

Milestone 2

Devops Project

Name : Ankit Tiwari

Ps No. : 10844395

Problem Statement :

Create an end to end CI/CD pipeline in AWS platform using Jenkins as the orchestration tool, Github as the SCM, Maven as the Build tool, Deploy in a docker instance and create a Docker image, Store the docker image in ECR, Achieve Kubernetes deployment using the ECR image. Build a sample java web app using maven.

Project Starts -----

Approach: Declarative pipeline

Requirements:

- ✓ **CI/CD pipeline System**
- ✓ **Git - local version control system.**
- ✓ **GitHub - As Distributed version control system.**
- ✓ **Jenkins - Continuous Integration tool.**
- ✓ **Maven - As a Build Tool.**
- ✓ **docker -Containerization**
- ✓ **Kubernetes - As Container Management Tool**

Steps :

Step 1: Source Code Setup

- **Created an EC2 instance for hosting source code.**
 - **Connected to the instance via SSH.**
 - **Generated SSH keys and added them to GitHub.**
 - **Cloned the repository containing:**
 - **Dockerfile**
 - **Jenkinsfile**
 - **deployment.yaml, service.yaml**
 - **Application source code**
-

Step 2: Jenkins CI Setup

- **Created a second EC2 instance for Jenkins.**
- **Connected via SSH and installed:**
 - **Jenkins**
 - **Maven**
 - **Git**
- **Configured Jenkins:**
 - **Set JDK and Maven paths**
 - **Connected GitHub repository using token/webhook**
- **Created a new Jenkins item (Pipeline project).**
- **Wrote a Declarative Jenkinsfile to automate:**
 - **Build process**
 - **Docker image creation**

- Deployment steps
-

Step 3: Tomcat Deployment Setup

- Created a third EC2 instance for Tomcat deployment.
 - Connected via SSH and installed Tomcat.
 - Started Tomcat and edited configuration files.
 - Created Jenkins credentials for Tomcat deployer.
 - Added deployment steps in Jenkinsfile (if applicable).
-

Step 4: Docker & AWS ECR Integration

- Created a fourth EC2 instance for Docker operations.
 - Installed Docker and configured AWS CLI using aws configure.
 - Enabled root login via SSH (sshd_config).
 - Created a private ECR repository using AWS CLI.
 - Added Jenkins pipeline steps to:
 - Build Docker image
 - Tag and push image to ECR
 - Verified successful image upload to ECR after Jenkins build.
-

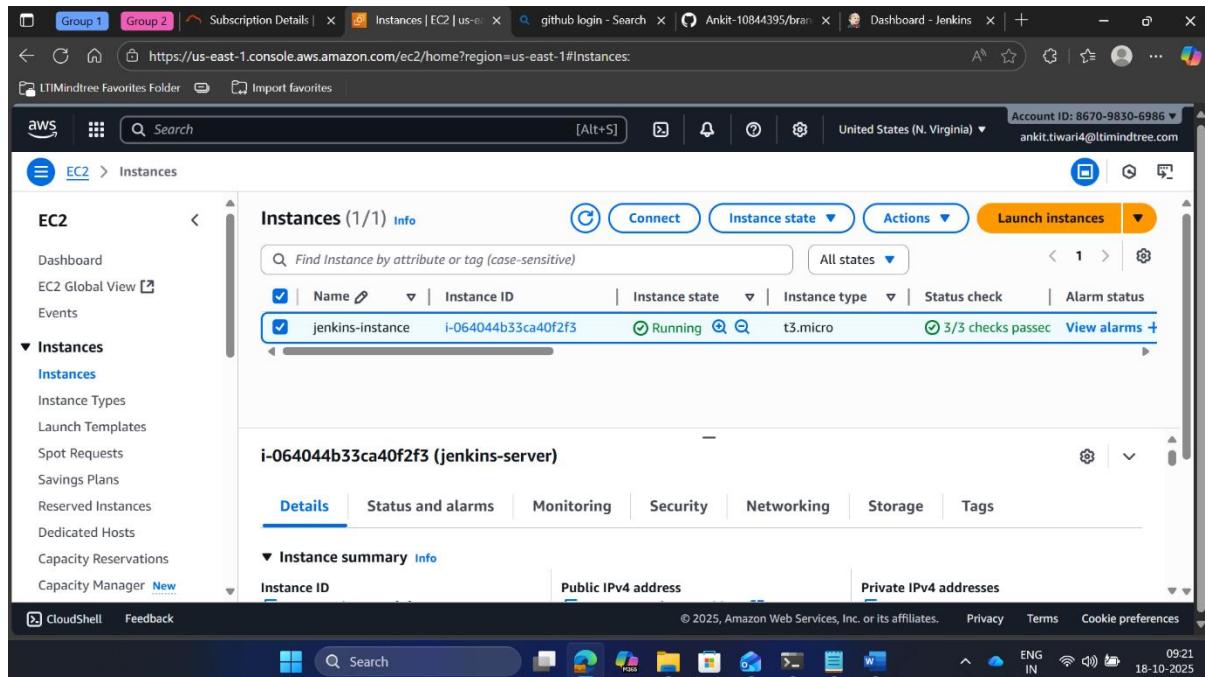
Step 5: Kubernetes Deployment on EKS

- Created a fifth EC2 instance for EKS deployment.
- Installed:
 - AWS CLI

- **kubectl**
- **eksctl**
- **Configured AWS credentials.**
- **Created an EKS cluster and node group using eksctl.**
- **Created deployment.yaml and service.yaml files.**
- **Added Jenkins pipeline steps to:**
 - **Update kubeconfig**
 - **Apply Kubernetes manifests**
 - **Monitor rollout status**
- **Verified pod creation and load balancer updates after GitHub changes.**

Solution:

→ Creating an instance to install Jenkins



→ Jenkins-installed successfully

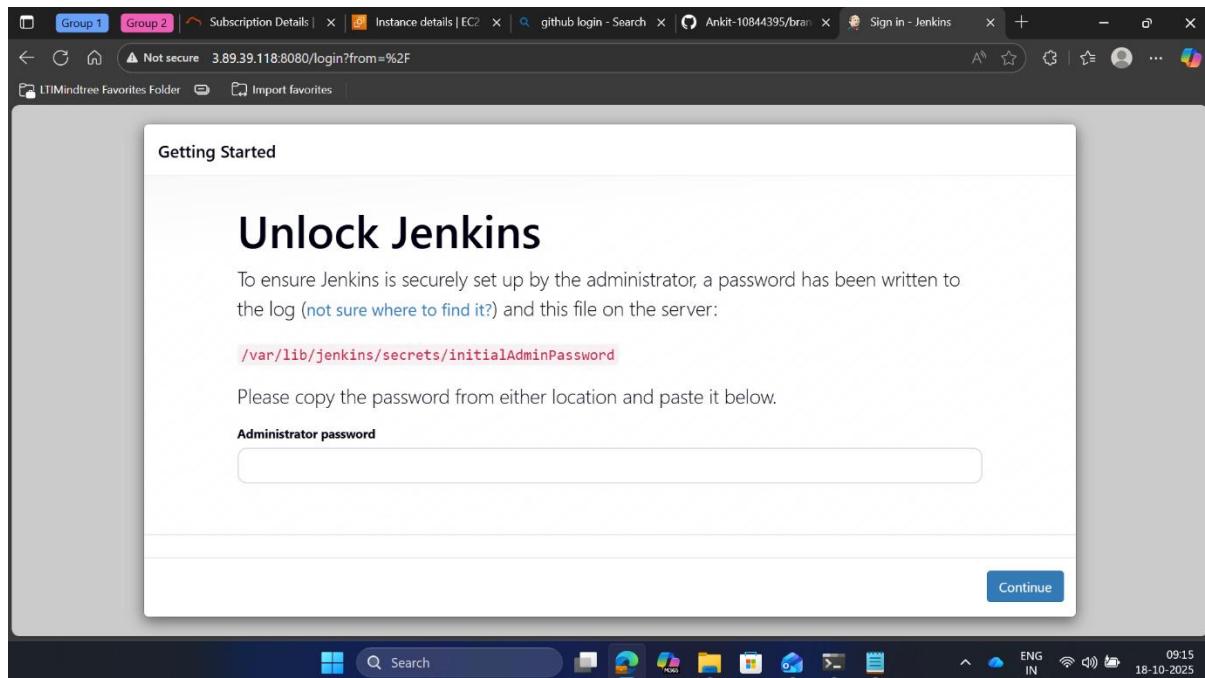
```
root@ip-172-31-40-207:~| + | - | x | 09:19
Transaction Summary
=====
Install 1 Package

Total download size: 91 M
Installed size: 91 M
Downloading Packages:
jenkins-2.528.1-1.1.noarch.rpm
Total                                         13 MB/s | 91 MB   00:06
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :                                         1/1
Running scriptlet: jenkins-2.528.1-1.1.noarch   1/1
Installing : jenkins-2.528.1-1.1.noarch        1/1
Running scriptlet: jenkins-2.528.1-1.1.noarch   1/1
Verifying   : jenkins-2.528.1-1.1.noarch        1/1

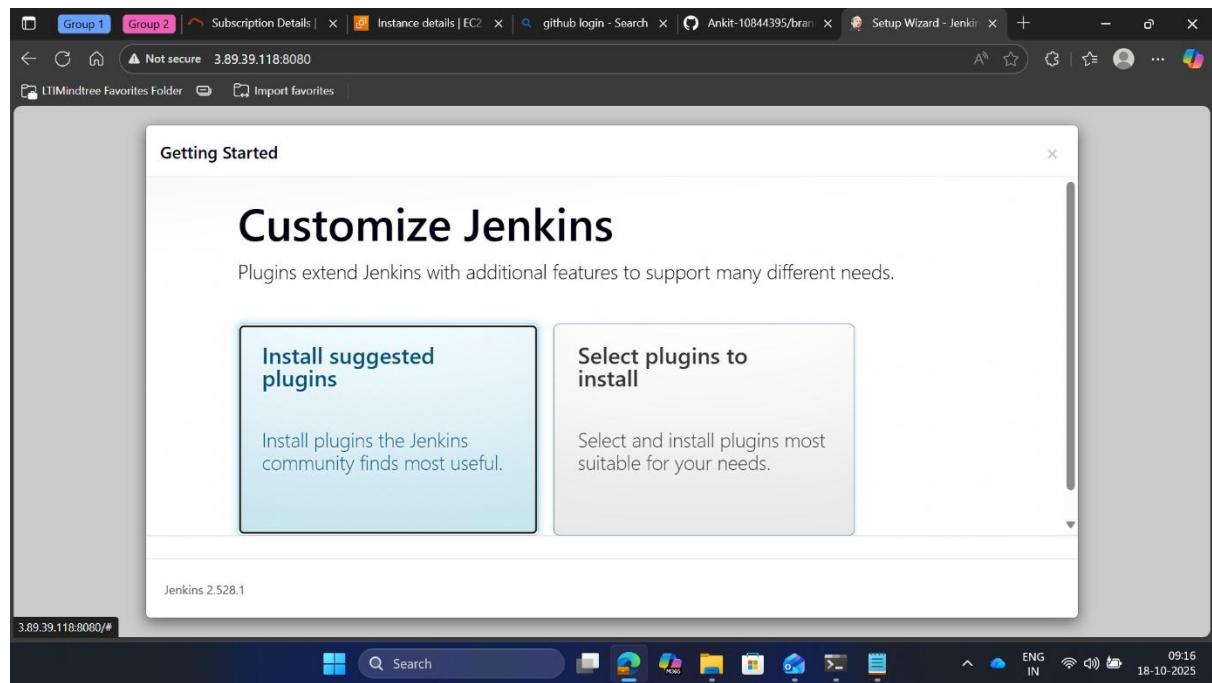
Installed:
 jenkins-2.528.1-1.1.noarch

Complete!
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
[root@jenkins-server ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
6e30e5b045934d11bec4f538b82206c3
[root@jenkins-server ~]#
```

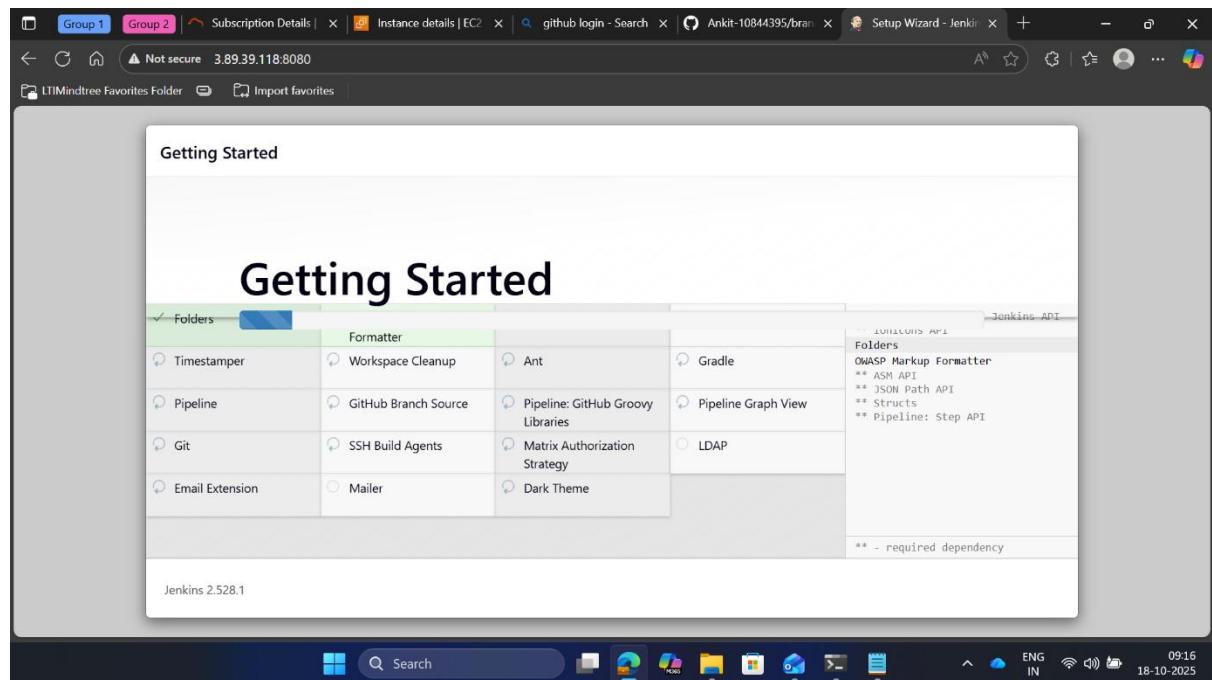
→ Authenticating password for Jenkins



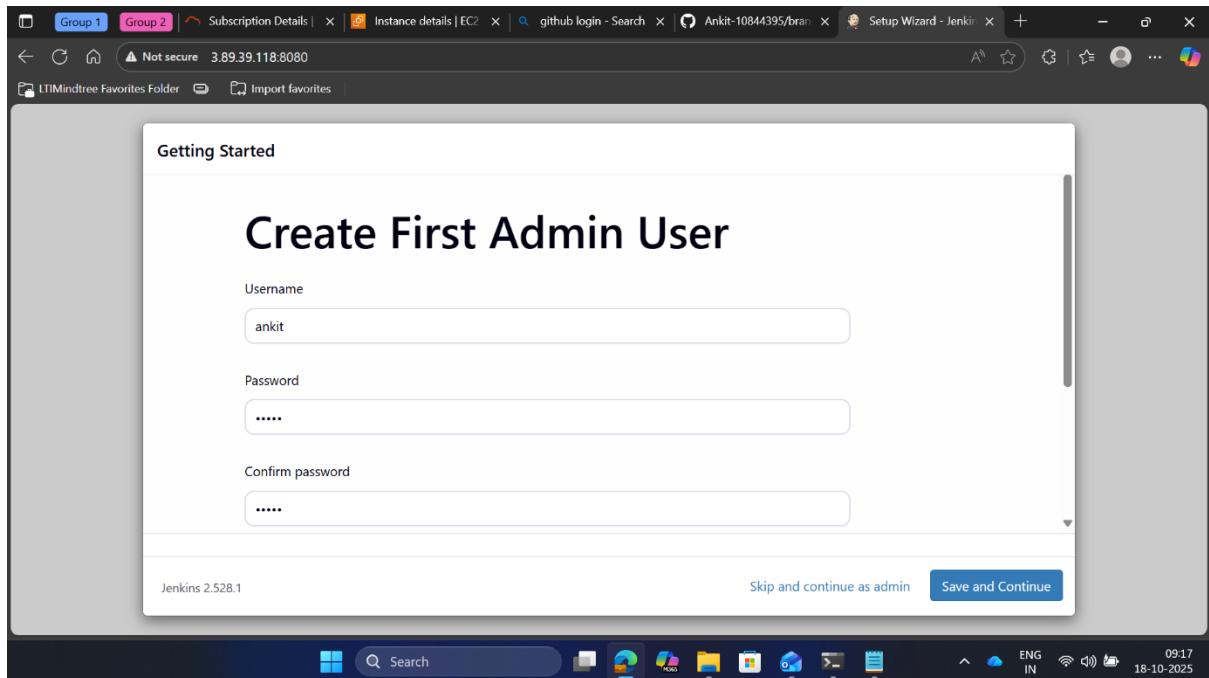
→ Installing suggested Pluggins



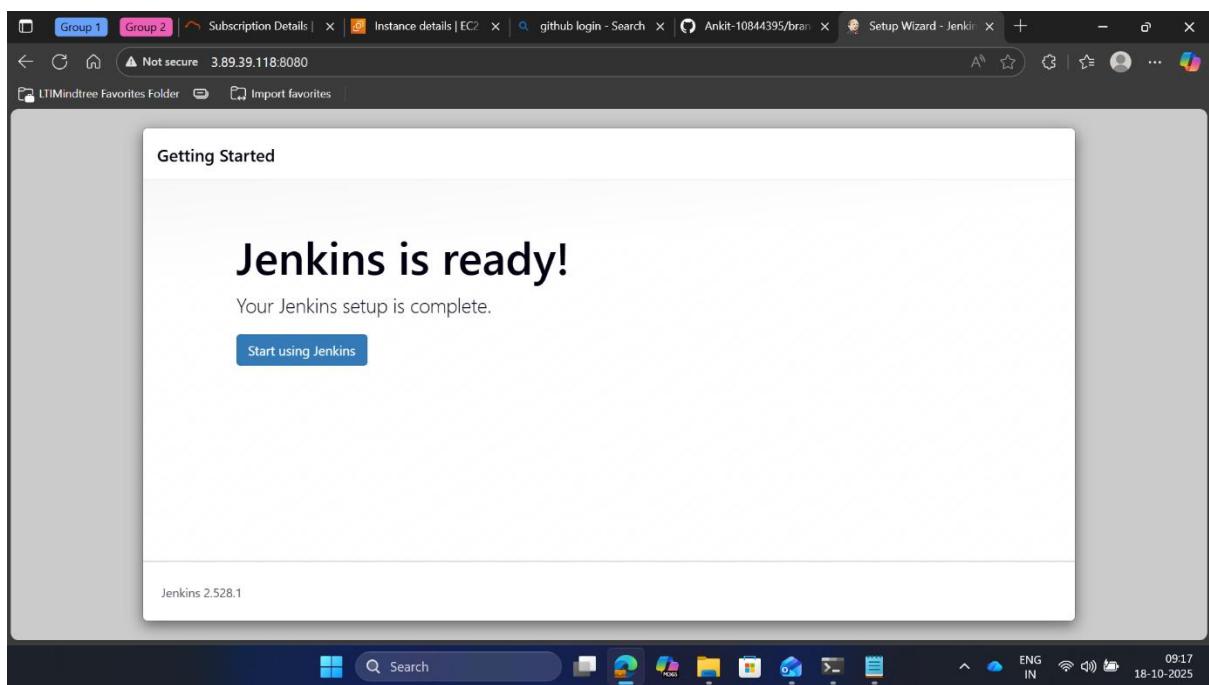
→ Jenkins getting started



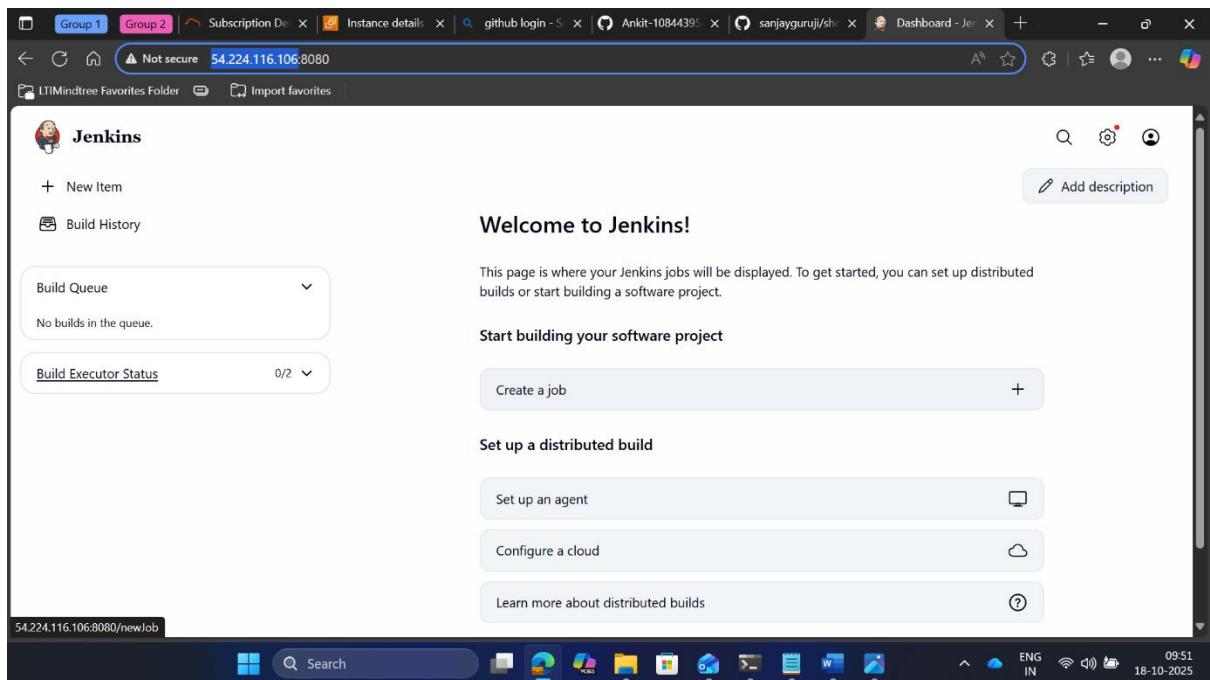
→ Providing user information to access the dashboard



→ Jenkins is ready

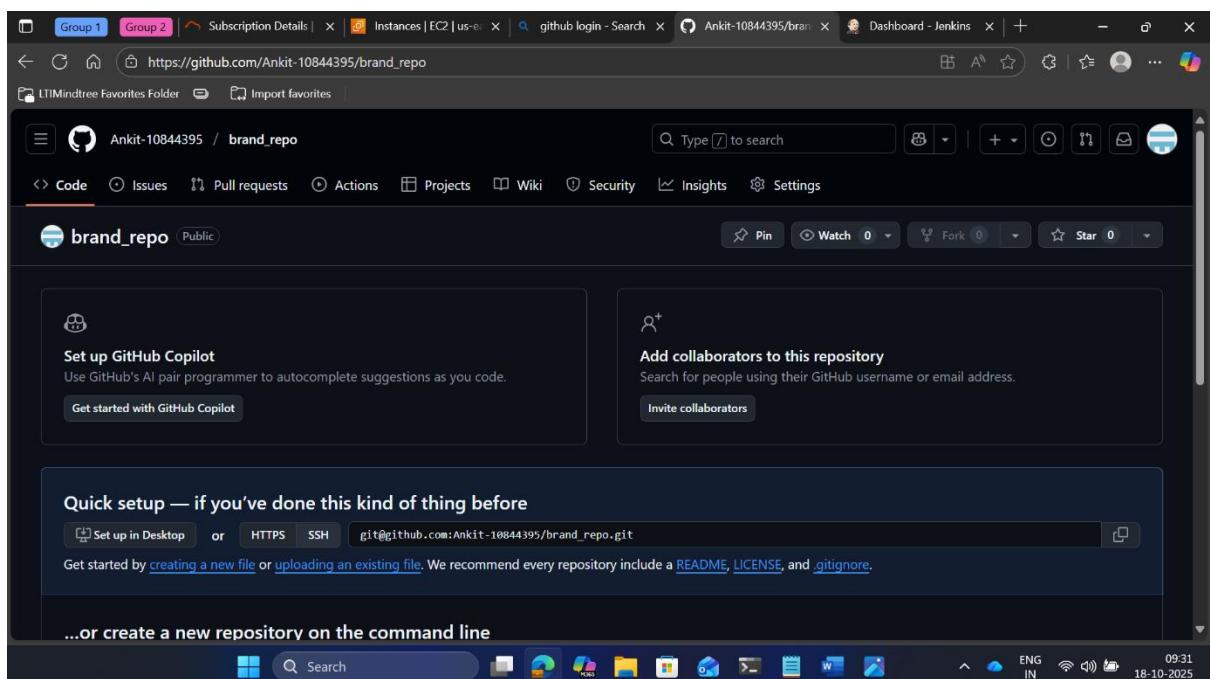


→ Jenkins Dashboard



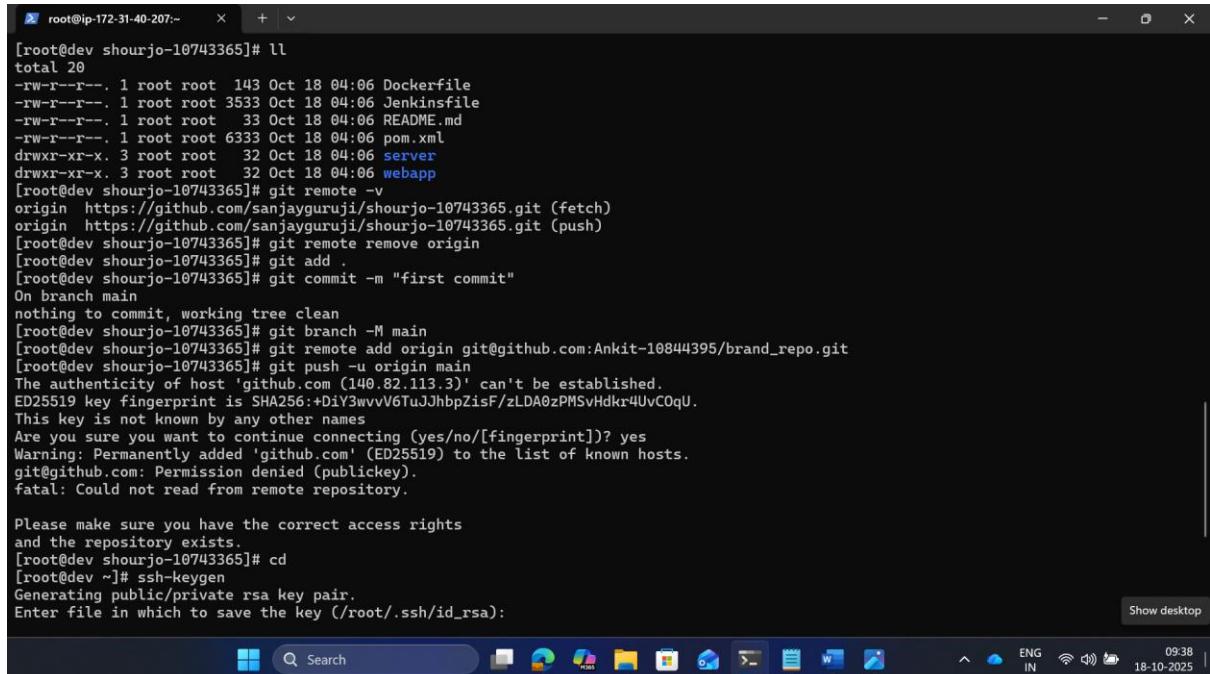
The screenshot shows the Jenkins Dashboard. At the top, there is a header with the Jenkins logo and a search bar. Below the header, there are two main sections: 'Build Queue' and 'Build Executor Status'. The 'Build Queue' section shows 'No builds in the queue.' The 'Build Executor Status' section shows '0/2'. To the right, there is a large 'Welcome to Jenkins!' message with a sub-section 'Start building your software project' containing a 'Create a job' button and a '+' icon. Below this, there is a section titled 'Set up a distributed build' with three buttons: 'Set up an agent' (with a monitor icon), 'Configure a cloud' (with a cloud icon), and 'Learn more about distributed builds' (with a question mark icon). The bottom of the screen shows a taskbar with various icons and a system tray indicating the date and time as 18-10-2025.

→ New github repository created



The screenshot shows a GitHub repository page for 'brand_repo'. The top navigation bar includes links for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', 'Settings', 'Pin', 'Watch', 'Fork', 'Star', and 'Settings'. The main content area features a 'Set up GitHub Copilot' section with a 'Get started with GitHub Copilot' button. To the right, there is an 'Add collaborators to this repository' section with a 'Invite collaborators' button. Below these, a 'Quick setup — if you've done this kind of thing before' section provides instructions for setting up the repository, including options to 'Set up in Desktop' or 'HTTPS' or 'SSH', and a URL 'git@github.com:Ankit-10844395/brand_repo.git'. A note at the bottom encourages creating a new file or uploading an existing file, mentioning README, LICENSE, and .gitignore. The bottom of the screen shows a taskbar with various icons and a system tray indicating the date and time as 18-10-2025.

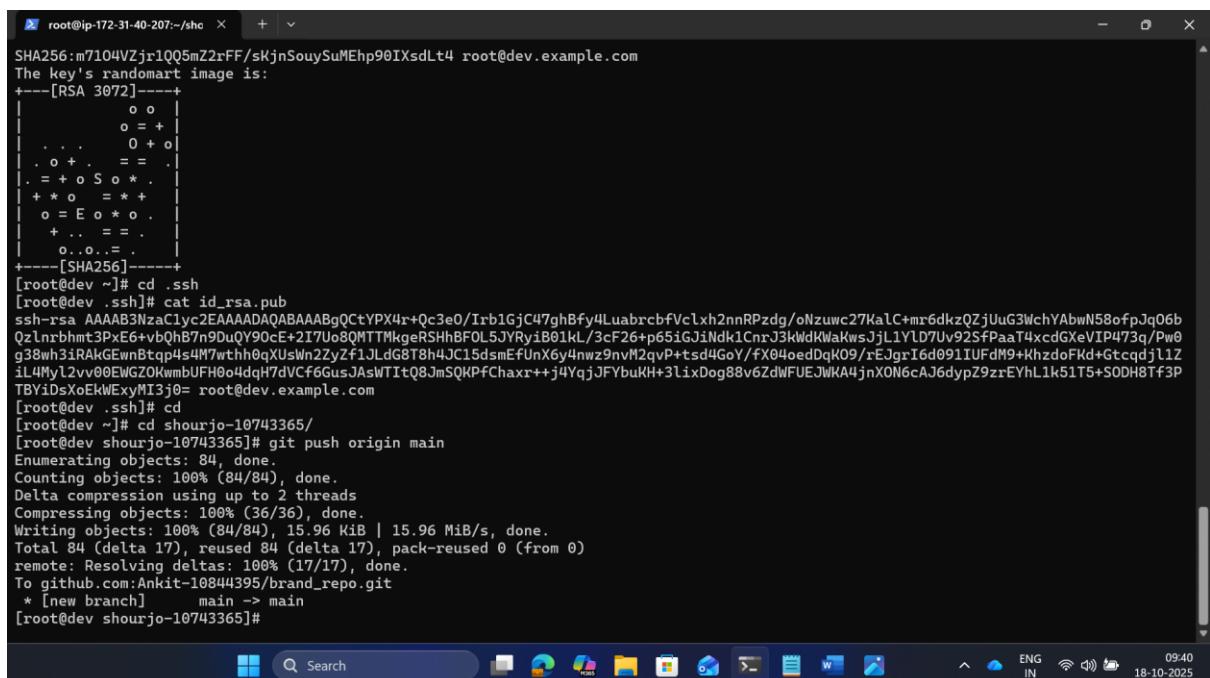
→ Cloning the repository consisting of java project



```
root@ip-172-31-40-207:~# ll
total 20
-rw-r--r--. 1 root root 143 Oct 18 04:06 Dockerfile
-rw-r--r--. 1 root root 3533 Oct 18 04:06 Jenkinsfile
-rw-r--r--. 1 root root 33 Oct 18 04:06 README.md
-rw-r--r--. 1 root root 6333 Oct 18 04:06 pom.xml
drwxr-xr-x. 3 root root 32 Oct 18 04:06 server
drwxr-xr-x. 3 root root 32 Oct 18 04:06 webapp
[root@dev shourjo-10743365]# git remote -v
origin https://github.com/sanjayguruji/shourjo-10743365.git (fetch)
origin https://github.com/sanjayguruji/shourjo-10743365.git (push)
[root@dev shourjo-10743365]# git remote remove origin
[root@dev shourjo-10743365]# git add .
[root@dev shourjo-10743365]# git commit -m "first commit"
On branch main
nothing to commit, working tree clean
[root@dev shourjo-10743365]# git branch -M main
[root@dev shourjo-10743365]# git remote add origin git@github.com:Ankit-108444395/brand_repo.git
[root@dev shourjo-10743365]# git push -u origin main
The authenticity of host 'github.com (140.82.113.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3vvvv6TuJhbpbZisFzLDA0zPMsvHdkr4UvC0qU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
git@github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

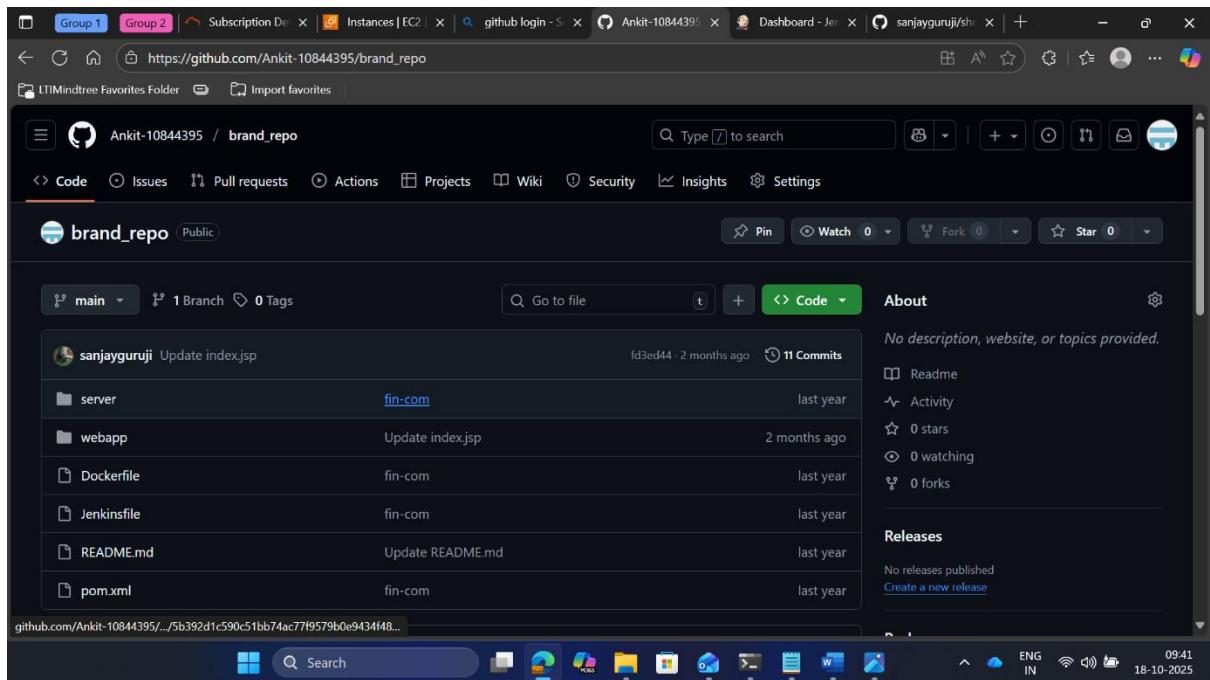
Please make sure you have the correct access rights
and the repository exists.
[root@dev shourjo-10743365]# cd
[root@dev ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
```

→ Github cloned Successfully using ssh-key authentication



```
root@ip-172-31-40-207:~# ll
total 20
-rw-r--r--. 1 root root 143 Oct 18 04:06 Dockerfile
-rw-r--r--. 1 root root 3533 Oct 18 04:06 Jenkinsfile
-rw-r--r--. 1 root root 33 Oct 18 04:06 README.md
-rw-r--r--. 1 root root 6333 Oct 18 04:06 pom.xml
drwxr-xr-x. 3 root root 32 Oct 18 04:06 server
drwxr-xr-x. 3 root root 32 Oct 18 04:06 webapp
[root@dev shourjo-10743365]# git remote -v
origin https://github.com/sanjayguruji/shourjo-10743365.git (fetch)
origin https://github.com/sanjayguruji/shourjo-10743365.git (push)
[root@dev shourjo-10743365]# git push origin main
Counting objects: 84, done.
Delta compression using up to 2 threads
Compressing objects: 100% (36/36), done.
Writing objects: 100% (84/84), 15.96 KiB | 15.96 MiB/s, done.
Total 84 (delta 17), reused 84 (delta 17), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (17/17), done.
To github.com:Ankit-108444395/brand_repo.git
 * [new branch]      main -> main
[root@dev shourjo-10743365]#
```

→ All java files obtained successfully via cloning repository

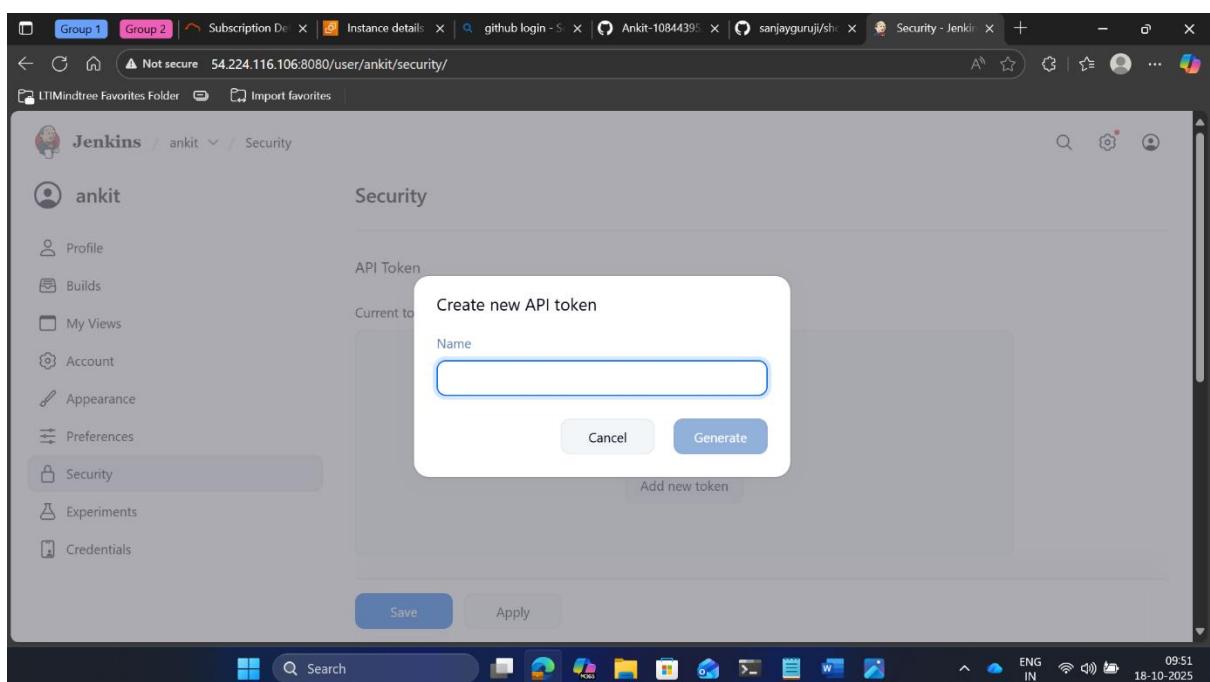


The screenshot shows a GitHub repository page for 'brand_repo'. The repository is public and contains 1 branch and 0 tags. The main branch has 11 commits. The commits are as follows:

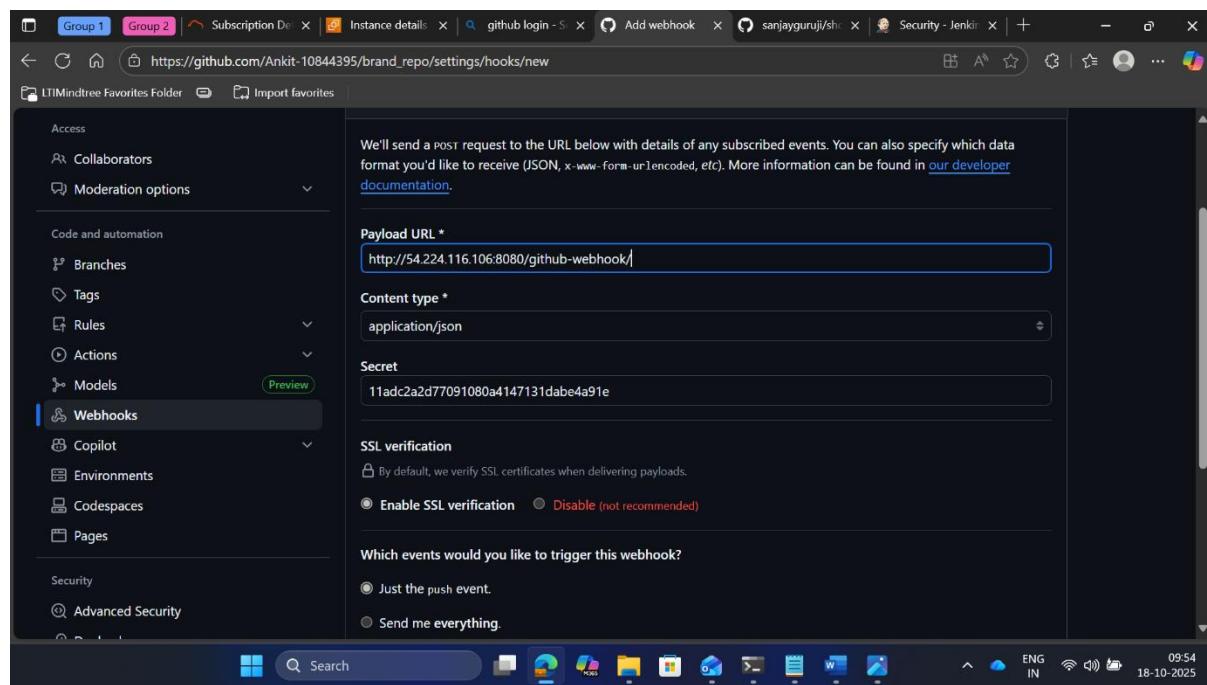
Author	Commit Message	Time
sanjayguruji	Update index.jsp	fd3ed44 · 2 months ago
	server	fin-com · last year
	webapp	Update index.jsp · 2 months ago
	Dockerfile	fin-com · last year
	Jenkinsfile	fin-com · last year
	README.md	Update README.md · last year
	pom.xml	fin-com · last year

The repository has 0 stars, 0 forks, and 0 releases. The status bar at the bottom shows the URL as github.com/Ankit-10844395/brand_repo, the date as 18-10-2025, and the time as 09:41.

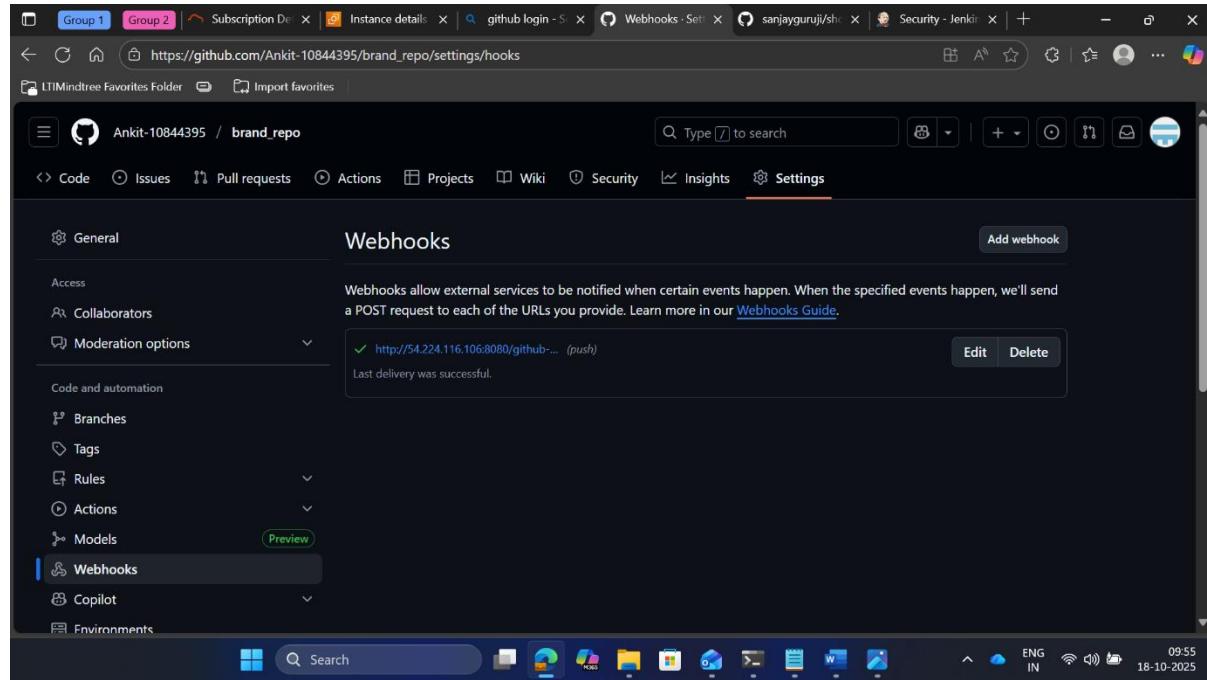
→ Connecting github to Jenkins via webhook using tokens



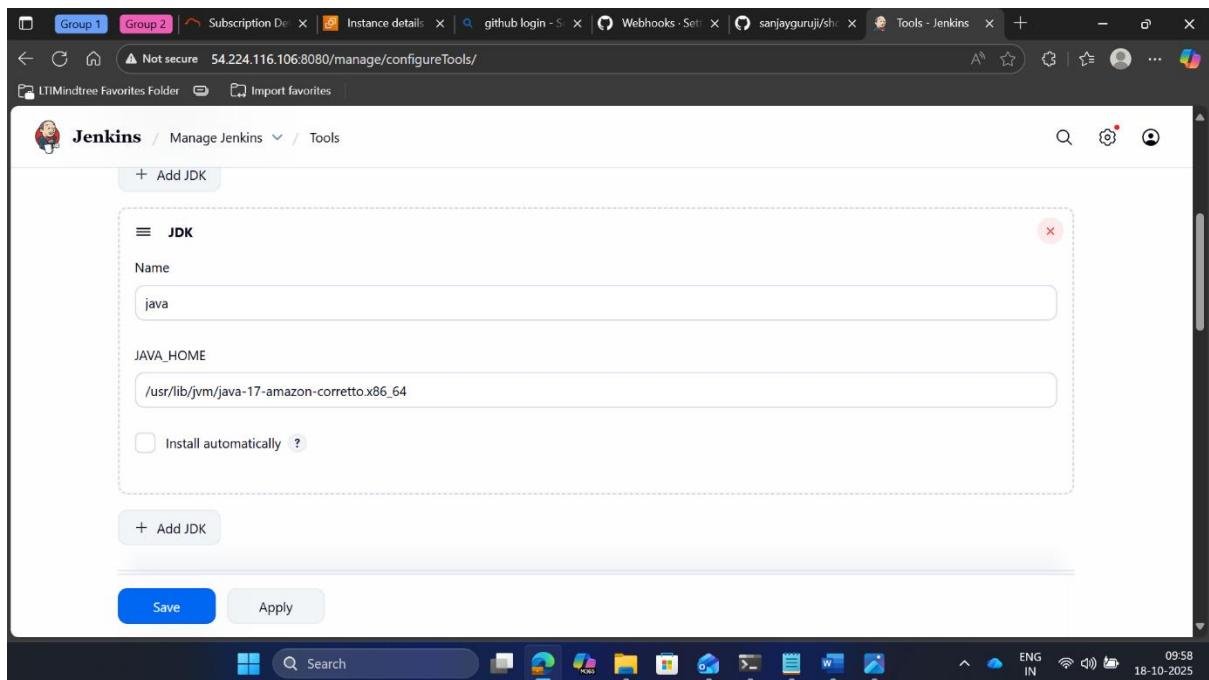
The screenshot shows the Jenkins Security page for user 'ankit'. The 'Security' tab is selected. A modal window titled 'Create new API token' is open, prompting for a 'Name'. The modal has 'Cancel' and 'Generate' buttons. Below the modal, there is a button 'Add new token'. At the bottom of the page are 'Save' and 'Apply' buttons. The status bar at the bottom shows the URL as <https://54.224.116.106:8080/user/ankit/security/>, the date as 18-10-2025, and the time as 09:51.



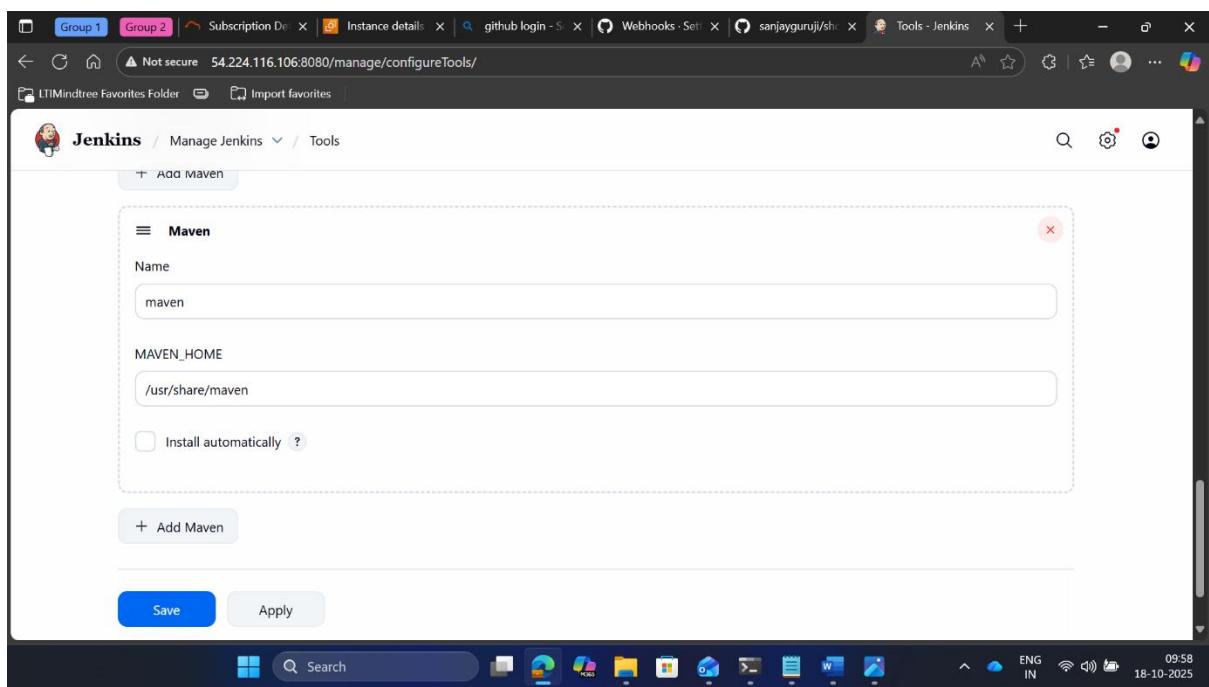
→ Webhook successfully connected



→ Configuring java and maven path to Jenkins tool

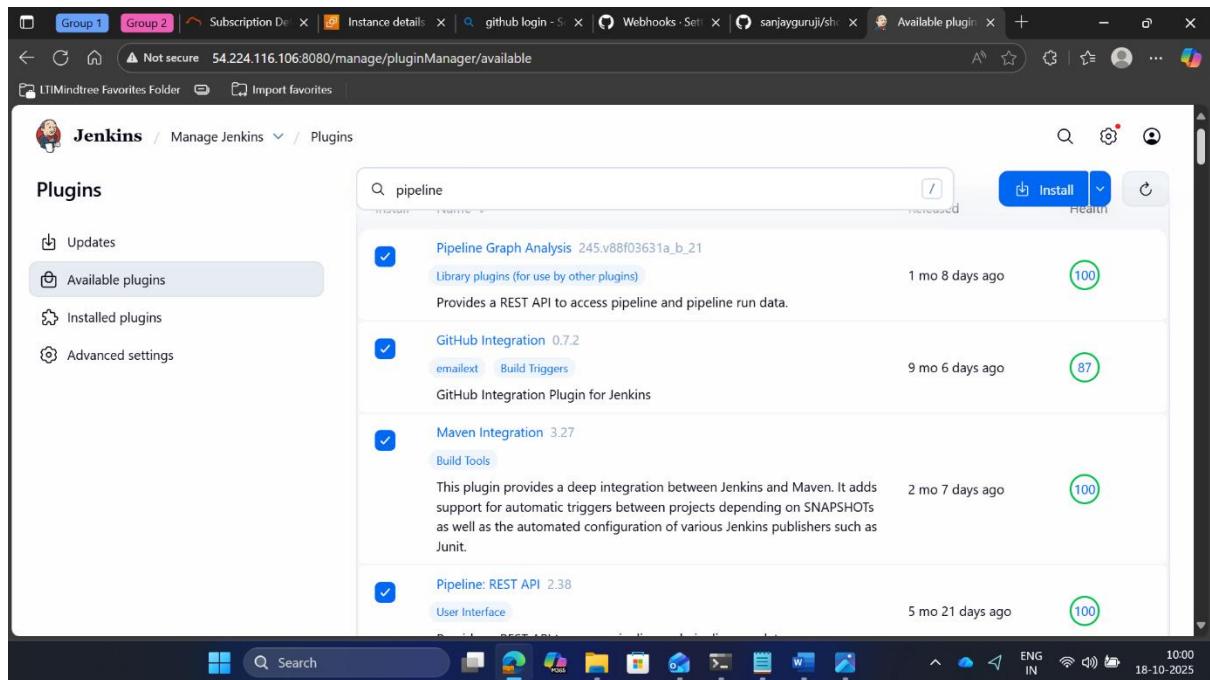


The screenshot shows the Jenkins 'Tools' configuration page. The URL in the browser is `54.224.116.106:8080/manage/configureTools/`. The page is titled 'Manage Jenkins / Tools'. A 'JDK' section is active, showing a configuration for 'java'. The 'Name' field contains 'java'. The 'JAVA_HOME' field contains `/usr/lib/jvm/java-17-amazon-corretto.x86_64`. There is an unchecked checkbox for 'Install automatically'. At the bottom are 'Save' and 'Apply' buttons.



The screenshot shows the Jenkins 'Tools' configuration page. The URL in the browser is `54.224.116.106:8080/manage/configureTools/`. The page is titled 'Manage Jenkins / Tools'. A 'Maven' section is active, showing a configuration for 'maven'. The 'Name' field contains 'maven'. The 'MAVEN_HOME' field contains `/usr/share/maven`. There is an unchecked checkbox for 'Install automatically'. At the bottom are 'Save' and 'Apply' buttons.

→ Installing essential Plugins required for deploying & integrating



Plugins

Updates

Available plugins

Installed plugins

Advanced settings

Search pipeline

Install

Pipeline Graph Analysis 245.v88f03631a_b_21

Provides a REST API to access pipeline and pipeline run data.

1 mo 8 days ago 100

GitHub Integration 0.7.2

emailext Build Triggers

GitHub Integration Plugin for Jenkins

9 mo 6 days ago 87

Maven Integration 3.2.7

Build Tools

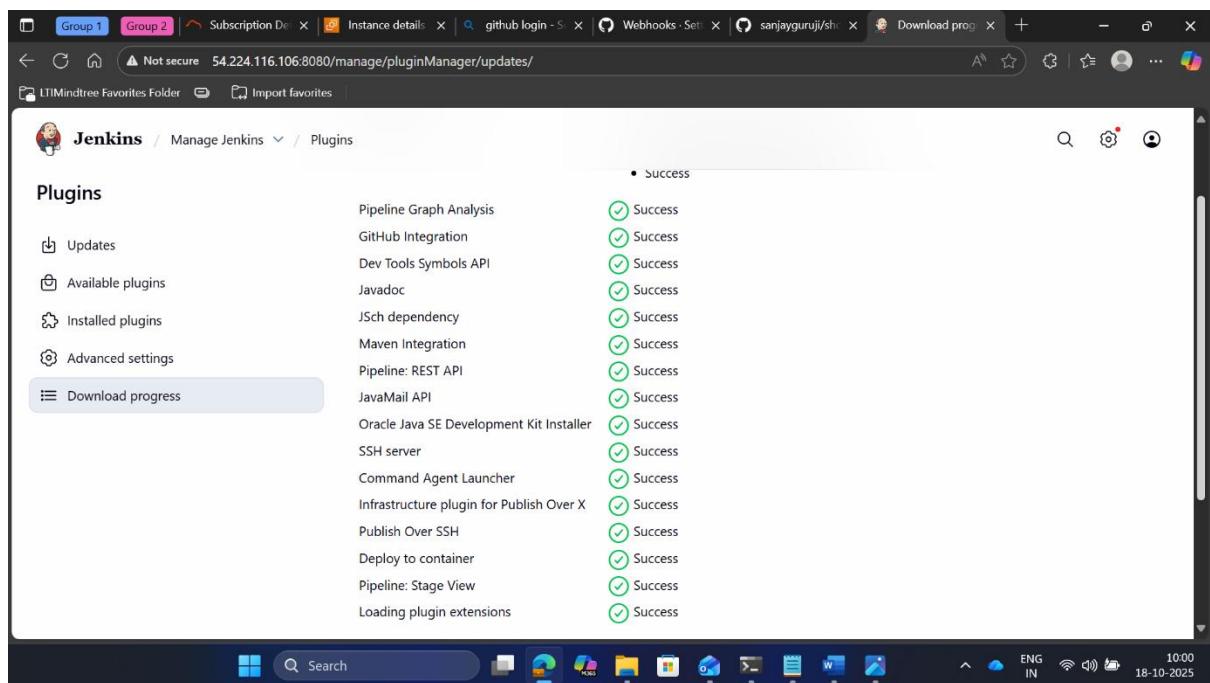
This plugin provides a deep integration between Jenkins and Maven. It adds support for automatic triggers between projects depending on SNAPSHOTs as well as the automated configuration of various Jenkins publishers such as JUnit.

2 mo 7 days ago 100

Pipeline: REST API 2.38

User Interface

5 mo 21 days ago 100



Plugins

Updates

Available plugins

Installed plugins

Advanced settings

Download progress

Success

Pipeline Graph Analysis Success

GitHub Integration Success

Dev Tools Symbols API Success

Javadoc Success

JSch dependency Success

Maven Integration Success

Pipeline: REST API Success

JavaMail API Success

Oracle Java SE Development Kit Installer Success

SSH server Success

Command Agent Launcher Success

Infrastructure plugin for Publish Over X Success

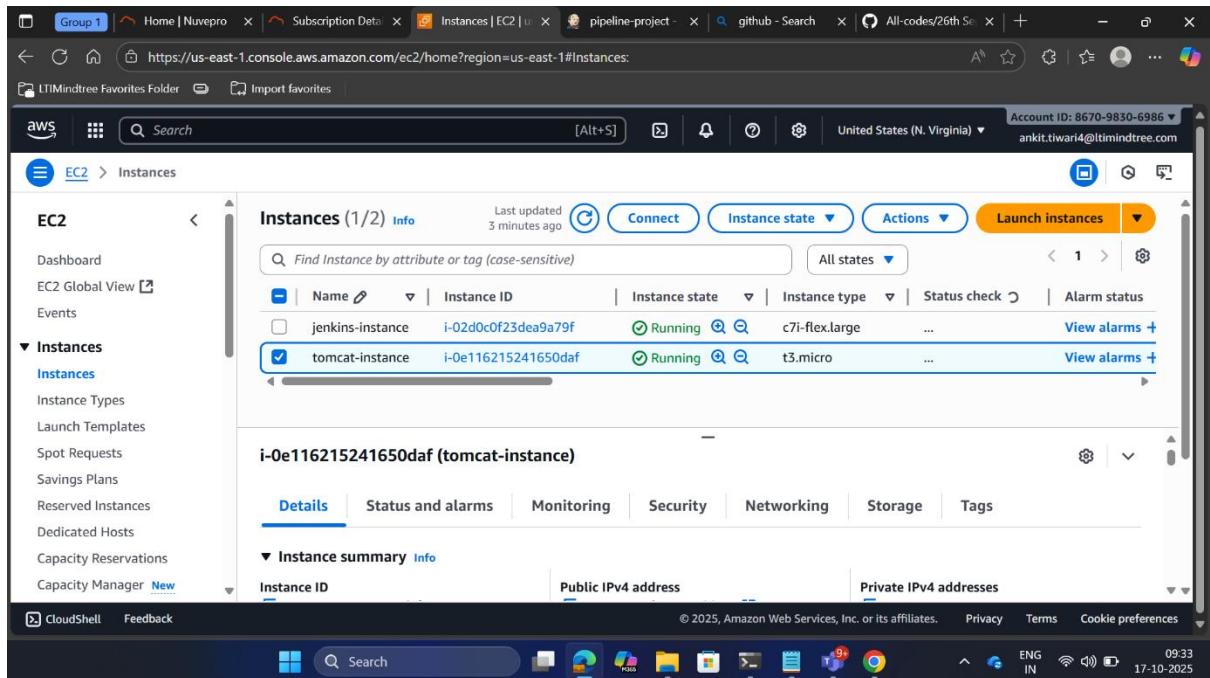
Publish Over SSH Success

Deploy to container Success

Pipeline: Stage View Success

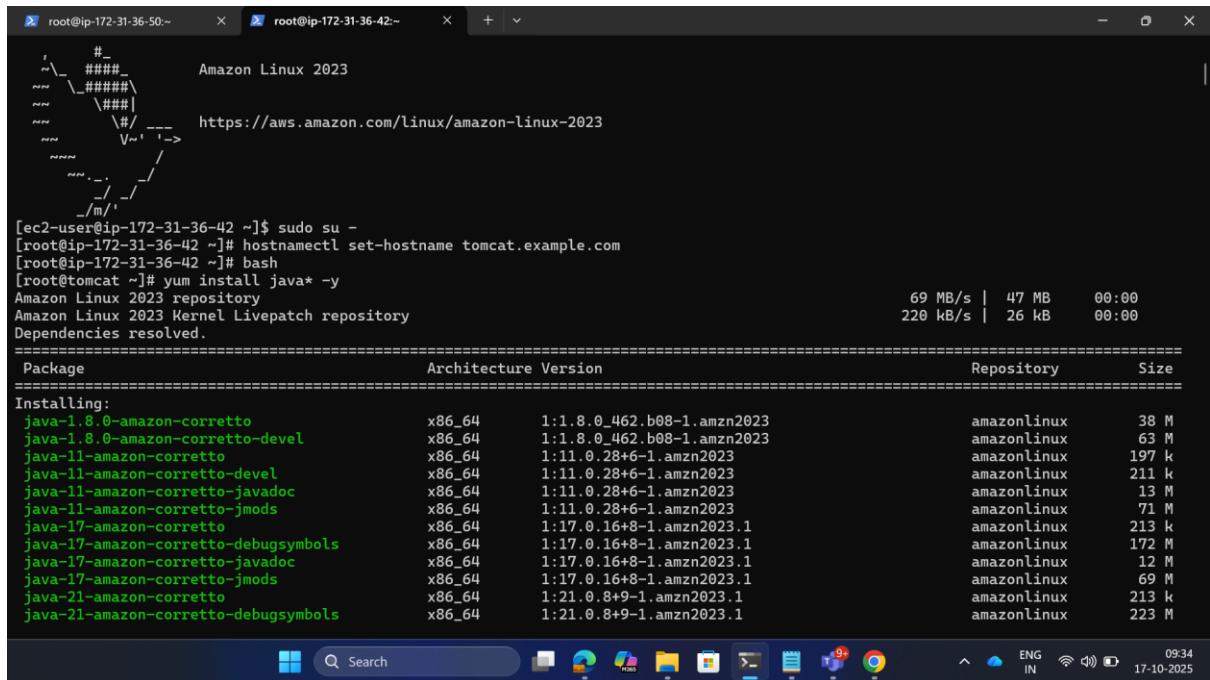
Loading plugin extensions Success

→ Creating another instance for tomcat



The screenshot shows the AWS EC2 Instances page. On the left, a sidebar navigation includes 'EC2', 'Dashboard', 'EC2 Global View', 'Events', and a 'Instances' section with 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', and 'Capacity Manager'. The main content area displays 'Instances (1/2) Info' with a table showing two instances: 'jenkins-instance' (i-02d0c0f23dea9a79f, Running, c7i-flex.large) and 'tomcat-instance' (i-0e116215241650daf, Running, t3.micro). Below the table, a detailed view for 'tomcat-instance' (i-0e116215241650daf) is shown, with tabs for 'Details', 'Status and alarms', 'Monitoring', 'Security', 'Networking', 'Storage', and 'Tags'. The 'Details' tab is selected, showing the instance ID, Public IPv4 address (172.31.36.42), and Private IPv4 addresses (172.31.36.42). The status bar at the bottom shows the date (17-10-2025) and time (09:33).

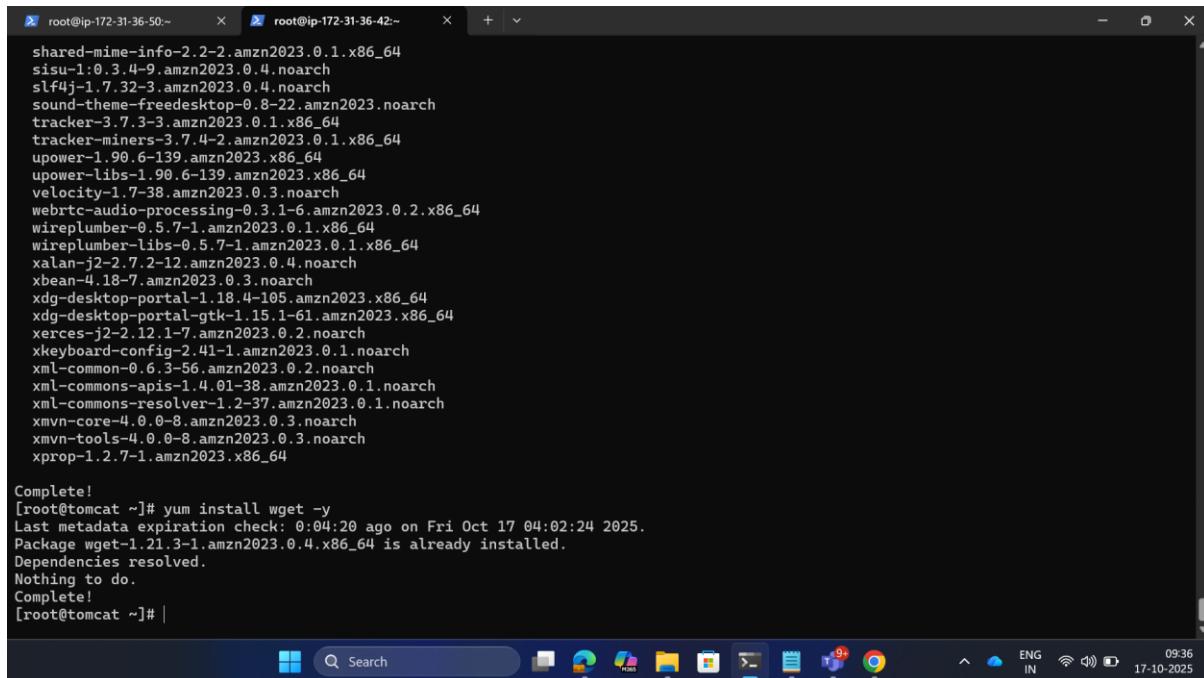
→ Installing all the java files on tomcat server



```
[ec2-user@ip-172-31-36-42 ~]$ sudo su -
[root@ip-172-31-36-42 ~]# hostnamectl set-hostname tomcat.example.com
[root@ip-172-31-36-42 ~]# bash
[root@tomcat ~]# yum install java* -y
Amazon Linux 2023 repository
Amazon Linux 2023 Kernel Livepatch repository
Dependencies resolved.
=====
Package           Architecture Version      Repository  Size
=====
Installing:
  java-1.8.0-amazon-corretto           x86_64    1:1.8.0_462.b08-1.amzn2023  amazonlinux  38 M
  java-1.8.0-amazon-corretto-devel     x86_64    1:1.8.0_462.b08-1.amzn2023  amazonlinux  63 M
  java-11-amazon-corretto             x86_64    1:11.0.28+6-1.amzn2023      amazonlinux  197 k
  java-11-amazon-corretto-devel       x86_64    1:11.0.28+6-1.amzn2023      amazonlinux  211 k
  java-11-amazon-corretto-javadoc    x86_64    1:11.0.28+6-1.amzn2023      amazonlinux  13 M
  java-11-amazon-corretto-jmod       x86_64    1:11.0.28+6-1.amzn2023      amazonlinux  71 M
  java-17-amazon-corretto            x86_64    1:17.0.16+8-1.amzn2023.1    amazonlinux  213 k
  java-17-amazon-corretto-debugsymbols x86_64    1:17.0.16+8-1.amzn2023.1    amazonlinux  172 M
  java-17-amazon-corretto-javadoc    x86_64    1:17.0.16+8-1.amzn2023.1    amazonlinux  12 M
  java-17-amazon-corretto-jmod       x86_64    1:17.0.16+8-1.amzn2023.1    amazonlinux  69 M
  java-21-amazon-corretto            x86_64    1:21.0.8+9-1.amzn2023.1     amazonlinux  213 k
  java-21-amazon-corretto-debugsymbols x86_64    1:21.0.8+9-1.amzn2023.1     amazonlinux  223 M

[ec2-user@ip-172-31-36-42 ~]$
```

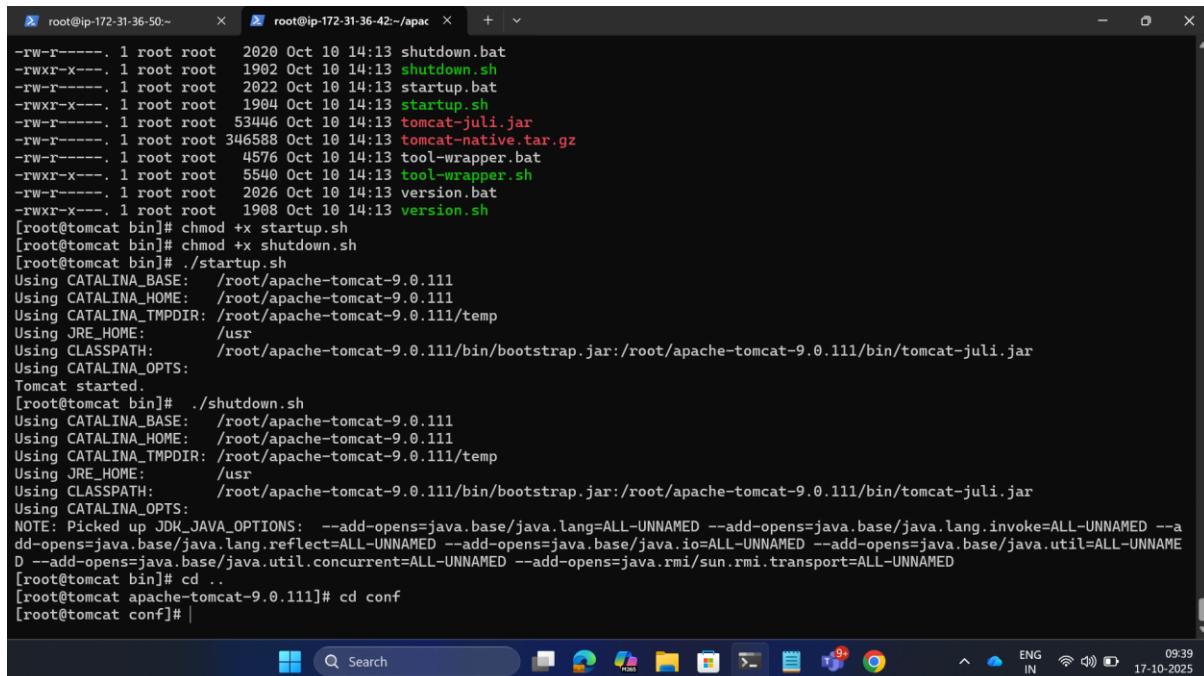
→ Installing tomcat 9 on the server



```
root@ip-172-31-36-50:~ x root@ip-172-31-36-42:~ + x
shared-mime-info-2.2-2.amzn2023.0.1.x86_64
sisu-1.0.3.4-9.amzn2023.0.4.noarch
slf4j-1.7.32-3.amzn2023.0.4.noarch
sound-theme-freedesktop-0.8-22.amzn2023.noarch
tracker-3.7.3-3.amzn2023.0.1.x86_64
tracker-miners-3.7.4-2.amzn2023.0.1.x86_64
upower-1.90.6-139.amzn2023.x86_64
upower-libs-1.90.6-139.amzn2023.x86_64
velocity-1.7.38.amzn2023.0.3.noarch
webrtc-audio-processing-0.3.1-6.amzn2023.0.2.x86_64
wireplumber-0.5.7-1.amzn2023.0.1.x86_64
wireplumber-libs-0.5.7-1.amzn2023.0.1.x86_64
xalan-j2-2.7.2-12.amzn2023.0.4.noarch
xbean-4.18-7.amzn2023.0.3.noarch
xdg-desktop-portal-1.18.4-105.amzn2023.x86_64
xdg-desktop-portal-gtk-1.15.1-61.amzn2023.x86_64
xerces-j2-2.12.1-7.amzn2023.0.2.noarch
xkeyboard-config-2.41-1.amzn2023.0.1.noarch
xml-common-0.6.3-56.amzn2023.0.2.noarch
xml-commons-apis-1.4.01-38.amzn2023.0.1.noarch
xml-commons-resolver-1.2-37.amzn2023.0.1.noarch
xmvn-core-4.0.0-8.amzn2023.0.3.noarch
xmvn-tools-4.0.0-8.amzn2023.0.3.noarch
xprop-1.2.7-1.amzn2023.x86_64

Complete!
[root@tomcat ~]# yum install wget -y
Last metadata expiration check: 0:04:20 ago on Fri Oct 17 04:02:24 2025.
Package wget-1.21.3-1.amzn2023.0.4.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@tomcat ~]# |
```

→ All the files are present in the tomcat apache server



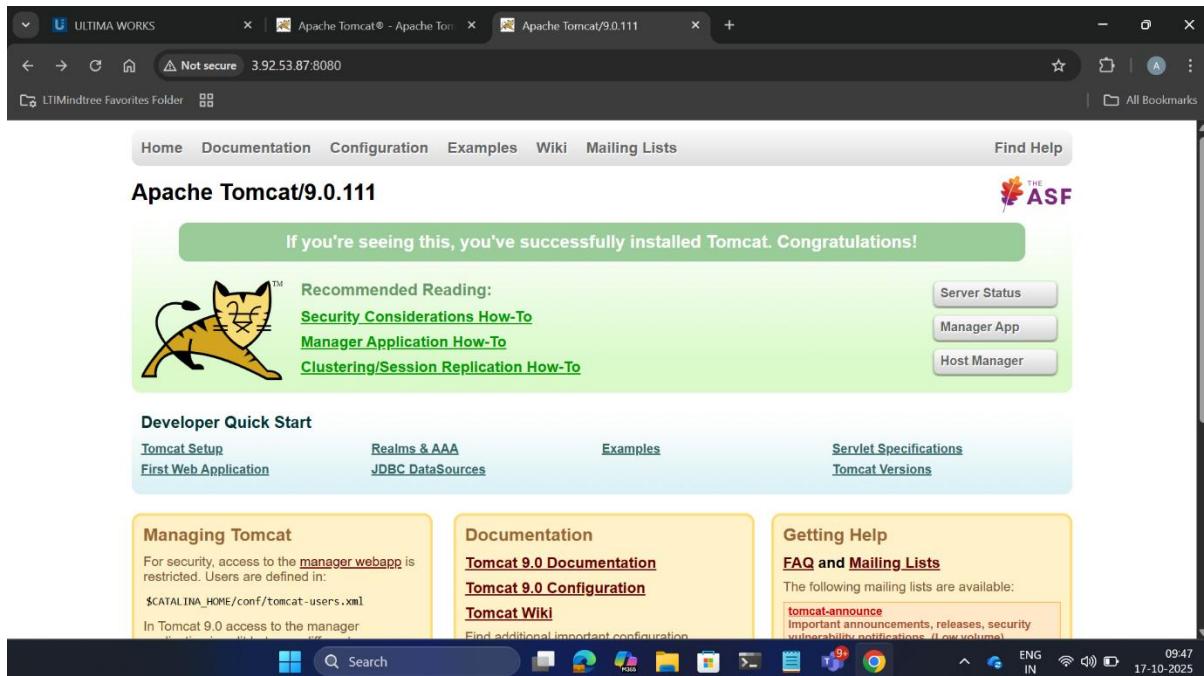
```
root@ip-172-31-36-50:~ x root@ip-172-31-36-42:~/apac x + x
-rw-r----- 1 root root 2020 Oct 10 14:13 shutdown.bat
-rwxr-x--- 1 root root 1902 Oct 10 14:13 shutdown.sh
-rw-r----- 1 root root 2022 Oct 10 14:13 startup.bat
-rwxr-x--- 1 root root 1984 Oct 10 14:13 startup.sh
-rw-r----- 1 root root 53446 Oct 10 14:13 tomcat-juli.jar
-rw-r----- 1 root root 346588 Oct 10 14:13 tomcat-native.tar.gz
-rw-r----- 1 root root 4576 Oct 10 14:13 tool-wrapper.bat
-rwxr-x--- 1 root root 5540 Oct 10 14:13 tool-wrapper.sh
-rw-r----- 1 root root 2026 Oct 10 14:13 version.bat
-rwxr-x--- 1 root root 1908 Oct 10 14:13 version.sh
[root@tomcat bin]# chmod +x startup.sh
[root@tomcat bin]# chmod +x shutdown.sh
[root@tomcat bin]# ./startup.sh
Using CATALINA_BASE: /root/apache-tomcat-9.0.111
Using CATALINA_HOME: /root/apache-tomcat-9.0.111
Using CATALINA_TMPDIR: /root/apache-tomcat-9.0.111/temp
Using JRE_HOME: /usr
Using CLASSPATH: /root/apache-tomcat-9.0.111/bin/bootstrap.jar:/root/apache-tomcat-9.0.111/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@tomcat bin]# ./shutdown.sh
Using CATALINA_BASE: /root/apache-tomcat-9.0.111
Using CATALINA_HOME: /root/apache-tomcat-9.0.111
Using CATALINA_TMPDIR: /root/apache-tomcat-9.0.111/temp
Using JRE_HOME: /usr
Using CLASSPATH: /root/apache-tomcat-9.0.111/bin/bootstrap.jar:/root/apache-tomcat-9.0.111/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --a
dd-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAME
D --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
[root@tomcat bin]# cd ..
[root@tomcat apache-tomcat-9.0.111]# cd conf
[root@tomcat conf]# |
```

→ Tomcat successfully started

```
root@ip-172-31-36-50:~ x root@ip-172-31-36-42:~/apac x + x
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --a
dd-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAME
D --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
Oct 17, 2025 4:13:05 AM org.apache.catalina.startup.Catalina stopServer
SEVERE: Could not contact [localhost:8005] (base port [8005] and offset [0]). Tomcat may not be running.
Oct 17, 2025 4:13:05 AM org.apache.catalina.startup.Catalina stopServer
SEVERE: Error stopping Catalina
java.net.ConnectException: Connection refused
    at java.base/sun.nio.ch.Net.connect0(Native Method)
    at java.base/sun.nio.ch.Net.connect(Net.java:546)
    at java.base/sun.nio.ch.Net.connect(Net.java:535)
    at java.base/sun.nio.ch.NioSocketImpl.connect(NioSocketImpl.java:585)
    at java.base/java.net.SocksSocketImpl.connect(SocksSocketImpl.java:284)
    at java.base/java.net.Socket.connect(Socket.java:659)
    at java.base/java.net.Socket.connect(Socket.java:597)
    at java.base/java.net.Socket.<init>(Socket.java:464)
    at java.base/java.net.Socket.<init>(Socket.java:276)
    at org.apache.catalina.startup.Catalina.stopServer(Catalina.java:630)
    at java.base/jdk.internal.reflect.DirectMethodHandleAccessor.invoke(DirectMethodHandleAccessor.java:104)
    at java.base/java.lang.reflect.Method.invoke(Method.java:565)
    at org.apache.catalina.startup.Bootstrap.stopServer(Bootstrap.java:390)
    at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:479)

[root@tomcat bin]# ./startup.sh
Using CATALINA_BASE: /root/apache-tomcat-9.0.111
Using CATALINA_HOME: /root/apache-tomcat-9.0.111
Using CATALINA_TMPDIR: /root/apache-tomcat-9.0.111/temp
Using JRE_HOME: /usr
Using CLASSPATH: /root/apache-tomcat-9.0.111/bin/bootstrap.jar:/root/apache-tomcat-9.0.111/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@tomcat bin]#
```

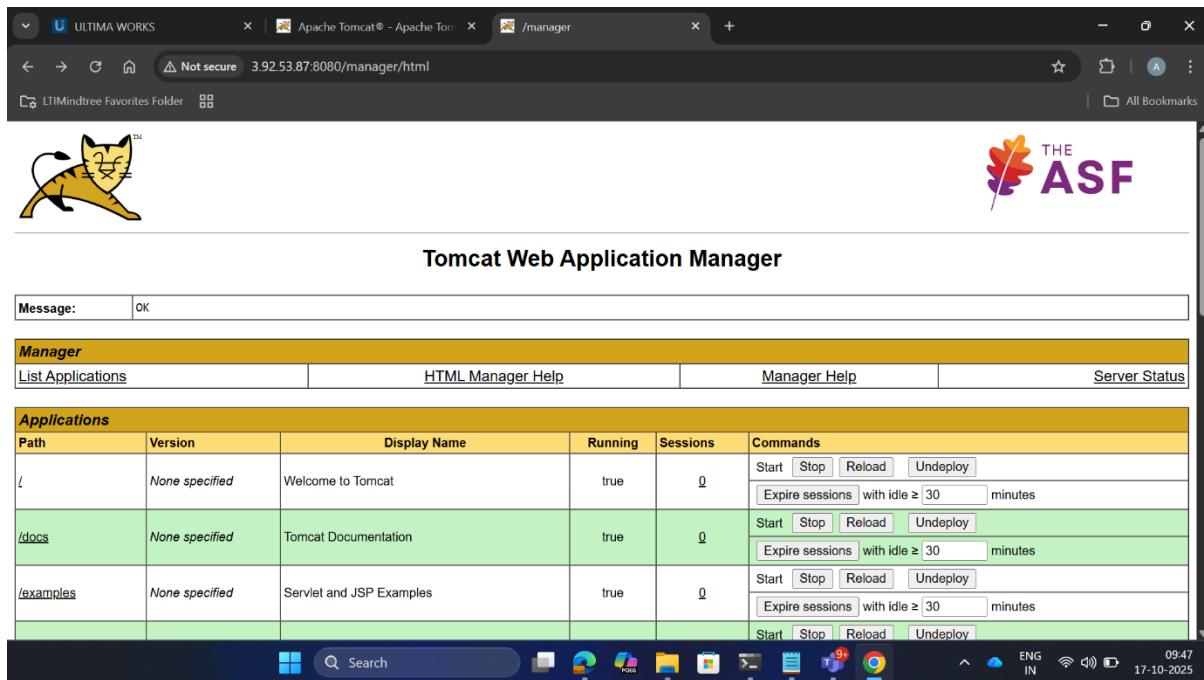
→ Tomcat Default page accessible



The screenshot shows a web browser window displaying the Apache Tomcat 9.0.111 default page. The URL in the address bar is `3.92.53.87:8080`. The page content includes:

- Apache Tomcat/9.0.111** logo and title.
- A green banner at the top stating: **If you're seeing this, you've successfully installed Tomcat. Congratulations!**
- A cartoon cat icon.
- Recommended Reading:**
 - [Security Considerations How-To](#)
 - [Manager Application How-To](#)
 - [Clustering/Session Replication How-To](#)
- Developer Quick Start** section with links:
 - [Tomcat Setup](#)
 - [First Web Application](#)
 - [Realms & AAA](#)
 - [JDBC DataSources](#)
 - [Examples](#)
 - [Servlet Specifications](#)
 - [Tomcat Versions](#)
- Documentation** section with links:
 - [Tomcat 9.0 Documentation](#)
 - [Tomcat 9.0 Configuration](#)
 - [Tomcat Wiki](#)
- Getting Help** section with links:
 - [FAQ and Mailing Lists](#)
 - The following mailing lists are available:
 - [tomcat-announce](#) (Important announcements, releases, security vulnerability notifications, if any volume)

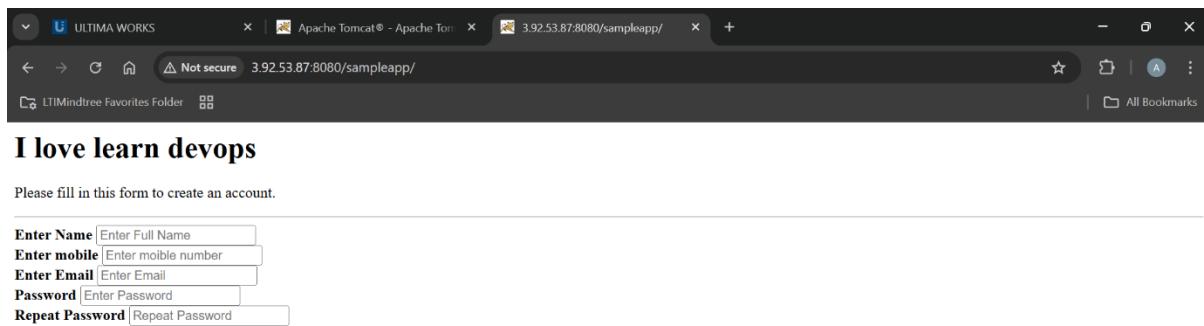
→ Accessing Tomcat manager to see if changes are made



The screenshot shows a web browser window with the URL <http://3.92.53.87:8080/manager/html>. The page title is "Tomcat Web Application Manager". The content area shows a table of applications with the following data:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

→ Changes are getting reflected on tomcat server



The screenshot shows a web browser window with the URL <http://3.92.53.87:8080/sampleapp/>. The page title is "Apache Tomcat - Apache Tomcat". The content area shows a registration form:

Please fill in this form to create an account.

Enter Name	<input type="text"/>
Enter mobile	<input type="text"/>
Enter Email	<input type="text"/>
Password	<input type="password"/>
Repeat Password	<input type="password"/>

By creating an account you agree to our [Terms & Privacy](#).

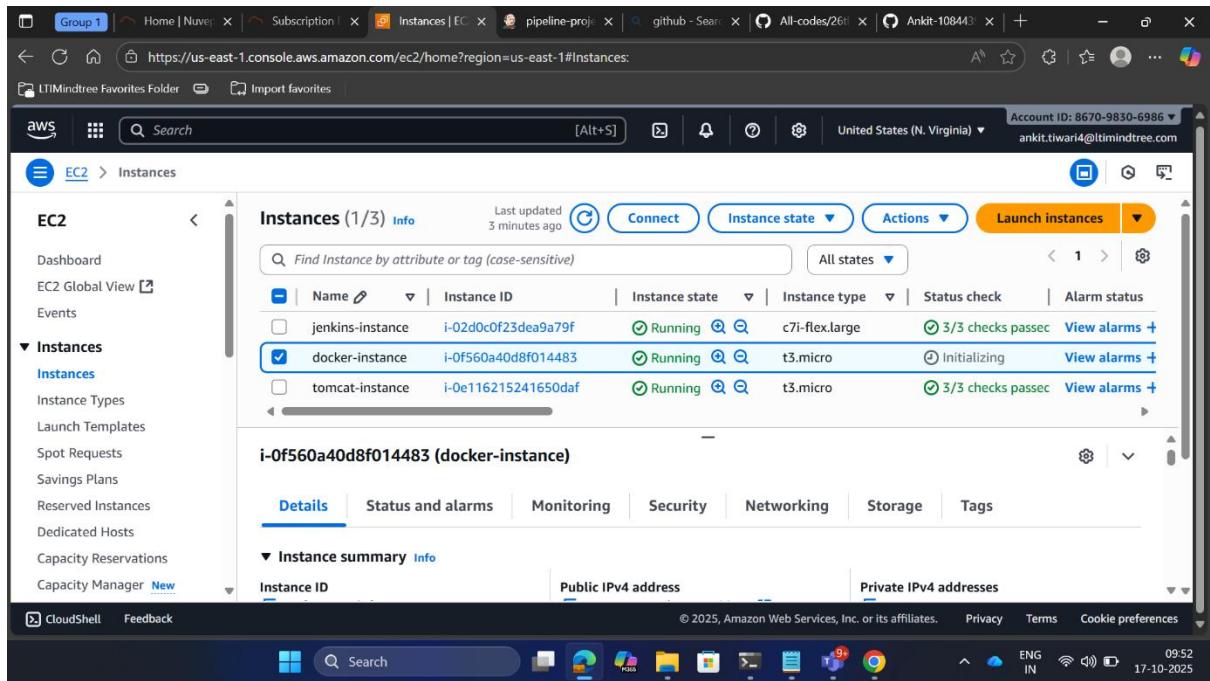
Already have an account? [Sign in](#).

Thank You

bye



→ Now creating docker instance

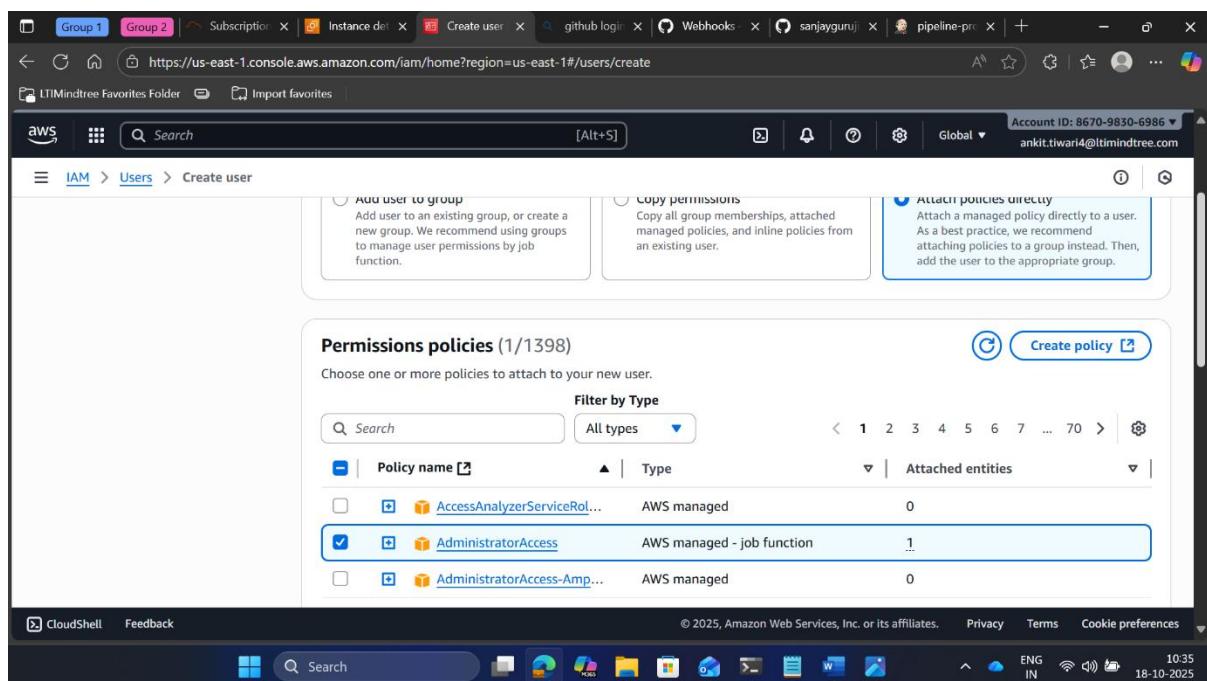
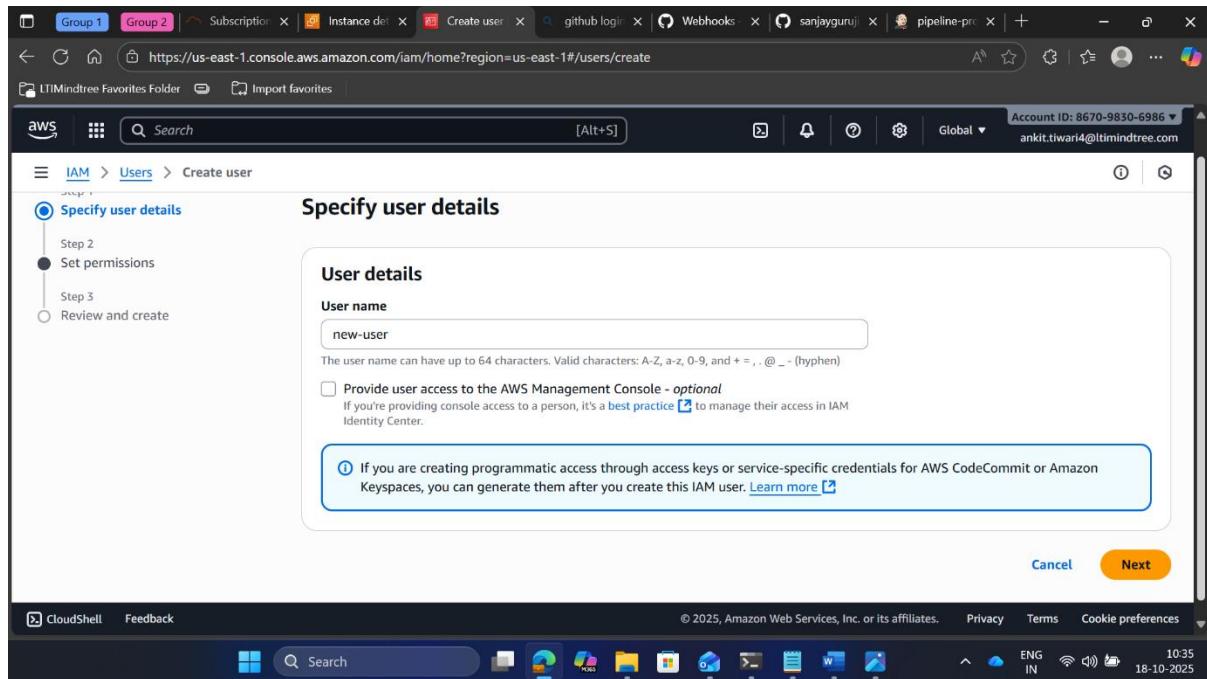


→ Installing docker

→ Creating new IAM user and role and providing policies

User : AdministorFullAcess

Roles: EKS-cluster , IAM full access, ec2aDMINRegistry full access



Group 1 Group 2 Subscription Instance del Users | IAM github logi Webhooks sanjayguruji pipeline-pro

https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/users

LTMindtree Favorites Folder Import favorites

aws Search [Alt+S] Account ID: 8670-9830-6986 ankit.tiwari4@ltimindtree.com

IAM > Users

Identity and Access Management (IAM)

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

View user

Users (2) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search

User name Path Group: Last activity MFA Password age

User name	Path	Group:	Last activity	MFA	Password age
ankit.tiwari4@ltimindtree.com	/	1	1 hour ago	-	26 days
new-user	/	0	-	-	-

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:35 18-10-2025

Search

Group 1 Group 2 Subscription Instance del new-role github logi Webhooks sanjayguruji pipeline-pro

https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/roles/details/new-role

LTMindtree Favorites Folder Import favorites

aws Search [Alt+S] Account ID: 8670-9830-6986 ankit.tiwari4@ltimindtree.com

IAM > Roles > new-role

Identity and Access Management (IAM)

new-role Info

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date: October 17, 2025, 11:48 (UTC+05:30)

ARN: arn:aws:iam::867098306986:role/new-role

Last activity: 16 hours ago

Maximum session duration: 1 hour

ARN: arn:aws:iam::867098306986:instance-profile/new-role

Permissions Trust relationships Tags Last Accessed Revoke sessions

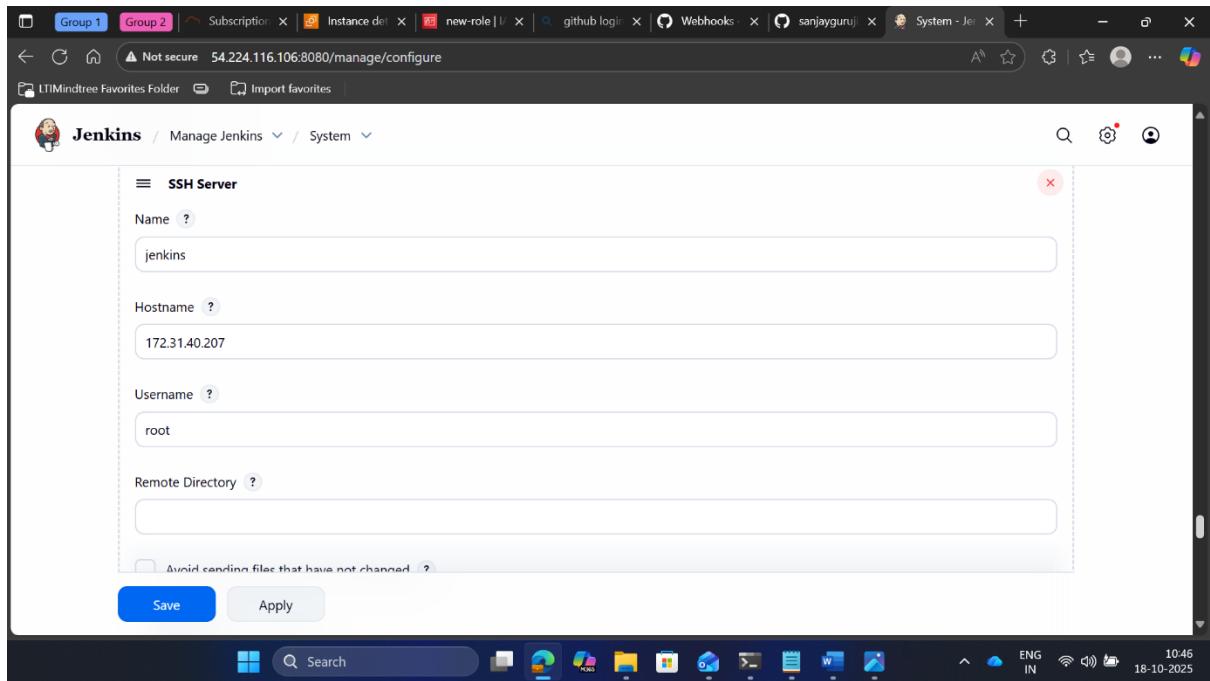
Permissions policies (0) Info

You can attach up to 10 managed policies.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 10:35 18-10-2025

Search

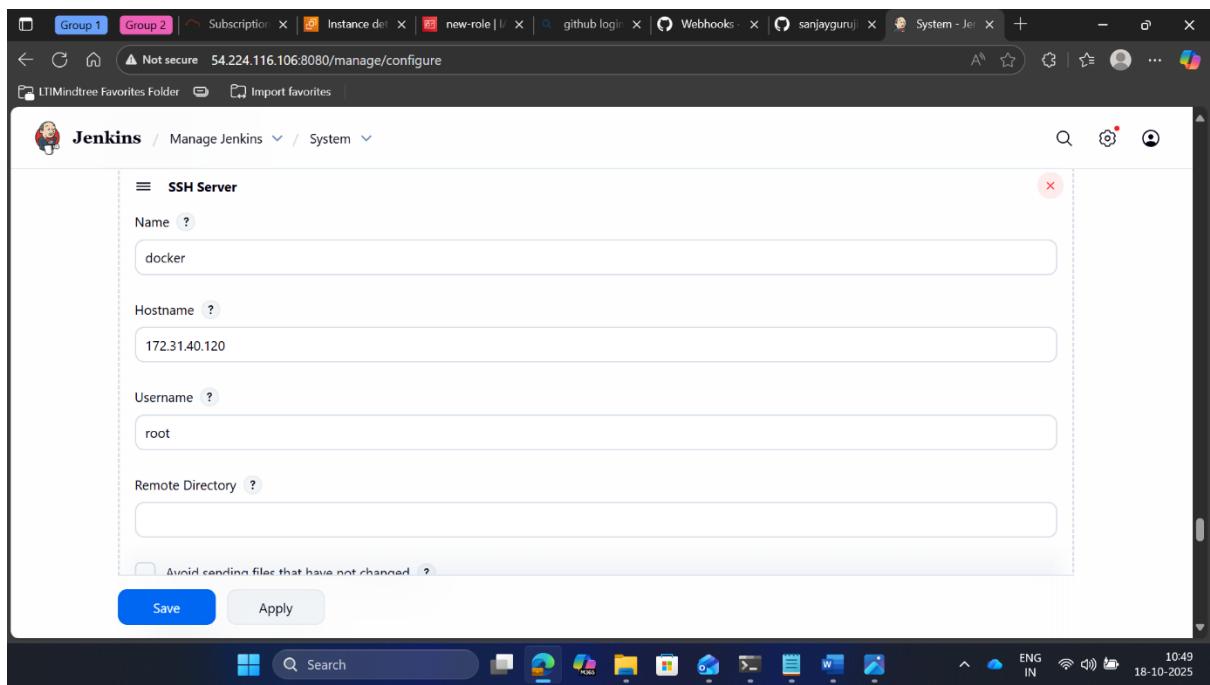
→ Adding SSH server for docker , Jenkins



The screenshot shows the Jenkins management interface. The URL is 54.224.116.106:8080/manage/configure. The page is titled 'System' and shows the 'SSH Server' configuration. The 'jenkins' server is listed with the following details:

- Name: jenkins
- Hostname: 172.31.40.207
- Username: root
- Remote Directory: (empty)

At the bottom, there are 'Save' and 'Apply' buttons. The status bar at the bottom right shows '10:46 18-10-2025'.



The screenshot shows the Jenkins management interface. The URL is 54.224.116.106:8080/manage/configure. The page is titled 'System' and shows the 'SSH Server' configuration. The 'docker' server is listed with the following details:

- Name: docker
- Hostname: 172.31.40.120
- Username: root
- Remote Directory: (empty)

At the bottom, there are 'Save' and 'Apply' buttons. The status bar at the bottom right shows '10:49 18-10-2025'.

→ Creating elastic container registry (ECR) and accessing the images in the ecr

```
root@ip-172-31-36-50-:~> root@ip-172-31-36-42-:~/apaci > root@ip-172-31-42-120-:~/dep >
+-----+
| imageTagMutability | MUTABLE
| registryId         | 867098306986
| repositoryArn      | arn:aws:ecr:us-east-1:867098306986:repository/tomcat-webapp
| repositoryName     | tomcat-webapp
| repositoryUri      | 867098306986.dkr.ecr.us-east-1.amazonaws.com/tomcat-webapp
+-----+
|           encryptionConfiguration
+-----+
| | encryptionType | AES256
+-----+
|           imageScanningConfiguration
+-----+
| | scanOnPush | False
+-----+
[root@docker ~]# passwd root
Changing password for user root.
New password:
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@docker ~]# mkdir /deploy
[root@docker ~]# systemctl restart docker
[root@docker ~]# systemctl enable docker
[root@docker ~]# cd deploy
[root@docker deploy]# ll
total 20
-rw-r--r--. 1 root root 143 Oct 17 04:50 Dockerfile
-rw-r--r--. 1 root root 3533 Oct 17 04:50 Jenkinsfile
-rw-r--r--. 1 root root 33 Oct 17 04:50 README.md
-rw-r--r--. 1 root root 6333 Oct 17 04:50 pom.xml
drwxr-xr-x. 4 root root 46 Oct 17 04:48 server
drwxr-xr-x. 4 root root 46 Oct 17 04:48 webapp
[root@docker deploy]#
```

→ Image accessible on ECR

Amazon Elastic Container Registry

Private registry

Images

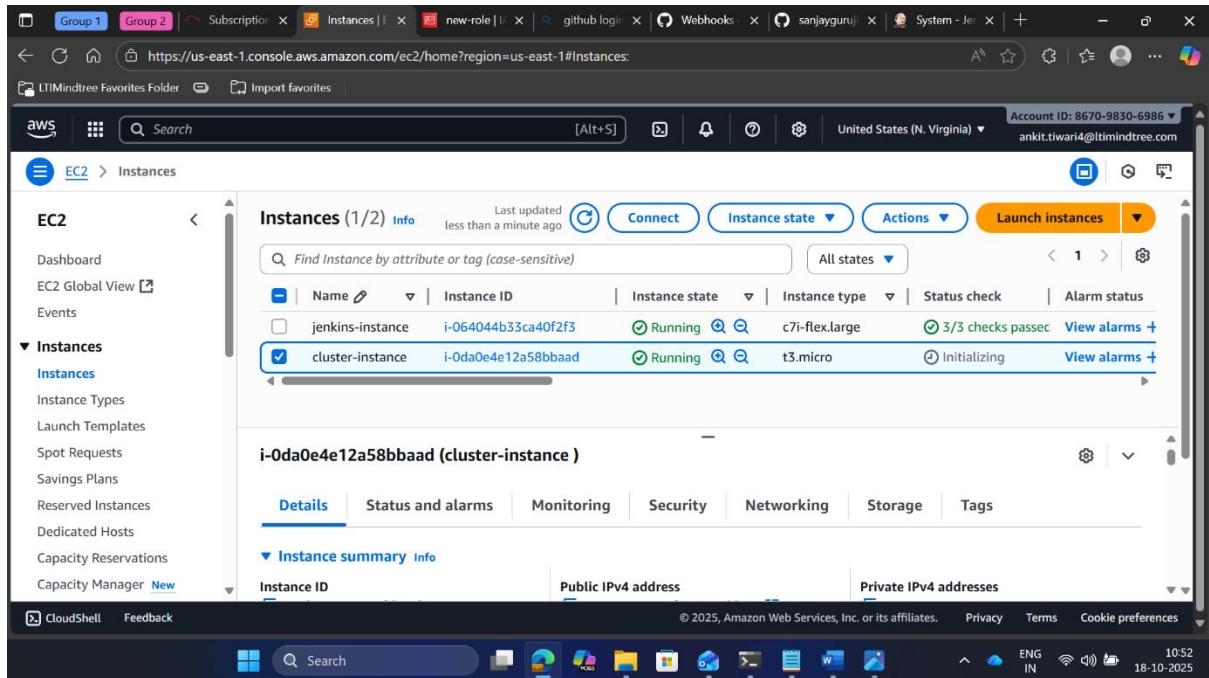
Images (1)

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Last recorded pull time
latest	Image	October 17, 2025, 10:21:34 (UTC+05.5)	156.86	<input type="button" value="Copy URI"/>	sha256:1095d60...	-

CloudShell Feedback

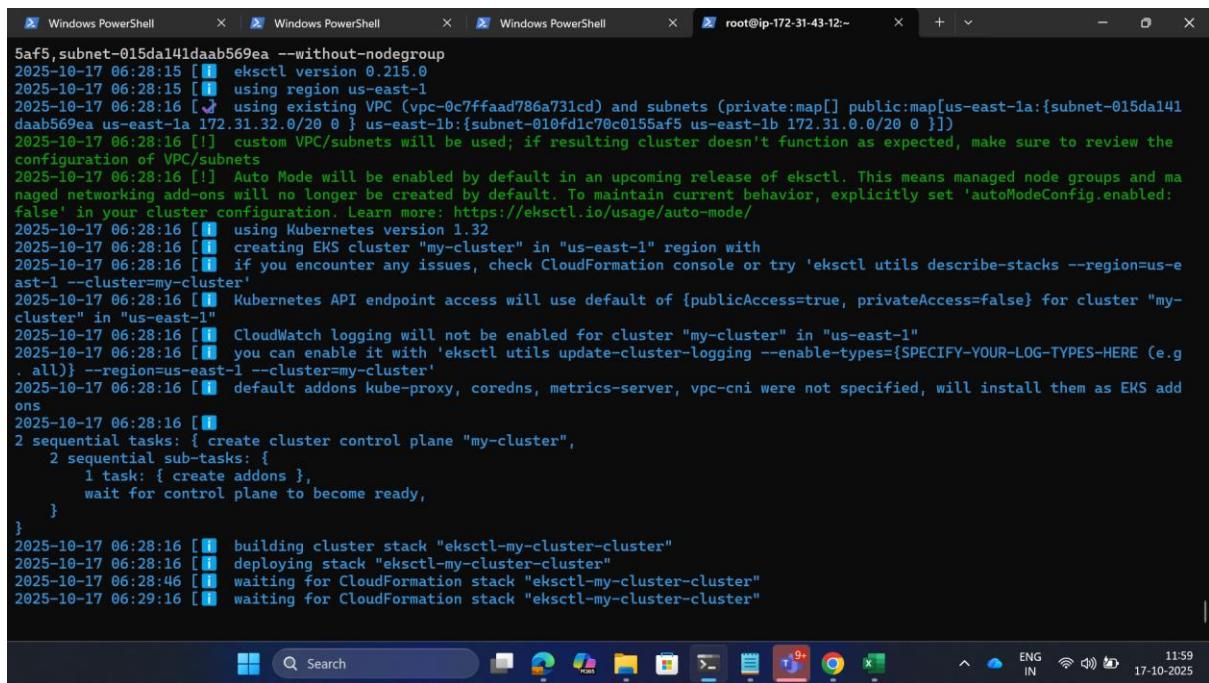
© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

→ Creating eks-cluster instance for making cluster and nodes



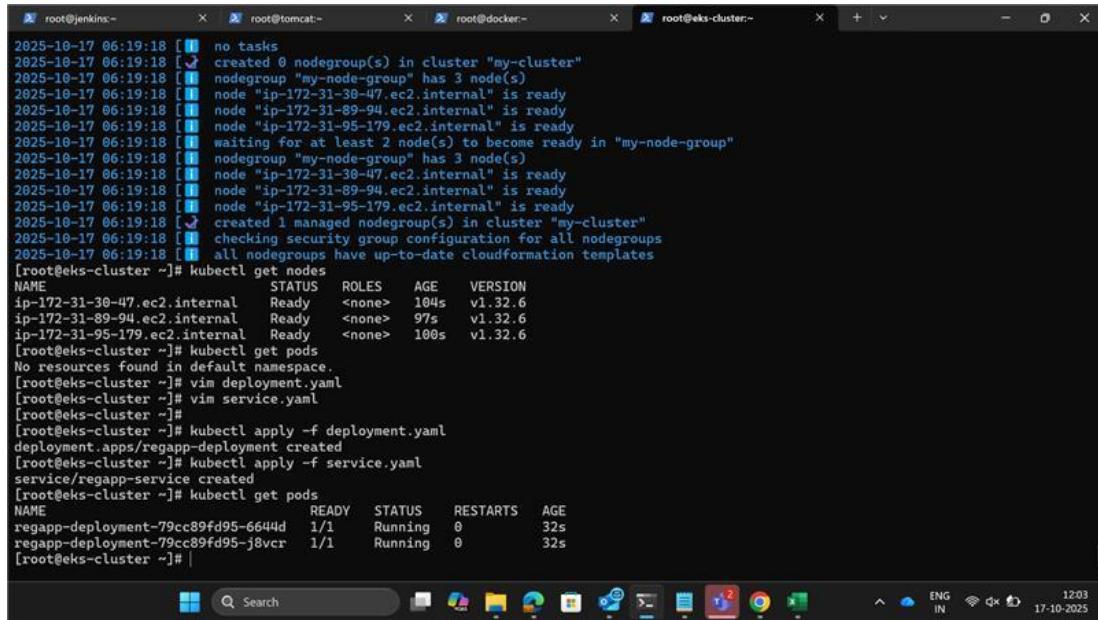
The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main area displays two instances: 'jenkins-instance' (running, c7i-flex.large) and 'cluster-instance' (running, t3.micro). The 'cluster-instance' is selected. Below the table, the instance details for 'i-0da0e4e12a58bbaad (cluster-instance)' are shown, including tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The Details tab is selected, showing the Instance ID, Public IPv4 address (172.31.32.0), and Private IPv4 addresses (172.31.32.1-2). The status is shown as 'Running' with a green checkmark and '3/3 checks passed'.

→ Cluster successfully created



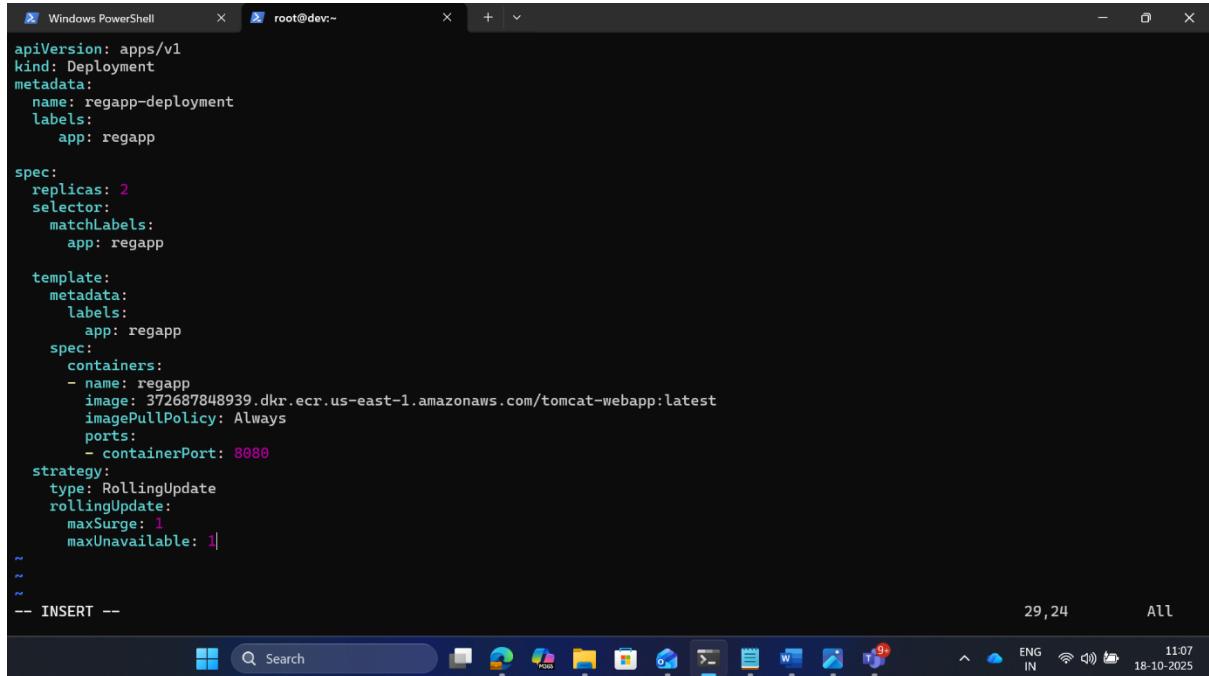
```
5af5,subnet-015dal141daab569ea --without-nodegroup
2025-10-17 06:28:15 [!] eksctl version 0.215.0
2025-10-17 06:28:15 [!] using region us-east-1
2025-10-17 06:28:16 [!] using existing VPC (vpc-0c7ffaad786a731cd) and subnets (private:map[] public:map[us-east-1a:{subnet-015dal141daab569ea us-east-1a 172.31.32.0/20 0 } us-east-1b:{subnet-010fdic70c0155af5 us-east-1b 172.31.0.0/20 0 }])
2025-10-17 06:28:16 [!] custom VPC/subnets will be used; if resulting cluster doesn't function as expected, make sure to review the configuration of VPC/subnets
2025-10-17 06:28:16 [!] Auto Mode will be enabled by default in an upcoming release of eksctl. This means managed node groups and managed networking add-ons will no longer be created by default. To maintain current behavior, explicitly set 'autoModeConfig.enabled: false' in your cluster configuration. Learn more: https://eksctl.io/usage/auto-mode/
2025-10-17 06:28:16 [!] using Kubernetes version 1.32
2025-10-17 06:28:16 [!] creating EKS cluster "my-cluster" in "us-east-1" region with
2025-10-17 06:28:16 [!] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-east-1 --cluster=my-cluster'
2025-10-17 06:28:16 [!] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "my-cluster" in "us-east-1"
2025-10-17 06:28:16 [!] CloudWatch logging will not be enabled for cluster "my-cluster" in "us-east-1"
2025-10-17 06:28:16 [!] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. . all)} --region=us-east-1 --cluster=my-cluster'
2025-10-17 06:28:16 [!] default addons kube-proxy, coredns, metrics-server, vpc-cni were not specified, will install them as EKS add-ons
2025-10-17 06:28:16 [!]
2 sequential tasks: { create cluster control plane "my-cluster",
  2 sequential sub-tasks: {
    1 task: { create addons },
    wait for control plane to become ready,
  }
}
2025-10-17 06:28:16 [!] building cluster stack "eksctl-my-cluster-cluster"
2025-10-17 06:28:16 [!] deploying stack "eksctl-my-cluster-cluster"
2025-10-17 06:28:46 [!] waiting for CloudFormation stack "eksctl-my-cluster-cluster"
2025-10-17 06:29:16 [!] waiting for CloudFormation stack "eksctl-my-cluster-cluster"
```

➔ Creating node group and checking if nodes created



```
2025-10-17 06:19:18 [root@eks-cluster ~]# no tasks
2025-10-17 06:19:18 [root@eks-cluster ~]# created 0 nodegroup(s) in cluster "my-cluster"
2025-10-17 06:19:18 [root@eks-cluster ~]# nodegroup "my-node-group" has 3 node(s)
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-30-47.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-89-94.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-95-179.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# waiting for at least 2 node(s) to become ready in "my-node-group"
2025-10-17 06:19:18 [root@eks-cluster ~]# nodegroup "my-node-group" has 3 node(s)
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-30-47.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-89-94.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# node "ip-172-31-95-179.ec2.internal" is ready
2025-10-17 06:19:18 [root@eks-cluster ~]# created 1 managed nodegroup(s) in cluster "my-cluster"
2025-10-17 06:19:18 [root@eks-cluster ~]# checking security group configuration for all nodegroups
2025-10-17 06:19:18 [root@eks-cluster ~]# all nodegroups have up-to-date cloudformation templates
[root@eks-cluster ~]# kubectl get nodes
NAME STATUS ROLES AGE VERSION
ip-172-31-30-47.ec2.internal Ready <none> 104s v1.32.6
ip-172-31-89-94.ec2.internal Ready <none> 97s v1.32.6
ip-172-31-95-179.ec2.internal Ready <none> 100s v1.32.6
[root@eks-cluster ~]# kubectl get pods
No resources found in default namespace.
[root@eks-cluster ~]# vim deployment.yaml
[root@eks-cluster ~]# vim service.yaml
[root@eks-cluster ~]#
[root@eks-cluster ~]# kubectl apply -f deployment.yaml
deployment.apps/regapp-deployment created
[root@eks-cluster ~]# kubectl apply -f service.yaml
service/regapp-service created
[root@eks-cluster ~]# kubectl get pods
NAME READY STATUS RESTARTS AGE
regapp-deployment-79cc89fd95-6644d 1/1 Running 0 32s
regapp-deployment-79cc89fd95-j8vcr 1/1 Running 0 32s
[root@eks-cluster ~]#
```

➔ Deployment File and service file for Autoscaling load balancer



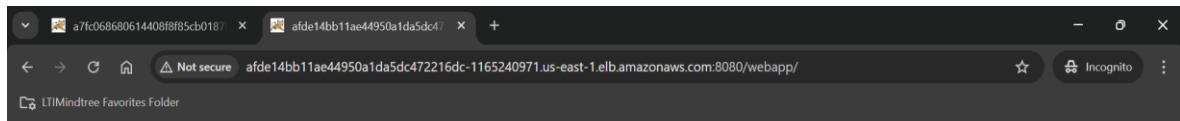
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: regapp-deployment
  labels:
    app: regapp

spec:
  replicas: 2
  selector:
    matchLabels:
      app: regapp

  template:
    metadata:
      labels:
        app: regapp
    spec:
      containers:
        - name: regapp
          image: 372687848939.dkr.ecr.us-east-1.amazonaws.com/tomcat-webapp:latest
          imagePullPolicy: Always
          ports:
            - containerPort: 8080
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 1
  ~
  ~
  ~
-- INSERT --
```

→ Finally Building the project

➔ The file is accessible on load balancer



I love learn devops teaching with sanjay sir loveeeee

Please fill in this form to create an account.

Enter Name	Enter Full Name
Enter mobile	Enter mobile number
Enter Email	Enter Email
Password	Enter Password
Repeat Password	Repeat Password

By creating an account you agree to our [Terms & Privacy](#).

[Register](#)

Already have an account? [Sign in](#).

Thank You

bye



➔ Making changes to the github file and checking if reflecting



Happy diwali!!!!I love learn devops teaching with sanjay sir loveeeee

Please fill in this form to create an account.

Enter Name	Enter Full Name
Enter mobile	Enter mobile number
Enter Email	Enter Email
Password	Enter Password
Repeat Password	Repeat Password

By creating an account you agree to our [Terms & Privacy](#).

[Register](#)

Already have an account? [Sign in](#).

Thank You

bye



Declarative Pipeline Script :

```
pipeline {

    agent any

    tools {
        maven 'maven'
    }

    environment {
        TOMCAT_USER = 'deployer'
        TOMCAT_PASS = 'deployer'
        TOMCAT_HOST = '3.92.53.87'
        TOMCAT_PORT = '8080'
    }

    stages {
        stage('Clone Repository') {
            steps {
                git branch: 'main', url: 'https://github.com/Ankit-10844395/brand-repo.git'
            }
        }

        stage('Build with Maven') {
            steps {
                sh 'mvn clean package -Dmaven.test.failure.ignore=true'
            }
        }

        stage('Deploy to Tomcat') {
    }
```

```

steps {
    sh """
        curl -u $TOMCAT_USER:$TOMCAT_PASS \
        --upload-file webapp/target/webapp.war \
        "http://$TOMCAT_HOST:$TOMCAT_PORT/manager/text/deploy?path=/sampleapp
        &update=true"
    """
}

stage('Deploy to Docker Host') {
    steps {
        sshPublisher(publishers: [
            sshPublisherDesc(
                configName: 'docker', // Must match SSH config name in Jenkins
                transfers: [
                    sshTransfer(
                        sourceFiles: '**/*',
                        removePrefix: '',
                        remoteDirectory: 'ultima',
                        execCommand: ""
                        cd ultima
                        aws ecr get-login-password --region us-east-1 | docker login --
                        username AWS --password-stdin 867098306986.dkr.ecr.us-east-1.amazonaws.com
                        docker build -t tomcat-webapp .
                        docker tag tomcat-webapp:latest 867098306986.dkr.ecr.us-east-
                        1.amazonaws.com/tomcat-webapp:latest
                        docker push 867098306986.dkr.ecr.us-east-
                        1.amazonaws.com/tomcat-webapp:latest
                    """
                ]
            )
        ])
    }
}

```

```

        )
    ]
)
D
}
}

stage('Deploy to EKS') {

steps {

    withCredentials([[$class: 'AmazonWebServicesCredentialsBinding',
credentialsId: 'awsconfig']]) {

        sh ''

            aws eks update-kubeconfig --region us-east-1 --name my-cluster
            kubectl delete -f /new/deployment.yaml
            kubectl apply -f /new/deployment.yaml
            kubectl apply -f /new/service.yaml
            kubectl rollout status deployment/regapp-deployment

        '''

    }
}

}
}

post {

    success {

        junit '**/target/surefire-reports/TEST-*.xml'
        archiveArtifacts artifacts: '**/target/*.war', fingerprint: true

    }
}

}

```

