

# **Prasad Tale - 10747894**

## **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 using maven.

### **Workflow to follow:**

#### **1. Source Code Management (SCM)**

- **Tool: GitHub**
- **Action: Store the source code of the sample Java web application in a GitHub repository.**

#### **2. Continuous Integration (CI)**

- **Tool: Jenkins**
- **Actions:**
  - **Setup Jenkins: Install Jenkins on an EC2 instance or use Jenkins on AWS.**
  - **Integrate GitHub with Jenkins: Use the GitHub plugin to connect Jenkins with your GitHub repository.**
  - **Create Jenkins Pipeline: Define a Jenkins pipeline using a Jenkins file stored in the GitHub repository.**
  - **Build Trigger: Configure Jenkins to trigger builds on code commits or pull requests.**

#### **3. Build Automation**

- **Tool: Maven**
- **Actions:**
  - **Define Build Steps: In the Jenkins file, use Maven to compile the code, run tests, and package the application into a JAR/WAR file.**

#### **4. Containerization**

- **Tool: Docker**
- **Actions:**
  - **Create Dockerfile:** Write a Dockerfile to containerize the Java application.
  - **Build Docker Image:** Use Jenkins to build the Docker image from the Dockerfile.

#### **5. Docker Image Storage**

- **Tool: Amazon Elastic Container Registry (ECR)**
- **Actions:**
  - **Create ECR Repository:** Set up a repository in ECR to store Docker images.
  - **Push Docker Image to ECR:** Use Jenkins to push the Docker image to the ECR repository.

#### **6. Continuous Deployment (CD)**

- **Tool: Kubernetes**
- **Actions:**
  - **Setup Kubernetes Cluster:** Use Amazon EKS (Elastic Kubernetes Service) to create a Kubernetes cluster.
  - **Deploy Application:** Use Kubernetes manifests (YAML files) to deploy the application using the Docker image stored in ECR.

**Approach:**

**SETTING GIT CLIENT:**

```

root@ip-172-31-9-94:~/project X + 
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\10743412\Downloads> ssh -i "Kubernetes.pem" ec2-user@ec2-18-222-23-121.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-222-23-121.us-east-2.compute.amazonaws.com (18.222.23.121)' can't be established.
ED25519 key fingerprint is SHA256:j1IA20W8R7luP5omUj8GI/SJ7oYg5YVJS00B+qIyq18.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-222-23-121.us-east-2.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-172-31-9-94 ~]$ sudo su -
[root@ip-172-31-9-94 ~]# yum install git
Last metadata expiration check: 0:02:19 ago on Tue Jul  9 04:05:50 2024.
Dependencies resolved.
=====
Package           Architecture   Version        Repository      Size
=====
Installing:
git              x86_64        2.40.1-1.amzn2023.0.3    amazonlinux      54 k
Installing dependencies:
git-core          x86_64        2.40.1-1.amzn2023.0.3    amazonlinux      4.3 M
git-core-doc      noarch       2.40.1-1.amzn2023.0.3    amazonlinux      2.6 M

```

## Git - Instance is connected in the Terminal

```

root@ip-172-31-9-94:~/project X + 
Initialized empty Git repository in /root/.git/
[root@ip-172-31-9-94 ~]# git clone https://github.com/sanjayguruji/registration-app-sanjaya.git
Cloning into 'registration-app-sanjaya'...
remote: Enumerating objects: 129, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 129 (delta 32), reused 29 (delta 28), pack-reused 87
Receiving objects: 100% (129/129), 12.41 MiB | 13.73 MiB/s, done.
Resolving deltas: 100% (35/35), done.
[root@ip-172-31-9-94 ~]# ll
total 0
drwxr-xr-x  5 root root 179 Jul  9 04:12 registration-app-sanjaya
[root@ip-172-31-9-94 ~]# cd registration-app-sanjaya
[root@ip-172-31-9-94 registration-app-sanjaya]# ll
total 14448
-rw-r--r-- 1 root root 130 Jul  9 04:12 Dockerfile
-rw-r--r-- 1 root root 29 Jul  9 04:12 README.md
-rw-r--r-- 1 root root 14768771 Jul  9 04:12 Snjay_Devops_projetc.pdf
-rw-r--r-- 1 root root 6333 Jul  9 04:12 pom.xml
-rw-r--r-- 1 root root 488 Jul  9 04:12 regapp-deploy.yml
-rw-r--r-- 1 root root 196 Jul  9 04:12 regapp-service.yml
drwxr-xr-x  3 root root 32 Jul  9 04:12 server
drwxr-xr-x  3 root root 32 Jul  9 04:12 webapp
[root@ip-172-31-9-94 registration-app-sanjaya]# rm -rf Snjay_Devops_projetc.pdf
[root@ip-172-31-9-94 registration-app-sanjaya]# ll
total 24
-rw-r--r-- 1 root root 130 Jul  9 04:12 Dockerfile
-rw-r--r-- 1 root root 29 Jul  9 04:12 README.md
-rw-r--r-- 1 root root 6333 Jul  9 04:12 pom.xml
-rw-r--r-- 1 root root 488 Jul  9 04:12 regapp-deploy.yml
-rw-r--r-- 1 root root 196 Jul  9 04:12 regapp-service.yml

```

Cloned

```

root@ip-172-31-9-94:~/project ~ + 
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:Eb6r+jLMTQ6zSiZQx1jyRUCqZ+QpRI6+vg24smIw0U8 root@ip-172-31-9-94.us-east-2.compute.internal
The key's randomart image is:
+---[RSA 3072]---+
| ..+.*+
| X + =
| + B * o
| o . = o
| ... o S
|=oE. B .
|o+= o o +
|o+.* = +
|*.o+.*+
+---[SHA256]---+
[root@ip-172-31-9-94 registration-app-sanjaya]# cd
[root@ip-172-31-9-94 ~]# cd .ssh/
[root@ip-172-31-9-94 .ssh]# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAgQcK2Tg98MfudL9HRzWybZhG46r2MvFicPnsrHy4cZ6hDknb70eWhmgmzWrJAn6jKPLRfxTSdlncoDX07wL7cs35
VzR9HdCyAlygT53FW/S+pwHC6//Xaxc1px/sJpInja1MPdLx3jb59w++hxokJgfqi3jnxEKGA2bVrKSoE/AIpcyNuGYsa7xR7/W/Iz3ekDqjYvZ71xpjhE8IVn/
WwVVkwikFGw+oWFYf7B6l78WP0niAVtV0JcLZEwdJWQ148r2FTC65D89eCSSIXnjrijnPIhM0gXRFr+/sSn9uWnxsDnPYEh1uPSK+3PSTiRm3nU8hnuZCRpw6F
HidRgBSLFMytU8qalilYgnXTx0n19S5aLA9pRsBc7xPjzL8JS8gN97Yaz4pOBHEhiudNNoEZi0tQ+m9TXLzILHpmXLptrxp7u9ltFycotudCVVT8CDVBHwJZAR
1vt9FepusSr0EGky+ByLS1+I9h0sxRE7eMB8wn0vCkdPhuefj9nBj0= root@ip-172-31-9-94.us-east-2.compute.internal
[root@ip-172-31-9-94 ~]# cd
[root@ip-172-31-9-94 ~]# git status
On branch master

No commits yet

```

Add this ssh in Github->Settings-> ssh

Git remote add origin git@github.com:Prasad-LTIMindtree/project-maven.git

Here the remote origin has been given.

## STEP 1: Building CI/CD Pipeline System

1. Create an machine for Jenkins (Note: If you are using ec2 kindly use medium machine to avoid deadlocks )
2. Install Java:
  - a. dnf update
  - b. dnf install java-17-amazon-corretto -y
  - c. java -version
  - d. wget -O /etc/yum.repos.d/jenkins.repo \u2014<https://pkg.jenkins.io/redhat-stable/jenkins.repo>
  - e. rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key>

```

Complete!
[root@ip-172-31-39-241 ~]# wget -O /etc/yum.repos.d/jenkins.repo \https://pkg.jenkins.io/redhat-stable/jenkins.repo
--2024-09-19 04:44:48-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.154.133, 2a04:4e42:24::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|151.101.154.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 85
Saving to: '/etc/yum.repos.d/jenkins.repo'

/etc/yum.repos.d/jenkins.repo    100%[=====] 85 --.-KB/s    in 0s

2024-09-19 04:44:48 (5.47 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

[root@ip-172-31-39-241 ~]# rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
[root@ip-172-31-39-241 ~]#

```

3. Install Jenkins

- a. dnf install jenkins -y
- b. systemctl enable jenkins
- c. systemctl start jenkins
- d. [http://your\\_amazon\\_linux\\_instance\\_ip:8080](http://your_amazon_linux_instance_ip:8080)

Note: Enable tcp port 8080 on your linux machine security groups inbound rules

```
[root@ip-172-31-39-241 ~]# dnf install jenkins -y
Jenkins-stable
Last metadata expiration check: 0:00:01 ago on Thu Sep 19 04:46:57 2024.
Dependencies resolved.
=====
Transaction Summary
=====
Installing:
jenkins.noarch 2.462.2-1.1 jenkins 89 M
Total download size: 89 M
Installed size: 89 M
Downloading Packages:
jenkins-2.462.2-1.1.noarch.rpm 49% [=====] 8.6 MB/s | 44 MB 00:05 ETA
```

## Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue

To unlock Jenkins copy the path highlighted in red

**cd /var/lib/Jenkins/secrets/initialAdminPassword**

```
[root@ip-172-31-39-241 ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
b1d72a0a94774796b1537d9a2e794f32
```

Create new Username and Password and your jenkins is ready

Getting Started

# Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Jenkins 2.462.2

## 4. Configure jenkin

To configure jenkin follow following steps

- Go to configure

Dashboard > Prasad Tale > Configure

- Go to API->Add New Token

### API Token

Current token(s) ?

There are no registered tokens for this user.

[Add new Token](#)

### Generate

Copy is only supported with a secure (HTTPS) connection

Token created on 2024-09-19T05:09:44.4

11ca9bf04bdac5719d8676e057517414d2



**⚠️ Copy this token now, because it cannot be recovered in the future.**

[Add new Token](#)

- c. To connect git and github repo with jenkins copy this secret key and paste this under webhook section which is under setting of repo

**Webhooks / Add webhook**

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

**Payload URL \***  
http://43.205.233.68:8080/github-webhook/

**Content type \***  
application/json

**Secret**  
11ca9bf04bdac5719d8676e057517414d2

**SSL verification**  
 By default, we verify SSL certificates when delivering payloads.  
 Enable SSL verification    Disable (not recommended)

**Which events would you like to trigger this webhook?**

Add jenkin url in Payload Url

In Context type select application/json

## 5. Install Maven Plugin

```
wget https://dlcdn.apache.org/maven/maven-3/3.9.9/binaries/apache-maven-3.9.9-bin.tar.gz
    7:01-- https://dlcdn.apache.org/maven/maven-3/3.9.9/binaries/apache-maven-3.9.9-bin.tar.gz
      apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
      dn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
      , awaiting response... 200 OK
      8.7M [application/x-gzip]
      e-maven-3.9.9-bin.tar.gz'

9-bin.tar.gz      100%[=====]     8.68M
02 (104 MB/s) - 'apache-maven-3.9.9-bin.tar.gz' saved [9102945/9102945]
```

# Plugins

 Updates

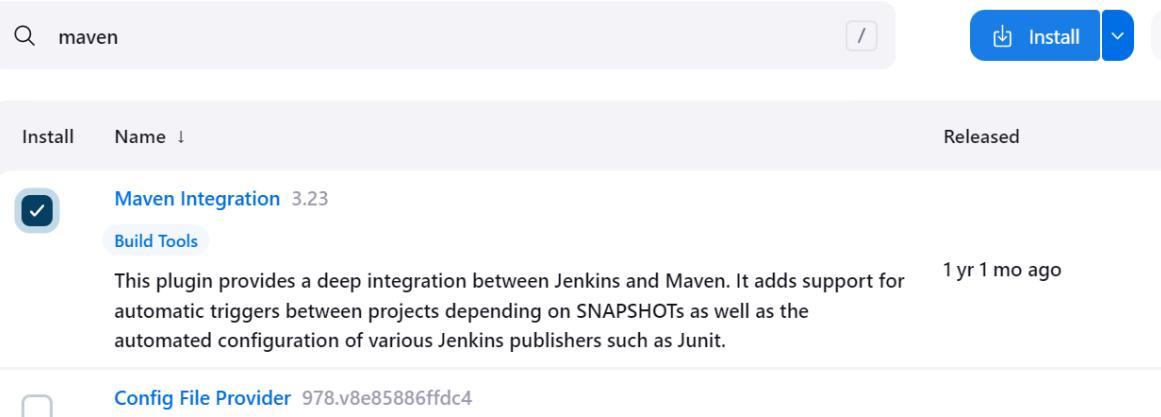
 Available plugins

 Installed plugins

 Advanced settings

 Download progress

On jenkin dashboard ->Manage Jenkin->Manage Jenkin->Plugins->Available Plugins



The screenshot shows the Jenkins Plugins page with a search bar containing "maven". A blue "Install" button is visible. The results table has columns for "Install", "Name ↓", and "Released". The first result is "Maven Integration 3.23" under "Build Tools", which is checked. The description states: "This plugin provides a deep integration between Jenkins and Maven. It adds support for automatic triggers between projects depending on SNAPSHOTs as well as the automated configuration of various Jenkins publishers such as Junit." The last result shown is "Config File Provider 978.v8e85886ffdc4".

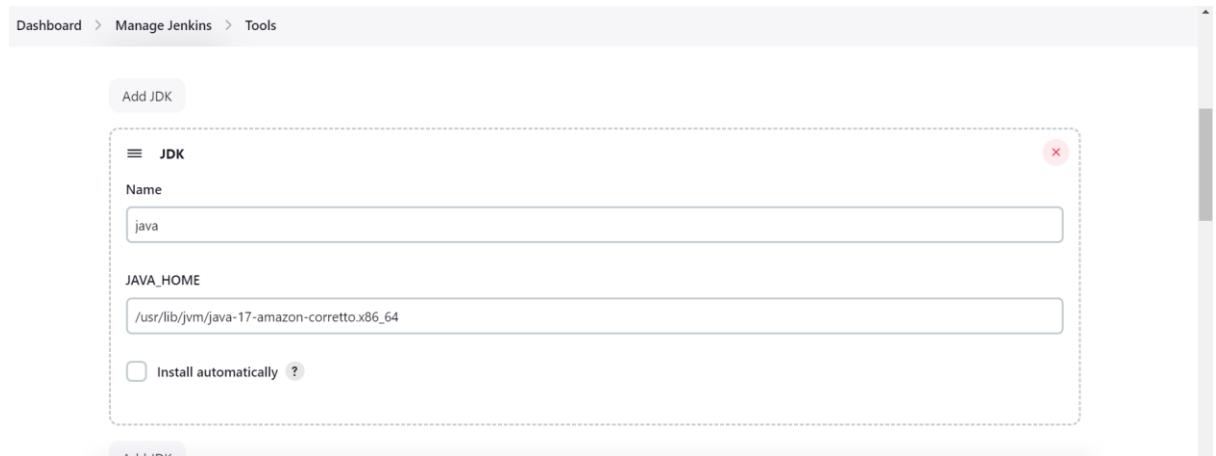
Install	Name ↓	Released
<input checked="" type="checkbox"/>	Maven Integration 3.23 Build Tools	1 yr 1 mo ago
<input type="checkbox"/>	Config File Provider 978.v8e85886ffdc4	

Type mvn -v on command prompt

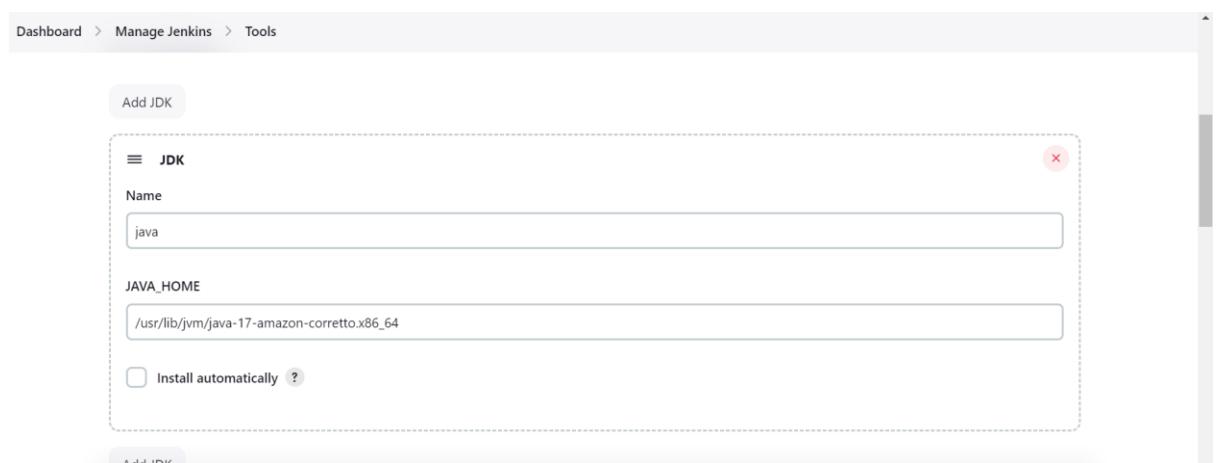
```
Apache Maven 3.8.4 (Red Hat 3.8.4-3.amzn2023.0.5)
Maven home: /usr/share/maven
Java version: 17.0.12, vendor: Amazon.com Inc., runtime: /usr/lib/jvm/java-17-amazon-corretto.x86_64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.109-118.189.amzn2023.x86_64", arch: "amd64", family: "unix"
[root@jenkins ~]# |
```

Copy java and mvn path

Paste it in Jenkins tools



The screenshot shows the Jenkins 'Manage Jenkins > Tools' page. A new JDK named 'java' is being added. The 'Name' field contains 'java', and the 'JAVA\_HOME' field contains '/usr/lib/jvm/java-17-amazon-corretto.x86\_64'. There is also an unchecked checkbox for 'Install automatically'.



The second screenshot is identical to the first, showing the 'Add JDK' configuration for 'java' with the same fields and settings.

## 6. creating a job

Enter an item name

test

Select an item type



Freestyle project

Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as

Select the following option

Git

Repositories

Repository URL

https://github.com/Prasad-LTIMindtree/project-maven.git

Credentials

- none -

+ Add

Advanced

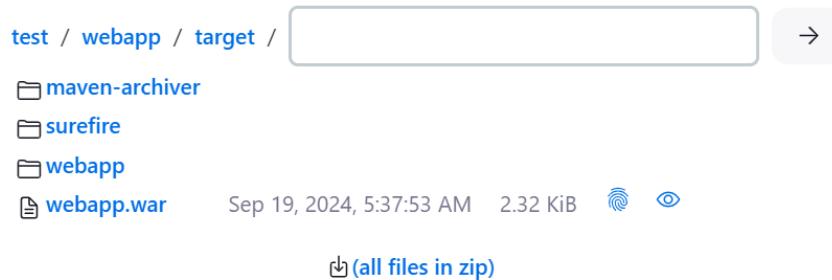
Once this is done You can start a build

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	test	4 min 23 sec #1	N/A	20 sec

Once the build is scuccessfull you can see the gren tick

For info click on test

## Workspace of test on Built-In Node



Inside test/webapp/target you will see the build file

Note: This is the manual build to build automatically select poll scm while creating an job so that after every push/commit it will automatically build new war

## 7. Tomcat Installation

a). get the .tar file from tomcat official website

```
wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.95/bin/apache-tomcat-9.0.95.tar.gz
```

```
[root@ip-172-31-11-237 ~]# wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.95/bin/apache-tomcat-9.0.95.tar.gz
--2024-09-19 08:52:27-- https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.95/bin/apache-tomcat-9.0.95.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12715996 (12M) [application/x-gzip]
Saving to: 'apache-tomcat-9.0.95.tar.gz'

apache-tomcat-9.0.95.tar.gz      100%[=====] 12.13M  --.-KB/s   in 0.08s
2024-09-19 08:52:27 (151 MB/s) - 'apache-tomcat-9.0.95.tar.gz' saved [12715996/12715996]

[root@ip-172-31-11-237 ~]# ll
total 12M
```

b.) Install Java

c.)go to tomcat file

cd tomcat/bin

run ./startup.sh

```
[root@ip-172-31-11-237 bin]# ./startup.sh
Using CATALINA_BASE:   /root/tomcat
Using CATALINA_HOME:   /root/tomcat
Using CATALINA_TMPDIR: /root/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /root/tomcat/bin/bootstrap.jar:/root/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
```

Now you tomcat is live .Copy your public and paste it in browser

[Home](#) [Documentation](#) [Configuration](#) [Examples](#) [Wiki](#) [Mailing Lists](#)

[Find Help](#)

## Apache Tomcat/9.0.95

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:  
[Security Considerations How-To](#)  
[Manager Application How-To](#)  
[Clustering/Session Replication How-To](#)

[Server Status](#)

[Manager App](#)

[Host Manager](#)

### Developer Quick Start

[Tomcat Setup](#)  
[First Web Application](#)

[Realms & AAA](#)  
[JDBC DataSources](#)

[Examples](#)

[Servlet Specifications](#)  
[Tomcat Versions](#)

### Managing Tomcat

For security, access to the manager webapp is

### Documentation

[Tomcat 9.0 Documentation](#)

### Getting Help

[FAQ and Mailing Lists](#)

d.) Now go to conf dir in tomcat

and do the following

vim tomcat-users.xml

Add following roles

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

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

ii.)Comment following in context.xml

```
find -name context.xml
```

```
./webapps/host-manager/META-INF/context.xml  
./webapps/manager/META-INF/context.xml
```

Edit this two files

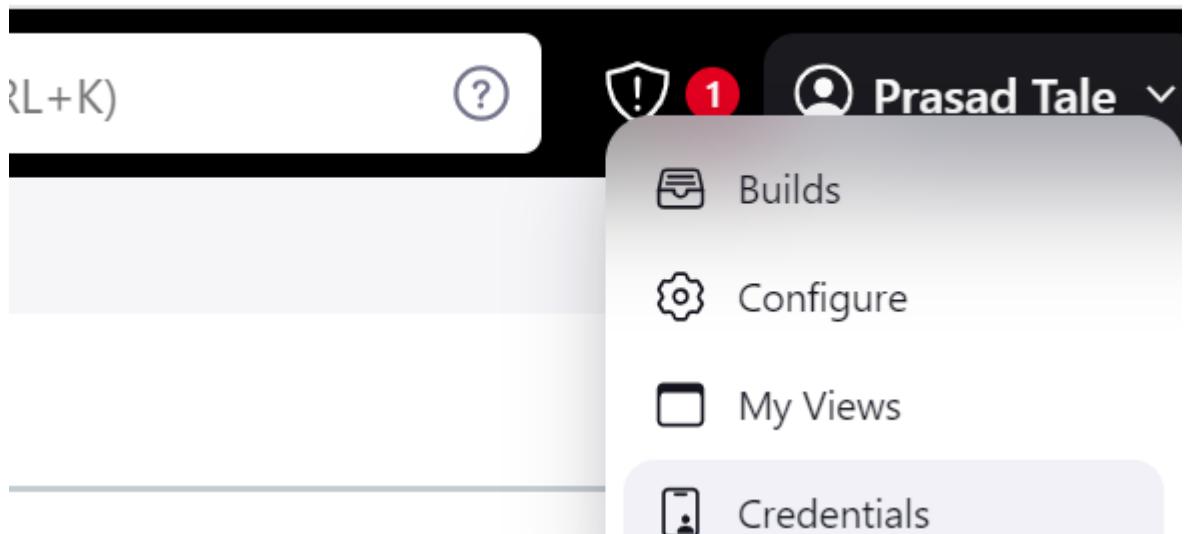
```
<Valve className="org.apache.catalina.valves.RemoteAddrValve"  
       allow="127\ \d+\ \d+\ \d+\.\cdot\1\0\.\0\.\0\.\0\.\0\.\1" />
```

Comment this value tag

Shutdown and startup tomcat to apply

```
[root@ip-172-31-11-237 bin]# ./shutdown.sh  
Using CATALINA_BASE:  /root/tomcat  
Using CATALINA_HOME:  /root/tomcat  
Using CATALINA_TMPDIR: /root/tomcat/temp  
Using JRE_HOME:      /usr  
Using CLASSPATH:     /root/tomcat/bin/bootstrap.jar:/root/tomcat/bin/tomcat-juli.jar  
Using CATALINA_OPTS:  
NOTE: Picked up JDK_JAVA_OPTIONS: --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.i  
java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.util.concurrent=ALL-UNNAMED --add-opens=java.rmi  
AMED  
[root@ip-172-31-11-237 bin]# ./startup.sh  
Using CATALINA_BASE:  /root/tomcat  
Using CATALINA_HOME:  /root/tomcat  
Using CATALINA_TMPDIR: /root/tomcat/temp  
Using JRE_HOME:      /usr  
Using CLASSPATH:     /root/tomcat/bin/bootstrap.jar:/root/tomcat/bin/tomcat-juli.jar  
Using CATALINA_OPTS:  
Tomcat started.
```

Now you have successfully configured tomcat



Now go to Credentials->Create new credentials

A screenshot of a "Create new credentials" form. The form has four fields: "Username" (with value "admin"), "Password" (with value "....."), "ID" (with value "admin"), and "Description". The "Description" field is currently selected, indicated by a blue underline. There is also a checkbox labeled "Treat username as secret".

Note: username and password should be

A screenshot of a search results page. A search bar at the top contains the text "deploy". Below the search bar, there is a list of plugins. The first plugin listed is "Deploy to container Plugin 1.16", which is described as allowing deployment to a container after a successful build. It is marked as "Enabled" and has a toggle switch next to it. There is also a red "X" icon to the right of the plugin name.

## New Item

Enter an item name

tom-job

Select an item type



Freestyle project

Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



Maven project

Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

OK

Git ?

Repositories ?

Repository URL ?

https://github.com/Prasad-LTIMindtree/project-maven.git



Credentials ?

- none -

+ Add ▾

Advanced ▾

POST-BUILD ACTIONS

☰ Deploy war/ear to a container



WAR/EAR files ?

\*\*/\*.war

Context path ?

Containers

Containers

**Tomcat 8.x Remote**

Credentials

admin/\*\*\*\*\*\*/ (tomcat\_server)

+ Add ▾

Tomcat URL ?

http://43.204.230.247:8080

Advanced ▾

## Console Output

Download

Copy

View as plain text

```

Started by user Prasad Tale
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/tom-job
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Prasad-LTIMindtree/project-maven.git
> git init /var/lib/jenkins/workspace/tom-job # timeout=10
Fetching upstream changes from https://github.com/Prasad-LTIMindtree/project-maven.git
> git --version # timeout=10
> git --version # 'git version 2.40.1'
> git fetch --tags --force --progress -- https://github.com/Prasad-LTIMindtree/project-maven.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/Prasad-LTIMindtree/project-maven.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10

```

Got to <tomcat\_url>:8080

The screenshot shows the Tomcat Manager application interface. At the top, there are tabs for 'List Applications', 'HTML Manager Help', 'Manager Help', and 'Server Status'. Below the tabs is a table titled 'Applications' with columns: Path, Version, Display Name, Running, Sessions, and Commands. The table lists several applications:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/webapp	None specified	Webapp	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

A yellow bar at the bottom contains the text 'Deploy'.

Go to /webapp

# New user Registration in Devops Project

Please fill in this form to create an account.

Enter Name	<input type="text" value="Enter Full Name"/>
Enter mobile Number	<input type="text" value="Enter mobile number"/>
Enter Email	<input type="text" value="Enter Email"/>
Password	<input type="password" value="Enter Password"/>
Repeat Password	<input type="password" value="Repeat Password"/>

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

[Register](#)

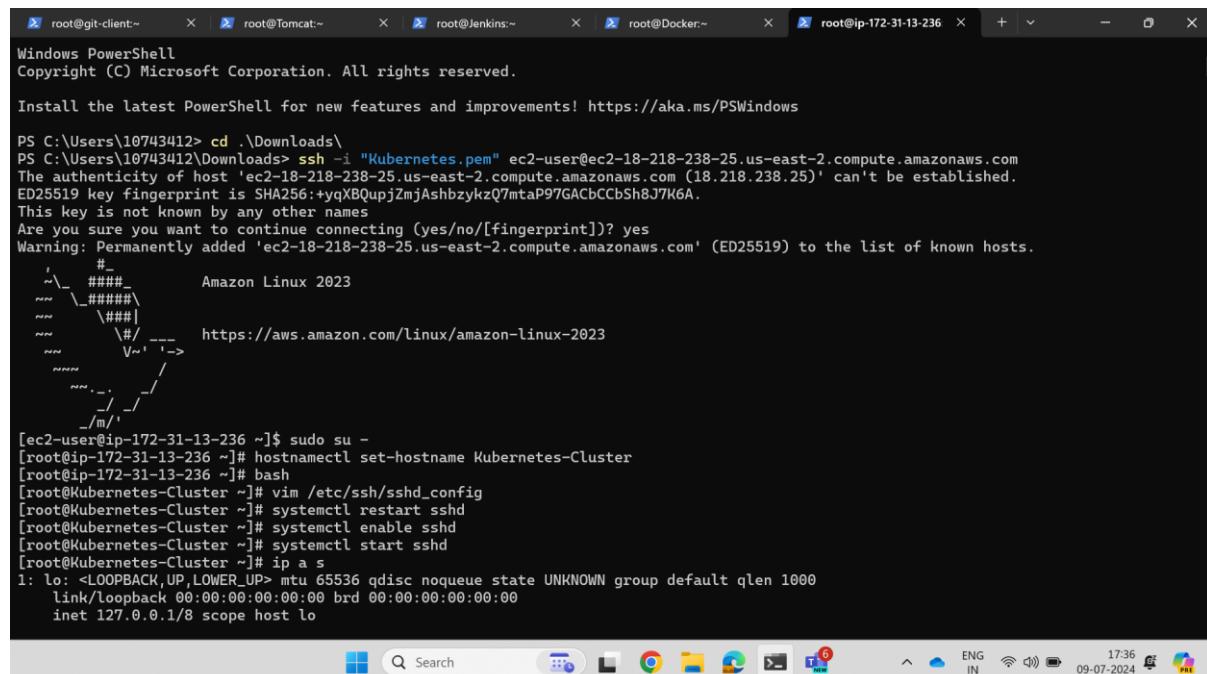
Already have an account? [Sign in](#).

## Thank You, Happy Learning

### See You Again

Project is Running

## 8.Docker Configuration with Jenkins



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

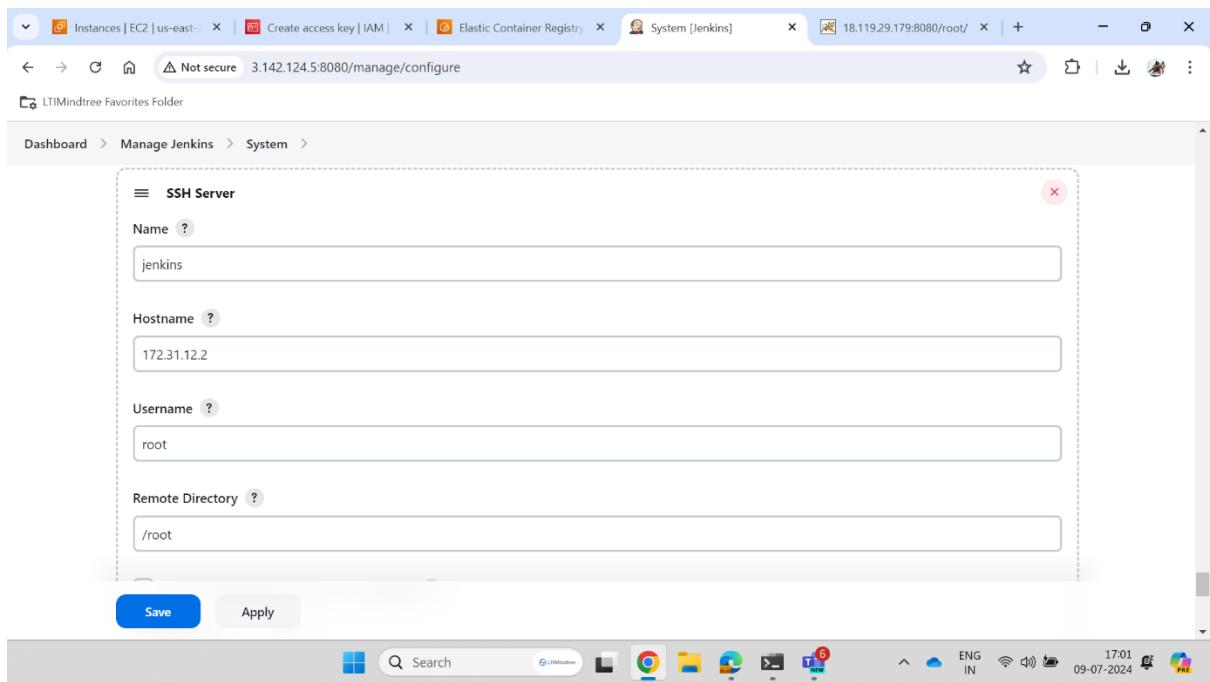
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\10743412> cd .\Downloads\
PS C:\Users\10743412\Downloads> ssh -i "Kubernetes.pem" ec2-user@ec2-18-218-238-25.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-218-238-25.us-east-2.compute.amazonaws.com (18.218.238.25)' can't be established.
ED25519 key fingerprint is SHA256:+yqXBQupjZmjAshbzykzQ7mtaP97GACbCCbSh8J7K6A.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-218-238-25.us-east-2.compute.amazonaws.com' (ED25519) to the list of known hosts.

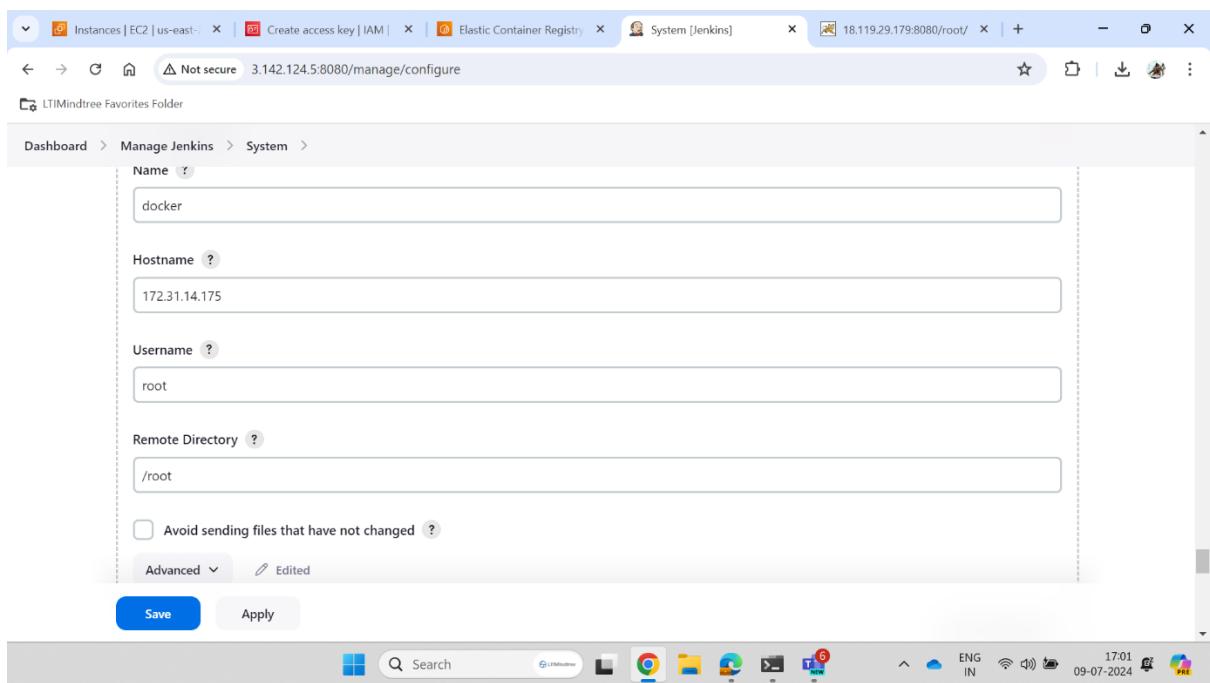
#_          Amazon Linux 2023
\_###_
\###|_
\##|_
\#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
 \~'-'>
   /_
  /_/
 _/m'

[ec2-user@ip-172-31-13-236 ~]$ sudo su -
[root@ip-172-31-13-236 ~]# hostnamectl set-hostname Kubernetes-Cluster
[root@ip-172-31-13-236 ~]# bash
[root@Kubernetes-Cluster ~]# vim /etc/ssh/sshd_config
[root@Kubernetes-Cluster ~]# systemctl restart sshd
[root@Kubernetes-Cluster ~]# systemctl enable sshd
[root@Kubernetes-Cluster ~]# systemctl start sshd
[root@Kubernetes-Cluster ~]# 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
        
```

Sshd is restarted after giving the permissions.



Jenkins has been connected with the SSH Server in Jenkins.



Docker has been connected with SSH Server in Jenkins.

The screenshot shows the Jenkins configuration interface for the 'Milestone-2' job. On the left, a sidebar lists various configuration sections: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. The 'Post-build Actions' section is currently selected and highlighted in grey. In the main panel, under 'Tomcat 9.x Remote', the 'WAR/EAR files' field contains the pattern '\*\*/\*.war'. The 'Context path' field is set to '/root'. Under the 'Containers' section, there is a single entry named 'Tomcat 9.x Remote' which includes a 'Credentials' dropdown set to 'deployer/\*\*\*'. Below this, the 'Tomcat URL' field is empty. At the bottom of the configuration panel are 'Save' and 'Apply' buttons.

Here .war file is created for tomcat.

The screenshot shows the Jenkins configuration interface for the 'Milestone-2' job. The 'Post-build Actions' section is selected. A new action has been added, showing the 'rsync -avh /var/lib/jenkins/workspace/Milestone-2/\* root@172.31.14.175:/opt' command in the 'Exec command' field. This command is used to sync the Jenkins workspace to a remote Docker host. The 'Advanced' dropdown menu is visible below the exec command field. The rest of the configuration sidebar and toolbars are visible at the top and bottom of the screen.

In project configuration we are connecting Jenkins with docker by providing docker IP.

Now apply deployment manually first from eks machine

**Kubectl apply -f deployment.yml**

**Kubectl apply -f service.yml**

The screenshot shows the Jenkins configuration interface for a job named 'Milestone-2'. On the left, a sidebar lists various configuration sections: General, Source Code Management, Build Triggers, Build Environment, Pre Steps, Build, Post Steps, Build Settings, and Post-build Actions. The 'Post-build Actions' section is currently selected. In the main panel, there are three configuration fields: 'Remote directory', 'Exec command', and 'Advanced'. The 'Exec command' field contains the following Jenkinsfile code:

```
cd /opt
aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin
public.ecr.aws/d4u9f1m1
docker build -t aryan10743412 .
docker tag aryan10743412:latest public.ecr.aws/d4u9f1m1/aryan10743412:latest
```

A note below the command states: "All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)". At the bottom of the configuration panel are 'Save' and 'Apply' buttons.

This commands are taken from ecr ->View push commands

The screenshot shows the Jenkins build history for the 'Milestone-2' job. The left sidebar includes options like Configure, Delete Maven project, Modules, Git Polling Log, and Rename. The main area displays the 'Build History' table, which lists the following builds:

#	Build Number	Date
1	#4	Jul 9, 2024, 11:29 AM
2	#3	Jul 9, 2024, 11:25 AM
3	#2	Jul 9, 2024, 9:02 AM
4	#1	Jul 9, 2024, 8:27 AM

To the right of the build history, there is a 'Permalinks' section with a 'Latest test result' link and a green rectangular placeholder for the result. A note at the bottom of the build history table says: "Last completed build (#4), 4 min 18 sec ago".

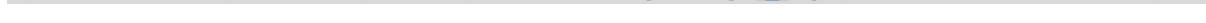
The screenshot shows the AWS ECR console interface. On the left, a sidebar navigation includes 'Amazon Elastic Container Registry', 'Private registry', 'Public registry' (selected), and 'ECR public gallery'. The main content area displays a message: 'Image scan overview, status, and full vulnerabilities has moved to the Image detail page. To access, click an image tag.' Below this, the repository name 'aryan10743412' is shown, along with 'View public listing', 'View push commands', and 'Edit' buttons. A table titled 'Images (1)' lists the single image tag: 'latest' (Image type, pushed at July 09, 2024, 17:00:02 UTC+05:5, size 220.92 MB, Image URI sha256:69e6252...). The bottom of the screen shows the Windows taskbar with various icons.

Here the image has been created in the docker and is stored inside ecr

## 9. Kubernetes EKS :

The screenshot shows the AWS CloudWatch Metrics Insights interface. It displays a list of log entries for the 'eks-manager' instance. One entry is highlighted, showing the instance ID 'i-0411f62359791a4af' and the text 'eks-manager'. Below the list, there is a summary section for the instance 'i-0411f62359791a4af (eks-manager)' with tabs for 'Details', 'Status and alarms', 'Monitoring', 'Security', 'Networking', 'Storage', and 'Tags'. At the bottom, there is a link to 'Instance summary'.

Instance for Kubernetes is created.



```
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\10743412> cd .\Downloads\
PS C:\Users\10743412\Downloads> ssh -i "Kubernetes.pem" ec2-user@ec2-18-218-238-25.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-218-238-25.us-east-2.compute.amazonaws.com (18.218.238.25)' can't be established.
ED25519 key fingerprint is SHA256:+yqXBQupjZmjAshbzykzQ7mtaP97GACbCCbSh8J7K6A.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-218-238-25.us-east-2.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-172-31-13-236 ~]$ sudo su -
[root@ip-172-31-13-236 ~]# hostnamectl set-hostname Kubernetes-Cluster
[root@ip-172-31-13-236 ~]# bash
[root@Kubernetes-Cluster ~]# |
```

Instance is connected in the Terminal



```
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:kWxs7l0576YIjtVEeNgKAInn/uai0HYTcyIin17CmI root@Docker
The key's randomart image is:
+---[RSA 3072]---+
|o.. .o.
|.=. .+..
|. +. B ..
|o+ + +...
|.+ o S.o
|.. o . . .
|oE. o ..o .
|.+ + oo ....
|.. +..+ ...*
+---[SHA256]---+
[root@Docker ~]# ssh-copy-id root@172.31.13.236
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.13.236' ('172.31.13.236') can't be established.
ED25519 key fingerprint is SHA256:+yqXBQupjZmjAshbzykzQ7mtaP97GACbCCbSh8J7K6A.
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@172.31.13.236's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.13.236'"
and check to make sure that only the key(s) you wanted were added.

[root@Docker ~]# |
```

Ssh key is generated and connected with the docker instance.

```

2024-09-21 15:47:23 [!] 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'
2024-09-21 15:47:23 [i] creating addon
2024-09-21 15:47:23 [i] successfully created addon
2024-09-21 15:49:24 [i] waiting for the control plane to become ready
2024-09-21 15:49:24 [✓] saved kubeconfig as "/root/.kube/config"
2024-09-21 15:49:24 [i] no tasks
2024-09-21 15:49:24 [✓] all EKS cluster resources for "my-cluster" have been created
2024-09-21 15:49:24 [✓] created 0 nodegroup(s) in cluster "my-cluster"
2024-09-21 15:49:25 [✓] created 0 managed nodegroup(s) in cluster "my-cluster"
2024-09-21 15:49:25 [i] kubectl command should work with "/root/.kube/config", try 'kubectl get nodes'
2024-09-21 15:49:25 [✓] EKS cluster "my-cluster" in "ap-south-1" region is ready
[root@ip-172-31-9-208 ~]# eksctl create nodegroup --cluster my-cluster --region ap-south-1 --name project-groups --node-ami-family Ubuntu2004 --node-type t2.small --subnet-ids subnet-0b49391b36fd53f89,subnet-05598a5487cefff3b --nodes 3 --nodes-min 2 --nodes-max 4 --ssh-access --ssh-public-key /root/.ssh/id_rsa.pub
2024-09-21 15:50:05 [i] will use version 1.29 for new nodegroup(s) based on control plane version
2024-09-21 15:50:06 [i] nodegroup "project-groups" will use "ami-0ab6dcbf35da05038" [Ubuntu2004/1.29]
2024-09-21 15:50:06 [i] using SSH public key "/root/.ssh/id_rsa.pub" as "eksctl-my-cluster-nodegroup-project-groups-1e:fc:fa:30:bb:0:ee:ac:9e:f8:c4:c2:40:6a:46:91"
2024-09-21 15:50:06 [i] 1 nodegroup (project-groups) was included (based on the include/exclude rules)
2024-09-21 15:50:06 [i] will create a CloudFormation stack for each of 1 managed nodegroups in cluster "my-cluster"
2024-09-21 15:50:06 [i] 2 sequential tasks: { fix cluster compatibility, 1 task: { 1 task: { create managed nodegroup "project-groups" } } }
2024-09-21 15:50:06 [i] checking cluster stack for missing resources
2024-09-21 15:50:07 [i] cluster stack has all required resources
2024-09-21 15:50:07 [i] building managed nodegroup stack "eksctl-my-cluster-nodegroup-project-groups"
2024-09-21 15:50:07 [i] deploying stack "eksctl-my-cluster-nodegroup-project-groups"
2024-09-21 15:50:07 [i] waiting for CloudFormation stack "eksctl-my-cluster-nodegroup-project-groups"

```

## Nodegroups and Cluster is created

Create two files

Deployment.yml

Service.yml

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: app-webapp
  labels:
    app: webapp
spec:
  replicas: 2
  selector:
    matchLabels:
      app: webapp
  template:
    metadata:
      labels:
        app: webapp
    spec:
      containers:
        - name: webapp
          image: public.ecr.aws/d4u9f1ml/aryan10743412:latest/webapp
          imagePullPolicy: Always
          ports:
            - containerPort: 8080

```

"deployment.yml" 23L, 414B

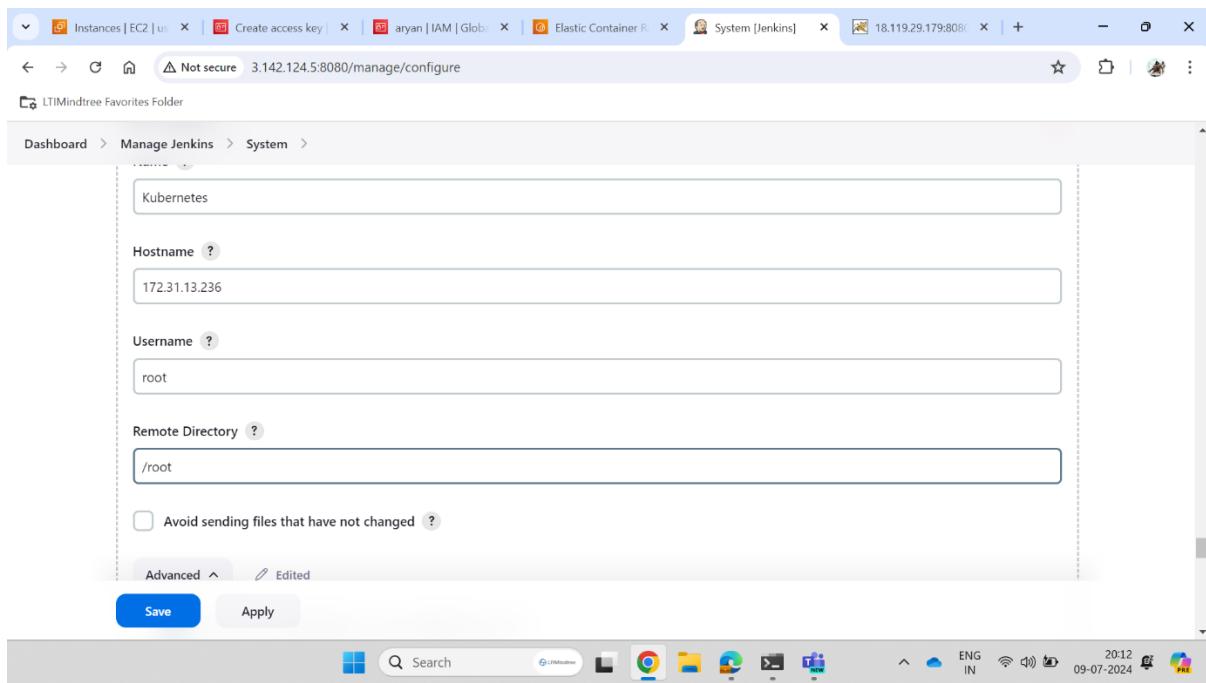
6,4 All

Windows Taskbar icons: File Explorer, Edge, Google Chrome, File Manager, PowerShell, Task View, Taskbar settings, Network, Battery, Volume, Date/Time (09-07-2024), Language (ENG IN), and a small icon.

```
apiVersion: v1
kind: Service
metadata:
  name: app-service
  labels:
    app: webapp
spec:
  selector:
    app: webapp
  ports:
    - port: 8080
      targetPort: 8080
  type: LoadBalancer
```

"service.yml" 15L, 192B

Content which has to be written in the service.yml



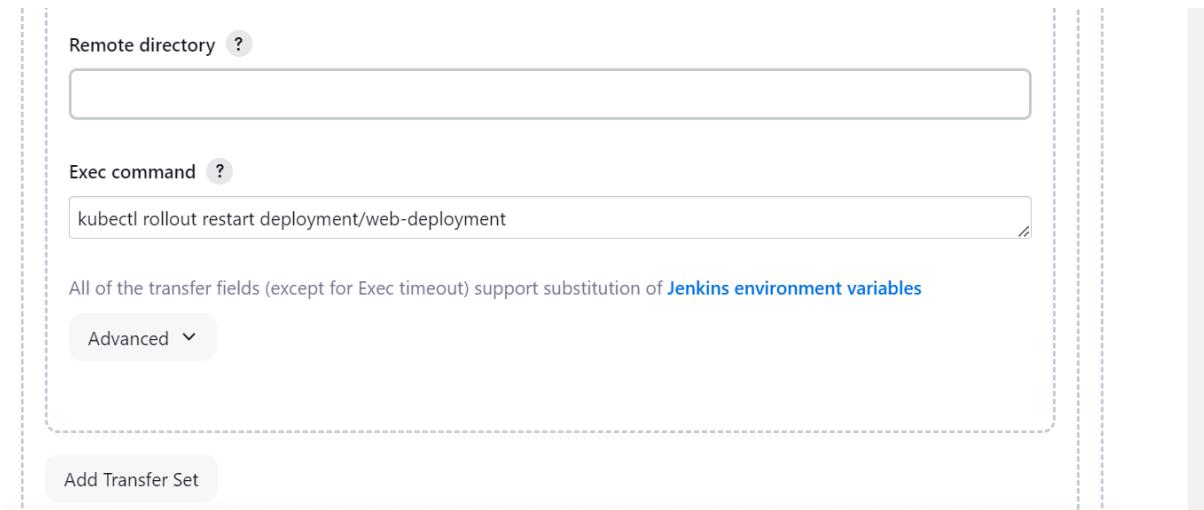
Kubernetes has been configured in the Jenkins.

Now run following command manually on eks machine

Kubectl apply -f deployment.yml

Kubectl apply -f service.yml

Now once the deployment is done



Add server in job -> and add this command

This command will make sure to update changed image from ecr whenever new build is committed

 Build Now

 Configure

P€

 Delete Maven project

 Modules

 GitHub Hook Log

 Rename



Build History

trend ▾

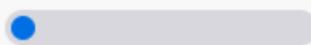
 Filter... 



#8



Sep 21, 2024, 4:53 PM



#7

The screenshot shows a build history interface with a sun icon. The title is "Build History" and there is a "trend" dropdown menu. A search bar says "Filter..." and has a placeholder "/". Below it, a green checkmark indicates "Build #8" and the date "Sep 21, 2024, 4:53 PM".

Build is successfull

```
[root@eks-manager ~]# kubectl get svc
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP
AGE
kubernetes   ClusterIP    10.100.0.1    <none>
71m
web-service  LoadBalancer  10.100.201.40  ac59dbbe3599a401d8e69a0cbd9c4d4b-99218763.ap-south-1.elb.amazonaws.com  8080:31797/TCP
54m
```

Copy the External ip and paste it in browser



## New user Registration in Devops Project

Please fill in this form to create an account.

Enter Name	<input type="text" value="Enter Full Name"/>
Enter mobile Number	<input type="text" value="Enter moible number"/>
Enter Email	<input type="text" value="Enter Email"/>
Password	<input type="password" value="Enter Password"/>
Repeat Password	<input type="password" value="Repeat Password"/>

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

[Register](#)

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

**Thank You, Happy Learning**

**See You Again**

Now the website is live

**Thank You**  
**-Prasad Tale**

