

DevOps Project

Problem Statement:

Create an end-to-end CI/CD pipeline in AWS platform using Jenkins as the orchestration tool, GitHub as scm, maven as the build tool, deploy in a docker instance and create a docker image, store the docker image in ECR, Kubernetes deployment using ECR image. Build sample java web app usi

Approach:

Requirements:

- ✓ CI/CD pipeline System
- ✓ Git - local version control system.
- ✓ GitHub - As Distributed version control system.
- ✓ Jenkins – Orchestration
- ✓ Maven - As a Build Tool.
- ✓ docker -Container Engine
- ✓ Kubernetes - As Container Management Tool

1) Developer

Launch an instance for developer to access the github from it.

The screenshot shows the AWS Management Console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances. The left sidebar shows 'Services' and 'Search'. The main area displays various AMI options: Amazon Linux, macOS, Ubuntu, Windows, and Red Hat. A search bar is present above the AMI list. To the right, the 'Summary' section shows 'Number of instances' set to 1. Below it, the 'Software Image (AMI)' section shows 'Amazon Linux 2023 AMI' with the ID 'ami-08718895af4dfa033'. The 'Virtual server type (instance type)' is set to 't2.medium'. A large orange 'Launch instance' button is at the bottom right.

This screenshot continues from the previous one, showing the 'Instance type' section where 't2.medium' is selected. It also shows the 'Key pair (login)' section with 'dev' as the key pair name. The rest of the interface is identical to the first screenshot, including the summary and launch buttons.

Choose the instance and connect to that instance.

Perform ssh-keygen at developer side. That key will be used to connect with github server.

[ec2-user@ip-172-31-17-72:~]\$ sudo su -
[root@ip-172-31-17-72 ~]# hostnamectl set-hostname developer
[root@ip-172-31-17-72 ~]# bash
[root@developer ~]# ssh-keygen
Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id_rsa): Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:IAOqJ7GGGP0ZRLFGetE5TZWkuq7uEMX7GpD3j36n55M root@developer
The key's randomart image is:
+---[RSA 3072]---+
| .oo +.oo . |
| ...+ ... |
| o +ooo . |
|+oo +o+. |
|*=.o = S |
|oo+ = . |
| + . = . |
| . o B o. E |
| oBo=.o=. |
+---[SHA256]---+
[root@developer ~]# yum install git -y
Last metadata expiration check: 0:18:11 ago on Thu Sep 19 04:20:36 2024.
Dependencies resolved.
=====

Package	Architecture	Version	Repository	Size
---------	--------------	---------	------------	------

Page 2 of 2 Undo Redo English (India) Text Predictions: On Accessibility: Investigate

10:11 19-09-2024

As we using git as version control system, install git and initialize in the desired directory where you want to code/code is available.

The screenshot shows a terminal window titled "root@ip-172-31-17-72:/data". The terminal output is as follows:

```
Verifying : git-core-2.40.1-1.amzn2023.0.3.x86_64 2/8
Verifying : git-core-doc-2.40.1-1.amzn2023.0.3.noarch 3/8
Verifying : perl-Error-1:0.17029-5.amzn2023.0.2.noarch 4/8
Verifying : perl-File-Find-1.37-477.amzn2023.0.6.noarch 5/8
Verifying : perl-Git-2.40.1-1.amzn2023.0.3.noarch 6/8
Verifying : perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64 7/8
Verifying : perl-lib-0.65-477.amzn2023.0.6.x86_64 8/8

Installed:
git-2.40.1-1.amzn2023.0.3.x86_64
git-core-doc-2.40.1-1.amzn2023.0.3.noarch
perl-File-Find-1.37-477.amzn2023.0.6.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64

Complete!
[root@developer ~]# mkdir /data
[root@developer ~]# cd /data
[root@developer data]# git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /data/.git/
[root@developer data]#
```

The terminal window has a dark theme. The status bar at the bottom shows "Page 2 of 2", "0 words", "English (India)", "Text Predictions: On", "Accessibility: investigate", "Focus", "ENG IN", "10:09", "19-09-2024", and a battery icon.

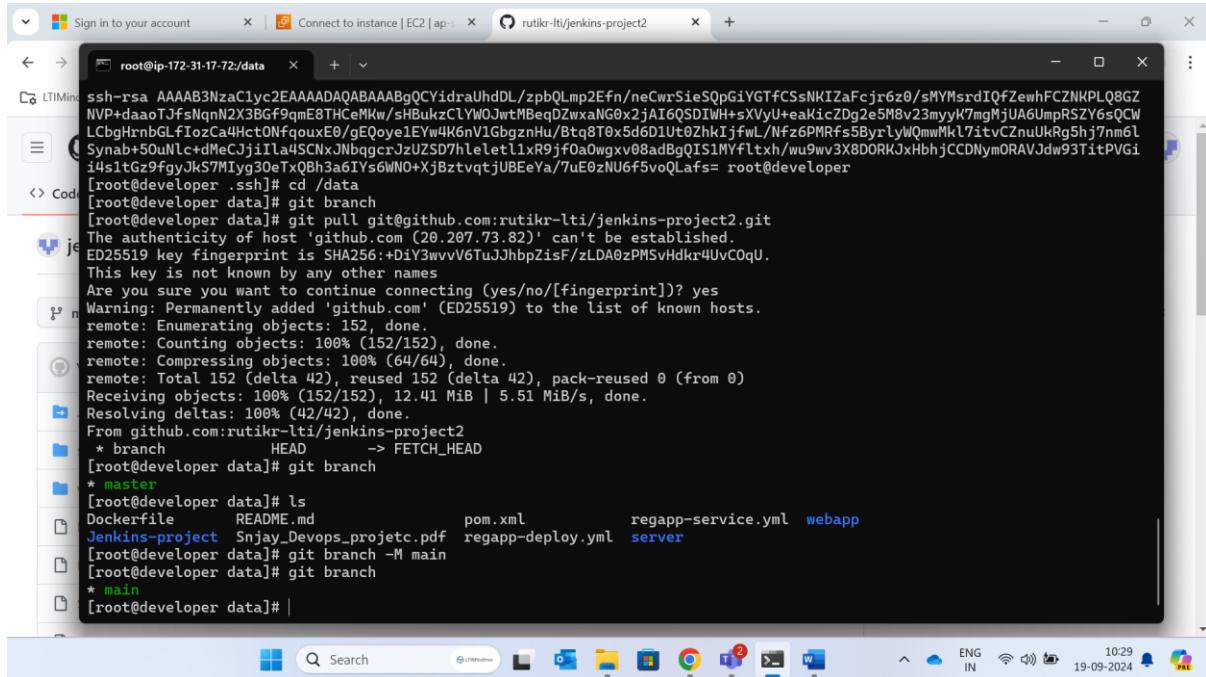
Copy public key of developer and add it to github. We have to add it to github ssh and gpg keys in github account settings.

The screenshot shows a Windows desktop environment with two main windows open:

- Terminal Window (root@ip-172-31-17-72:~/ssh):** Displays the command-line process of generating an SSH key pair. It shows the creation of a Git repository, navigating to the .ssh directory, listing authorized_keys, and displaying the contents of id_rsa.pub. The public key is copied to the clipboard.
- Web Browser (github.com/settings/keys):** Shows the GitHub account settings page for SSH keys. It lists two existing SSH keys, both of which have "Delete" buttons next to them. A "New SSH key" button is visible at the top right of the list.

The taskbar at the bottom of the screen shows various application icons, and the system tray indicates the date and time as 19-09-2024 at 10:26.

Now pull the code where it is available in our github repository. And make changes if required.



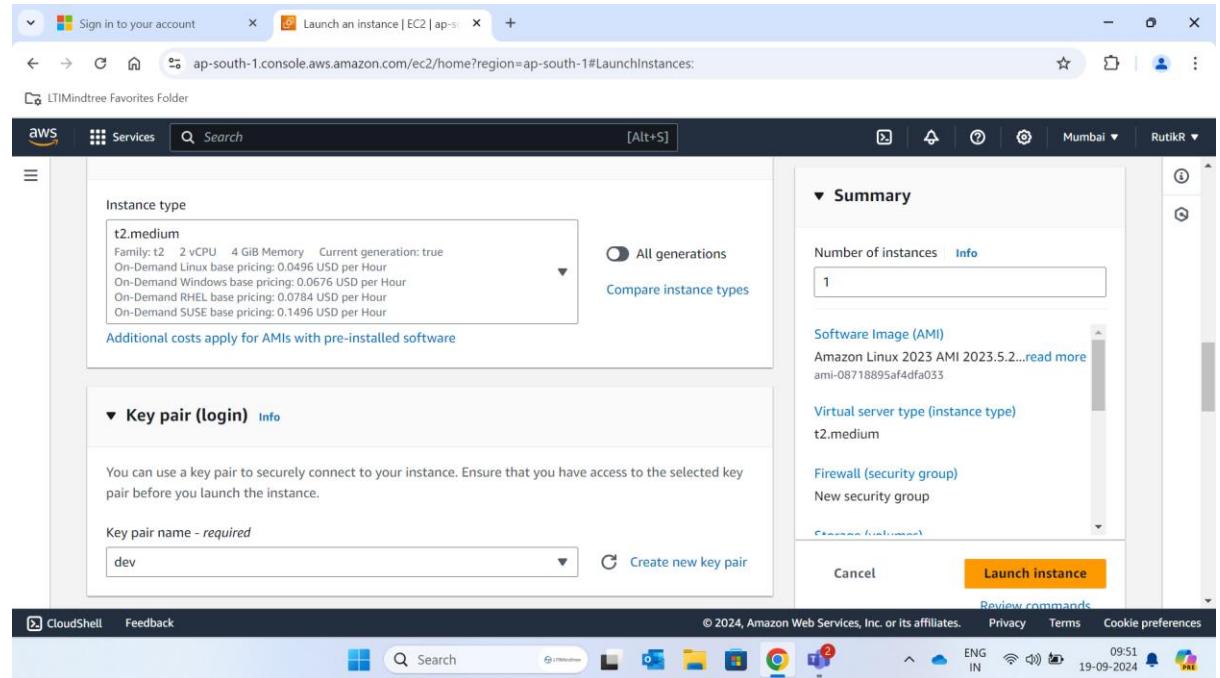
The screenshot shows a Windows terminal window titled "root@ip-172-31-17-72:/data". The terminal is displaying an SSH session to a host at ip-172-31-17-72. The user has run several commands:

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCYidraUhdDL/zpbQLmp2EfFn/neCwrsieSpqGiYGTfCSsNKIZaFcjr6z0/sMYMsrdIQfZewhFCZNKPLQ8GZ
NVP+daaoTJfsNqnN2X3BGF9qmE8THCeMKw/sHBukzClyW0JwrtMBeqDZwxnNG0x2jAI60SD1WH+sXVyu+eaKicZDg2e5M8v23myk7mgMjUA6UmprSZY6sQCW
LCbgHrnbgLfl0zCa4HctONfquoxE0/gEQoye1EW4k6nV1gbgznHu/Btq8T0x5d6D1Ut0ZhkiJfwL/Nfz6PMRfs5ByrlyWQmwMkl7itvCznuUkr5hj7nm6l
Synabv5OuNlc+dMeCjjiIla4SCNxJNbqgcrJzUZSD7hleletl1xR9jf0aQwgxv08adBgQIS1MYfltxh/wu9wv3X8DORKJxHbhjCCDNymORAVJdw93T1tPVG
i4s1tg29fgyJk57Mlyg30eTxQbh3a61Ys6WNO+XjbztvqtjUBEEeYa/TuE0zNU6f5voQlafs= root@developer
[root@developer ssh]# cd /data
[root@developer data]# git branch
[root@developer data]# git pull git@github.com:rutikr-lti/jenkins-project2.git
The authenticity of host 'github.com (20.207.73.82)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJhbhpZisF/zLDA0zPMsvHdkr4UvCoqu.
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.
remote: Enumerating objects: 152, done.
remote: Counting objects: 100% (152/152), done.
remote: Compressing objects: 100% (64/64), done.
remote: Total 152 (delta 42), reused 152 (delta 42), pack-reused 0 (from 0)
Receiving objects: 100% (152/152), 12.41 MiB | 5.51 MiB/s, done.
Resolving deltas: 100% (42/42), done.
From github.com:rutikr-lti/jenkins-project2
 * branch           HEAD      -> FETCH_HEAD
 [root@developer data]# git branch
* master
 [root@developer data]# ls
 Dockerfile      README.md      pom.xml      regapp-service.yml  webapp
 Jenkins-project  Snjay_Devops_projectc.pdf  regapp-deploy.yml  server
 [root@developer data]# git branch -M main
 [root@developer data]# git branch
* main
 [root@developer data]# |
```

The terminal window is part of a larger desktop environment, with icons for File Explorer, Task View, and Start menu visible at the bottom. The status bar at the bottom right shows the date (19-09-2024) and time (10:29).

2) Jenkins

Connect instance where we need to install Jenkins and choose t2.medium for Jenkins server. As it require minimum 2 core cpus and 4gb ram to work.



Install Jenkins.

```
root@ip-172-31-17-72:/data ~ % root@ip-172-31-40-234:~ % 
=====
Install 1 Package
=====
Total download size: 89 M
Installed size: 89 M
Downloading Packages:
jenkins-2.462.2-1.1.noarch.rpm
=====
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Running scriptlet: jenkins-2.462.2-1.1.noarch 1/1
  Installing  : jenkins-2.462.2-1.1.noarch 1/1
  Running scriptlet: jenkins-2.462.2-1.1.noarch 1/1
  Verifying   : jenkins-2.462.2-1.1.noarch 1/1
=====
Installed:
  jenkins-2.462.2-1.1.noarch
=====
Complete!
[root@jenkins ~]# sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
[root@jenkins ~]# sudo systemctl start jenkins
[root@jenkins ~]#
```

Install git on the Jenkins server. As we connect Jenkins to docker so git must be on the both systems.

```
root@developer:~# Running transaction test
Transaction test succeeded.
root@ip-172-31-7-173:~# Running transaction
Preparing : 1/1
Installing : git-core-2.40.1-1.amzn2023.0.3.x86_64 1/8
Installing : git-core-doc-2.40.1-1.amzn2023.0.3.noarch 2/8
Installing : perl-lib-0.65-477.amzn2023.0.6.x86_64 3/8
Installing : perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64 4/8
Installing : perl-File-Find-1.37-477.amzn2023.0.6.noarch 5/8
Installing : perl-Error-1.0.17029-5.amzn2023.0.2.noarch 6/8
Installing : perl-Git-2.40.1-1.amzn2023.0.3.noarch 7/8
Installing : git-2.40.1-1.amzn2023.0.3.x86_64 8/8
Running scriptlet: git-2.40.1-1.amzn2023.0.3.x86_64
Verifying : git-2.40.1-1.amzn2023.0.3.x86_64 8/8
Verifying : git-core-2.40.1-1.amzn2023.0.3.x86_64 1/8
Verifying : perl-Git-2.40.1-1.amzn2023.0.3.noarch 2/8
Verifying : perl-Error-1.0.17029-5.amzn2023.0.2.noarch 3/8
Verifying : perl-File-Find-1.37-477.amzn2023.0.6.noarch 4/8
Verifying : perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64 5/8
Verifying : perl-lib-0.65-477.amzn2023.0.6.x86_64 6/8
Verifying : perl-Git-2.40.1-1.amzn2023.0.3.noarch 7/8
Verifying : git-2.40.1-1.amzn2023.0.3.x86_64 8/8
Installed:
git-2.40.1-1.amzn2023.0.3.x86_64 git-core-2.40.1-1.amzn2023.0.3.x86_64
git-core-doc-2.40.1-1.amzn2023.0.3.noarch perl-Error-1.0.17029-5.amzn2023.0.2.noarch
perl-File-Find-1.37-477.amzn2023.0.6.noarch perl-Git-2.40.1-1.amzn2023.0.3.noarch
perl-TermReadKey-2.38-9.amzn2023.0.2.x86_64 perl-lib-0.65-477.amzn2023.0.6.x86_64
Complete!
[root@jenkins ~]#
```

Now start Jenkins. Also after starting the Jenkins it will give the path where password is stored. So go to Jenkins systems and copy that password by hitting that command and paste it on the Jenkins server.

```
root@developer:~# Install 1 Package
=====
Total download size: 89 M
Installed size: 89 M
Downloading Packages:
jenkins-2.462.2-1.1.noarch.rpm
13 MB/s | 89 MB 00:06
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/1
Running scriptlet: jenkins-2.462.2-1.1.noarch 1/1
Installing : jenkins-2.462.2-1.1.noarch 1/1
Running scriptlet: jenkins-2.462.2-1.1.noarch 1/1
Verifying : jenkins-2.462.2-1.1.noarch 1/1
Installed:
jenkins-2.462.2-1.1.noarch
Complete!
[root@jenkins ~]# sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
[root@jenkins ~]# sudo systemctl start jenkins
[root@jenkins ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
70a4d5eeac79498b985ec785bbe936de
[root@jenkins ~]#
```

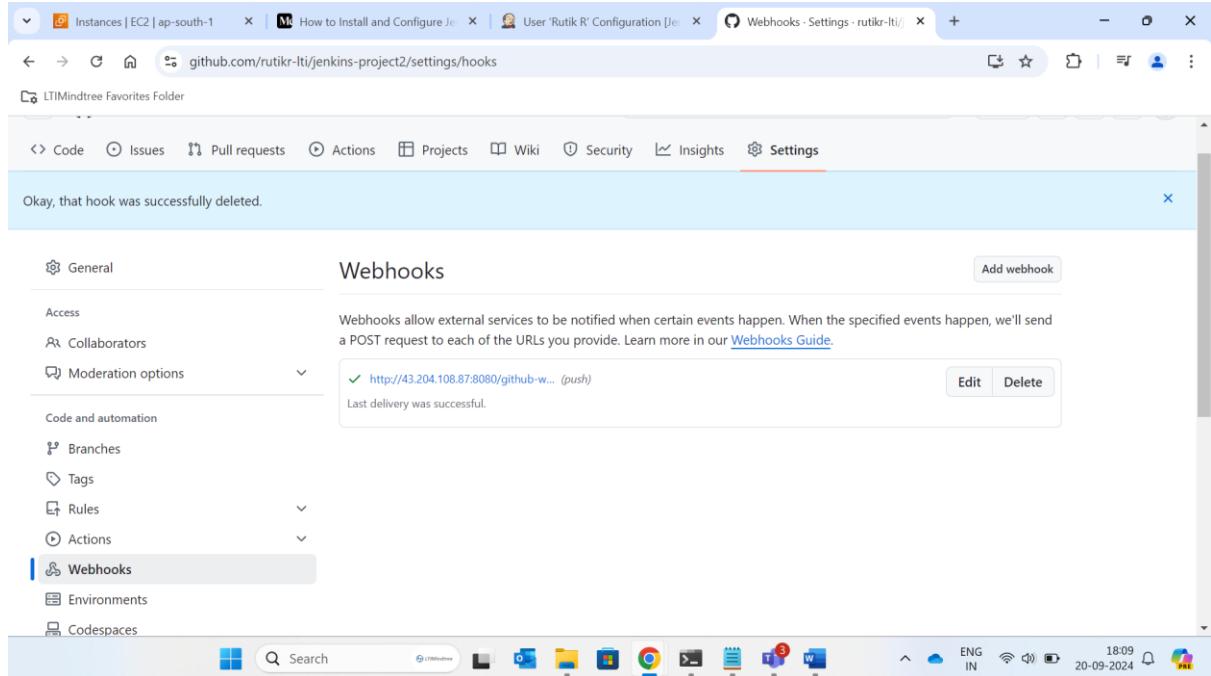
Paste copied password here.

The screenshot shows a browser window titled 'Sign in [Jenkins]' with the URL '43.204.108.87:8080/login?from=%2F'. The page is titled 'Getting Started' and features a large heading 'Unlock Jenkins'. It instructs the user to copy the password from a log file located at '/var/lib/jenkins/secrets/initialAdminPassword'. A text input field is provided for pasting the password, and a 'Continue' button is at the bottom right.

The screenshot shows a browser window titled 'Setup Wizard [Jenkins]' with the URL '43.204.108.87:8080'. The page is titled 'Getting Started' and displays a grid of Jenkins plugins. The grid has two columns: 'Folders' and 'Formatter'. The 'Folders' column includes Timestamper, Pipeline, Git, and LDAP. The 'Formatter' column includes Workspace Cleanup, GitHub Branch Source, SSH Build Agents, Email Extension, Ant, Pipeline: GitHub Groovy Libraries, Matrix Authorization Strategy, Mailer, Gradle, Pipeline Graph View, PAM Authentication, and Dark Theme. A note at the bottom right states '** - required dependency'. The Jenkins version '2.462.2' is mentioned at the bottom.

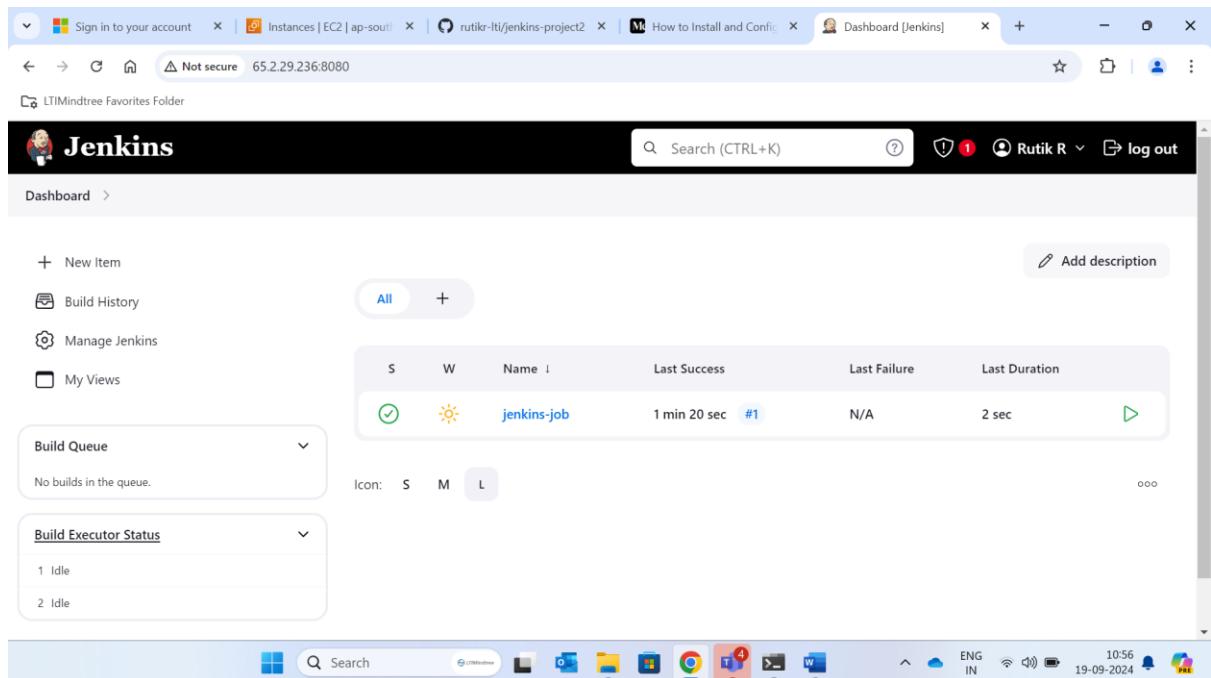
To connect Jenkins with github repository, we need to add webhook in github. So go to Jenkins and generate token , copy that token and paste it onto the github webhook.

Add the Jenkins ip/github-webhook/ in the url and at token paste copied token from Jenkins server.



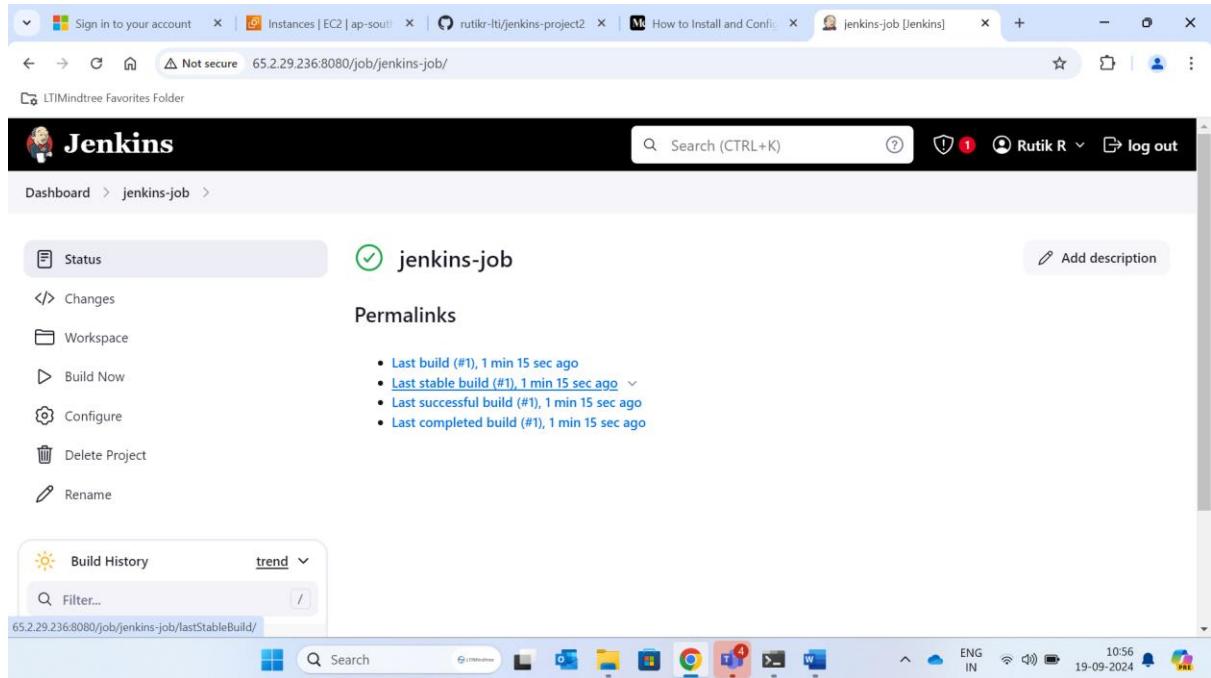
The screenshot shows the GitHub Settings page for a repository named 'jenkins-project2'. The 'Webhooks' section is active. A message at the top says 'Okay, that hook was successfully deleted.' Below this, there is a list of webhooks with one entry: 'http://43.204.108.87:8080/github-w... (push)'. This entry has a green checkmark and the status 'Last delivery was successful.'. There are 'Edit' and 'Delete' buttons next to the URL. On the left sidebar, 'Webhooks' is highlighted under the 'Code and automation' section. The bottom of the screen shows a Windows taskbar with various icons and a date/time indicator.

Create freestyle project and check if it is working.

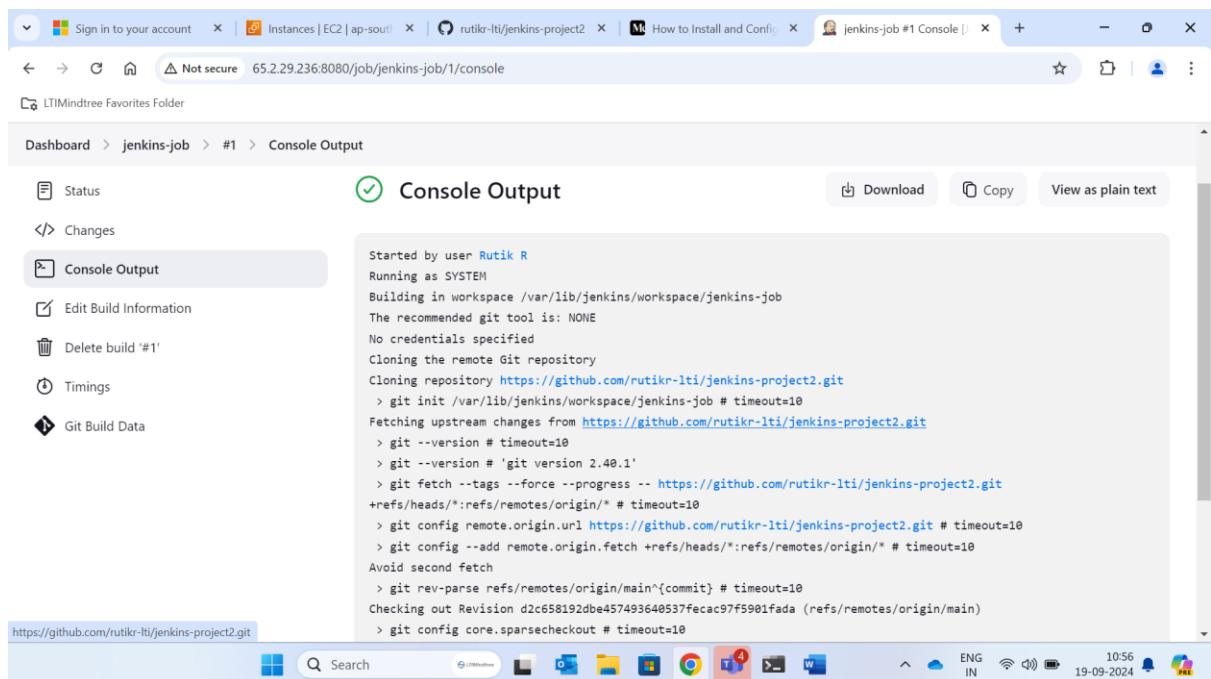


The screenshot shows the Jenkins Dashboard. At the top, there is a search bar and a user profile for 'Rutik R'. Below the dashboard header, there are several links: 'Dashboard', 'Build History', 'Manage Jenkins', and 'My Views'. A 'Build Queue' section indicates 'No builds in the queue.' A 'Build Executor Status' section shows '1 Idle' and '2 Idle'. The main area displays a table for a build named 'jenkins-job'. The table columns are 'S' (Status), 'W' (Waiting), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. The row for 'jenkins-job' shows a green circle icon, a yellow sun icon, the name 'jenkins-job', '1 min 20 sec', 'N/A', and '2 sec'. There is also a 'More' button (three dots) next to the duration. The bottom of the screen shows a Windows taskbar with various icons and a date/time indicator.

We have successfully integrated Jenkins with Github as job run successfully.



The screenshot shows the Jenkins job configuration page for 'jenkins-job'. The left sidebar lists options like Status, Changes, Workspace, Build Now, Configure, Delete Project, and Rename. The main area displays the 'Permalinks' section with four recent builds: Last build (#1), Last stable build (#1), Last successful build (#1), and Last completed build (#1). Below this is the 'Build History' section, which shows a single entry for the last stable build. The status bar at the bottom indicates the build was run from 'https://github.com/rutikr-lti/jenkins-project2.git'.

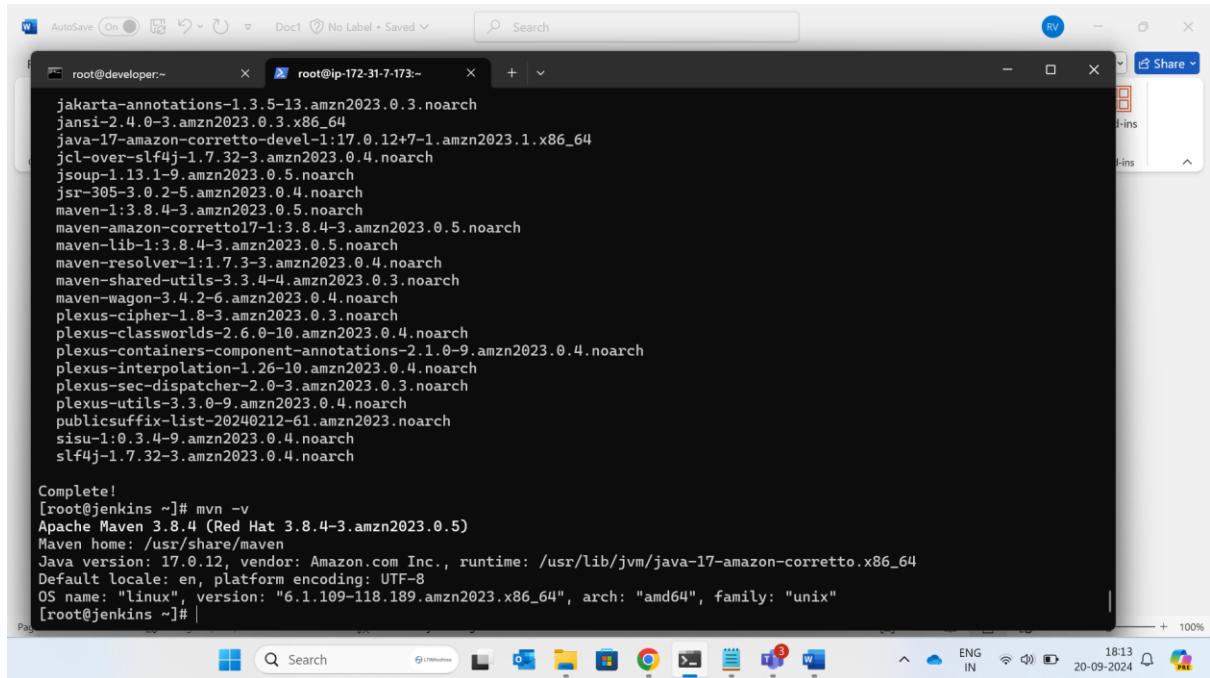


The screenshot shows the Jenkins console output for build #1. The left sidebar shows the 'Console Output' option selected. The main area displays the command-line log of the build process, starting with the user 'Rutik R' and detailing the cloning of the repository from GitHub. The status bar at the bottom indicates the build was run from 'https://github.com/rutikr-lti/jenkins-project2.git'.

Maven Configuration:

On the same Jenkins machine install maven using yum install maven command.

Also install java package.



```
AutoSave On 🔍 ⌂ ⌂ Doc1 No Label • Saved Search Share
root@developer:~ x root@ip-172-31-7-173:~ x + 
jakarta-annotations-1.3.5-13.amzn2023.0.3.noarch
jansi-2.4.0-3.amzn2023.0.3.x86_64
java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
jcl-over-slf4j-1.7.32-3.amzn2023.0.4.noarch
jsoup-1.13.1-9.amzn2023.0.5.noarch
jsr-305-3.0.2-5.amzn2023.0.4.noarch
maven-1:3.8.4-3.amzn2023.0.5.noarch
maven-amazon-corretto17-1:3.8.4-3.amzn2023.0.5.noarch
maven-lib-1:3.8.4-3.amzn2023.0.5.noarch
maven-resolver-1:1.7.3-3.amzn2023.0.4.noarch
maven-shared-utils-3.3.4-4.amzn2023.0.3.noarch
maven-wagon-3.4.2-6.amzn2023.0.4.noarch
plexus-cipher-1.8-3.amzn2023.0.3.noarch
plexus-classworlds-2.6.0-10.amzn2023.0.4.noarch
plexus-containers-component-annotations-2.1.0-9.amzn2023.0.4.noarch
plexus-interpolation-1.26-10.amzn2023.0.4.noarch
plexus-sec-dispatcher-2.0-3.amzn2023.0.3.noarch
plexus-utils-3.3.0-9.amzn2023.0.4.noarch
publicsuffix-list-20240212-61.amzn2023.noarch
sisu-1:0.3.4-9.amzn2023.0.4.noarch
slf4j-1.7.32-3.amzn2023.0.4.noarch
Complete!
[root@jenkins ~]# mvn -v
Apache Maven 3.8.4 (Red Hat 3.8.4-3.amzn2023.0.5)
Maven home: /usr/share/maven
Java version: 17.0.12, vendor: Amazon.com Inc., runtime: /usr/lib/jvm/java-17-amazon-corretto.x86_64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.109-118.189.amzn2023.x86_64", arch: "amd64", family: "unix"
[root@jenkins ~]# |
```

The screenshot shows a terminal window with two tabs open. The first tab is 'root@developer:~' and the second is 'root@ip-172-31-7-173:~'. The terminal displays the output of a Maven installation command, showing various dependency packages like jakarta-annotations, jansi, java-17-amazon-corretto-devel, and slf4j. It also shows the completion of the Maven configuration with the command 'mvn -v', which prints the Maven version (3.8.4), Java version (17.0.12), and the operating system details (Linux, 6.1.109-118.189.amzn2023.x86_64). The terminal is running on a Windows desktop environment, as indicated by the taskbar at the bottom.

Copy java and maven path and paste it into tools. Add java and maven tools in the tools settings. Also go to plugins and add the maven integration to Jenkins and install the same. Then restart the Jenkins.

The screenshot shows the Jenkins 'Tools' configuration page. In the top navigation bar, there are tabs for 'Instances | EC2 | ap-south-1', 'How to Install and Configure Jenkins', 'Tools [Jenkins]', and 'Webhooks - Settings - rutikr-lt/'. The current tab is 'Tools [Jenkins]'. Below the tabs, the URL is 43.204.108.87:8080/manage/configureTools/. The main content area is titled 'Manage Jenkins > Tools'. A sub-section titled 'Add JDK' is displayed, containing fields for 'Name' (set to 'java') and 'JAVA_HOME' (set to '/usr/lib/jvm/java-17-amazon-corretto.x86_64'). There is also an unchecked checkbox for 'Install automatically'. At the bottom of this section are 'Save' and 'Apply' buttons. The status bar at the bottom right shows the time as 18:14 and the date as 20-09-2024.

This screenshot shows the 'Add Maven' section of the Jenkins 'Tools' configuration page. It has a similar structure to the 'Add JDK' section, with fields for 'Name' (set to 'Maven') and 'MAVEN_HOME' (set to '/usr/share/maven'). An unchecked checkbox for 'Install automatically' is present. Below these fields are 'Save' and 'Apply' buttons. The status bar at the bottom right shows the time as 18:14 and the date as 20-09-2024.

Also add github plugin.

The screenshot shows the Jenkins plugin manager interface. The search bar at the top has 'github' typed into it. Below the search bar, there is a table of installed GitHub-related plugins:

Name	Enabled
GitHub API Plugin	Enabled
GitHub Branch Source Plugin	Disabled
GitHub plugin	Enabled
Pipeline: GitHub Groovy Libraries	Enabled

The left sidebar has a 'Installed plugins' tab selected, along with other options like 'Updates', 'Available plugins', 'Advanced settings', and 'Download progress'.

Also maven integration plugin

The screenshot shows the Jenkins plugin manager interface. The search bar at the top has 'maven' typed into it. Below the search bar, there is a table of available Maven-related plugins:

Install	Name	Released
<input checked="" type="checkbox"/>	Maven Integration 3.23	1 yr 1 mo ago
<input type="checkbox"/>	Config File Provider	11 days ago
<input type="checkbox"/>	Jira	3.13

The left sidebar has an 'Available plugins' tab selected, along with other options like 'Updates', 'Installed plugins', 'Advanced settings', and 'Download progress'.

Screenshot of a web browser showing the Jenkins plugin manager page. The URL is 43.204.108.87:8080/manage/pluginManager/updates/. The page displays a list of installed Jenkins plugins with their status as 'Success' indicated by green checkmarks.

Plugin	Status
Email Extension	Success
Mailer	Success
Theme Manager	Success
Dark Theme	Success
Loading plugin extensions	Success
Javadoc	Success
JSch dependency	Success
Maven Integration	Success
Loading plugin extensions	Success

Below the table, there are two links:

- [Go back to the top page](#) (you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running

Screenshot of a web browser showing the Jenkins restart progress page. The URL is 43.204.108.87:8080/manage/pluginManager/updates/. The page features a large Jenkins logo at the top and the text "Please wait while Jenkins is restarting ...". Below this, it says "Your browser will reload automatically when Jenkins is ready." A green button labeled "Safe Restart" with the subtext "Builds on agents can usually continue." is present.



Create maven project to check if maven is working successfully.

The screenshot shows the Jenkins 'New Item' creation interface. At the top, there are four tabs: 'Instances | EC2 | ap-south-1', 'How to Install and Configure Jenkins', 'New Item [Jenkins]', and 'Webhooks - Settings - rutikr-lti/'. Below the tabs, the URL is 43.204.108.87:8080/view/all/newJob. The main area is titled 'New Item' and has a sub-section 'Enter an item name' with the value 'maven-job'. Under 'Select an item type', three options are listed: 'Freestyle project', 'Maven project' (which is selected), and 'Pipeline'. Each option has a brief description. At the bottom right of the main window is an 'OK' button. The taskbar at the bottom of the screen shows various application icons and the date/time 20-09-2024 18:18.

Add git repo url to the job and then build the job.

The screenshot shows the Jenkins 'maven-job Configuration' page. The top navigation bar includes tabs for 'Instances | EC2 | ap-south-1', 'How to Install and Configure Jenkins', 'maven-job Config [Jenkins]', and 'rutikr-lti/jenkins-project2'. The URL is 43.204.108.87:8080/job/maven-job/configure. The main content area is titled 'Configure' and shows the 'Git' configuration section. On the left, a sidebar lists configuration categories: General, Source Code Management (which is selected and highlighted in grey), Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. The 'Source Code Management' section contains fields for 'Repository URL' (set to 'https://github.com/rutikr-lti/jenkins-project2.git') and 'Credentials' (set to '- none -'). There is also an 'Advanced' dropdown and 'Save' and 'Apply' buttons at the bottom. The taskbar at the bottom of the screen shows various application icons and the date/time 20-09-2024 18:19.

The screenshot shows the Jenkins job configuration interface. On the left, a sidebar lists various configuration sections: General, Source Code Management, Build Triggers, Build Environment, Pre Steps (which is currently selected and highlighted in grey), Post Steps, Build Settings, and Post-build Actions. The main panel is titled 'Configure' and contains two main sections: 'Build' and 'Post Steps'. Under 'Build', there are fields for 'Root POM' (set to 'pom.xml') and 'Goals and options'. Under 'Post Steps', there is a dropdown menu set to 'Advanced'. At the bottom of the configuration panel are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins job console output page. The top navigation bar includes links for Instances | EC2 | ap-south-1, How to Install and Configure Jenkins, maven-job #1 Console [Jenkins], and rutikr-lti/jenkins-project2. The main content area is titled 'Console Output' and displays the build log for job #1. The log starts with 'Started by user Rutik R' and details the git cloning process from a GitHub repository. The log ends with a note about avoiding a second fetch. On the left, a sidebar provides links to Status, Changes, Console Output (which is currently selected and highlighted in grey), Edit Build Information, Delete build '#1', Timings, Git Build Data, Redeploy Artifacts, Test Result, and See Fingerprints. On the right, there are buttons for Download, Copy, and View as plain text. The system tray at the bottom shows standard icons like battery level, signal strength, and system status.

Maven built the project successfully and created the webapp.war file available in workspace/webapp/target/webapp.war

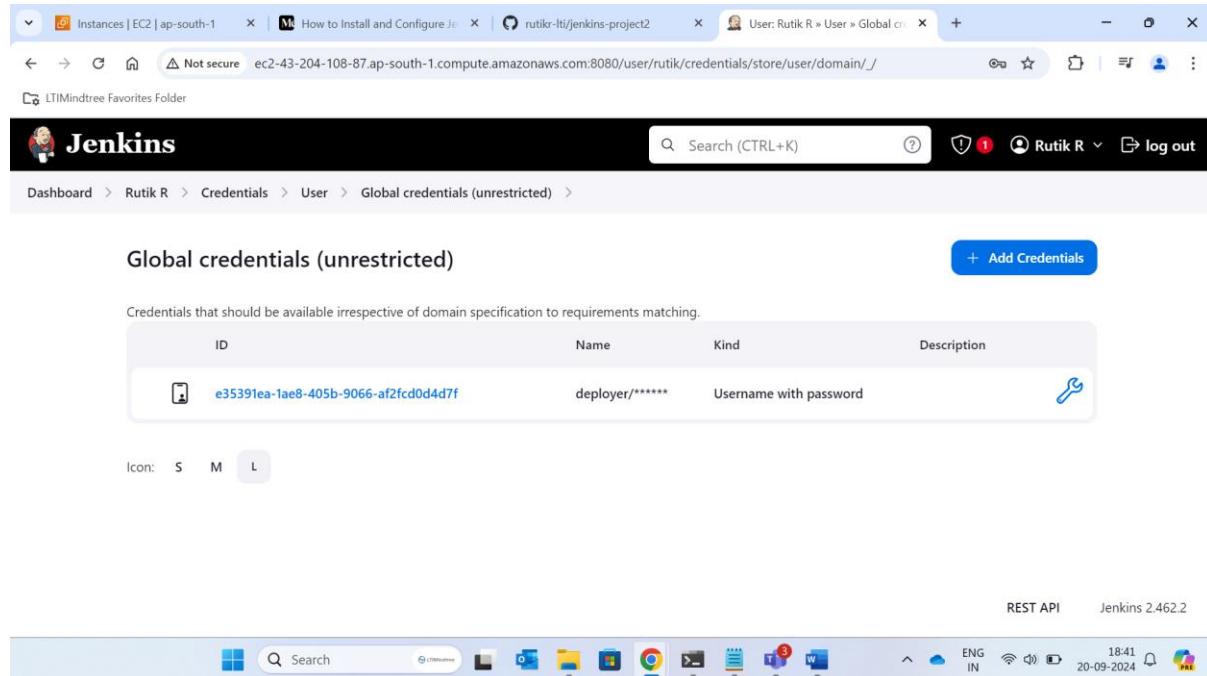
The screenshot shows a browser window with multiple tabs open. The active tab is 'maven-job [Jenkins]' at the URL '43.204.108.87:8080/job/maven-job/'. The page title is 'Maven project maven-job'. On the left, there's a sidebar with links like 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Maven project', 'Modules', 'GitHub Hook Log', and 'Rename'. The main content area shows a 'Latest Test Result (no failures)' section with a green icon and a 'Test Result Trend' chart. The chart has two points: one green dot labeled 'Passed' and one red dot labeled 'Failed'. Below the chart is a list of recent builds:

- Last build (#1), 32 sec ago
- Last stable build (#1), 32 sec ago
- Last successful build (#1), 32 sec ago
- Last completed build (#1), 32 sec ago

The browser's address bar shows the URL '43.204.108.87:8080/job/maven-job/'. The system tray at the bottom right indicates the date as '20-09-2024' and the time as '18:20'. The taskbar at the bottom shows various application icons.

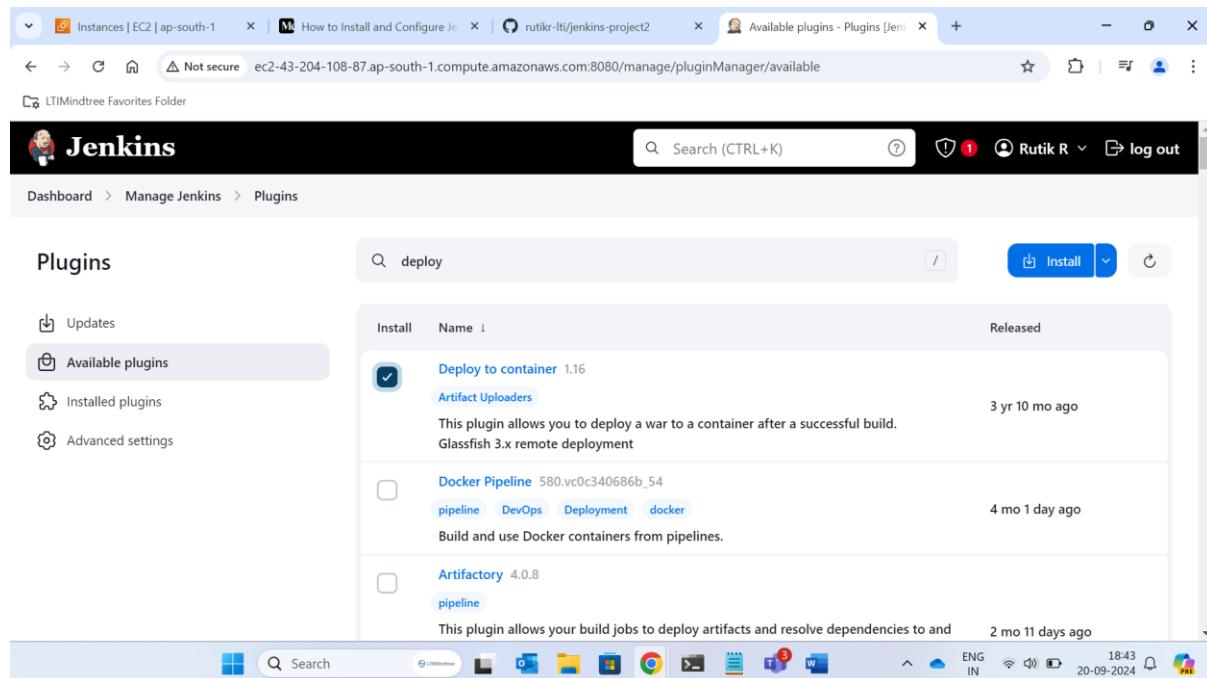
3) Tomcat

Create an instance and connect to it. Go to Jenkins and add global credentials to the Jenkins account . the credentials should be same as tomcat configuration.



The screenshot shows a browser window with multiple tabs open. The active tab is 'Global credentials (unrestricted)' under 'User' in the Jenkins 'Credentials' section. The page displays a table of credentials with one entry: ID 'e35391ea-1ae8-405b-9066-af2fc0d4d7f', Name 'deployer/*****', Kind 'Username with password', and a wrench icon for editing. Above the table, a note says 'Credentials that should be available irrespective of domain specification to requirements matching.' The browser's address bar shows the URL 'ec2-43-204-108-87.ap-south-1.compute.amazonaws.com:8080/user/rutik/credentials/store/user/domain/_/'. The top right of the browser shows Jenkins version 'Jenkins 2.462.2' and a timestamp '20-09-2024 18:41'.

Add another plugin deploy to container and restart Jenkins.



The screenshot shows a browser window with the 'Available plugins' tab selected in the Jenkins 'Plugin Manager' section. A search bar at the top contains the text 'deploy'. The results table lists three plugins: 'Deploy to container' (version 1.16), 'Docker Pipeline' (version 580.vc0c340686b_54), and 'Artifactory' (version 4.0.8). The 'Deploy to container' plugin is checked for installation. The browser's address bar shows the URL 'ec2-43-204-108-87.ap-south-1.compute.amazonaws.com:8080/manage/pluginManager/available'. The top right of the browser shows Jenkins version 'Jenkins 2.462.2' and a timestamp '20-09-2024 18:43'.

The screenshot shows the Jenkins 'Manage Jenkins > Plugins' section. On the left, there's a sidebar with links like 'Updates', 'Available plugins', 'Installed plugins', 'Advanced settings', and 'Download progress'. The 'Download progress' link is highlighted with a grey background. The main content area is titled 'Download progress' and includes a 'Preparation' section with three items: 'Checking internet connectivity' (Success), 'Checking update center connectivity' (Success), and 'Success'. Below this is a table with several rows: 'JavaMail API' (Success), 'SSH server' (Pending), 'Deploy to container' (Pending), 'Loading plugin extensions' (Pending), and 'Restarting Jenkins' (Pending). At the bottom, there are two buttons: 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.

Now go to item, create job with maven project.

The screenshot shows the Jenkins 'Dashboard > tomcat > Configuration' page. On the left, there's a sidebar with 'General', 'Source Code Management', 'Build Triggers', 'Build Environment', 'Pre Steps', 'Build', 'Post Steps', and 'Build Settings'. The 'General' link is highlighted with a grey background. The main content area is titled 'General' and includes a 'Description' field with a large empty text area. Below it are two checkboxes: 'Discard old builds' (unchecked) and 'GitHub project' (unchecked). At the bottom are 'Save' and 'Apply' buttons. To the right of the 'General' title is a 'Enabled' toggle switch which is checked. The status bar at the bottom shows the date and time as 20-09-2024 18:47.

Connect to instance and download tomcat and configure it.

Sign in to your account x Connect to instance | EC... x rutikr-lt/jenkins-project2 x How to Install and Config x Dashboard [Jenkins] x +

← → Command Prompt x Windows PowerShell x root@ip-172-31-18-139:~ x + v

LTIMini AWS

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

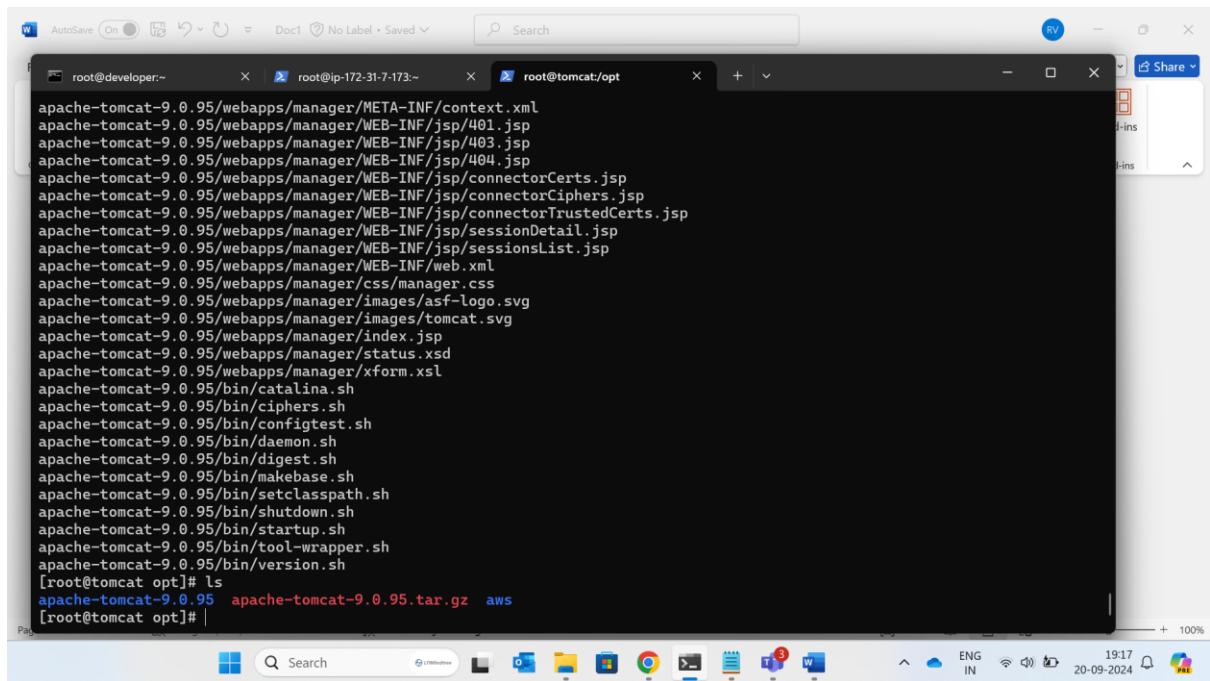
PS C:\Users\10748011> cd .\Downloads
PS C:\Users\10748011\Downloads> ssh -i "dev.pem" ec2-user@ec2-3-110-58-143.ap-south-1.compute.amazonaws.com
The authenticity of host 'ec2-3-110-58-143.ap-south-1.compute.amazonaws.com (3.110.58.143)' can't be established.
ED25519 key fingerprint is SHA256:wCRAZeogCSqY+JG0yU58y8ijelOa0uoMxjRT1wWhU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-110-58-143.ap-south-1.compute.amazonaws.com' (ED25519) to the list of known hosts.

`_ ##### Amazon Linux 2023
~~ \#####
~~ \|##|
~~ \|#/ __ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~' '-'>
~~ /
~~ .-' /
~~ / /
~/m', /
[ec2-user@ip-172-31-18-139 ~]\$ sudo su -
[root@ip-172-31-18-139 ~]# hostnamectl set-hostname tomcat
[root@ip-172-31-18-139 ~]# bash
[root@tomcat ~]# |

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 11:57 19-09-2024

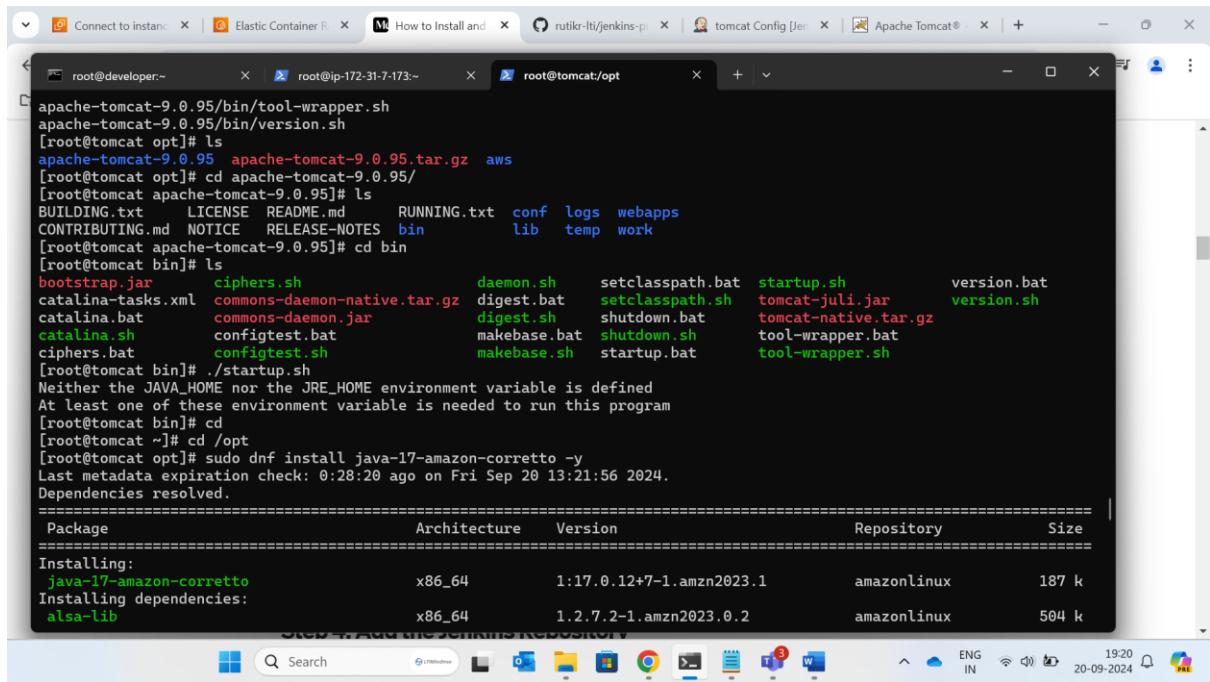
Downlodaing tomcat and configuring it.

Downloaded tar inside /opt folder And extracted using tar -xvf command.



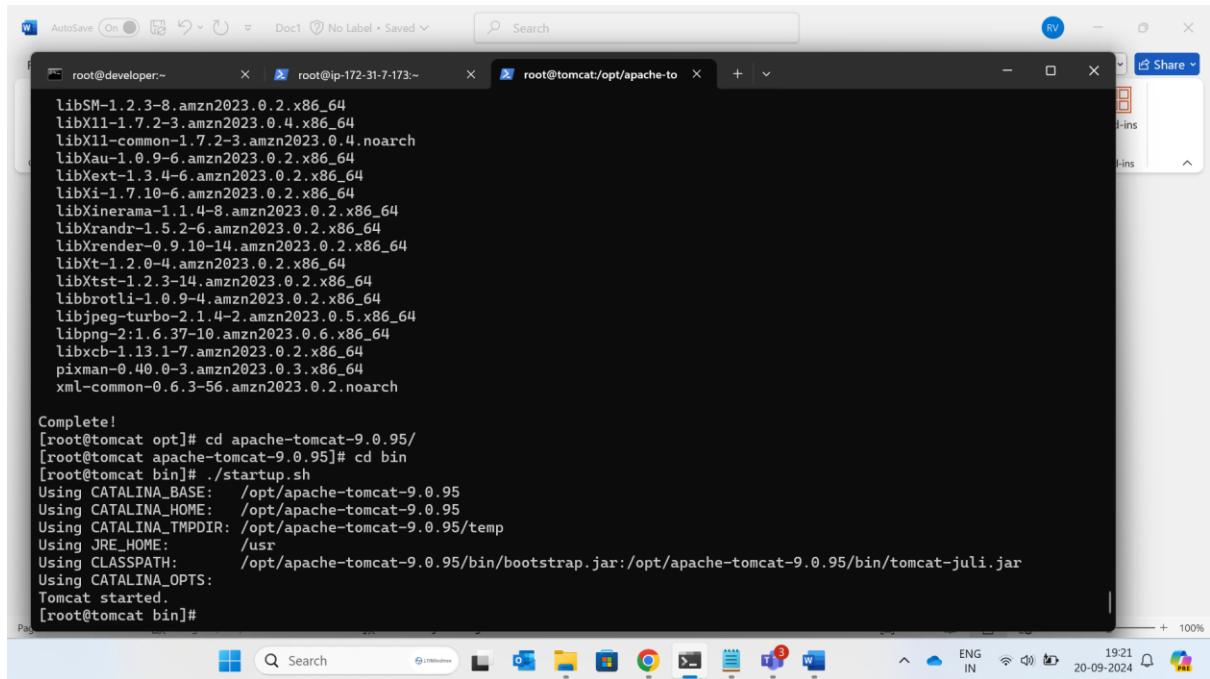
```
root@developer:~          x  root@ip-172-31-7-173:~          x  root@tomcat:/opt          x + v
apache-tomcat-9.0.95/webapps/META-INF/context.xml
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/401.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/403.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/404.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/connectorCerts.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/connectorCiphers.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/connectorTrustedCerts.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/sessionDetail.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/jsp/sessionsList.jsp
apache-tomcat-9.0.95/webapps/manager/WEB-INF/web.xml
apache-tomcat-9.0.95/webapps/manager/css/manager.css
apache-tomcat-9.0.95/webapps/manager/images/asf-logo.svg
apache-tomcat-9.0.95/webapps/manager/images/tomcat.svg
apache-tomcat-9.0.95/webapps/manager/index.jsp
apache-tomcat-9.0.95/webapps/manager/status.xsd
apache-tomcat-9.0.95/webapps/manager/xform.xsl
apache-tomcat-9.0.95/bin/catalina.sh
apache-tomcat-9.0.95/bin/ciphers.sh
apache-tomcat-9.0.95/bin/configtest.sh
apache-tomcat-9.0.95/bin/daemon.sh
apache-tomcat-9.0.95/bin/digest.sh
apache-tomcat-9.0.95/bin/makebase.sh
apache-tomcat-9.0.95/bin/setclasspath.sh
apache-tomcat-9.0.95/bin/shutdown.sh
apache-tomcat-9.0.95/bin/startup.sh
apache-tomcat-9.0.95/bin/tool-wrapper.sh
apache-tomcat-9.0.95/bin/version.sh
[root@tomcat opt]# ls
apache-tomcat-9.0.95 apache-tomcat-9.0.95.tar.gz aws
[root@tomcat opt]# |
```

Installed java which is requirement for tomcat



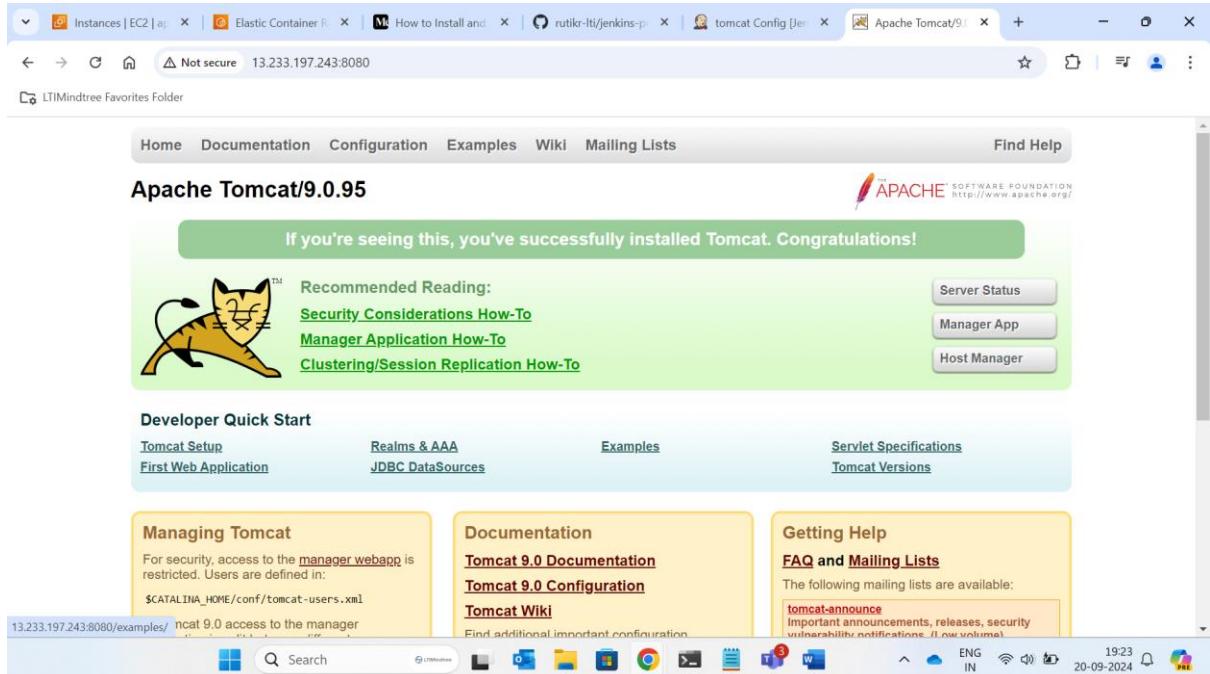
```
root@developer:~          x  Connect to instance x  Elastic Container R x  How to Install and x  rutikr-lti/jenkins-p x  tomcat Config [jenkins] x  Apache Tomcat® x + v
apache-tomcat-9.0.95/bin/tool-wrapper.sh
apache-tomcat-9.0.95/bin/version.sh
[root@tomcat opt]# ls
apache-tomcat-9.0.95 apache-tomcat-9.0.95.tar.gz aws
[root@tomcat opt]# cd apache-tomcat-9.0.95/
[root@tomcat apache-tomcat-9.0.95]# ls
BUILDING.txt LICENSE README.md RUNNING.txt conf logs webapps
CONTRIBUTING.md NOTICE RELEASE-NOTES bin lib temp work
[root@tomcat apache-tomcat-9.0.95]# cd bin
[root@tomcat bin]# ls
bootstrap.jar ciphers.sh daemon.sh setclasspath.bat startup.sh version.bat
catalina-tasks.xml commons-daemon-native.tar.gz digest.bat setclasspath.sh tomcat-juli.jar version.sh
catalina.bat commons-daemon.jar digest.sh shutdown.bat tomcat-native.tar.gz
catalina.sh configtest.bat makebase.bat shutdown.sh tool-wrapper.bat
ciphers.bat configtest.sh makebase.sh startup.bat tool-wrapper.sh
[root@tomcat bin]# ./startup.sh
Neither the JAVA_HOME nor the JRE_HOME environment variable is defined
At least one of these environment variable is needed to run this program
[root@tomcat bin]# cd
[root@tomcat ~]# cd /opt
[root@tomcat ~]# sudo dnf install java-17-amazon-corretto -y
Last metadata expiration check: 0:28:20 ago on Fri Sep 20 13:21:56 2024.
Dependencies resolved.
=====
Package           Architecture Version       Repository      Size
=====
Installing:
java-17-amazon-corretto x86_64      1:17.0.12+7-1.amzn2023.1   amazonlinux    187 k
Installing dependencies:
alsa-lib           x86_64      1.2.7.2-1.amzn2023.0.2   amazonlinux    504 k
=====
[root@tomcat ~]# |
```

Tomcat started



```
root@developer:~ root@ip-172-31-7-173:~ root@tomcat:/opt/apache-to libSM-1.2.3-8.amzn2023.0.2.x86_64 libX11-1.7.2-3.amzn2023.0.4.x86_64 libX11-common-1.7.2-3.amzn2023.0.4.noarch libXau-1.0.9-6.amzn2023.0.2.x86_64 libXext-1.3.4-6.amzn2023.0.2.x86_64 libXi-1.7.10-6.amzn2023.0.2.x86_64 libXinerama-1.1.4-8.amzn2023.0.2.x86_64 libXrandr-1.5.2-6.amzn2023.0.2.x86_64 libXrender-0.9.10-14.amzn2023.0.2.x86_64 libXt-1.2.0-4.amzn2023.0.2.x86_64 libXtst-1.2.3-14.amzn2023.0.2.x86_64 libbrotli-1.0.9-4.amzn2023.0.2.x86_64 libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64 libpng-2.1.6.37-10.amzn2023.0.6.x86_64 libxcb-1.13.1-7.amzn2023.0.2.x86_64 pixman-0.40.0-3.amzn2023.0.3.x86_64 xml-common-0.6.3-56.amzn2023.0.2.noarch Complete! [root@tomcat opt]# cd apache-tomcat-9.0.95/ [root@tomcat apache-tomcat-9.0.95]# cd bin [root@tomcat bin]# ./startup.sh Using CATALINA_BASE: /opt/apache-tomcat-9.0.95 Using CATALINA_HOME: /opt/apache-tomcat-9.0.95 Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.95/temp Using JRE_HOME: /usr Using CLASSPATH: /opt/apache-tomcat-9.0.95/bin/bootstrap.jar:/opt/apache-tomcat-9.0.95/bin/tomcat-juli.jar Using CATALINA_OPTS: Tomcat started. [root@tomcat bin]#
```

Now configure it . using the ip address of instance where the tomcat is running. Running it on the port no. 8080.



The screenshot shows a web browser window with the URL 13.233.197.243:8080. The page title is "Apache Tomcat/9.0.95". The main content area displays a green banner stating "If you're seeing this, you've successfully installed Tomcat. Congratulations!" Below the banner is a cartoon cat icon. To the right of the banner are three buttons: "Server Status", "Manager App", and "Host Manager". The page also features a "Developer Quick Start" section with links to "Tomcat Setup", "First Web Application", "Realms & AAA", "JDBC DataSources", "Examples", and "Servlet Specifications". Below this are three columns: "Managing Tomcat", "Documentation", and "Getting Help". The "Managing Tomcat" column contains a note about restricted access to the manager webapp. The "Documentation" column links to "Tomcat 9.0 Documentation", "Tomcat 9.0 Configuration", and "Tomcat Wiki". The "Getting Help" column links to "FAQ and Mailing Lists" and lists the "tomcat-announce" mailing list. The browser's status bar at the bottom shows the URL 13.233.197.243:8080/examples/, the time 19:23, and the date 20-09-2024.

Then go to manager app to configure the tomcat. Change the files mentioned in the tomcat documentation below.

The screenshot shows a browser window with the URL `13.233.197.243:8080/manager/html`. The title bar includes tabs for 'Instances | EC2 | ap...', 'Elastic Container R...', 'How to Install and...', 'rutikr-lti/jenkins-p...', 'tomcat Config [Jen...', and '403 Access Denied'. The main content area displays a '403 Access Denied' error message with the following text:

You are not authorized to view this page.
By default the Manager is only accessible from a browser running on the same machine as Tomcat. If you wish to modify this restriction, you'll need to edit the Manager's `context.xml` file.
If you have already configured the Manager application to allow access and you have used your browser's back button, used a saved bookmark or similar then you may have triggered the cross-site request forgery (CSRF) protection that has been enabled for the HTML interface of the Manager application. You will need to reset this protection by returning to the main Manager page. Once you return to this page, you will be able to continue using the Manager application's HTML interface normally. If you continue to see this access denied message, check that you have the necessary permissions to access this application.
If you have not changed any configuration files, please examine the file `conf/tomcat-users.xml` in your installation. That file must contain the credentials to let you use this webapp.
For example, to add the `manager-gui` role to a user named `tomcat` with a password of `s3cret`, add the following to the config file listed above.

```
<role rolename="manager-gui"/>
<user username="tomcat" password="s3cret" roles="manager-gui"/>
```

Note that for Tomcat 7 onwards, the roles required to use the manager application were changed from the single `manager` role to the following four roles. You will need to assign the role(s) required for the functionality you wish to access.

- `manager-gui` - allows access to the HTML GUI and the status pages
- `manager-script` - allows access to the text interface and the status pages
- `manager-jmx` - allows access to the JMX proxy and the status pages
- `manager-status` - allows access to the status pages only

The HTML interface is protected against CSRF but the text and JMX interfaces are not. To maintain the CSRF protection:

- Users with the `manager-gui` role should not be granted either the `manager-script` or `manager-jmx` roles.
- If the text or jmx interfaces are accessed through a browser (e.g. for testing since these interfaces are intended for tools not humans) then the browser must be closed afterwards to terminate the session.

For more information - please see the [Manager App How-To](#).

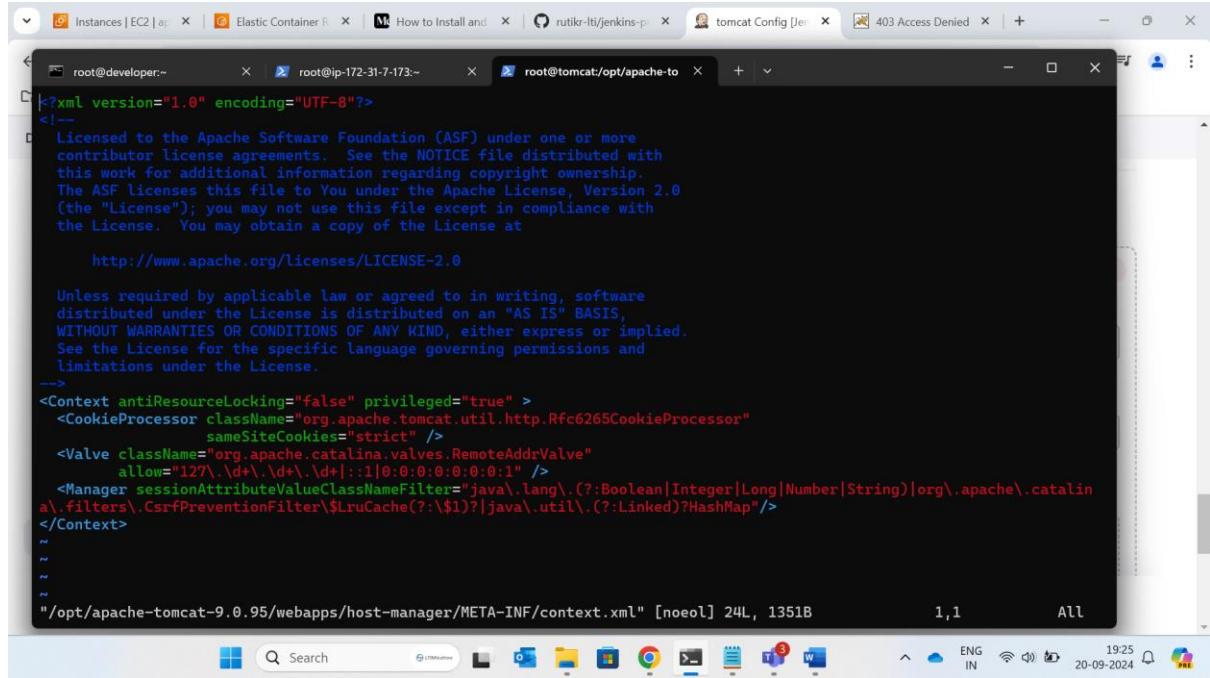
These two files we have to edit.

The screenshot shows a terminal window with the root user executing commands. The terminal tabs are labeled 'root@developer:~', 'root@ip-172-31-7-173:~', and 'root@tomcat:/opt/apache-tomcat-9.0.95/bin'. The terminal output shows:

```
libXrandr-1.5.2-6.amzn2023.0.2.x86_64
libXrender-0.9.10-14.amzn2023.0.2.x86_64
libXt-1.2.0-4.amzn2023.0.2.x86_64
libXtst-1.2.3-14.amzn2023.0.2.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libpng-2.1.6.37-10.amzn2023.0.6.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64
xml-common-0.6.3-56.amzn2023.0.2.noarch

Complete!
[root@tomcat opt]# cd apache-tomcat-9.0.95/
[root@tomcat apache-tomcat-9.0.95]# cd bin
[root@tomcat bin]# ./startup.sh
Using CATALINA_BASE: /opt/apache-tomcat-9.0.95
Using CATALINA_HOME: /opt/apache-tomcat-9.0.95
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.95/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/apache-tomcat-9.0.95/bin/bootstrap.jar:/opt/apache-tomcat-9.0.95/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@tomcat bin]# find / -name context.xml
/opt/apache-tomcat-9.0.95/conf/context.xml
/opt/apache-tomcat-9.0.95/webapps/docs/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/examples/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/host-manager/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/manager/META-INF/context.xml
[root@tomcat bin]# vim /opt/apache-tomcat-9.0.95/webapps/host-manager/META-INF/context.xml
[root@tomcat bin]# vim /opt/apache-tomcat-9.0.95/webapps/manager/META-INF/context.xml
```

We need to comment value

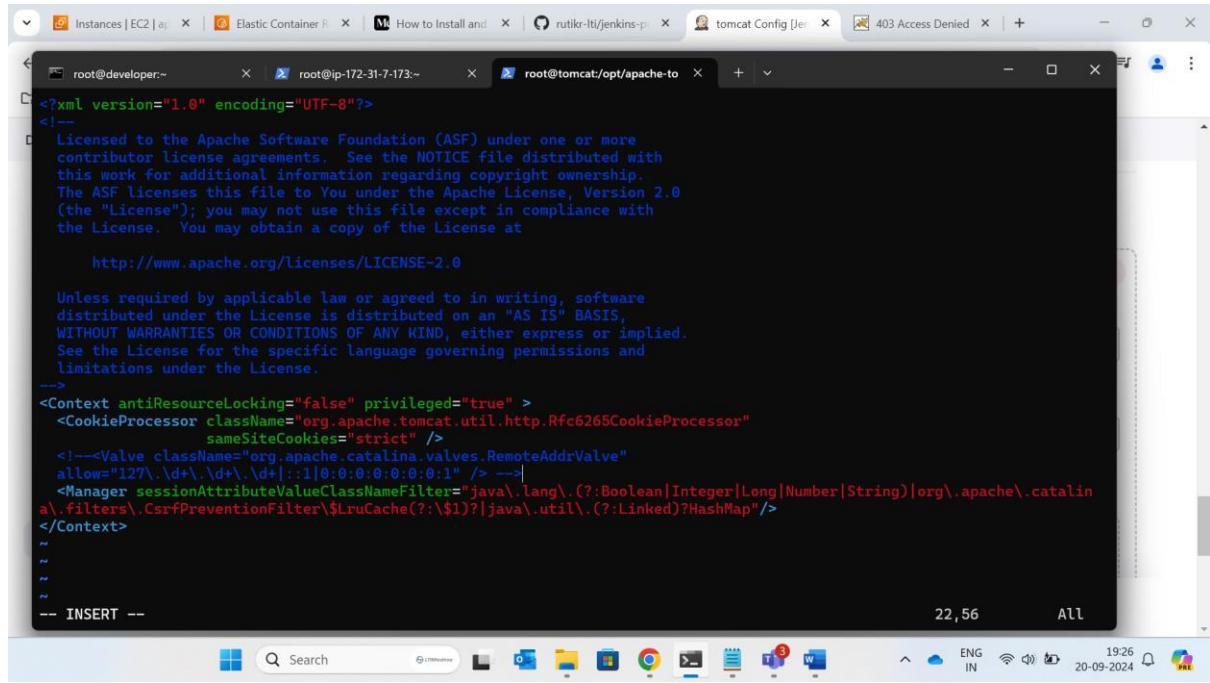


```
<?xml version="1.0" encoding="UTF-8"?>
<!--
 Licensed to the Apache Software Foundation (ASF) under one or more
 contributor license agreements. See the NOTICE file distributed with
 this work for additional information regarding copyright ownership.
 The ASF licenses this file to You under the Apache License, Version 2.0
 (the "License"); you may not use this file except in compliance with
 the License. You may obtain a copy of the License at
 http://www.apache.org/licenses/LICENSE-2.0

 Unless required by applicable law or agreed to in writing, software
 distributed under the License is distributed on an "AS IS" BASIS,
 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the specific language governing permissions and
 limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
 <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
 <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\\.\\d+\\.\\d+|::1|0:0:0:0:0:0:1" />
 <Manager sessionAttributeValueClassNameFilter="java\\.lang\\.\\?(Boolean|Integer|Long|Number|String)|org\\.apache\\.catalin
 a\\.filters\\.CsrfPreventionFilter\\$LruCache\\(\\?\\:\\$1\\)?|java\\.util\\.\\?(Linked)HashMap\\?"/>
</Context>
~
~
~
~

"/opt/apache-tomcat-9.0.95/webapps/host-manager/META-INF/context.xml" [noeol] 24L, 1351B
```

Commented the value as below.



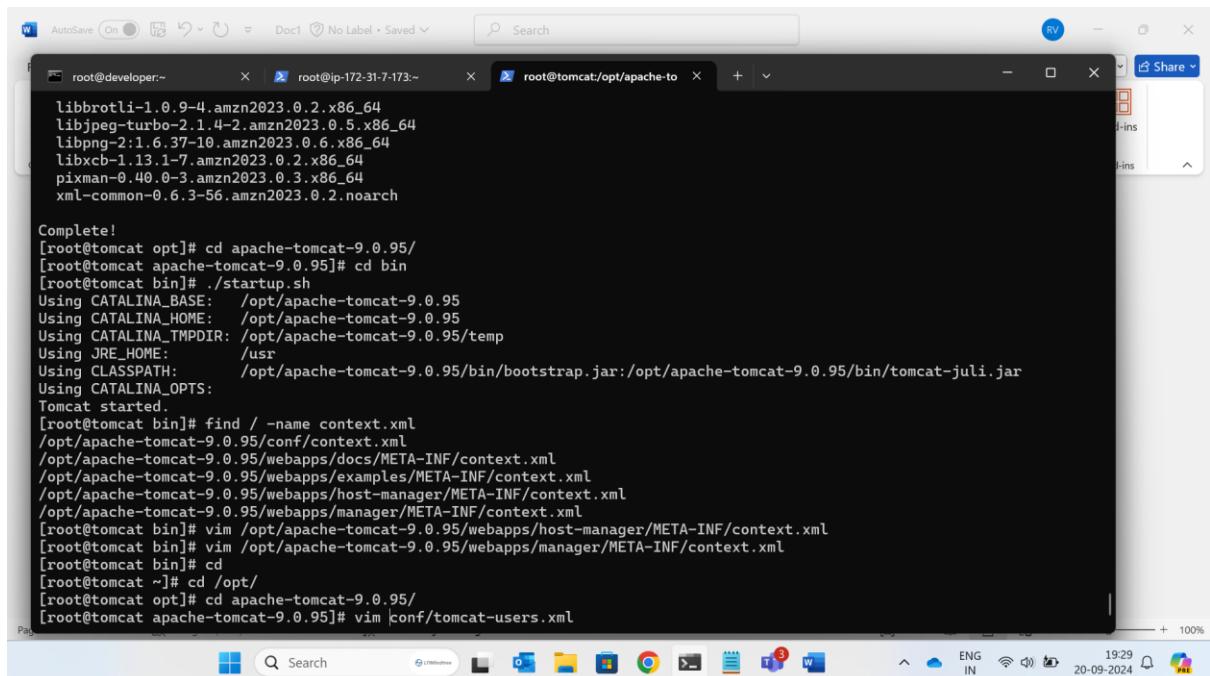
```
<?xml version="1.0" encoding="UTF-8"?>
<!--
 Licensed to the Apache Software Foundation (ASF) under one or more
 contributor license agreements. See the NOTICE file distributed with
 this work for additional information regarding copyright ownership.
 The ASF licenses this file to You under the Apache License, Version 2.0
 (the "License"); you may not use this file except in compliance with
 the License. You may obtain a copy of the License at
 http://www.apache.org/licenses/LICENSE-2.0

 Unless required by applicable law or agreed to in writing, software
 distributed under the License is distributed on an "AS IS" BASIS,
 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 See the License for the specific language governing permissions and
 limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
 <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
    sameSiteCookies="strict" />
 <!--<Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\\.\\d+\\.\\d+|::1|0:0:0:0:0:0:1" /> -->
 <Manager sessionAttributeValueClassNameFilter="java\\.lang\\.\\?(Boolean|Integer|Long|Number|String)|org\\.apache\\.catalin
 a\\.filters\\.CsrfPreventionFilter\\$LruCache\\(\\?\\:\\$1\\)?|java\\.util\\.\\?(Linked)HashMap\\?"/>
</Context>
~
~
~
~

-- INSERT --
```

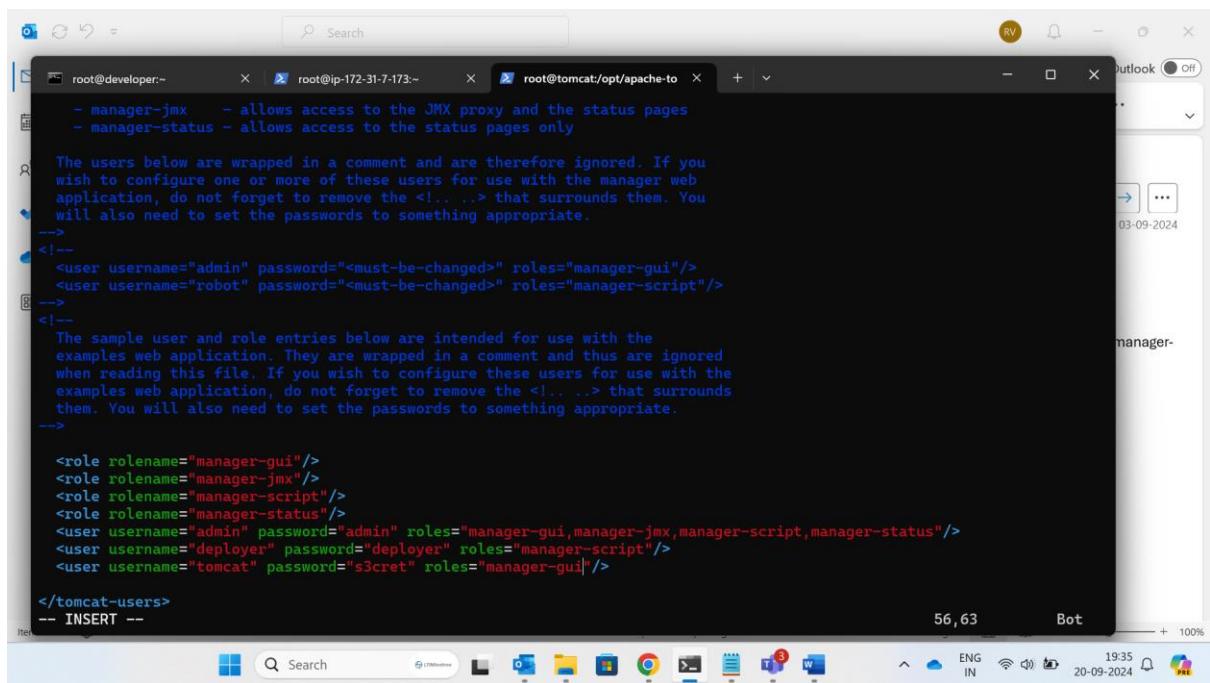
Same for another file.

Edit below file to give users access. And assign correct roles.



```
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libpng-2:1.6.37-10.amzn2023.0.6.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64
xml-common-0.6.3-56.amzn2023.0.2.noarch

Complete!
[root@tomcat opt]# cd apache-tomcat-9.0.95/
[root@tomcat apache-tomcat-9.0.95]# cd bin
[root@tomcat bin]# ./startup.sh
Using CATALINA_BASE: /opt/apache-tomcat-9.0.95
Using CATALINA_HOME: /opt/apache-tomcat-9.0.95
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.95/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/apache-tomcat-9.0.95/bin/bootstrap.jar:/opt/apache-tomcat-9.0.95/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@tomcat bin]# find / -name context.xml
/opt/apache-tomcat-9.0.95/conf/context.xml
/opt/apache-tomcat-9.0.95/webapps/docs/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/examples/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/host-manager/META-INF/context.xml
/opt/apache-tomcat-9.0.95/webapps/manager/META-INF/context.xml
[root@tomcat bin]# vim /opt/apache-tomcat-9.0.95/webapps/host-manager/META-INF/context.xml
[root@tomcat bin]# vim /opt/apache-tomcat-9.0.95/webapps/manager/META-INF/context.xml
[root@tomcat bin]# cd
[root@tomcat ~]# cd /opt/
[root@tomcat opt]# cd apache-tomcat-9.0.95/
[root@tomcat apache-tomcat-9.0.95]# vim conf/tomcat-users.xml
```



```
- manager-jmx - allows access to the JMX proxy and the status pages
- manager-status - allows access to the status pages only

The users below are wrapped in a comment and are therefore ignored. If you
wish to configure one or more of these users for use with the manager web
application, do not forget to remove the <!...> that surrounds them. You
will also need to set the passwords to something appropriate.
-->
<!--
<user username="admin" password=<must-be-changed>" roles="manager-gui"/>
<user username="robot" password=<must-be-changed>" roles="manager-script"/>
-->
<!--
The sample user and role entries below are intended for use with the
examples web application. They are wrapped in a comment and thus are ignored
when reading this file. If you wish to configure these users for use with the
examples web application, do not forget to remove the <!...> that surrounds
them. You will also need to set the passwords to something appropriate.
-->

<role rolename="manager-gui"/>
<role rolename="manager-jmx"/>
<role rolename="manager-script"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-gui,manager-jmx,manager-script,manager-status"/>
<user username="deployer" password="deployer" roles="manager-script"/>
<user username="tomcat" password="s3cret" roles="manager-gui"/>

</tomcat-users>
-- INSERT --
```

Shutdown and start tomcat to see applied changes .

The screenshot shows a browser window with multiple tabs open. The active tab is titled '/manager' and displays the Apache Tomcat Web Application Manager interface. The title bar includes the Apache logo and the text 'THE APACHE SOFTWARE FOUNDATION'. The main content area is titled 'Tomcat Web Application Manager'. A message box at the top says 'Message: OK'. Below it is a navigation bar with tabs: 'Manager', 'List Applications', 'HTML Manager Help', 'Manager Help', and 'Server Status'. The 'List Applications' tab is selected. The application list table has columns: Path, Version, Display Name, Running, Sessions, and Commands. It lists three applications: 'Welcome to Tomcat' (Path: /, Version: None specified), 'Tomcat Documentation' (Path: /docs, Version: None specified), and 'Servlet and JSP Examples' (Path: /examples, Version: None specified). Each row in the table contains buttons for Start, Stop, Reload, and Undeploy, along with an 'Expire sessions with idle ≥ [30] minutes' button.

Create tomcat job item.

The screenshot shows a browser window with multiple tabs open. The active tab is titled '/manager' and displays the Jenkins configuration for a 'tomcat' job. The title bar includes the Jenkins logo and the text 'JENKINS'. The main content area shows the 'Configuration' screen for the 'tomcat' job. On the left, there is a sidebar with various configuration sections: General, Source Code Management (selected), Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. The 'Source Code Management' section is currently active. It shows a 'Repository URL' input field containing 'https://github.com/rutikr-lti/jenkins-project2.git' and a 'Credentials' dropdown set to '- none -'. There is also a '+ Add' button and an 'Advanced' dropdown. At the bottom of this section are 'Save' and 'Apply' buttons. The status bar at the bottom of the browser window shows the date and time as '20-09-2024 19:42'.

The screenshot shows the Jenkins configuration interface for a Tomcat job. The left sidebar lists various configuration sections: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build (which is selected), Post Steps, Build Settings, and Post-build Actions. The main panel is titled 'Build' and contains fields for 'Root POM' (set to 'pom.xml'), 'Goals and options' (empty), and an 'Advanced' dropdown. Below this is the 'Post Steps' section, which includes a radio button for 'Run only if build succeeds'. At the bottom are 'Save' and 'Apply' buttons.

Select add container and select version of tomcat which has been downloaded. And provide the credentials which we added in settings of Jenkins.

The screenshot shows the Jenkins configuration interface for a Tomcat container. The left sidebar lists the same configuration sections as the previous screenshot. The main panel is titled 'Containers' and shows a single entry: 'Tomcat 9.x Remote'. This entry includes a 'Credentials' field containing 'deployer/*****' and an 'Add' button. Below it is a 'Tomcat URL' field set to 'http://13.233.197.243:8080/'. An 'Advanced' dropdown is also present. At the bottom are 'Save' and 'Apply' buttons.

We successfully build and deployed tomcat job

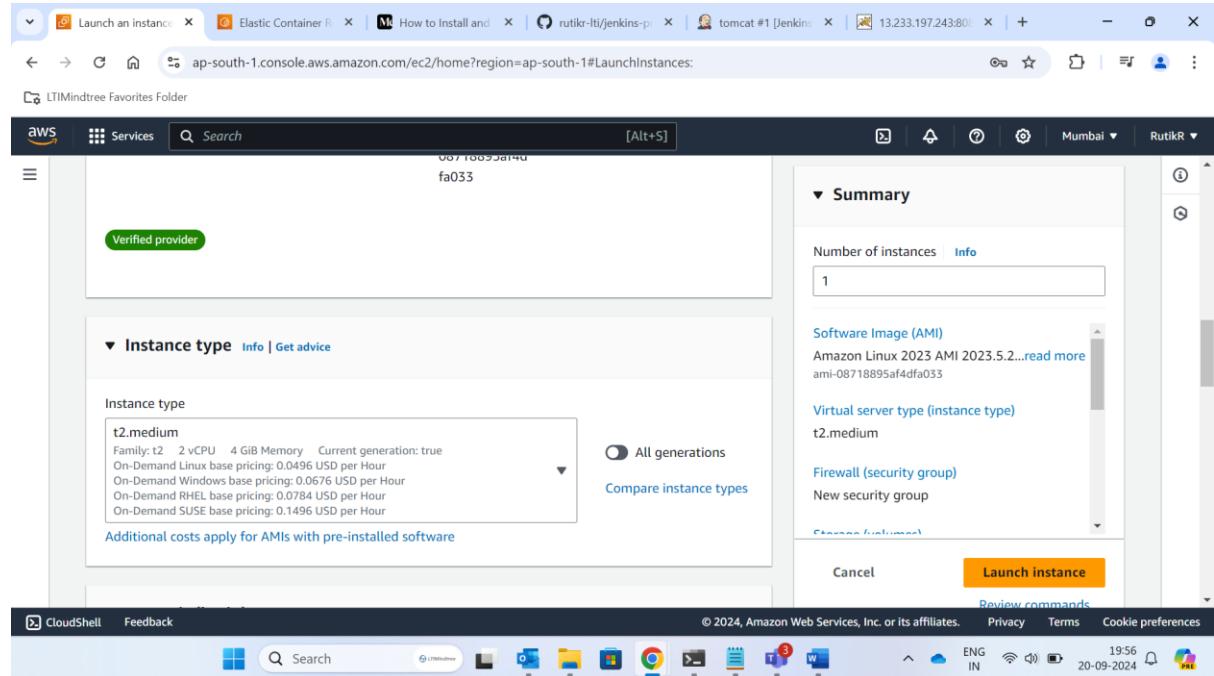
The screenshot shows the Jenkins interface for a job named 'tomcat'. The top navigation bar includes tabs for 'Instances | EC2 | ap...', 'Elastic Container R...', 'How to Install and...', 'rutikr-lti/jenkins-p...', 'tomcat #1 [Jenkins]', and '13.233.197.243:80...'. The main content area displays the build details for '#1 (Sep 20, 2024, 2:20:12 PM)'. It shows the build was started by 'Rutik R' 52 seconds ago and took 14 seconds. The build status is green. On the left, there's a sidebar with links like 'Status', 'Changes', 'Console Output', 'Edit Build Information', 'Delete build #1', 'Timings', 'Git Build Data', 'Redeploy Artifacts', 'Test Result', and 'See Fingerprints'. Below the status bar, there's a 'Module Builds' section. The bottom of the screen shows a Windows taskbar with various icons and system status.

Our webapp is live on tomcat instance public ip:8080/webapp/

The screenshot shows a web browser window with the URL '13.233.197.243:8080/webapp/'. The page title is 'This is Rutik Rekhawar ,Welcomes You! Again'. Below the title, there's a heading 'Please fill in this form to create an account.' followed by a horizontal line. The form contains five input fields: 'Enter Name' (placeholder 'Enter Full Name'), 'Enter mobile' (placeholder 'Enter mobile number'), 'Enter Email' (placeholder 'Enter Email'), 'Password' (placeholder 'Enter Password'), and 'Repeat Password' (placeholder 'Repeat Password'). At the bottom of the form, there's a note 'By creating an account you agree to our [Terms & Privacy](#)'. Below the note is a 'Register' button. The bottom of the screen shows a Windows taskbar with various icons and system status.

4) Docker Configuration

Launch instance with t2.medium package.



Connect to instance and setup docker on it.

```
Installing : iptables-nft-1.8-8-3.amzn2023.0.2.x86_64
Running scriptlet: iptables-nft-1.8-8-3.amzn2023.0.2.x86_64
Installing : libcgroup-3.0-1.amzn2023.0.1.x86_64
Running scriptlet: docker-25.0.6-1.amzn2023.0.2.x86_64
Installing : docker-25.0.6-1.amzn2023.0.2.x86_64
Running scriptlet: docker-25.0.6-1.amzn2023.0.2.x86_64
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.

Verifying : containerd-1.7.20-1.amzn2023.0.1.x86_64
Verifying : docker-25.0.6-1.amzn2023.0.2.x86_64
Verifying : iptables-libs-1.8-8-3.amzn2023.0.2.x86_64
Verifying : iptables-nft-1.8-8-3.amzn2023.0.2.x86_64
Verifying : libcgroup-3.0-1.amzn2023.0.1.x86_64
Verifying : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
Verifying : libnftnl-1.0.1-19.amzn2023.0.2.x86_64
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64
Verifying : pigz-2.5-1.amzn2023.0.3.x86_64
Verifying : runc-1.1.13-1.amzn2023.0.1.x86_64

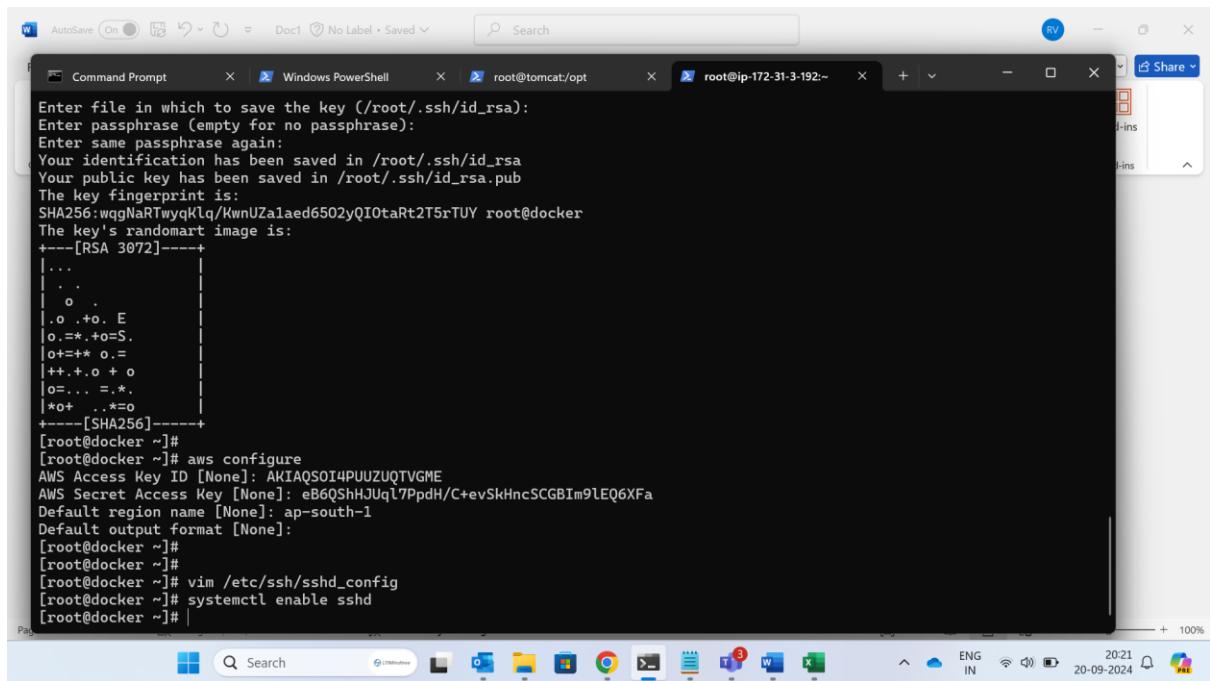
Installed:
containerd-1.7.20-1.amzn2023.0.1.x86_64
iptables-libs-1.8-8-3.amzn2023.0.2.x86_64
libcgroup-3.0-1.amzn2023.0.1.x86_64
libnftnl-1.0.1-19.amzn2023.0.2.x86_64
pigz-2.5-1.amzn2023.0.3.x86_64

docker-25.0.6-1.amzn2023.0.2.x86_64
iptables-nft-1.8-8-3.amzn2023.0.2.x86_64
libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
libnftnl-1.2.2-2.amzn2023.0.2.x86_64
runc-1.1.13-1.amzn2023.0.1.x86_64

Complete!
[root@docker ~]# sudo service docker start
Redirecting to /bin/systemctl start docker.service
[root@docker ~]# |
```

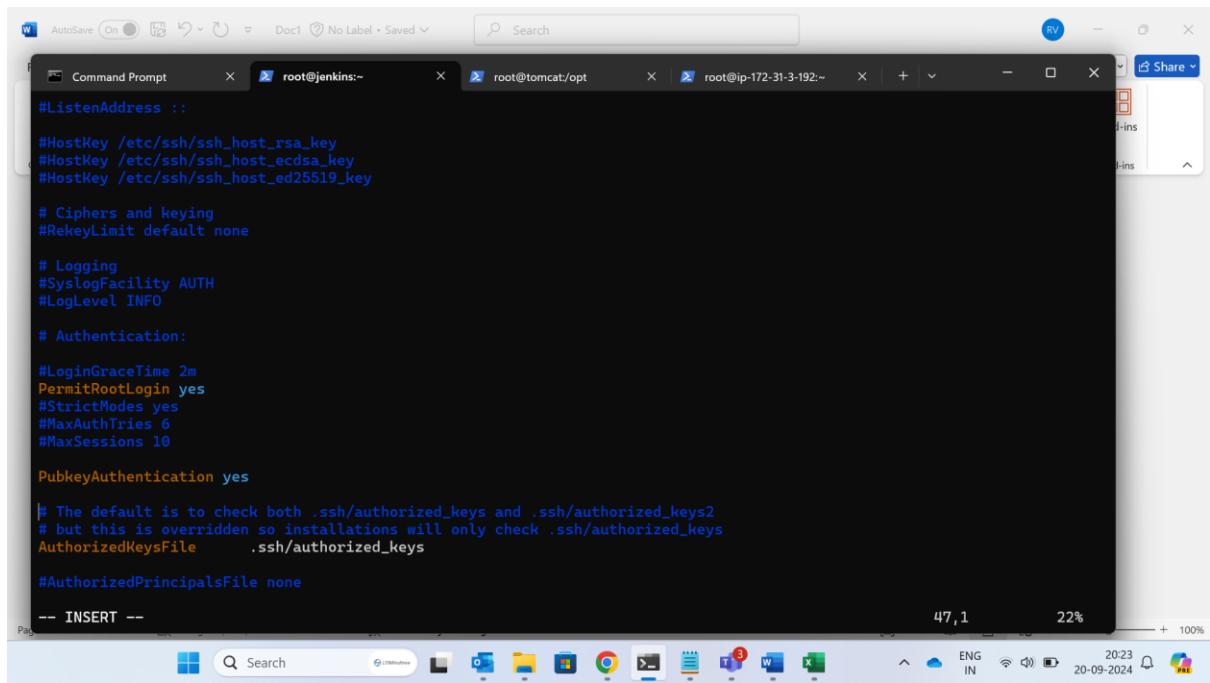
The terminal session shows the successful installation of Docker packages and the startup of the Docker service. The session is run as root on an AWS instance. The bottom status bar shows the date as 20-09-2024 and the time as 19:56.

Configure sshd after creating ssh-keygen and also aws configure.



```
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:wqqNaRTwyqKlq/KwnUZa1aed6502yQIotaRt2T5rTUY root@docker
The key's randomart image is:
+---[RSA 3072]---+
|...
| ..
| o .
| .o ..+o. E
|o.=*,+o=S.
|o+=** o =
|++,.o + o
|o=... =*.
|*o+ ..*=o
+---[SHA256]---+
[root@docker ~]# aws configure
AWS Access Key ID [None]: AKIAQSOI4PUUZUQTVGME
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
Default output format [None]:
[root@docker ~]#
[root@docker ~]#
[root@docker ~]# vim /etc/ssh/sshd_config
[root@docker ~]# systemctl enable sshd
[root@docker ~]#
```

Same at Jenkins server



```
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

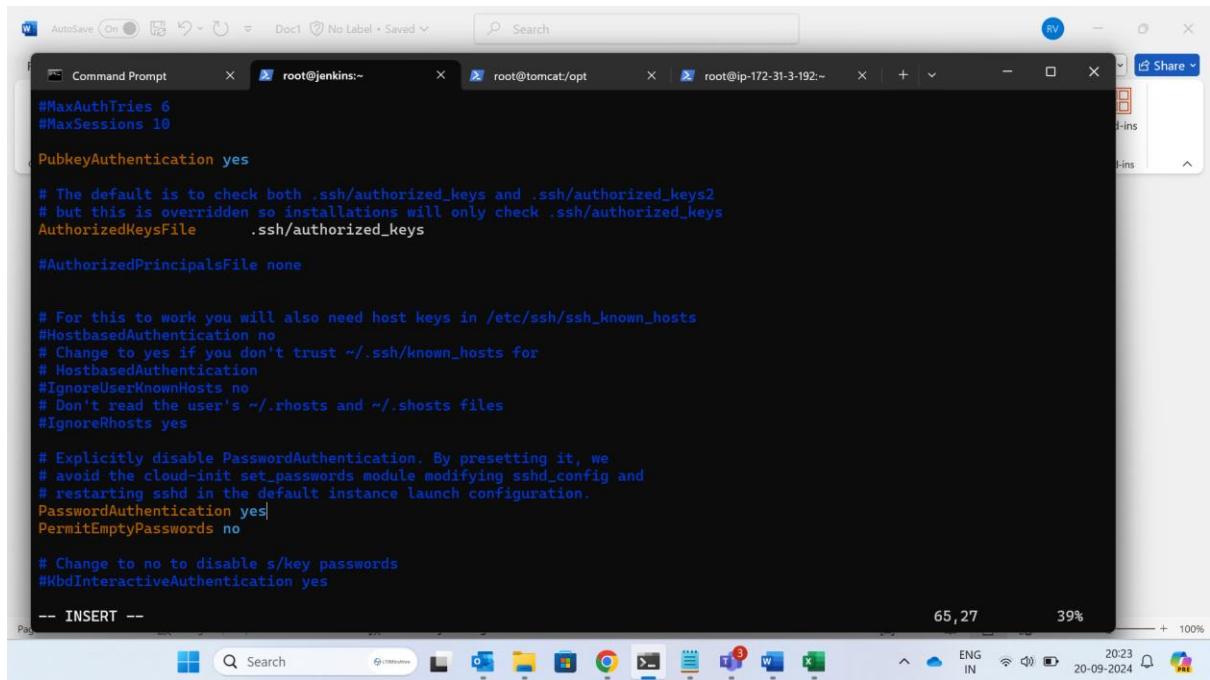
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none

-- INSERT --
```



```
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none

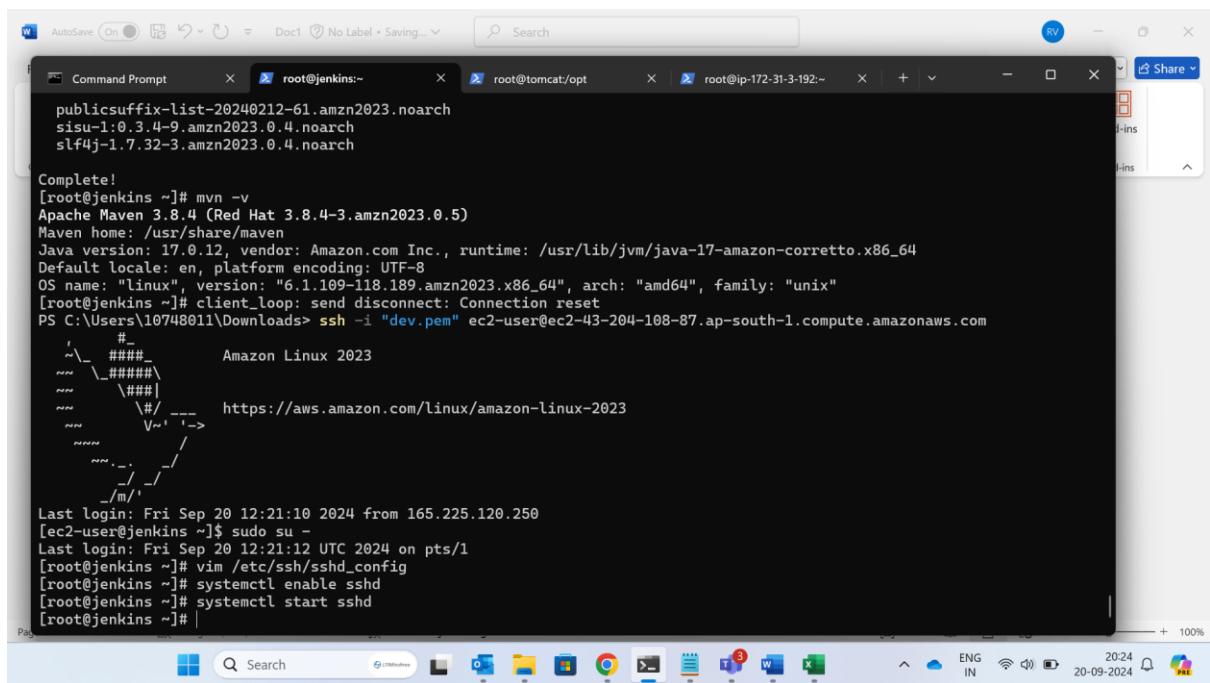
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# Explicitly disable PasswordAuthentication. By presetting it, we
# avoid the cloud-init set_passwords module modifying sshd_config and
# restarting sshd in the default instance launch configuration.
PasswordAuthentication yes
PermitEmptyPasswords no

# Change to no to disable s/key passwords
#KbdInteractiveAuthentication yes

-- INSERT --
```

After this restart sshd whenever you change configuration of sshd service.



```
publicsuffix-list-20240212-61.amzn2023.noarch
sisu-1:0.3.4-9.amzn2023.0.4.noarch
slf4j-1.7.32-3.amzn2023.0.4.noarch

Complete!
[root@jenkins ~]# mvn -v
Apache Maven 3.8.4 (Red Hat 3.8.4-3.amzn2023.0.5)
Maven home: /usr/share/maven
Java version: 17.0.12, vendor: Amazon.com Inc., runtime: /usr/lib/jvm/java-17-amazon-corretto.x86_64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.109-118.189.amzn2023.x86_64", arch: "amd64", family: "unix"
[root@jenkins ~]# client_loop: send disconnect: Connection reset
PS C:\Users\10748011\Downloads> ssh -i "dev.pem" ec2-user@ec2-43-204-108-87.ap-south-1.compute.amazonaws.com
  _#
 /###_
Amazon Linux 2023
~~ \###\
~~ \##|
~~ \#/ __ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~'`'>
~~ /
~~ _`_/
~~ _/_/
~~ _m/
Last login: Fri Sep 20 12:21:10 2024 from 165.225.120.250
[ec2-user@jenkins ~]$ sudo su -
Last login: Fri Sep 20 12:21:12 UTC 2024 on pts/1
[root@jenkins ~]# vim /etc/ssh/sshd_config
[root@jenkins ~]# systemctl enable sshd
[root@jenkins ~]# systemctl start sshd
[root@jenkins ~]# |
```

Exchange docker and Jenkins keys

```
[root@docker ~]# cd .ssh
[root@docker .ssh]# ssh-copy-id root@172.31.7.173
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.7.173 (172.31.7.173)' can't be established.
ED25519 key fingerprint is SHA256:KnBTxMvxHFiE7M+5QMBfp15azz1rM1CQ36LBuYK9I.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.7.173's password:
Permission denied, please try again.
root@172.31.7.173's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.7.173'"
and check to make sure that only the key(s) you wanted were added.

[root@docker .ssh]# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \\"ec2-user\\" rather than the user \\"root\\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtwMXE0FEUAY1GHzH13uLxTwoTV2DiorG
rpZhcKxxSYmRBWly/pDQkWP7+jSeXerkwKxBnfhgxCmDhj/f/rzPySNMk2fNgv8beC1a/5myj3Gn5d7M6v18zqGRXLJMXXe1M8n7gcTp1HUF/zOj+4772Uht
IcYHH2LWySgweiFNG3dBcmHkR/D4t73EBcCB1ga5eRoLL36zXs2tMFxkwfl/78zUjaE3g0yLAPU2hr64Y0UmjlPuCzaTb++NKS8cq+Hx4bQ9f1N8eXdjN1
SmBt9Jz6AbuNCRBtQ3n2utnfudy07VvEtA0SzX022AWE68zvqf5yxuhL5Vct3RhQf dev
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtwMXE0FEUAY1GHzH13uLxTwoTV2DiorG
I+XMP+GjakSPH6+xocfjEJVVTv1l7Klp48g5xEKzln3MaFg50IgvicVdGeaI3i0hLIo5rxsuqTOUY6NxRnmVch7yeak5rJH4PKlgHYZiS6LZs1W1pjW
4wW3XraT17YXnuKa88r6MYiL/m0rxJJmG8V6AsqCTKktwN7Qdv5tjy+eqRORxu6yo26710WL7ES5ewSc17Eb6Xg0qUv0DztpZSPWncFQzk8LM4Esycgg
R1nKkrFyH+0LFXMnhSjk9oCs5Ry7DYbTovQr04frLygWcMLPNk40Kk7WmbXZhmd5DSVoc= root@jenkins
[root@docker .ssh]#
```

```
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 0a:b8:25:b7:b1:5b brd ff:ff:ff:ff:ff:ff
    altname eni-098dac183ff8e8acb
    altname device-number-0-0
    inet 172.31.7.173/20 metric 512 brd 172.31.15.255 scope global dynamic enX0
        valid_lft 2565sec preferred_lft 2565sec
    inet6 fe80::8b8:25ff:feb7:b15b/64 scope link
        valid_lft forever preferred_lft forever
[root@jenkins ~]# cd
[root@jenkins ~]# passwd root
Changing password for user root.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@jenkins ~]# cat authorized_keys
cat: authorized_keys: No such file or directory
[root@jenkins ~]# cd .ssh
[root@jenkins .ssh]# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \\"ec2-user\\" rather than the user \\"root\\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtwMXE0FEUAY1GHzH13uLxTwoTV2DiorG
rpZhcKxxSYmRBWly/pDQkWP7+jSeXerkwKxBnfhgxCmDhj/f/rzPySNMk2fNgv8beC1a/5myj3Gn5d7M6v18zqGRXLJMXXe1M8n7gcTp1HUF/zOj+4772Uht
IcYHH2LWySgweiFNG3dBcmHkR/D4t73EBcCB1ga5eRoLL36zXs2tMFxkwfl/78zUjaE3g0yLAPU2hr64Y0UmjlPuCzaTb++NKS8cq+Hx4bQ9f1N8eXdjN1
SmBt9Jz6AbuNCRBtQ3n2utnfudy07VvEtA0SzX022AWE68zvqf5yxuhL5Vct3RhQf dev
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtwMXE0FEUAY1GHzH13uLxTwoTV2DiorG
kV1H8DiSH3moluhYLogFOBVYbdkwff01d1D8P0y0u9yN9QzWhhiy4rc72hXCPLzWAWr+kWEcmajw16Te93bEkhkkrW+WoPTOPr+G6/v1/6LZEUIIB/BT3ZB2
UnbJga6vkhthdS1j/Cm6nzXxuQwAacq2M7mHwtqHTffttxllbl68rD2VTLqSRWSF4kTRX5kWbCYXMWQPUSqAnZm5B6MScE/xRuKdQkqTjJ2k5DUstR6FLz
k8EZ/apGEd2ldy6A8IEBo/vzLGTGkkLjeUkydhXM/FexE4E8MFredi4xdBG4B7Voy11l25X6EVvCECQ22juhJZSUZXyP2vV0mu/MYyU6fSUbiY80rwY8kSi
x10rSgw0xgXapew2Cte97W06F/9TY94mlMh/6iFLARjZH1qgygWBgk6ZJ2ikz+DRGjs= root@docker
[root@jenkins .ssh]#
```

```
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.7.173's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.7.173'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins .ssh]# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-pty,command="echo 'Please login as the user \"ec2-user\" rather than the user \"root\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtWaMXE0FEUAY1GHrH13uIxTwoTV2DiorGrpZhcKxxSYmR8Wq/pDQkWP7+JSeXerkwKxBnfhgxCmhj/f/rzPySnMk2FNgv8beCia/5myj3Gn5d7M6v18zqGRXLJM3e1M8n7gcTp1HUF/zOj+4772UhtIcYHH2LWySgweiFNG3dBcmHKR/D4t73EcBCB1ga5eRoLL36zXs2tzMFxwnfl/78zUjaE3gdY LAPU2hr64Y0UmjlPuCzaTb++NKS8cq+Hx4bQ9f1N8eXdjN1SmBt9Jz6AbuNCRBtQ3n2utnfudye07VvEtA0SzX022AwE68zvf5yxuhL5Yct3RhQF dev
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQBgQxCs5637TKzKtp+BYoYTzkoJMSy5nJmWE+1kis+Vh//sq6m2L6FM26MJB1cqQhWCHqG2k4iOMPvUBLN1v4kV1H8DiSH3moUhYUogFOBVYbdkwff0id1D8P0q8uy9Nx9QzWHhiy4rc72hXCPlZwAWr+kWEcmajwU61e93bEkhkkrK+WoPTOpG6Vrl/6LZEUIIB/BT3ZB2UnbJga6vkhUdSlj/CM6nzXxuYQwAacq2M7mhWtqHTffttxLlbL68rD2VTLqSRWSF4hTRXj5kWcYXNWQPU5qAnZm5Bh6Msce/xRukQkqTj2k5DUstR6FLzk8EZ/ apGEq2Ldy68IEBo/vzLGTGkkLjeUkyDHXM/FexE4E8MFrEdi4rdBG4B7Voy11l25X6EVvCECQ22juhJZSUZxyP2vV0mu/MYyU6fSUbiY80rwYkSiyxj1oRSgW0xgXqpew2Cte97W06f/9TY94mllMh//6iFLARjZH1qgygWBogk6ZJ2iJk+DRGjs= root@docker
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQBgCzbDifMfwTTzAqrAQZjHDCsKaZUkipzkl/B5uCUrZDAATZD/nvr052Ujcac63C1MVQeSdXD/a75dUmPh0d1RTJ6tRLYUpar+kvxDWR3iryuY2kmoIAAUcgq19iq3m98tZDMA1v76HQzCxKwnBx100cpd5BNLc022igdbyt/LBm0Dr/bp6ntZBxFdDg8KnVdXF4o/lqy7I+XMX+GjakspH6+xocfjEJVvtTv117KLqp48g5xEkzLn3MaFg50IgvicVdGeaI3iohLi05rxsuvgTOULY6NxRnmVch7yeak5rJH4PKlgHYZiS6Lzs1w1pjW4wWM3XraT17YXnukaB8r6AMYil/m0rxJJjmG8V6AsqCTKktwN7Qdv5tyj+eqRORxu6yoze710WL7ES5ewSc17Eb6Xg0qUV0dZtpZSPWncFQzk8lM4EsyccgR1nKrrFlyH+OLFMXnhsJh9oCs5RyV7DYbToqvRqo4frLygWcMLPNk40Kk7wMbKXZhM5DSVoc= root@jenkins
[root@jenkins .ssh]#
[root@jenkins .ssh]# cd
[root@jenkins ~]#
[root@jenkins ~]#
```

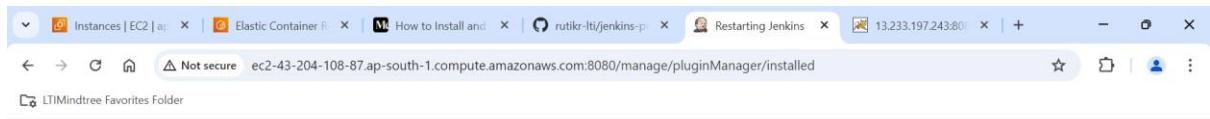
```
[root@jenkins .ssh]# cd
[root@jenkins ~]# cd .ssh
[root@jenkins .ssh]# ls
authorized_keys
[root@jenkins .ssh]# cd
[root@jenkins ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+lbkn48LQ/9culpbKxCk30YUBEnYP+rd3QfuYOCHRkk root@jenkins
The key's randomart image is:
+---[RSA 3072]---+
| +o+.
| . o +
| . B .
| =.B
| EoO o
| . =o* .
| . ..*.Bo.+o|
| ... *oOoo=|
| ... ++Oo. |
+---[SHA256]---+
[root@jenkins ~]#
[root@jenkins ~]#
[root@jenkins ~]# cd .ssh
[root@jenkins .ssh]#
```

Go to Jenkins add one more plugin “publish over ssh” to the Jenkins plugin.

The screenshot shows the Jenkins plugin manager interface. A search bar at the top contains the text "publish over". Below the search bar, there is a table with two rows. The first row is for the "Infrastructure plugin for Publish Over X" version 0.22, which is described as "Send build artifacts somewhere." and is currently "Enabled". The second row is for the "Publish Over SSH" version 1.25, which is described as "Send build artifacts over SSH" and has a link "Report an issue with this plugin". Both rows have a toggle switch icon to their left.

Install it.

The screenshot shows the Jenkins plugin manager interface with the "Download progress" tab selected. On the left, there is a sidebar with links: Updates, Available plugins, Installed plugins, Advanced settings, and Download progress (which is highlighted). The main area is titled "Download progress" and contains two sections: "Preparation" and a list of plugins. In the "Preparation" section, there are three items: "Checking internet connectivity" (Success), "Checking update center connectivity" (Success), and "Success". Below this, a list of plugins shows their download status: Oracle Java SE Development Kit Installer (Success), Command Agent Launcher (Pending), Infrastructure plugin for Publish Over X (Pending), Publish Over SSH (Pending), Loading plugin extensions (Pending), and Restarting Jenkins (Pending). At the bottom, there is a link "→ Go back to the top page (you can start using the installed plugins right away)".



Copy private key of Jenkins to the Jenkins server in publish over ssh tool in system settings.

```
dLRTj6rLYUpar+kvxWR3irYv2kmoIAAUccqg19ioq3m98tZDMA1v76HqZCxkwBxI00cpd5BNlC022igbyt/LBm0Dr/bp0ntZBxFdDg9knVdXF4o/lqy7I+XMX+GjakspH6+xocfjeJVVTv1l7Klpq48g5xEKzLn3Mafg56IgvicVdGeaI310hLi5frisuqvTOU1Y6NrNmVchg7yeak5rJH4PKlgHYZiS6LzsLw1pjW4wWM3XraT17YXnwka88r6AMyil@m0rxJjmGBV6AsqCTKktwN70d5tyj+eqRORxu6yoz6710WL7ES5ewSc17Eb6x9@qUV0dZtpZSPWncFQzk8lM4E$syccgRlnkrFlyH+0LFMXnhsJk9oCs5Ry7DYbTovQro4frlygWcMLPNk40Kk7wMbXZh5DSVoc= root@jenkins
[root@jenkins .ssh]#
[root@jenkins .ssh]# cd
[root@jenkins ~]#
[root@jenkins ~]# cd .ssh
[root@jenkins .ssh]# cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaClZXktdjEAAAAABG5vbmUAQAAAEBm9uZQAAAAAAAAAAABAwAAAAdzc2gtcnNhAAAAAwEAAQAYEAm2XShzH1rU7WQKqwEGYwxrCgM1Coqc8SpwebglK2QwAE2Q/576zudlI3Gn0twtTFUHknVw/+xVJj4dHZUUYerUS2FKWq/p1801kd4q8xmNpJqCAAFHIktfYqk5t5fLWQzAnB+fH6QsSsJwcjnKxeQTzQtNtoofHw8rf5QZjg6/26dJ7WQaxX04PJJ1XVxeKP5asuyPlzF/h02pLKR+saH4hCVVb79ZeypaqePIOcRcs5Z9zGn40dCIL4gLXRnmh4UjoSykOa8blqr@z1lW0jcuZ5LXYI08nniuayR+dypYB2GYku12bVtaY1uMFjN162ky02F57imgFk+gDGtpf5tK8SSY5hvFegLkgkyircDe0Hb+bco/nqkTkcbsqM+u9Tli+xEuXsEnNxG+L4NKLTnWbaWUj1p3BUn5PJT0BEzMnIEdzYq6xZch/tCxtF54bcZPaAx0Uclew2G06L0h60h65wIFnDCzzONCp08DgyL2YZzuQ0laAAAF1kG8eFOhvhTAaaaB3NzaClcy2EAAAGBAJtLoh8x9a101kCqsBbmMcMkwDNOqknPOT8hm4StkMABNkP+r+s7nZSNxpzrcLUXvB5J1cP9rv1LSY+HR2vFMnq1Eths1lqv65/ENZHeKvK5jaSaggABryCrX2Kireb3y1kMwDW/voepkLERCcHeg45yL3kE2ULtbaKB1vk3+UGY40vunSe1kGsv900DySdv1cx1j+lwLsj5cf4aLqNqSykfr7Gh+x+MQLW10/WxsqWqnjyDnQc+0Wfcxp+DnQjC+iJV0Z5ojeI6EsijmvGy6q9M55Vj03FgezVvGDvJ5ormskfg8wWAdhmJLtotmyVbwlmNbjeByzdetpMjthee4poHyvoAxikX+bSvElkm0YbxOyoJMoq3A3tB2/m3KP56pE5HG7rKjPrvUyVsRL17BjzXsRvpeDSpRU51m2l19adwVD0tUzgRKzJxyBHwqusWX1f7QsUxeeeGwmT2gKzlHjsNht0i9Cujh+uViBZwms82TjQqTvAxspdmGbKJNjWhAAAAMBAEAAAGAFwAHDTZtYeddrzs8qjC2Bmw+5Zl7118xYCMAszCKF3zN0uNmdtb9ph2eskeU3FprvN/ox81wbHOzo46+Pnzcb4DQALZs7MMmm1tBVR/gMat/A2dZG62VcEEzFI0/qxGkkM2ee7Z5dXYuLMgzH+/lx+f0BxK20f9L/4YThW5gpEkThQcmqsyYa9F85Ep1duqBB9BhWIGDrbAzUWCWj0wNURLE4JK/x60/cCY3xtVt
```

Path to key ?

```
-----BEGIN OPENSSH PRIVATE KEY-----  
b3BlbnNzaC1rZXktdjEAAAABG5vbmUAAAAEbmc9uZQAAAAAAAAABAABlwAAAAdzc2gtcn  
NhAAAAAwEAAQAAAYEam2XSHzH1rU7WQKqwEGYxwwrCgM1Coqc85PwebglK2QwAE2Q/576z  
udlI3Gn0wtTFUHknVw/2u+XVj4dHZUUYerUS2FKWq/p1.8Q1kd4q8rmNpjCAAFHIktY  
qKT5vFLWQzANb++h6mQsSjwvSDjnKxeQTZQtNtooHW8rf5QZjg6/26dJ7WQaxXQ4PJ1X  
-----END OPENSSH PRIVATE KEY-----
```

Disable exec ?

SSH Servers

SSH Server
Name : docker
Hostname : 172.31.3.192
Username : root
Remote Directory : /root

Save Apply

Now in system add ssh server of Jenkins as well as docker server.

Dashboard > Manage Jenkins > System >

SSH Server

Name ? docker

Hostname ? 172.31.3.192

Username ? root

Remote Directory ? /root

Save Apply

Now go to the tomcat job and send over ssh artifacts to the job configuration.

The screenshot shows the Jenkins job configuration page for a 'tomcat' job. On the left, a sidebar lists various build steps: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. 'Post-build Actions' is currently selected. In the main panel, under the 'SSH Server' section, the 'Name' field is set to 'jenkins'. Below it, the 'Transfers' section is expanded, showing a 'Transfer Set' with fields for 'Source files' and 'Remove prefix'. At the bottom of the panel are 'Save' and 'Apply' buttons. The browser's address bar shows the URL as 'ec2-43-204-108-87.ap-south-1.compute.amazonaws.com:8080/job/tomcat/configure'. The system tray at the bottom right indicates the date as 20-09-2024 and the time as 21:37.

This screenshot shows the same Jenkins job configuration page for the 'tomcat' job. The 'Post-build Actions' sidebar is still selected. In the main panel, the 'SSH Server' section has been modified. The 'Remove prefix' field now contains 'rsync -avh /var/lib/jenkins/workspace/tomcat/* root@172.31.3.192:/opt'. The 'Advanced' dropdown menu is open, and a note at the bottom states: 'All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)'. The 'Save' and 'Apply' buttons are visible at the bottom. The browser's address bar and system tray are identical to the previous screenshot.

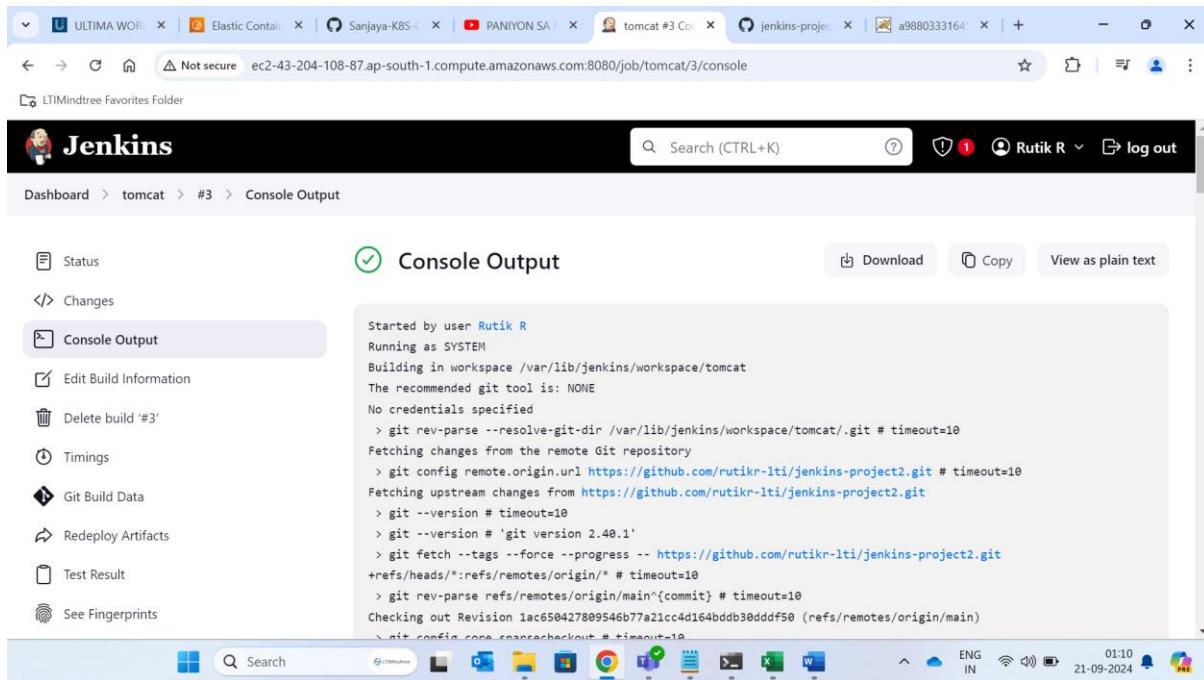
The screenshot shows the Jenkins configuration interface for a job named 'tomcat'. The left sidebar lists various build steps: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. The 'Post-build Actions' step is currently selected. On the right, under the heading 'SSH Server', there is a section titled 'Transfers' with a 'Transfer Set' configuration. It includes fields for 'Source files' and 'Remove prefix'. At the bottom of the page are 'Save' and 'Apply' buttons.

These commands copy from ecr where we create repository.

The screenshot shows the Jenkins configuration interface for a job named 'tomcat'. The left sidebar lists various build steps, with 'Post-build Actions' selected. On the right, under the heading 'Post-build Actions', there is a 'Exec command' section containing the following script:

```
cd /opt
aws ecr get-login-password --region ap-south-1 | docker login --username AWS --password-stdin
039612874025.dkr.ecr.ap-south-1.amazonaws.com
docker build -t my-data .
docker tag my-data:latest 039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data:latest
```

Below the command, a note states: "All of the transfer fields (except for Exec timeout) support substitution of Jenkins environment variables". At the bottom of the page are 'Save' and 'Apply' buttons.



So docker image is created

```
root@kubernetes:~ root@jenkins:~ ssh root@docker/opt
~~ \_#####
~~ \|##|
~~ #/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~ ' ->
~~ .-
~~ / \
~~ _/ _/
~~ _m/_\

Last login: Fri Sep 20 18:22:06 2024 from 136.226.232.244
[ec2-user@docker ~]$ sudo su -
Last login: Fri Sep 20 18:22:09 UTC 2024 on pts/3
[root@docker ~]# cd /opt
[root@docker opt]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   latest    ac107b70cled  29 minutes ago  474MB
my-data             latest   ac107b70cled  29 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   79c520c43d42  31 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   d6197bf4fb05  34 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   878728dd1765  36 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   93f9a6bf46bf  38 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   599581b38b2f  44 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   87691da25807  51 minutes ago  474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   ceadb6994023  4 hours ago   474MB
039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data   <none>   d6e6994e6f29  4 hours ago   474MB
[root@docker opt]# docker iamge
docker: 'iamge' is not a docker command.
See 'docker --help'.
[root@docker opt]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
[root@docker opt]#
```

Image is created in docker and pushed into ecr repo.

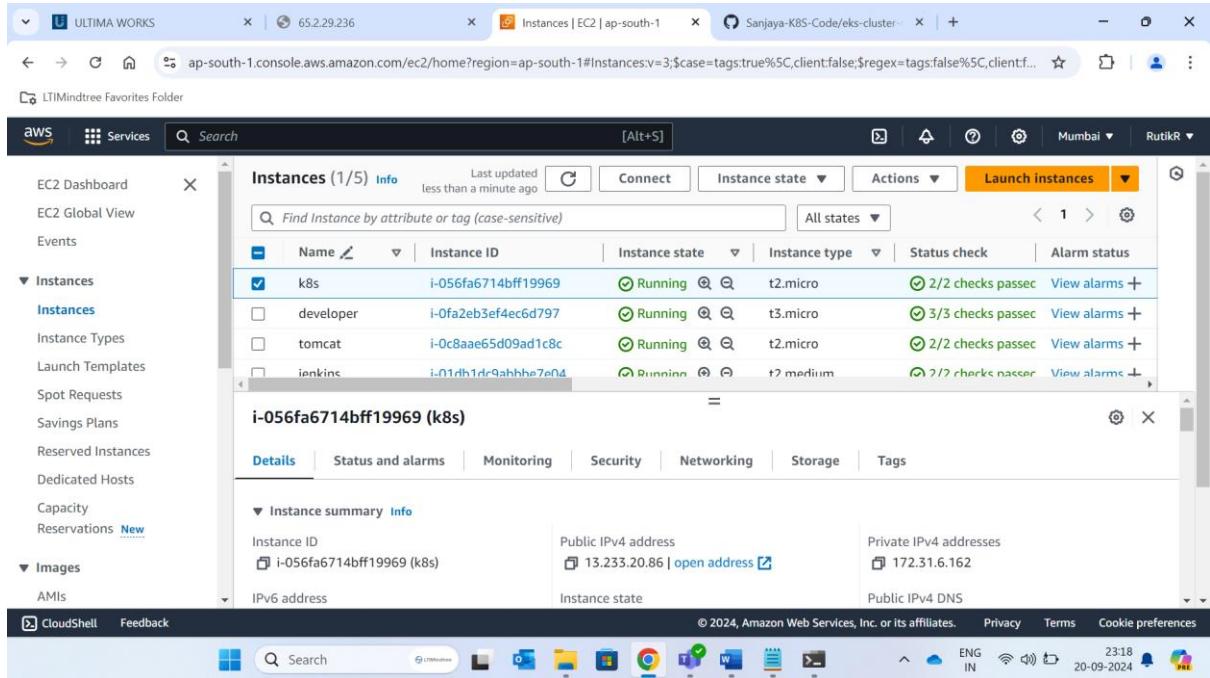
The screenshot shows the AWS ECR console interface. The left sidebar has a tree view with 'Private registry' expanded, showing 'Repositories', 'Summary', 'Images' (which is selected), 'Permissions', 'Lifecycle Policy', 'Repository tags', and 'Features & Settings'. Below that is 'Public registry' with 'Repositories' and 'Settings'. The main content area shows the 'my-data' repository details. At the top right is a 'View push commands' button. Below it is a table titled 'Images (10)' with columns: Image tag, Artifact type, Pushed at, Size (MB), Image URI, and Digest. There are two rows visible:

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest
latest	image	September 21, 2024, 00:42:03 (UTC+05:5)	227.35	Copy URI	sha256:7286891...
-	Image	September 21, 2024, 00:39:18 (UTC+05:5)	227.35	Copy URI	sha256:c462f724...

At the bottom of the page, the URL is https://ap-south-1.console.aws.amazon.com/ecr/repositories/private/039612874025/my-data/_/image/sha256:72868914e9045e284101a2186592a06719b67ab0f96c30a23f4b36c09112152/details?region=ap-south-1. The status bar shows the date as 21-09-2024 and time as 01:07.

5) Kubernetes

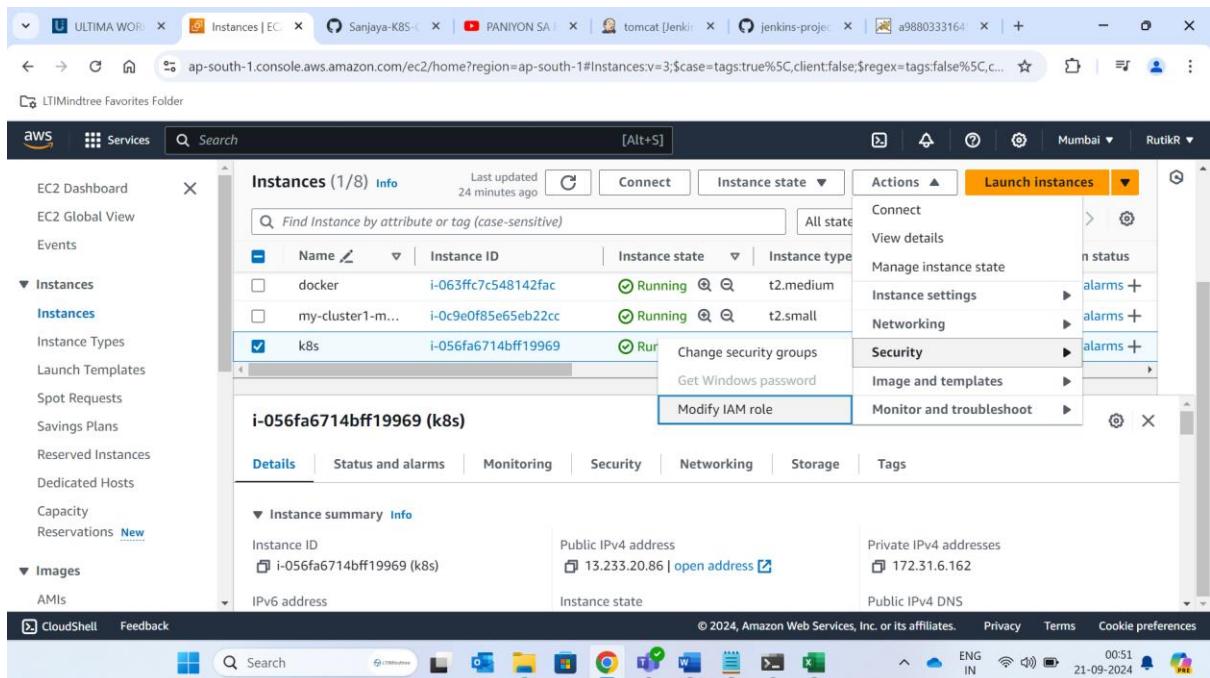
Connect to instance to set Kubernetes on it.



The screenshot shows the AWS EC2 Instances page. There are four instances listed:

Name	Instance ID	Instance state	Instance type	Status check
k8s	i-056fa6714bff19969	Running	t2.micro	2/2 checks passed
developer	i-0fa2eb3ef4ec6d797	Running	t3.micro	3/3 checks passed
tomcat	i-0c8aae65d09ad1c8c	Running	t2.micro	2/2 checks passed
jenkins	i-01dh1dr9ahhbe7e04	Running	t2.micro	2/2 checks passed

Attach IAM role to it. With eksfull access, ecr full access and iam full access.



The screenshot shows the AWS EC2 Instances page with the 'Actions' dropdown menu open for the 'k8s' instance. The 'Modify IAM role' option is highlighted.

The screenshot shows the AWS IAM Modify IAM Role interface. At the top, the URL is ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ModifyIAMRole:instanceId=i-056fa6714bff19969. The page title is "Modify IAM role". The breadcrumb navigation shows "EC2 > Instances > i-056fa6714bff19969 > Modify IAM role". The main section is titled "Modify IAM role" with a "Info" link. It says "Attach an IAM role to your instance." Below this, there's a form with fields for "Instance ID" (set to "i-056fa6714bff19969 (k8s)") and "IAM role" (set to "latestrol"). There are buttons for "Create new IAM role" and "Update IAM role". The bottom right of the window has "Cancel" and "Update IAM role" buttons.

Perform ssh-keygen at k8s cluster side.

The screenshot shows a terminal window titled "root@ip-172-31-6-162:~". The user runs the command "sudo su -" and then "hostnamectl set-hostname kubernetes". They then run "ssh-keygen" and generate a public/private RSA key pair. The terminal shows the key generation process, including prompts for saving the key and entering a passphrase. The generated key is saved to "/root/.ssh/id_rsa". The terminal window is part of a desktop environment with a taskbar at the bottom.

```
sud[ec2-user@ip-172-31-6-162 ~]$ sudo su -
[root@ip-172-31-6-162 ~]# hostnamectl set-hostname kubernetes
[root@ip-172-31-6-162 ~]# bash
[root@kubernetes ~]# ssh-keygen
Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id_rsa): Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:OPMiFOzDCtnh3udf0NLkrs1LS5c9trffIYzZX4yjSs root@kubernetes
The key's randomart image is:
+---[RSA 3072]---+
| . o |
| * o |
| . = 0 o oo |
| + + = + +o |
| . o S o o+++o |
| . . o +o.++o |
| .ooE.o . |
| o= *... |
| ..o...o |
+---[SHA256]---+
[root@kubernetes ~]#
[root@kubernetes ~]#
```

Configure sshd at Kubernetes side. And also copy ssh keys both side. On k8s and Jenkins. Also on docker.

```
+----[SHA256]----+
[root@kubernetes ~]# [root@kubernetes ~]# vim /etc/ssh/sshd_config
[root@kubernetes ~]# systemctl enable sshd
[root@kubernetes ~]# systemctl restart sshd
[root@kubernetes ~]# cd .ssh
[root@kubernetes .ssh]# ls
authorized_keys id_rsa id_rsa.pub
[root@kubernetes .ssh]# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ec2-user\" rather than the user \"root\"'.;echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtWaMXE0FEUAY1GHrH13uUxTwoTV2DiorG
rpZhcKxxSYmRBWyo/pDQkWp7+JSeXerkwKxBnfhgxCmDhj/f/rzPySnMk2fNvg8beC1a/5myj3Gn5d7M6v18zqGRXLJMXX3e1M8n7gcTp1HUF/zOj+4772Uht
IcYHH2LWySgweiFNG3dBcmHkR/D4t73EBcB1lg5eRoLL36zXs2tzMFxwnfL/78zUjaE3gDyLAPU2hr64Y0UmjLPuCzaTb++NKS8cq+Hx4bQ9f1N8eXdjN1
SmBt9Jz6AbuNCRbTQ3n2utnfudeyQ7VvEtA0SzX022AW68zvqf5syuH5YCt3RhQF dev
[root@kubernetes .ssh]# ssh-copy-id root@172.31.7.173
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.7.173' (172.31.7.173) can't be established.
ED25519 key fingerprint is SHA256:KnbtXoMvxHFie7M+5QMbfpx15azzlrm1CQ36LBuYK9I.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.7.173's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.7.173'"
and check to make sure that only the key(s) you wanted were added.

[root@kubernetes .ssh]#
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 23:25 ENG IN 20-09-2024

Setup eks cluster.

```
Now try logging into the machine, with: "ssh 'root@172.31.3.192'"
and check to make sure that only the key(s) you wanted were added.

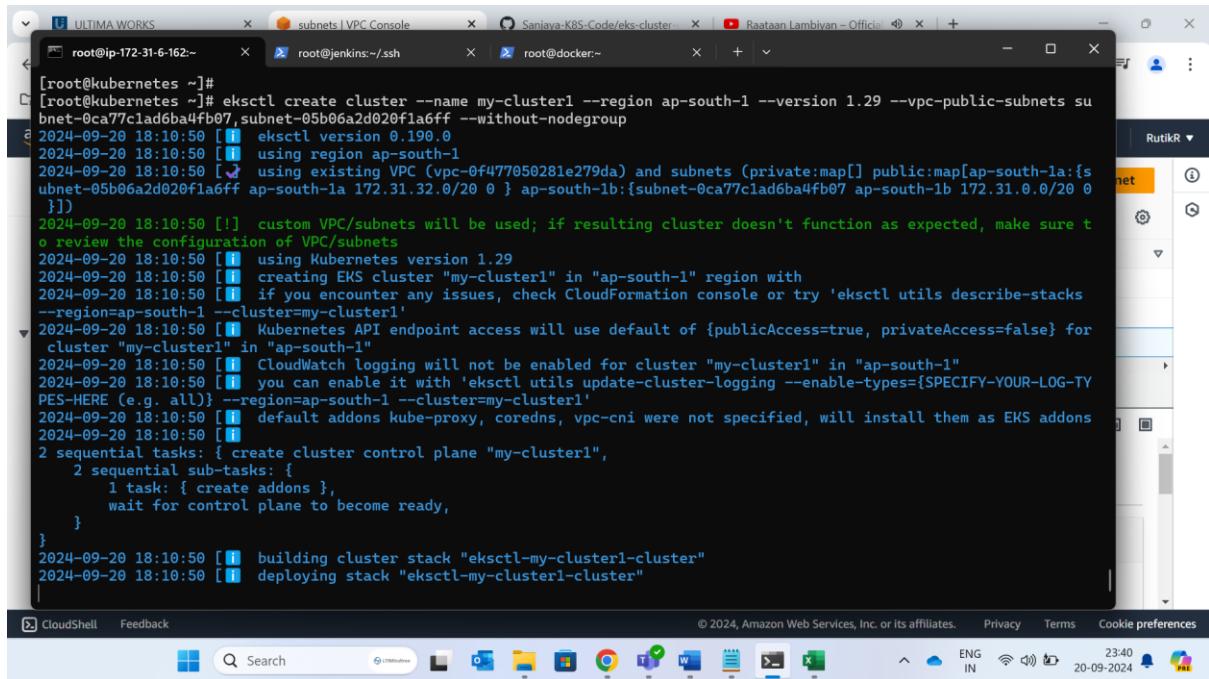
[root@kubernetes .ssh]# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 0a:b1:17:3c:db:f5 brd ff:ff:ff:ff:ff:ff
    altname eni-0d6b825afdecacceb
    altname device-number=0
    inet 172.31.6.162/20 metric 512 brd 172.31.15.255 scope global dynamic enX0
        valid_lft 2879sec preferred_lft 2879sec
    inet6 fe80::8b1:17ff:fe3c:dbf5/64 scope link
        valid_lft forever preferred_lft forever
[root@kubernetes .ssh]# passwd root
Changing password for user root.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@kubernetes .ssh]# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ec2-user\" rather than the user \"root\"'.;echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtWaMXE0FEUAY1GHrH13uUxTwoTV2DiorG
rpZhcKxxSYmRBWyo/pDQkWp7+JSeXerkwKxBnfhgxCmDhj/f/rzPySnMk2fNvg8beC1a/5myj3Gn5d7M6v18zqGRXLJMXX3e1M8n7gcTp1HUF/zOj+4772Uht
IcYHH2LWySgweiFNG3dBcmHkR/D4t73EBcB1lg5eRoLL36zXs2tzMFxwnfL/78zUjaE3gDyLAPU2hr64Y0UmjLPuCzaTb++NKS8cq+Hx4bQ9f1N8eXdjN1
Page 39 of 39 189 words English (India) Text Predictions: On Accessibility: Investigate Focus 23:38 ENG IN 20-09-2024
```

```
AutoSave  Do1  Mo1 Label  Sound  Search
root@ip-172-31-6-162~  x  root@jenkins:~/ssh  x  root@docker:~  -  X
Share  D-ins  i-ins  ^

no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ec2-user\" rather than the user \"root\"';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVFtWaMxE0FEUAY1GhrH13uUxTwoTV2DiorG rpzKhCxXwBnMk2PqDQ8eIa/ErJewkWuBnfhgqCmDhJ/r/xzPySNlk2FNgvbeIa/5myj3gn5d7M6v18zqGRXLJMX3e1M8r7gTp1HUF/Zo+4772Uht IcYHh2LWLySgewiFGN3gdBcmhK/R/D4t3Zx2zTzMFxkwfnL/78zUjaE3gDyLAPU2hr64Y0UmjLpuCzA Tb++NHS8cq+Hx4bQ9f1N8eXdjN1 SmBt9Jz6AbuNCRbTQ3n2utnfudye07VvTeAt05zX022AWE62zvqf5vhuyL5tC3RhQF dev
ssh rsa AAAAB3NzaC1yc2EAAAQABAAQCBzIDfMMWtZTAqrAQZjHDCsKAzUKipzzk/B5uCuRzDAATZD/nvr052Ujcac63C1MVQe5dXD/a75duPh0 dLR7J6tRLYUpar+kvxDW3r1ryuY2kmIAUAcq91ioq398tZDMA1v76HqZCxKwnBx1Ocp5BNlC022igdbyt/LBm0Dr/bp0ntZBxFdBg8knVdFx4o/lqy7 I+XqkphSPh3TJyJGvJvtTbQ58x5Ge2kzl3MaFg50IgvicVdgear13i0hIo5rxsuvyTOULY6NxRnmVchg7yeak5rJH4PKLgHYZis6LzsLw1pjW h4Wm3XraTT7YXnuKaB8r6AMY1il/mehrJxJjmB6AsqcTKtKwTN7Qdv5tyj+eqORxu6yo26710WL7E5ewSc17Eb6xg6qVU0dZtpzSPWnfcFzk8lM4Esyccg RInkrFrphyH+0LFMxnhjsK9oCs5Ry7DybTv0qro4frlYgWcMLPNK40K7hWMBXZh5MSVoc= root@jenkins
ssh rsa AAAAB3NzaC1yc2EAAAQABAAQCBzIDfMMWtZTAqrAQZjHDCsKAzUKipzzk/B5uCuRzDAATZD/nvr052Ujcac63C1MVQe5dXD/a75duPh0 k1Vh8DiSH3moUyUhQyFOBVBybdkuffF0id1D8P0y0uy9Nx90zWWhhi4rc272XCPLzlwAwR+rWEcmaJuW6le93bEHkmKR+wPoPTpG6Vrl/6LZEUIIB/TB3ZB2 UnbjGa6KbhJldS1j/cM6nZ50IgvicVdgear13i0hIo5rxsuvyTOULY6NxRnmVchg7yeak5rJH4PKLgHYZis6LzsLw1pjW k8E7/apG2E2dy6A8TEBo/vzLGTGkk1jeUkyDHXM/FexE4E8MFredi4rdBG4B7v0y11l25X6EVvEcQ22juhJZSUZYp2vV0mu/MYyU6fsUBiY80rwy8kSi yxjloRSGw0xgXqpeW2Cte97W06f/9TY94mlHn/6iFLARjZH1ggygWBogk6Z2JikK+dRCjs= root@docker
[root@kubernetes ~]# cd
[root@kubernetes ~]# apt-get update -y
bash: apt-get: command not found
[root@kubernetes ~]# yum update -y
Last metadata expiration check: 0:15:45 ago on Fri Sep 20 17:47:25 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@kubernetes ~]# apt install unzip -y
bash: apt: command not found
[root@kubernetes ~]# yum install unzip -y
Last metadata expiration check: 0:16:02 ago on Fri Sep 20 17:47:25 2024.
Package unzip-6.0-57.amzn2023.0.2.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
```

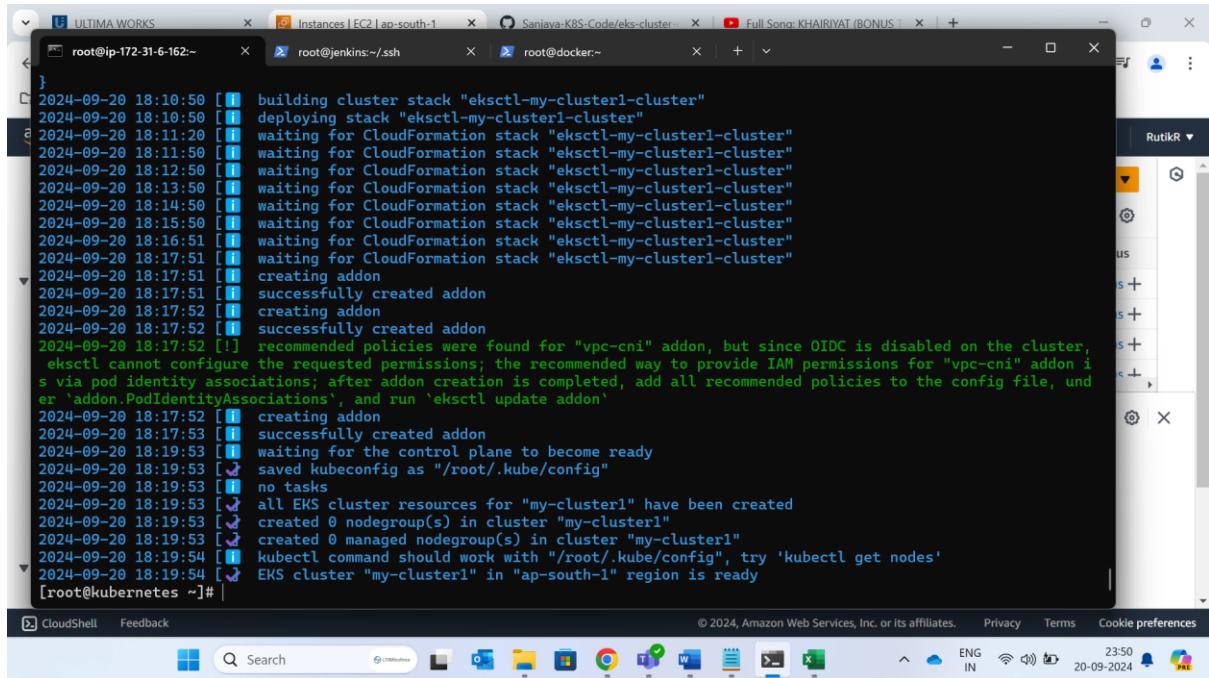
Aws configuration on the kubernetes side.

Create cluster using create cluster command and providing subnet and region properly.



```
[root@kubernetes ~]# eksctl create cluster --name my-cluster1 --region ap-south-1 --version 1.29 --vpc-public-subnets subnet-0ca77c1ad6ba4fb07,subnet-05b06a2d020f1a6ff --without-nodegroup
2024-09-20 18:10:50 [!] eksctl version 0.190.0
2024-09-20 18:10:50 [!] using region ap-south-1
2024-09-20 18:10:50 [!] using existing VPC (vpc-0f477050281e279da) and subnets (private:map[] public:map[ap-south-1:{subnet-0ca77c1ad6ba4fb07 ap-south-1a 172.31.32.0/20 0 } ap-south-1b:{subnet-0ca77c1ad6ba4fb07 ap-south-1b 172.31.0.0/20 0 }])
2024-09-20 18:10:50 [!] custom VPC/subnets will be used; if resulting cluster doesn't function as expected, make sure to review the configuration of VPC/subnets
2024-09-20 18:10:50 [!] using Kubernetes version 1.29
2024-09-20 18:10:50 [!] creating EKS cluster "my-cluster1" in "ap-south-1" region with
2024-09-20 18:10:50 [!] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=ap-south-1 --cluster=my-cluster1'
2024-09-20 18:10:50 [!] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "my-cluster1" in "ap-south-1"
2024-09-20 18:10:50 [!] CloudWatch logging will not be enabled for cluster "my-cluster1" in "ap-south-1"
2024-09-20 18:10:50 [!] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=ap-south-1 --cluster=my-cluster1'
2024-09-20 18:10:50 [!] default addons kube-proxy, coredns, vpc-cni were not specified, will install them as EKS addons
2024-09-20 18:10:50 [!] 2 sequential tasks: { create cluster control plane "my-cluster1",
  2 sequential sub-tasks:
    1 task: { create addons },
    wait for control plane to become ready,
  }
}
2024-09-20 18:10:50 [!] building cluster stack "eksctl-my-cluster1-cluster"
2024-09-20 18:10:50 [!] deploying stack "eksctl-my-cluster1-cluster"
```

Cluster is created successfully.



```
[root@kubernetes ~]#
2024-09-20 18:10:50 [!] building cluster stack "eksctl-my-cluster1-cluster"
2024-09-20 18:10:50 [!] deploying stack "eksctl-my-cluster1-cluster"
2024-09-20 18:11:20 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:11:50 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:12:50 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:13:50 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:14:50 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:15:50 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:16:51 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:17:51 [!] waiting for CloudFormation stack "eksctl-my-cluster1-cluster"
2024-09-20 18:17:51 [!] creating addon
2024-09-20 18:17:51 [!] successfully created addon
2024-09-20 18:17:52 [!] creating addon
2024-09-20 18:17:52 [!] successfully created addon
2024-09-20 18:17:53 [!] recommended policies were found for "vpc-cni" addon, but since OIDC is disabled on the cluster, eksctl cannot configure the requested permissions; the recommended way to provide IAM permissions for "vpc-cni" addon is via pod identity associations; after addon creation is completed, add all recommended policies to the config file, under 'addon.PodIdentityAssociations', and run 'eksctl update addon'
2024-09-20 18:17:52 [!] creating addon
2024-09-20 18:17:53 [!] successfully created addon
2024-09-20 18:19:53 [!] waiting for the control plane to become ready
2024-09-20 18:19:53 [!] saved kubeconfig as "/root/.kube/config"
2024-09-20 18:19:53 [!] no tasks
2024-09-20 18:19:53 [!] all EKS cluster resources for "my-cluster1" have been created
2024-09-20 18:19:53 [!] created 0 nodegroup(s) in cluster "my-cluster1"
2024-09-20 18:19:53 [!] created 0 managed nodegroup(s) in cluster "my-cluster1"
2024-09-20 18:19:54 [!] kubectl command should work with "/root/.kube/config", try 'kubectl get nodes'
2024-09-20 18:19:54 [!] EKS cluster "my-cluster1" in "ap-south-1" region is ready
[root@kubernetes ~]#
```

Create node groups.

```
No resources found
[root@kubernetes ~]# kubectl get deployment
No resources found in default namespace.
[root@kubernetes ~]# eksctl create nodegroup \
--cluster my-cluster1 \
--region ap-south-1 \
--name my-node-group \
--node-ami-family Ubuntu2004 \
--node-type t2.small \
--subnet-ids subnet-0ca77c1ad6ba4fb07,subnet-05b06a2d020f1a6ff \
--nodes 3 \
--nodes-min 2 \
--nodes-max 4 \
--ssh-access \
--ssh-public-key /root/.ssh/id_rsa.pub
2024-09-20 18:25:11 [INFO] will use version 1.29 for new nodegroup(s) based on control plane version
2024-09-20 18:25:12 [INFO] nodegroup "my-node-group" will use "ami-0ab6dcfb35da05038" [Ubuntu2004/1.29]
2024-09-20 18:25:12 [INFO] using SSH public key "/root/.ssh/id_rsa.pub" as "eksctl-my-cluster1-nodegroup-my-node-group-2c:24:b8:b3:8b:e8:1c:55:2b:6a:d2:14:8d:10:85:e9"
2024-09-20 18:25:12 [INFO] 1 nodegroup (my-node-group) was included (based on the include/exclude rules)
2024-09-20 18:25:12 [INFO] will create a CloudFormation stack for each of 1 managed nodegroups in cluster "my-cluster1"
2024-09-20 18:25:12 [INFO] 2 sequential tasks: { fix cluster compatibility, 1 task: { 1 task: { create managed nodegroup "my-node-group" } } }
2024-09-20 18:25:12 [INFO] checking cluster stack for missing resources
2024-09-20 18:25:12 [INFO] cluster stack has all required resources
2024-09-20 18:25:12 [INFO] building managed nodegroup stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-20 18:25:12 [INFO] deploying stack "eksctl-my-cluster1-nodegroup-my-node-group"
2024-09-20 18:25:13 [INFO] waiting for CloudFormation stack "eksctl-my-cluster1-nodegroup-my-node-group"
```

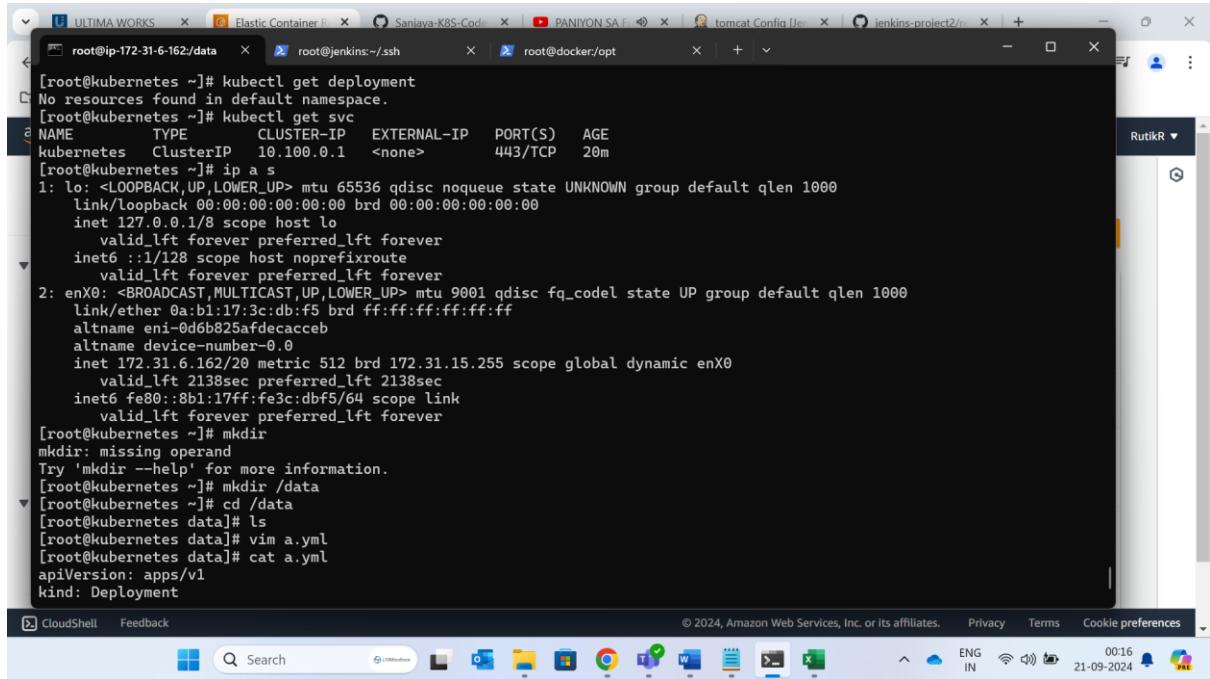
Set kubernetes on Jenkins dashboard. Add Kubernetes server to Jenkins server.

The screenshot shows a Jenkins configuration page for adding a new SSH server. The fields filled are:

- Name: kubernetes
- Hostname: 172.31.6.162
- Username: root
- Remote Directory: /root

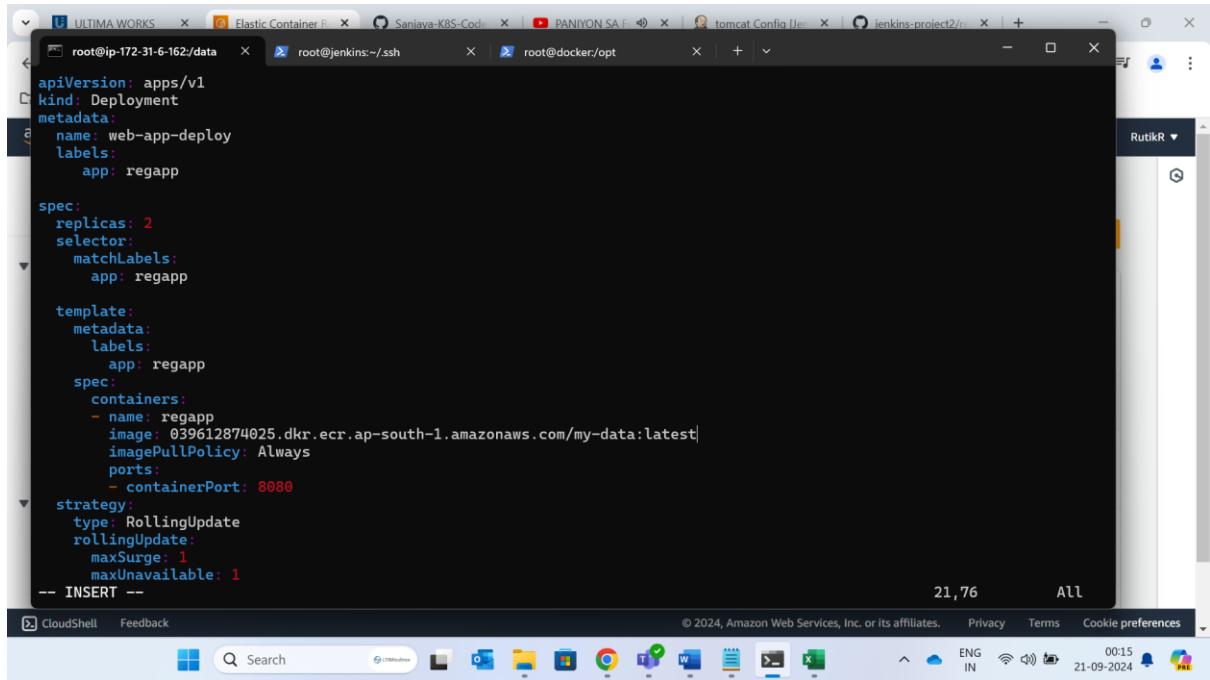
At the bottom, there are "Save" and "Apply" buttons.

Create deployment.yaml file also service.yaml file to create deployment and expose it automatically.



```
[root@kubernetes ~]# kubectl get deployment
No resources found in default namespace.
[root@kubernetes ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP  10.100.0.1   <none>        443/TCP   20m
[root@kubernetes ~]# ip a s
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
        valid_lft forever preferred_lft forever
2: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 0a:b1:17:3c:db:f5 brd ff:ff:ff:ff:ff:ff
    altname eni-0d6b825afdecacceb
    altname device-number-0
    inet 172.31.6.162/20 brd 172.31.15.255 scope global dynamic enX0
        valid_lft 2138sec preferred_lft 2138sec
    inet6 fe80::8b1:17ff:fe3c:dbf5/64 brd fe80::ff:ffff:ffff:ffff scope link
        valid_lft forever preferred_lft forever
[root@kubernetes ~]# mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
[root@kubernetes ~]# mkdir /data
[root@kubernetes ~]# cd /data
[root@kubernetes data]# ls
[root@kubernetes data]# vim a.yml
[root@kubernetes data]# cat a.yml
apiVersion: apps/v1
kind: Deployment
```

Here are the scripts available.



```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web-app-deploy
  labels:
    app: regapp
spec:
  replicas: 2
  selector:
    matchLabels:
      app: regapp
  template:
    metadata:
      labels:
        app: regapp
    spec:
      containers:
        - name: regapp
          image: 039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data:latest
          imagePullPolicy: Always
          ports:
            - containerPort: 8080
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 1
-- INSERT --
```

```
replicas: 2
selector:
  matchLabels:
    app: regapp

template:
  metadata:
    labels:
      app: regapp
spec:
  containers:
    - name: regapp
      image: 039612874025.dkr.ecr.ap-south-1.amazonaws.com/my-data:latest
      imagePullPolicy: Always
      ports:
        - containerPort: 8080
strategy:
  type: RollingUpdate
  rollingUpdate:
    maxSurge: 1
    maxUnavailable: 1
[root@kubernetes data]# vim b.yml
[root@kubernetes data]# ls
a.yml b.yml
[root@kubernetes data]# cd
[root@kubernetes ~]# ls
aws awscliv2.zip kubectl
[root@kubernetes ~]# cd /data
[root@kubernetes data]# vim b.yml
[root@kubernetes data]# |
```

Again configure the tomcat job to add kubernetes artifacts to the job.

The screenshot shows the Jenkins configuration interface for a 'tomcat' job. The left sidebar lists various configuration sections: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. 'Post-build Actions' is currently selected. In the main panel, under the 'Transfers' section, there is a 'Transfer Set' configuration. It includes fields for 'Source files', 'Remove prefix', and 'Remote directory'. At the bottom of the panel are 'Save' and 'Apply' buttons. The browser address bar shows the URL `ec2-43-204-108-87.ap-south-1.compute.amazonaws.com:8080/job/tomcat/configure`. The system tray at the bottom right indicates the date as 21-09-2024 and the time as 00:22.

This screenshot shows the same Jenkins configuration interface as the previous one, but with an additional 'Exec command' field added to the 'Transfer Set' configuration. The 'Exec command' field contains the following text:
cd /data
kubectl apply -f a.yaml
kubectl apply -f b.yaml

If job is run successful then go to k8s server to get the svc.

The screenshot shows a browser window with multiple tabs open. The active tab is 'Console Output' for a Jenkins job named 'tomcat #4'. The output shows a successful build process, starting with 'Started by user Rutik R' and running as SYSTEM. It details the git fetch and checkout steps, including the revision 'a988033164524ee18997f1e7822bfeb-1907797259.ap-south-1.elb.amazonaws.com'. The Jenkins interface includes a sidebar with options like Status, Changes, Console Output (which is selected), Edit Build Information, Delete build '#4', Timings, Git Build Data, Redeploy Artifacts, Test Result, and See Fingerprints.

Get svc external ip using kubectl get svc command.

The screenshot shows a terminal window with several tabs. The active tab displays the output of the 'kubectl get svc' command. It lists a service named 'webapp-service' with a single port mapping from port 8080 to targetPort 8080. The service type is LoadBalancer. Below this, the terminal shows the results of 'kubectl get deployment' and 'kubectl get pods', both of which show two healthy pods for the 'web-app-deploy' deployment. Finally, the 'kubectl get svc' command is run again, showing the external IP '10.100.78.135' for the service.

Paste this IP:8080/webapp/ into the browser to see the deployed webapp.

The screenshot shows a Microsoft Edge browser window with multiple tabs open at the top. The active tab's URL is <http://a988033164524ee18997f1e7822bfeb-1907797259.ap-south-1.elb.amazonaws.com:8080/webapp/>. The page content is as follows:

This is Rutik Rekhawar ,Welcomes You! Again

Please fill in this form to create an account.

Enter Name

Enter mobile

Enter Email

Password

Repeat Password

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

At the bottom of the browser window, there is a toolbar with icons for file operations, a search bar, and system status indicators like battery level, signal strength, and date/time (00:21, 21-09-2024).

Webapp deployment is successful.