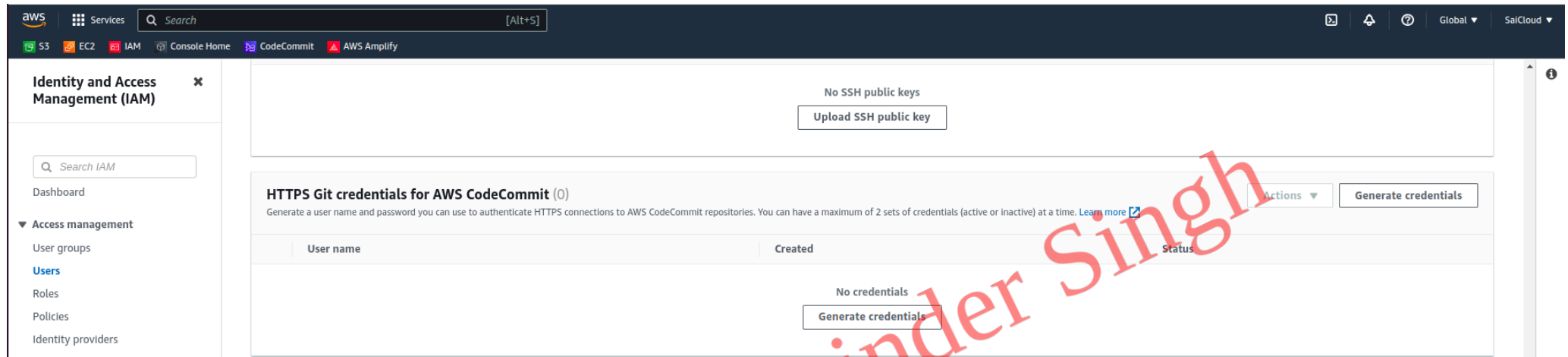
The screenshot displays the AWS IAM console interface. The left-hand navigation pane shows the 'Identity and Access Management (IAM)' section with a search bar and a list of options including 'Access management', 'Access reports', and 'Service control policies (SCPs)'. The main content area is titled 'test' and shows the user's details. The 'Summary' section includes the ARN 'arn:aws:iam::414310061589:user/test', console access status 'Disabled', and last console sign-in '-'. The 'Security credentials' tab is selected, showing 'Console sign-in' with a link to 'https://414310061589.signin.aws.amazon.com/console' and a status of 'Not enabled'. The 'Multi-factor authentication (MFA)' section shows 0 devices and options to 'Remove', 'Resync', or 'Assign MFA device'. A large red watermark 'Maninder Singh' is overlaid on the image.

Device type	Identifier	Created on
-------------	------------	------------

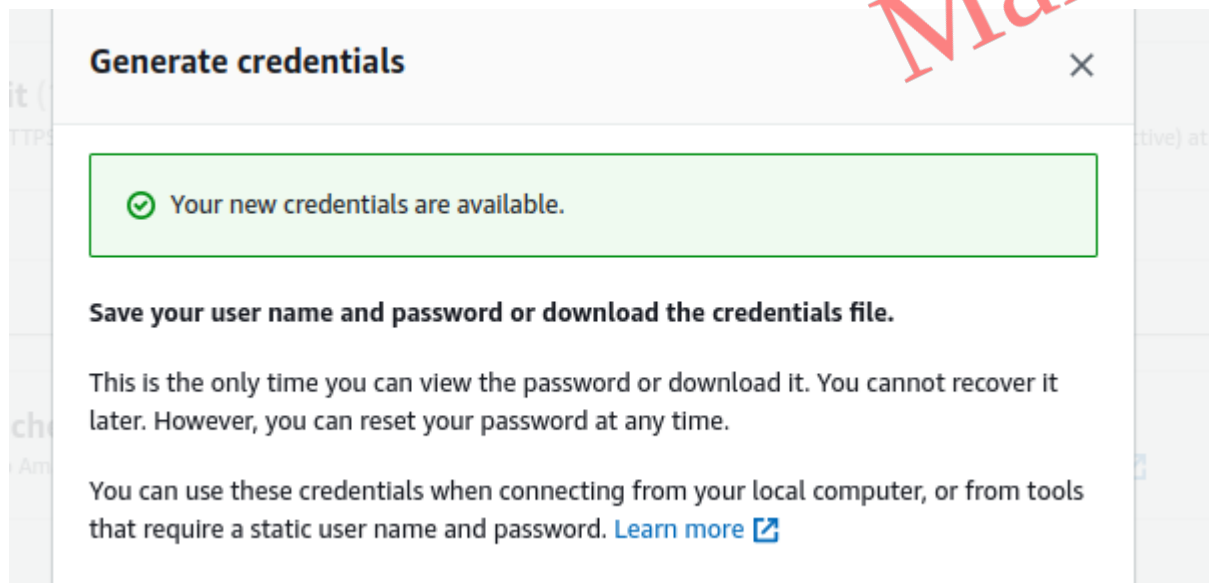


Scroll down to the security credentials section Diagram.

Under the security credentials,scroll down and come to HTTPS Git credentials for AWS CodeCommit section click on Generate Credentials.



After Click on the Download Credentials button your Git credentials and click on Close.



User name

test-at-414310061589

Password

|jfbYORg/eMXfwVuDctc/sTcPs7jYaFZoRxU1qixsD0= [Hide](#)

Download credentials

Close

HTTPS Git credentials for AWS CodeCommit section click on Generate Credentials Diagram.

Now your Git Credentials are created.

aws

Services

Search

[Alt+S]

S3 EC2 IAM Console Home CodeCommit AWS Amplify

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

SSH Key ID

Uploaded

Status

No SSH public keys

Upload SSH public key

HTTPS Git credentials for AWS CodeCommit (1)

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](#)

Actions

Generate credentials

User name	Created	Status
test-at-414310061589	Now	Active

Now your Git Credentials are created. Diagram.

Use the Credentials in your local and then clone the repository form the codecommit.

In the code commit,Go inside your repository that you created in above steps , In right hand side click Clone URL and choose Clone HTTPS.

How would you rate your experience with this service console? ☆ ☆ ☆ ☆ ☆

aws

Services

Search

[Alt+S]

S3 EC2 IAM Console Home CodeCommit AWS Amplify

Developer Tools

CodeCommit

Source • CodeCommit

Developer Tools > CodeCommit > Repositories > NODE-APP

NODE-APP

Clone URL

Clone HTTPS

The screenshot shows the AWS CodeCommit console interface. On the left is a navigation sidebar with options: Getting started, Repositories, Code (highlighted), Pull requests, Commits, Branches, Git tags, Settings, Approval rule templates, Artifacts (with CodeArtifact sub-option), Build (with CodeBuild sub-option), Deploy (with CodeDeploy sub-option), Pipeline (with CodePipeline sub-option), and Settings. The main content area is titled 'Connection steps' and has tabs for HTTPS (selected), SSH, and HTTPS (GRC). A warning box states: 'You are signed in using a root account. You cannot configure SSH connections for a root account, and HTTPS connections for a root account are not recommended. Consider signing in as an IAM user and then setting up your connection.' Below this, 'Step 1: Prerequisites' explains that a Git client (version 1.7.9 or later) and an AWS CodeCommit managed policy are required. 'Step 2: Set up the AWS CLI Credential Helper' states that this is the only connection method for AWS CodeCommit repositories that does not require an IAM user. On the right side of the console, there are buttons for 'Clone SSH' and 'Clone HTTPS (GRC)'.

In right hand side click Clone URL and choose Clone HTTPS Diagram.

Open a terminal on your local machine.

Navigate to the directory where you want to clone the repository.

Run the following Commands

you will be prompted to enter your Git Credentials. Enter the username and password that you downloaded earlier.

you will be prompted to enter your Git Credentials. Enter the username and password that you downloaded earlier Diagram.

You have now set up a CodeCommit repository and cloned it on your local machine using Git Credentials in AWS IAM. (Note: Your user has aws codecommit permission to perform this action).

Task-02

Add a new file form the local and commit to your local branch

Create a new file in the local repository directory.

Create a new file in the local repository directory Diagram.

Check the status using the using command "git status"

```
ubuntu@ip-172-31-3-140:~/NODE-APP$ ls
file.txt
ubuntu@ip-172-31-3-140:~/NODE-APP$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    file.txt

nothing added to commit but untracked files present (use "git add" to track)
ubuntu@ip-172-31-3-140:~/NODE-APP$
```

Check the status using the using command "git status Diagram.

Add the new file to your local branch using the following commads.

```
ubuntu@ip-172-31-3-140:~/NODE-APP$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    file.txt

nothing added to commit but untracked files present (use "git add" to track)
ubuntu@ip-172-31-3-140:~/NODE-APP$
ubuntu@ip-172-31-3-140:~/NODE-APP$ git add file.txt
ubuntu@ip-172-31-3-140:~/NODE-APP$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   file.txt

ubuntu@ip-172-31-3-140:~/NODE-APP$
```

Commit the changes to your local branch using the following commands.

```
ubuntu@ip-172-31-3-140:~/NODE-APP$
ubuntu@ip-172-31-3-140:~/NODE-APP$ git commit -m "added file.txt"
[master (root-commit) 7615dd9] added file.txt
  Committer: maninder <ubuntu@ip-172-31-3-140.ap-south-1.compute.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:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

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 file.txt
ubuntu@ip-172-31-3-140:~/NODE-APP$ git status
On branch master
Your branch is based on 'origin/master', but the upstream is gone.
  (use "git branch --unset-upstream" to fixup)

nothing to commit, working tree clean
ubuntu@ip-172-31-3-140:~/NODE-APP$
```

Commit the changes to your local branch using the following commands Diagram.

Push the local changes to CodeCommit repository.

Push the changes from your local branch to the codeCommit repository using the following commands:

```
System information as of Mon May 29 14:36:22 UTC 2023

System load:  0.0          Processes:            103
Usage of /:   21.9% of 7.57GB Users logged in:      1
Memory usage: 24%          IPv4 address for eth0: 172.31.3.140
Swap usage:   0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.
```

```

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '22.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon May 29 14:20:16 2023 from 110.224.70.186
ubuntu@ip-172-31-3-140:~$
ubuntu@ip-172-31-3-140:~$ ls
NODE-APP  aws  codecommit
ubuntu@ip-172-31-3-140:~$ cd NODE-APP/
ubuntu@ip-172-31-3-140:~/NODE-APP$ ls
file.txt
ubuntu@ip-172-31-3-140:~/NODE-APP$ git push origin master
Username for 'https://git-codecommit.ap-south-1.amazonaws.com': test-at-414310061589
Password for 'https://test-at-414310061589@git-codecommit.ap-south-1.amazonaws.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote: Validating objects: 100%
To https://git-codecommit.ap-south-1.amazonaws.com/v1/repos/NODE-APP
 * [new branch]      master -> master
ubuntu@ip-172-31-3-140:~/NODE-APP$

```

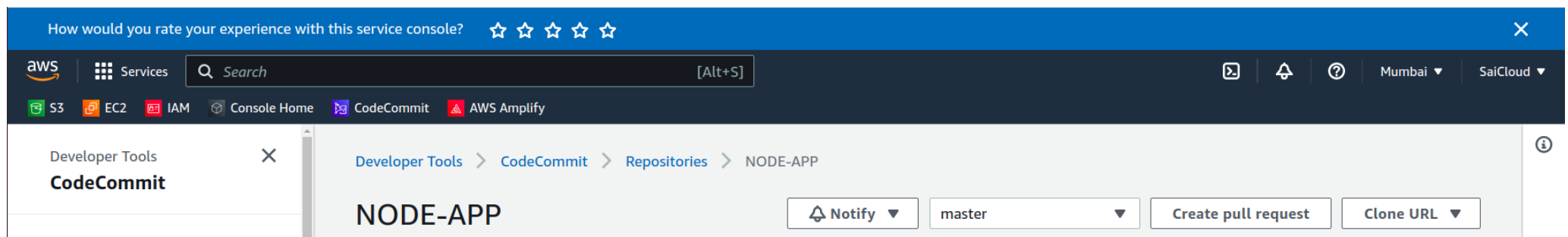
Push the changes from your local branch to the codeCommit repository using the following commands: `git push origin master` Diagram.

Verify that the changes have been pushed to the codeCommit repository.

Go to the code commit repository that you created earlier, you should see the new file listed in the repository files.

Go to the code commit repository that you created earlier, you should see the new file listed in the repository files Diagram.

You can see the content of the file.



▼ Source • CodeCommit

- Getting started
- Repositories
- Code**
- Pull requests
- Commits
- Branches
- Git tags
- Settings
- Approval rule templates

► Artifacts • CodeArtifact

► Build • CodeBuild

► Deploy • CodeDeploy

► Pipeline • CodePipeline

► Settings

NODE-APP / file.txt Info Edit

```
1 Hello Everyone !!
2
3 This is my Day 50 task of #90 days of devops challenge. under the guideline of shubham londhe sir
4
```

CloudShell Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

You can see the content of the file Diagram.

Thank you for reading !! I hope you find this article helpful!!

Maninder Singh

Next Topic:

Day 51:- we have completed AWS [CodeCommit](#).Next few days we'll learn these tools/ Services:-