

Project 1

Building a CI/CD Pipeline for a Retail Company

Business Challenge/Requirement

ABC technologies is a leading online retail store. ABC has recently acquired a large retail offline business store. The business store has large number of stores across the globe but is following conventional pattern of development and deployment. As a result, it has landed to great loss and are facing below challenges.

- low available
- low scalable
- low Performance
- Hard built and maintained
- Developed and deployed is time consuming

ABC will acquire the data from all these storage systems and plan to use it for analytics and prediction of the firm's growth and sales prospect. In the first phase ABC has to create the servlets to Add a product and Display product details. Add servlet dependencies required to compile the servlets. Create an HTML page which will be used to add a product. Team is using git to keep all the source code.

ABC has decided to use DevOps model and Once source code is available in Github, we need to integrate it with Jenkins and provide continuous build generation for continuous Delivery, integrate with Ansible and Kubernetes for deployment. Use docker hub to pull and push images between ansible and Kubernetes. Implement CICD such that ABC Company.

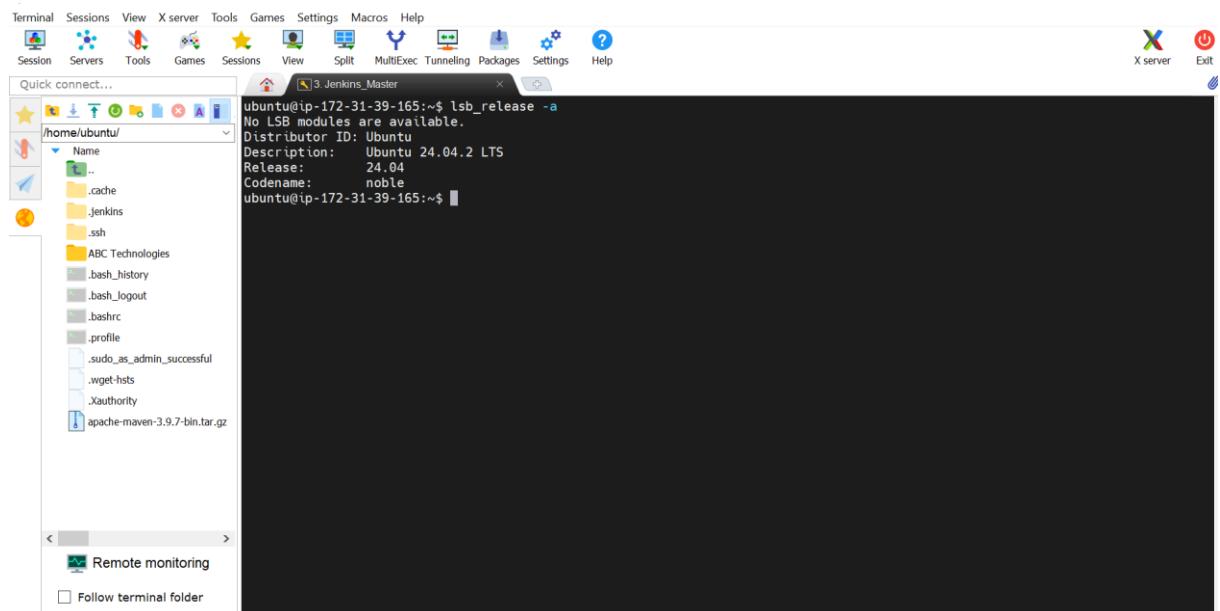
Verify following software is installed in the working machine:

- Java
- Maven
- Git
- Jenkins
- Docker
- Ansible
- Kubernetes
- Prometheus
- Grafana

https://github.com/semsra06/ABC_project1

I will follow “git config >> created project repo on github, initialise git and push the project on github>> Installed Jenkins on ec2 (Manually) and Through Ansible Play book >> Creating pipeline for CI on jenkins gui (Compile, test, package) and creating CI on jenkins with jenkinsfile written in groovy script >> Installation of K8s manually and with ansible playbook >> deployment the artifacts on K8s kluster>> monitoring tools Prometheus and grafana tools integration”

To find out which version of AWS Ubuntu is installed:



1. Java

```
sudo apt update && sudo apt upgrade -y
```

```
sudo apt install openjdk-21-jdk -y
```

```
java -version
```

Copy the path (e.g., /usr/lib/jvm/java-21-openjdk-amd64/) and set it in your environment variables:

```
echo 'export JAVA_HOME=/usr/lib/jvm/java-21-openjdk-amd64' >> ~/.bashrc
```

```
echo 'export PATH=$JAVA_HOME/bin:$PATH' >> ~/.bashrc
```

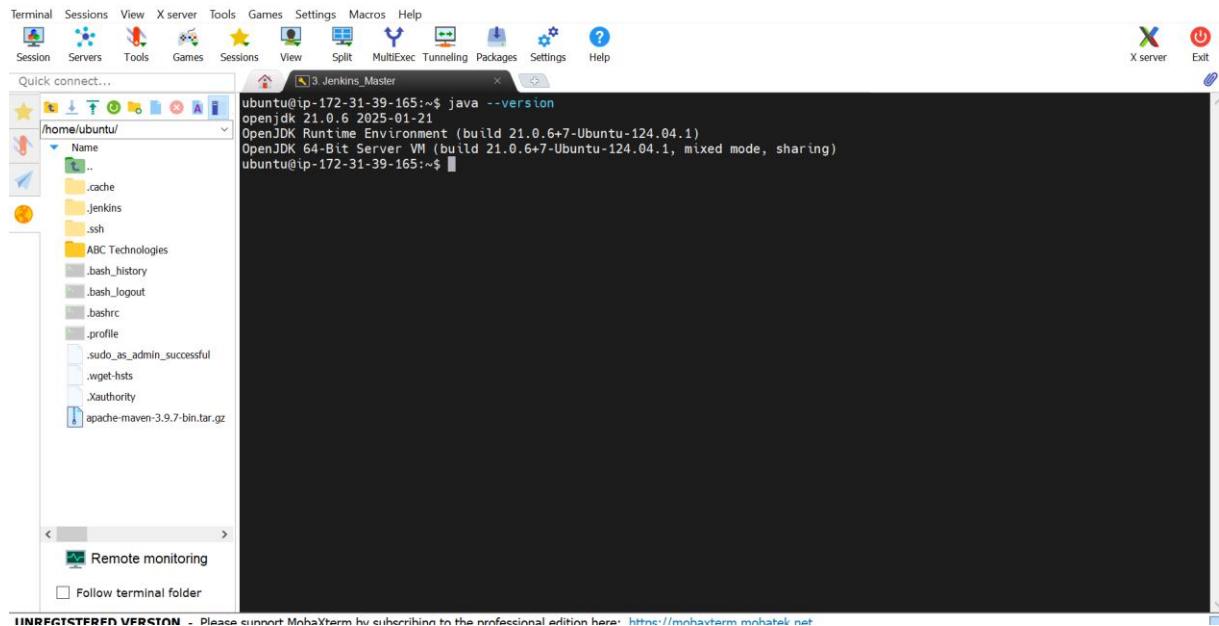
```
source ~/.bashrc
```

Verify:

```
echo $JAVA_HOME
```

```
> *** System restart required ***
Last login: Tue Mar 25 06:28:22 2025 from 176.240.124.183
ubuntu@ip-172-31-39-165:~$ echo $JAVA_HOME
/usr/lib/jvm/java-21-openjdk-amd64
ubuntu@ip-172-31-39-165:~$
```

I installed 21.0.6 version of Java.



2. Maven

Apache Maven is a popular build automation and dependency management tool for Java-based projects. Maven makes it easy to compile, test, package, and deploy code, especially for large-scale projects.

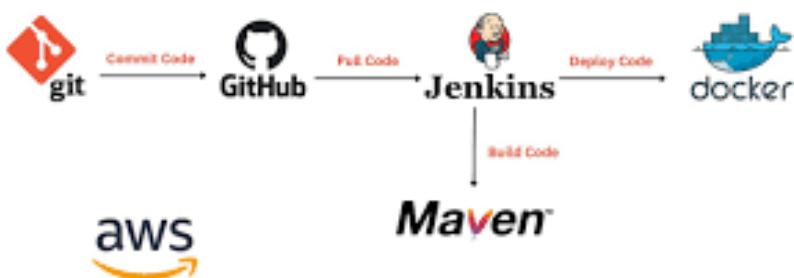
To install Apache Maven 3.9.7:

```
wget https://mirrors.estointernet.in/apache/maven/maven-3/3.9.7/binaries/apache-maven-3.9.7-bin.tar.gz  
sudo mkdir -p /opt/maven  
sudo tar -xvzf apache-maven-3.9.7-bin.tar.gz -C /opt/maven/ --strip-components=1  
sudo ln -s /opt/maven/bin/mvn /usr/bin/mvn  
mvn --version
```

Apache Maven 3.9.7

```
ubuntu@ip-172-31-39-165:~$ mvn --version
Apache Maven 3.9.7 (8b094c9513efc1b9ce2d952b3b9c8aeaaf8cbf0)
Maven home: /opt/maven
Java version: 21.0.6, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "Linux", version: "6.8.0-1024-aws", arch: "amd64", family: "unix"
ubuntu@ip-172-31-39-165:~$
```

3. Git



```
sudo apt update && sudo apt upgrade -y
sudo apt install git -y
git --version
```

```
ubuntu@ip-172-31-39-165:~$ echo $JAVA_HOME
/usr/lib/jvm/java-21-openjdk-amd64
ubuntu@ip-172-31-39-165:~$ git --version
git version 2.43.0
ubuntu@ip-172-31-39-165:~$
```

Go to Set User Information:

```
git config --global user.name "semra06"
git config --global user.email smrknnc@gmail.com
```

```
ubuntu@ip-172-31-39-165:~$ git config --list
ubuntu@ip-172-31-39-165:~$ git config --global user.name "semra06"
git config --global user.email smrknnc@gmail.com
ubuntu@ip-172-31-39-165:~$ git config --list
user.name=semra06
user.email=smrknnc@gmail.com
ubuntu@ip-172-31-39-165:~$
```

Clone Repository

In the terminal, change to the directory where you will download the project and run the following command(used HTTPS):

```
git clone https://github.com/semra06/ABC\_project1.git
```

Enter the Folder and Check Branches

```
cd ABC\ Technologies/
```

```
ubuntu@ip-172-31-39-165:~/cd ABC\ Technologies/
ubuntu@ip-172-31-39-165:~/ABC Technologies$ ls
README.md pom.xml pom.xml.bak src
ubuntu@ip-172-31-39-165:~/ABC Technologies$
```

To see all branches:

```
ubuntu@ip-172-31-39-165:~/ABC Technologies$ git branch -a
* main
  remotes/origin/main
ubuntu@ip-172-31-39-165:~/ABC Technologies$
```

Initialize a New Git Repository (git init)

```
git init
```

```
git add .
```

```
git commit -m "" --> local commit
```

```
git remote add origin https://github.com/semra06/ABC_project1.git
```

```
git remote -v
```

```
git push origin master
```

After all these processes, all documents were pushed to GitHub.

The screenshot shows the GitHub repository page for 'ABC_project1'. The repository is public and was created by 'semra06'. It contains 1 branch ('main') and 0 tags. The repository has 33 commits from 1860cf3 (2 minutes ago) to the first commit (first commit). The commits include adding Jenkinsfile, .settings, ABC_project1, src, .classpath, .project, Dockerfile, Jenkinsfile, README.md, pom.xml, and pom.xml.bak. The repository has 0 stars, 1 watching, and 0 forks. It also has 0 releases, 0 packages, and Java as the primary language (90.3%). A 'Suggested workflows' section is present at the bottom.

Github to make webhook setting.

Webhooks / Manage webhook

Payload URL *
http://51.21.162.117:8080/github-webhook/

Content type *
application/x-www-form-urlencoded

Secret

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?

Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ http://51.21.162.117:8080/github-w... (push)
Last delivery was successful.

Add webhook

4. Jenkins

```
sudo apt update && sudo apt upgrade -y
```

We will install by adding the official Jenkins repository.

Add GPG Key:

I added jenkins user as first

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt update
sudo apt install jenkins -y
```

Start Jenkins:

```
sudo systemctl start jenkins
```

To enable Jenkins to start automatically on every startup:

```
sudo systemctl enable jenkins
```

To check status of Jenkins:

```
sudo systemctl status jenkins
```

The screenshot shows a MobaXterm window titled "17. Master_Node". The terminal session displays the output of the command "sudo systemctl status jenkins". The Jenkins service is listed as loaded, active (running), and enabled. It has a Main PID of 1149919, running Java tasks, and consuming 699.2M of memory. The service is located in the /system.slice/jenkins.service cgroup. The log output shows Jenkins starting up and initializing its reactor runner. The terminal interface includes a sidebar with project files like pom.xml, pom.xml.bak, and README.md, and a bottom bar with "Remote monitoring" and "Follow terminal folder" checkboxes.

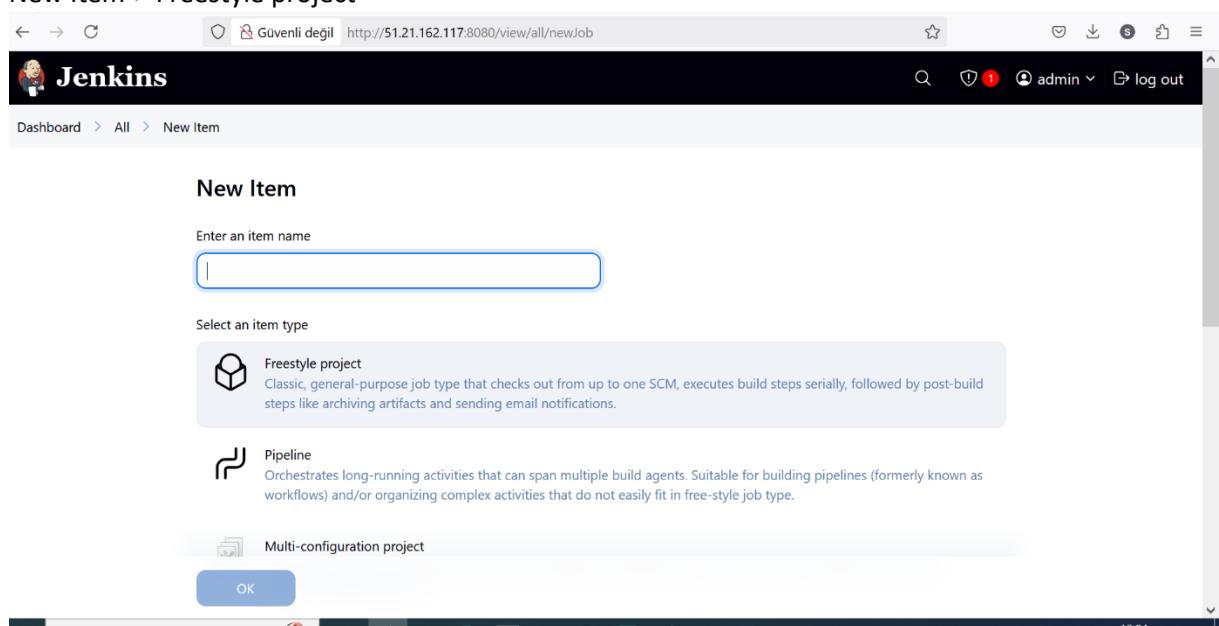
```
jenkins@master:~$ sudo systemctl status jenkins
[sudo] password for jenkins:
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-04-02 14:59:30 UTC; 33min ago
     Main PID: 1149919 (java)
       Tasks: 61 (limit: 9365)
      Memory: 699.2M (peak: 1.0G)
        CPU: 7min 34.524s
      CGroup: /system.slice/jenkins.service
              └─1149919 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war

Apr 02 14:59:29 master jenkins[1149919]: 2025-04-02 14:59:29.912+0000 [id=40]           INFO    jenkins.InitReactorRunner$1@
Apr 02 14:59:29 master jenkins[1149919]: 2025-04-02 14:59:29.949+0000 [id=36]           INFO    jenkins.InitReactorRunner$1@
Apr 02 14:59:30 master jenkins[1149919]: 2025-04-02 14:59:30.037+0000 [id=37]           INFO    jenkins.InitReactorRunner$1@
Apr 02 14:59:30 master jenkins[1149919]: 2025-04-02 14:59:30.107+0000 [id=30]           INFO    hudson.lifecycle.Lifecycle@
Apr 02 14:59:30 master systemd[1]: Started Jenkins Continuous Integration Server.
Apr 02 15:00:01 master jenkins[1149919]: 2025-04-02 15:00:01.508+0000 [id=74]           INFO    h.triggers.SCMTigger$Runner@
Apr 02 15:02:01 master jenkins[1149919]: 2025-04-02 15:02:01.479+0000 [id=413]          INFO    h.triggers.SCMTigger$Runner@
Apr 02 15:05:01 master jenkins[1149919]: 2025-04-02 15:05:01.351+0000 [id=652]          INFO    h.triggers.SCMTigger$Runner@
Apr 02 15:08:01 master jenkins[1149919]: 2025-04-02 15:08:01.403+0000 [id=899]          INFO    h.triggers.SCMTigger$Runner@
Apr 02 15:10:01 master jenkins[1149919]: 2025-04-02 15:10:01.356+0000 [id=963]          INFO    h.triggers.SCMTigger$Runner@
```

Accessing Jenkins via Web Interface:

<http://16.171.37.119:8080/>

New Item-> Freestyle project



I made github setting on the Jenkins.

The screenshots show three configuration pages for a Jenkins job named "maven_project".

Source Code Management (Top Screenshot):

- General tab selected.
- Source Code Management section: Git is selected, and the Repository URL is set to `https://github.com/semta06/ABC_project1`.
- Credentials dropdown shows "- none -".
- Buttons: Save, Apply.

Triggers (Middle Screenshot):

- Triggers tab selected.
- Branches to build: Branch Specifier is set to `*/main`.
- Repository browser: (Auto).
- Additional Behaviours: Add dropdown.
- Buttons: Save, Apply.

Triggers (Bottom Screenshot):

- Triggers tab selected.
- Automated actions:
 - Trigger builds remotely (e.g., from scripts) ?
 - Build after other projects are built ?
 - Build periodically ?
 - GitHub hook trigger for GITScm polling ?
 - Poll SCM ?
- Environment section: Configure settings and variables that define the context in which your build runs, like credentials, paths, and global parameters.
- Buttons: Save, Apply.

I saved and applied the my new project.

The screenshot shows the Jenkins dashboard at <http://51.21.162.117:8080>. The top navigation bar includes links for 'Dashboard', 'Build History', 'Manage Jenkins', and 'My Views'. A sidebar on the left lists 'Build Queue' (No builds in the queue) and 'Build Executor Status' (0/2). The main content area displays a table for the 'maven_project' job, which has a status icon (green circle with a checkmark), a build status icon (yellow sun), the name 'maven_project', the last success time ('10 hr #2'), and the last failure time ('N/A'). A 'Last Duration' of '0.62 sec' is also shown. Below the table, there are icons for 'S' (Stable), 'M' (Medium), and 'L' (Large). A button labeled 'Add description' is located in the top right corner.

The screenshot shows the Jenkins job details page for 'maven_project' at http://51.21.162.117:8080/job/maven_project/. The top navigation bar includes links for 'Dashboard' and 'maven_project'. The left sidebar provides options like 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', 'GitHub Hook Log', and 'Rename'. The main content area is titled 'Permalinks' and lists four items: 'Last build (#2), 12 hr ago', 'Last stable build (#2), 12 hr ago', 'Last successful build (#2), 12 hr ago', and 'Last completed build (#2), 12 hr ago'. Below this, the 'Builds' section shows a table for March 24, 2025, with two entries: '#2 8:21PM' and '#1 8:19PM'. A 'Filter' input field is present in the table header.

Güvenli değil http://51.21.162.117:8080/job/maven_project/lastBuild/console

Dashboard > maven_project > #2 > Console Output

Status Changes Console Output Edit Build Information Delete build '#2' Polling Log Timings Git Build Data Previous Build

Console Output

```

Started by GitHub push by semra06
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/maven_project
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/maven_project/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/semra06/ABC_project1 # timeout=10
Fetching upstream changes from https://github.com/semra06/ABC_project1
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- https://github.com/semra06/ABC_project1 +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 396032bb0f8cf1590bbe9e7fbdc543fc32cac8a6 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 396032bb0f8cf1590bbe9e7fbdc543fc32cac8a6 # timeout=10
Commit message: "Update RetailModule.java"
> git rev-list --no-walk d0e3tba58bd5d63d95042341225a372a5804c86e # timeout=10
Finished: SUCCESS

```

I created new items:
ABC_project_Compile

Güvenli değil http://16.171.37.119:8080

Dashboard > new item

Add description

Build History Manage Jenkins My Views

All +

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀️	ABC_project_Compile	46 sec #21	N/A	5.4 sec
✓	☀️	ABC_project_Package	16 sec #34	N/A	8.7 sec
✓	🌧	ABC_project_Pipeline	3 min 6 sec #39	6 min 44 sec #38	49 sec
✓	☀️	ABC_project_testing	31 sec #28	N/A	8.2 sec
✓	☀️	maven_project	1 day 2 hr #2	N/A	0.58 sec

Icon: S M L

docker-registry-credentials

Type here to search

The image displays two nearly identical screenshots of a web-based configuration interface for a CI/CD pipeline, likely Jenkins. Both screenshots show the 'Configuration' screen for a job named 'ABC_project_Compiler'.

Top Screenshot (Left): Configuration - Source Code Management

- Repository URL:** https://github.com/semra06/ABC_project1.git
- Credentials:** - none -
- Advanced:** Add Repository
- Branches to build:** */main

Bottom Screenshot (Right): Configuration - Triggers

- Branch Specifier (blank for 'any'):** */main
- Repository browser:** (Auto)
- Additional Behaviours:** Add

Both screenshots include a standard Windows taskbar at the bottom with various application icons like File Explorer, Task View, and Edge browser.

GitHub hook trigger for GITScm polling ?

Poll SCM ?

Schedule ?

H/5 * * * *

Would last have run at Wednesday, April 2, 2025, 3:10:00 PM Coordinated Universal Time; would next run at Wednesday, April 2, 2025, 3:15:00 PM Coordinated Universal Time.

Ignore post-commit hooks ?

Environment

Configure settings and variables that define the context in which your build runs, like credentials, paths, and global parameters.

Save

Apply

= INVOKE top-level MAVEN targets =

Goals

compile

Advanced ▾

Add build step ▾

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

= Build other projects =

Projects to build

Save

Apply

Configure

General Source Code Management Triggers Environment Build Steps Post-build Actions

Build other projects

Projects to build ABC_project_testing

Trigger only if build is stable

Add post-build action

Save Apply



ABC_project_Package

General Source Code Management Triggers Environment Build Steps Post-build Actions

None

Git

Repository URL https://github.com/semra06/ABC_project1.git

Credentials - none -

+ Add Advanced

Save Apply

docker-registry-credentials

Type here to search

18:13 2.04.2025

The screenshot shows the Jenkins configuration interface for a job named "ABC_project_Package". The left sidebar has "Triggers" selected. In the main area, under "Branches to build", there is a field containing "/main". Below it is a "Repository browser" dropdown set to "(Auto)". At the bottom are "Save" and "Apply" buttons.

The screenshot shows the Jenkins configuration interface for the same job. Under "Triggers", two options are checked: "GitHub hook trigger for GITScm polling" and "Poll SCM". The "Poll SCM" section includes a "Schedule" field with the value "H/5 * * * *". A note below states: "Would last have run at Wednesday, April 2, 2025, 3:13:00 PM Coordinated Universal Time; would next run at Wednesday, April 2, 2025, 3:18:00 PM Coordinated Universal Time." There is also an unchecked checkbox for "Ignore post-commit hooks".

Configure

General
Source Code Management
Triggers
Environment
Build Steps
Post-build Actions

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Invoke top-level Maven targets

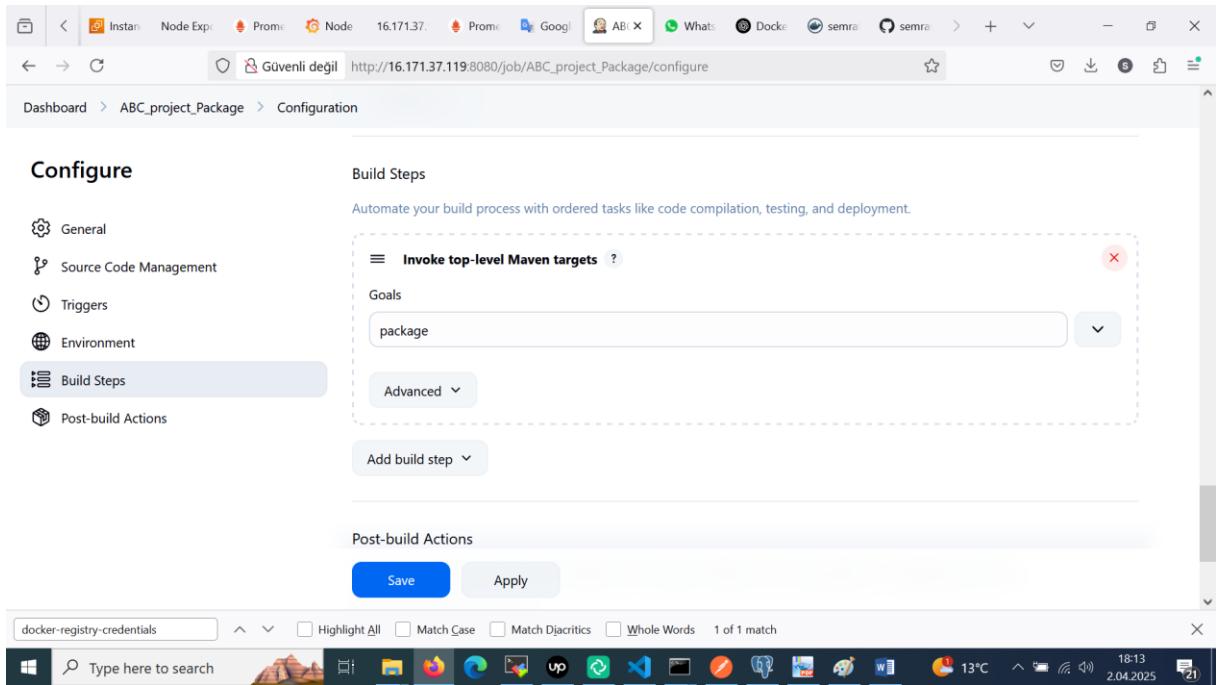
Goals: package

Advanced

Add build step

Post-build Actions

Save Apply



The screenshot shows the configuration interface for the 'ABC_project_Package'. On the left, there's a sidebar with icons for General, Source Code Management, Triggers, Environment, Build Steps (which is selected), and Post-build Actions. The main area is titled 'Build Steps' with a sub-section 'Invoke top-level Maven targets'. It shows a 'Goals' field containing 'package'. Below it is an 'Advanced' dropdown and a 'Add build step' button. At the bottom, there are 'Save' and 'Apply' buttons. The status bar at the bottom indicates a search for 'docker-registry-credentials' and shows system information like temperature (13°C) and date (2.04.2025).

ABC_project_Pipeline

General
Triggers
Pipeline
Advanced

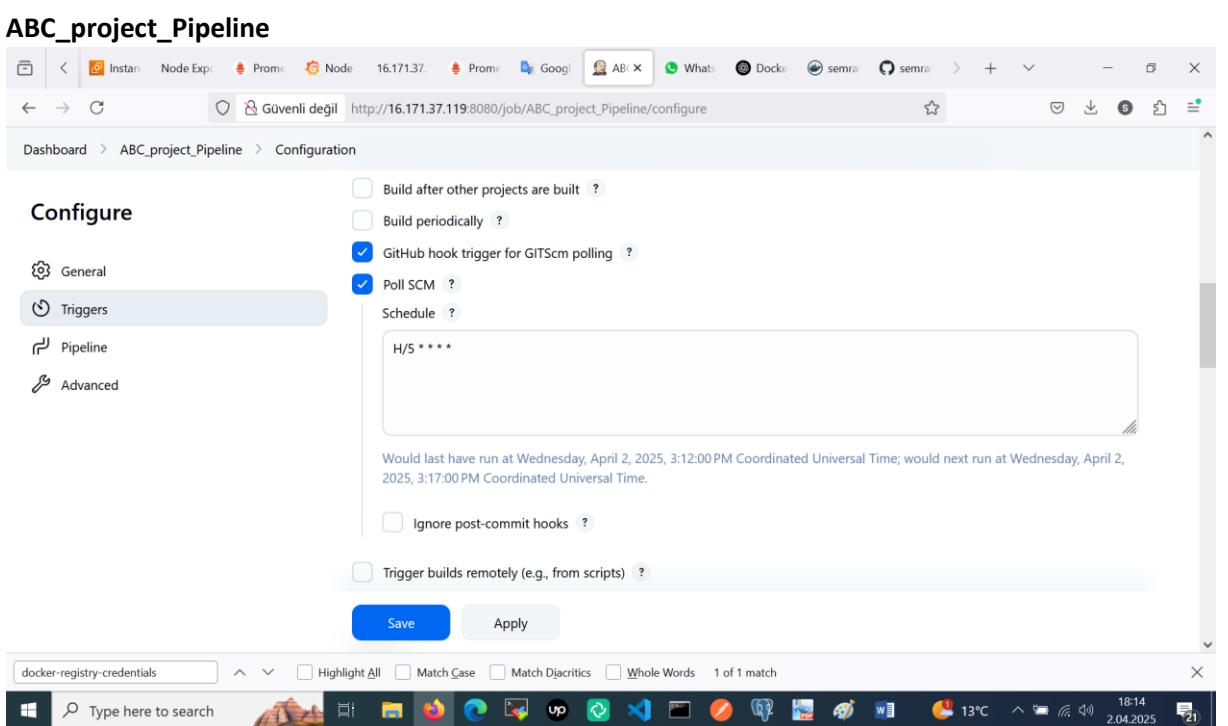
Build after other projects are built ?
 Build periodically ?
 GitHub hook trigger for GITScm polling ?
 Poll SCM ?

Schedule ?
H/5 * * * *

Would last have run at Wednesday, April 2, 2025, 3:12:00 PM Coordinated Universal Time; would next run at Wednesday, April 2, 2025, 3:17:00 PM Coordinated Universal Time.

Ignore post-commit hooks ?
 Trigger builds remotely (e.g., from scripts) ?

Save Apply



The screenshot shows the configuration interface for the 'ABC_project_Pipeline'. On the left, there's a sidebar with icons for General, Triggers (selected), Pipeline, and Advanced. The main area is titled 'Triggers' with sections for 'Build after other projects are built', 'Build periodically', 'GitHub hook trigger for GITScm polling' (which is checked), and 'Poll SCM' (which is also checked). A 'Schedule' section shows 'H/5 * * * *'. Below it, a note says it would last have run at Wednesday, April 2, 2025, 3:12:00 PM Coordinated Universal Time, and would next run at Wednesday, April 2, 2025, 3:17:00 PM Coordinated Universal Time. There are also checkboxes for 'Ignore post-commit hooks' and 'Trigger builds remotely'. At the bottom, there are 'Save' and 'Apply' buttons. The status bar at the bottom indicates a search for 'docker-registry-credentials' and shows system information like temperature (13°C) and date (2.04.2025).

Dashboard > ABC_project_Pipeline > Configuration

Configure

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/semra06/ABC_project1.git

Credentials ?

- none -

Save Apply

docker-registry-credentials

Type here to search

Güvenli değil http://16.171.37.119:8080/job/ABC_project_Pipeline/configure

Dashboard > ABC_project_Pipeline > Configuration

Configure

General

Triggers

Pipeline (selected)

Advanced

Branches to build ?

Branch Specifier (blank for 'any') ?

*/*main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add

Save Apply

The screenshot shows the Jenkins Pipeline configuration page for the project 'ABC_project_Pipeline'. The left sidebar has tabs: General, Triggers, Pipeline, and Advanced (which is selected). The main area has tabs: Add Branch, Repository browser (set to Auto), Additional Behaviours (with an 'Add' dropdown), Script Path (set to Jenkinsfile), and Pipeline Syntax. At the bottom are Save and Apply buttons.

Dashboard > ABC_project_Pipeline > Configuration

Configure

General

Triggers

Pipeline

Advanced

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add

Script Path ?

Jenkinsfile

Lightweight checkout ?

Pipeline Syntax

Save Apply

The screenshot shows the Jenkins Pipeline configuration page for the project 'ABC_project_testing'. The left sidebar has tabs: General, Source Code Management (selected), Triggers, Environment, Build Steps, and Post-build Actions. The main area shows a 'Git' configuration with 'None' selected. Under 'Source Code Management', there is a 'Repositories' section with a 'Repository URL' field containing 'https://github.com/semtar06/ABC_project1.git' and a 'Credentials' dropdown set to '- none -'. At the bottom are Save and Apply buttons.

Dashboard > ABC_project_testing > Configuration

Configure

General

Source Code Management

Triggers

Environment

Build Steps

Post-build Actions

None

Git ?

Repositories ?

Repository URL ?

https://github.com/semtar06/ABC_project1.git

Credentials ?

- none -

+ Add

Advanced

Save Apply

The screenshot shows the Jenkins configuration interface for a job named "ABC_project_testing". The left sidebar has tabs for General, Source Code Management, Triggers, Environment, Build Steps, and Post-build Actions. The "Triggers" tab is selected. In the main panel, under "Branches to build", there is a field containing "/main". Below it is an "Add Branch" button. Under "Repository browser", the dropdown is set to "(Auto)". Under "Additional Behaviours", there is an "Add" button. At the bottom are "Save" and "Apply" buttons.

The screenshot shows the Jenkins configuration interface for the same job. The "Triggers" tab is selected. It shows two checked options: "GitHub hook trigger for GITScm polling" and "Poll SCM". Under "Poll SCM", the "Schedule" field contains "H/5 * * * *". A note below says: "Would last have run at Wednesday, April 2, 2025, 3:12:00 PM Coordinated Universal Time; would next run at Wednesday, April 2, 2025, 3:17:00 PM Coordinated Universal Time." There is also an unchecked checkbox for "Ignore post-commit hooks". At the bottom are "Save" and "Apply" buttons.

The screenshot shows the Jenkins configuration interface for a job named "ABC_project_testing".

Build Steps:

- Invoke top-level Maven targets: Goals: test

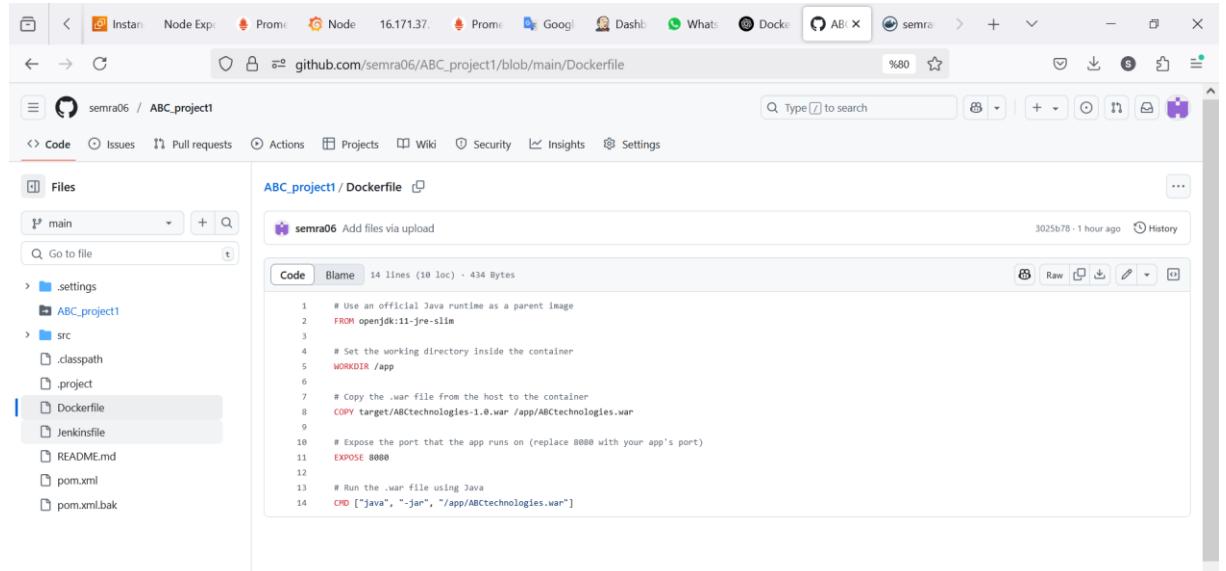
Post-build Actions:

- Build other projects: Projects to build: ABC_project_Package
- Trigger only if build is stable (radio button selected)
- Trigger even if the build is unstable
- Trigger even if the build fails

Buttons: Save, Apply

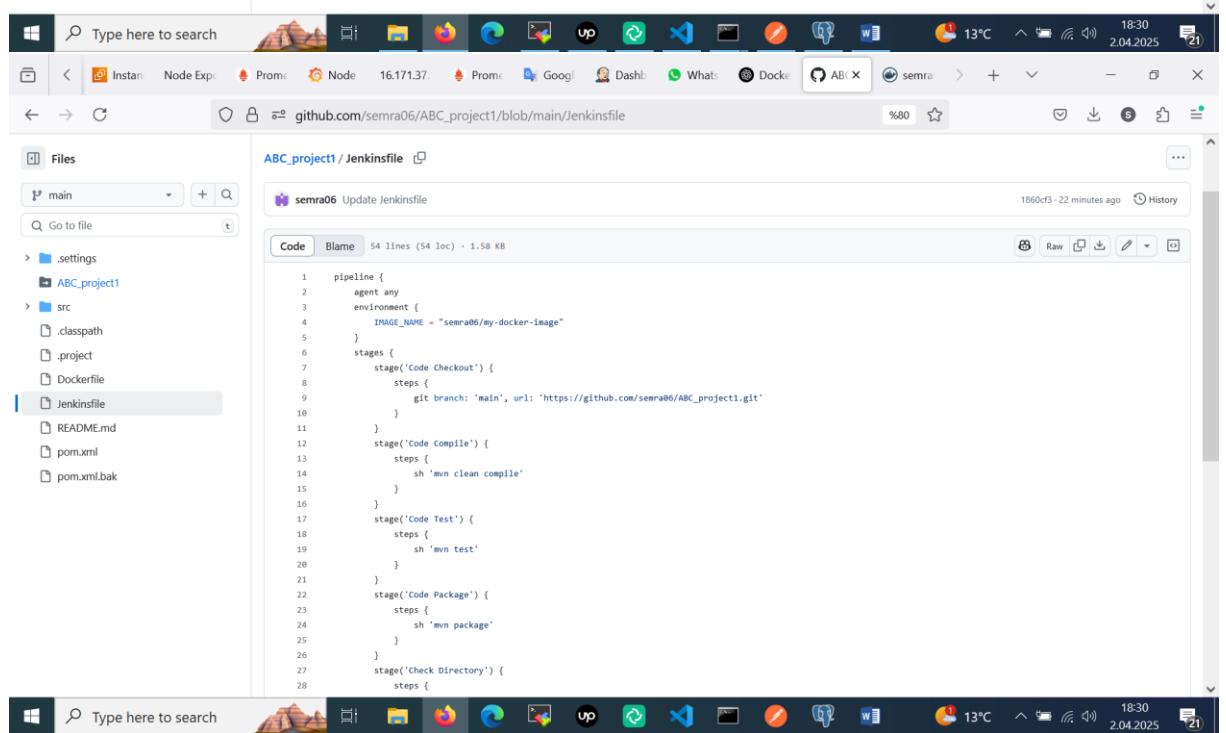
Header: REST API Jenkins 2.503

I also added two file (Jenkinsfile and Dockerfile):



The screenshot shows a GitHub repository page for 'ABC_project1'. The left sidebar lists files: settings, ABC_project1, src, classpath, project, Dockerfile, Jenkinsfile, README.md, pom.xml, and pom.xml.bak. The 'Dockerfile' file is selected. The main content area displays the Dockerfile code:

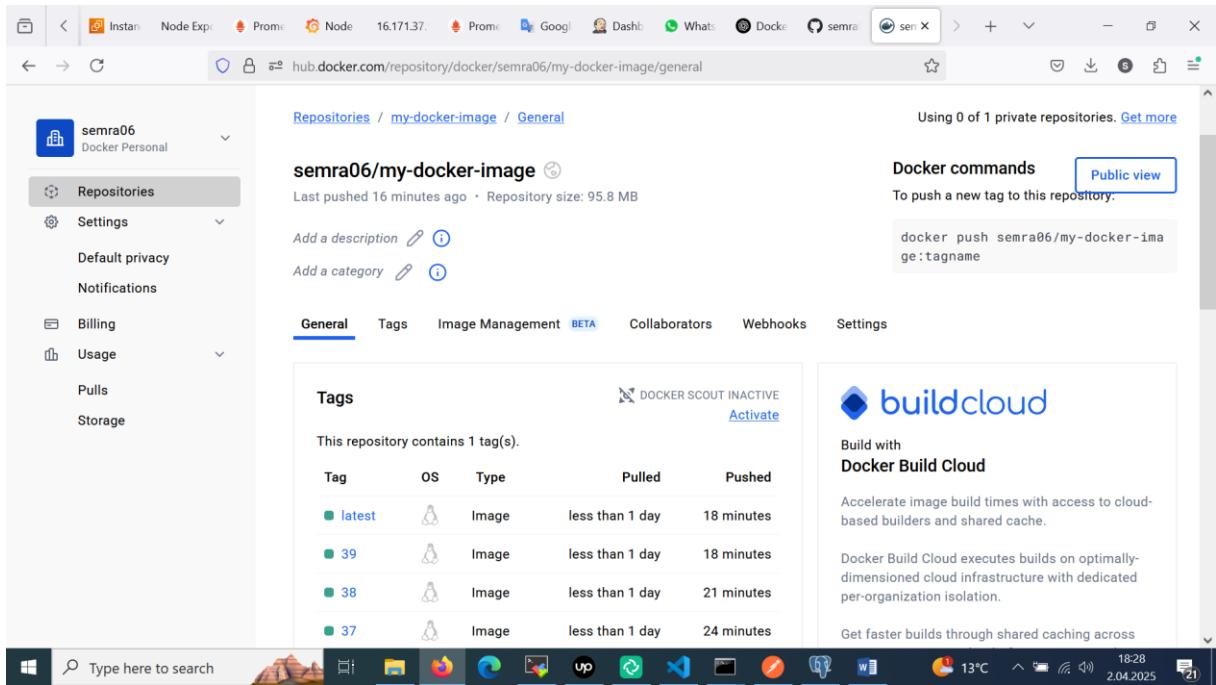
```
1 # Use an official Java runtime as a parent image
2 FROM openjdk:11-jre-slim
3
4 # Set the working directory inside the container
5 WORKDIR /app
6
7 # Copy the .war file from the host to the container
8 COPY target/ABCtechnologies-1.0.war /app/ABCtechnologies.war
9
10 # Expose the port that the app runs on (replace 8080 with your app's port)
11 EXPOSE 8080
12
13 # Run the .war file using Java
14 CMD ["java", "-jar", "/app/ABCtechnologies.war"]
```



The screenshot shows a GitHub repository page for 'ABC_project1'. The left sidebar lists files: settings, ABC_project1, src, classpath, project, Dockerfile, Jenkinsfile, README.md, pom.xml, and pom.xml.bak. The 'Jenkinsfile' file is selected. The main content area displays the Jenkinsfile code:

```
1 pipeline {
2   agent any
3   environment {
4     IMAGE_NAME = "semra06/my-docker-image"
5   }
6   stages {
7     stage('Code Checkout') {
8       steps {
9         git branch: 'main', url: 'https://github.com/semra06/ABC_project1.git'
10      }
11    }
12    stage('Code Compile') {
13      steps {
14        sh 'mvn clean compile'
15      }
16    }
17    stage('Code Test') {
18      steps {
19        sh 'mvn test'
20      }
21    }
22    stage('Code Package') {
23      steps {
24        sh 'mvn package'
25      }
26    }
27    stage('Check Directory') {
28      steps {
```

Jenkins push latest image automatically to dockerhub.



5. Docker

Install using the apt repository

Set up Docker's apt repository:

```
# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
"deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "${UBUNTU_CODENAME}:-$VERSION_CODENAME}") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

To install the latest version, run:

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-
compose-plugin
```

```
ansibleadministrator@master:/usr/local/tomcat/webapps$ sudo docker start dc4ad0e6ce87
dc4ad0e6ce87
ansibleadministrator@master:/usr/local/tomcat/webapps$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
dc4ad0e6ce87 my-tomcat-app:latest "catalina.sh run" 2 hours ago Up About a minute 0.0.0.0:8082->8080/tcp tomcat-co
ntainer
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

```
ansibleadministrator@master:/usr/local/tomcat/webapps$ sudo docker start dc4ad0e6ce87
dc4ad0e6ce87
ansibleadministrator@master:/usr/local/tomcat/webapps$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
dc4ad0e6ce87 my-tomcat-app:latest "catalina.sh run" 2 hours ago Up About a minute 0.0.0.0:8082->8080/tcp tomcat-co
ntainer
ansibleadministrator@master:/usr/local/tomcat/webapps$ docker --version
docker: '-version' is not a docker command.
See 'docker --help'.
ansibleadministrator@master:/usr/local/tomcat/webapps$ docker -version
Docker version 26.1.3, build 26.1.3-Ubuntu1~24.04.1
ansibleadministrator@master:/usr/local/tomcat/webapps$ ls
ABCtechnologies ABCtechnologies.war
ansibleadministrator@master:/usr/local/tomcat/webapps$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

```
ansibleadministrator@master:~/ansible_playbooks$ ls
docker-deploy.yml docker_service.yml inventory.ini jenkins-install.yml kubernetes_service.yml
ansibleadministrator@master:~/ansible_playbooks$ cd docker_deploy
```

Build the Docker Image:

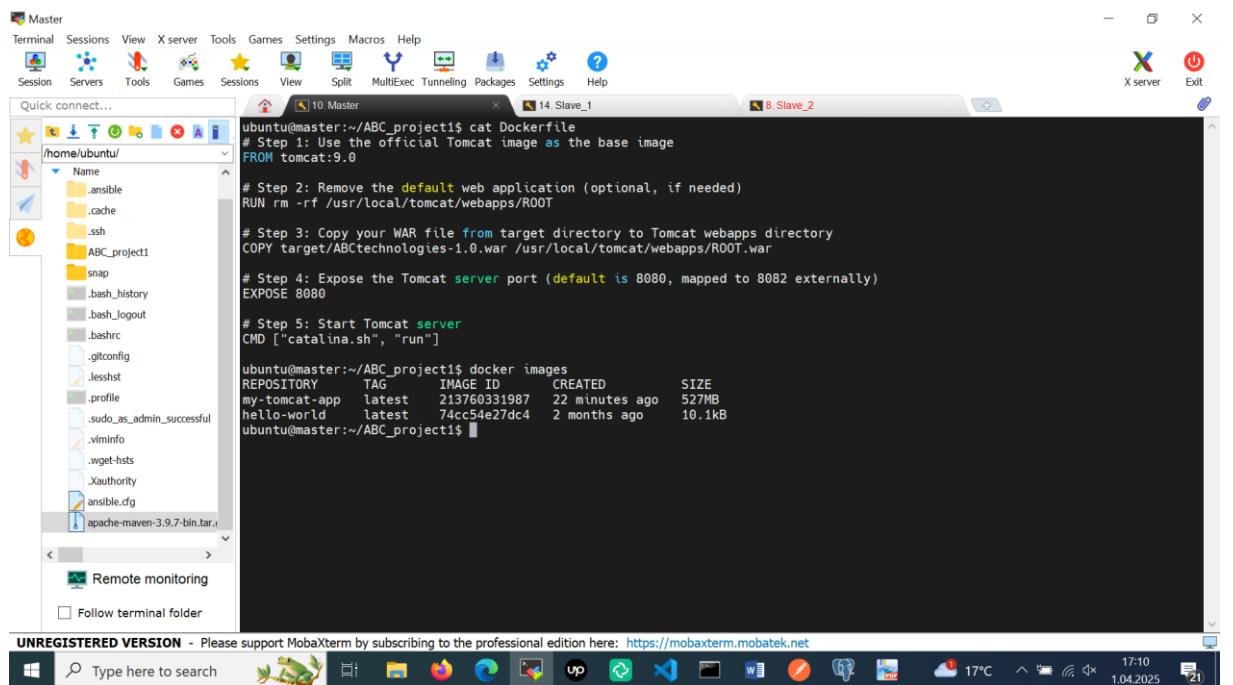
After creating the Dockerfile, you can build your Docker image using the following command:

```
docker build -t my-tomcat-app .
```

Run the Docker Image:

Once the image is successfully built, you can run it using the following command:

```
docker run -d -p 8082:8080 my-tomcat-app
```



The screenshot shows a MobaXterm window titled 'Master'. The terminal session displays the following command and its output:

```
ubuntu@master:~/ABC_project1$ cat Dockerfile
# Step 1: Use the official Tomcat image as the base image
FROM tomcat:9.0

# Step 2: Remove the default web application (optional, if needed)
RUN rm -rf /usr/local/tomcat/webapps/ROOT

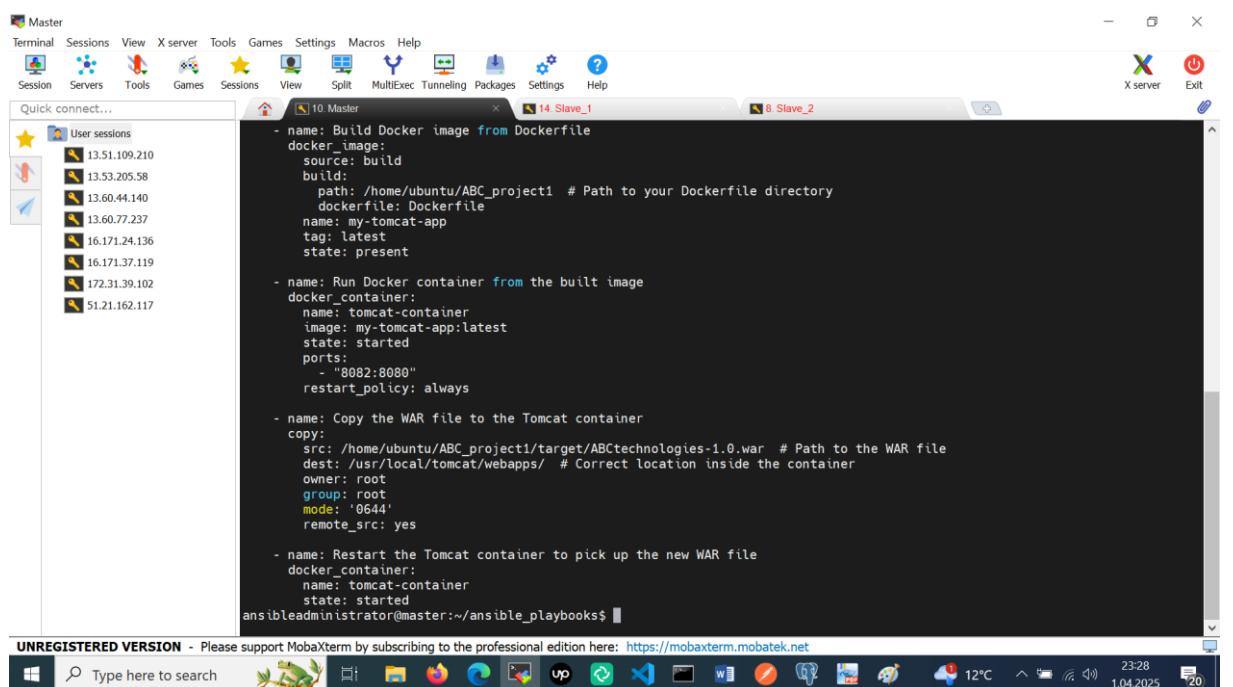
# Step 3: Copy your WAR file from target directory to Tomcat webapps directory
COPY target/ABCtechnologies-1.0.war /usr/local/tomcat/webapps/ROOT.war

# Step 4: Expose the Tomcat server port (default is 8080, mapped to 8082 externally)
EXPOSE 8080

# Step 5: Start Tomcat server
CMD ["catalina.sh", "run"]

ubuntu@master:~/ABC_project1$ docker images
REPOSITORY      TAG      IMAGE ID      CREATED        SIZE
my-tomcat-app   latest   213760331987   22 minutes ago  527MB
hello-world     latest   74cc54e27dc4   2 months ago   10.1kB
ubuntu@master:~/ABC_project1$
```

The terminal window also shows the message 'UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>'.



The screenshot shows a MobaXterm window titled 'Master'. The terminal session displays the following Ansible playbook code:

```
- name: Build Docker image from Dockerfile
  docker_image:
    source: build
    build:
      path: /home/ubuntu/ABC_project1 # Path to your Dockerfile directory
      dockerfile: Dockerfile
    name: my-tomcat-app
    tag: latest
    state: present

- name: Run Docker container from the built image
  docker_container:
    name: tomcat-container
    image: my-tomcat-app:latest
    state: started
    ports:
      - "8082:8080"
    restart_policy: always

- name: Copy the WAR file to the Tomcat container
  copy:
    src: /home/ubuntu/ABC_project1/target/ABCtechnologies-1.0.war # Path to the WAR file
    dest: /usr/local/tomcat/webapps/ # Correct location inside the container
    owner: root
    group: root
    mode: '0644'
    remote_src: yes

- name: Restart the Tomcat container to pick up the new WAR file
  docker_container:
    name: tomcat-container
    state: started
  ansibleleadadministrator@master:~/ansible_playbooks$
```

The terminal window also shows the message 'UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>'.

```
ansibleadministrator@master:~/ansible_playbooks$ ansible-playbook -i inventory.ini
docker-deploy.yml
```

```
state: started
ansibleadministrator@master:~/ansible_playbooks$ ansible-playbook -i inventory.ini docker-deploy.yml
PLAY [Deploy WAR file to Tomcat using Docker] ****
[WARNING]: Platform linux on host 16.171.37.119 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [16.171.37.119]

TASK [Install Docker if not installed] ****
ok: [16.171.37.119]

TASK [Ensure Docker service is running] ****
ok: [16.171.37.119]

TASK [Build Docker image from Dockerfile] ****
ok: [16.171.37.119]

TASK [Run Docker container from the built image] ****
ok: [16.171.37.119]

TASK [Copy the WAR file to the Tomcat container] ****
changed: [16.171.37.119]

TASK [Restart the Tomcat container to pick up the new WAR file] ****
ok: [16.171.37.119]

PLAY RECAP ****
16.171.37.119 : ok=7    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ansibleadministrator@master:~/ansible_playbooks$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>



I will show you below how to install ansible.

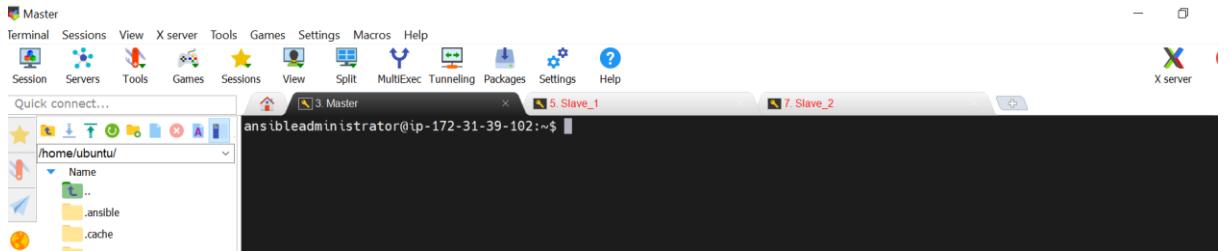
6. Ansible

https://docs.ansible.com/ansible/latest/installation_guide/installation_distros.html#installing-ansible-on-ubuntu

Create 3 EC2 Instances 1 Controller/Master and 2 Slaves/Nodes.

Instances (1/3) Info		Last updated less than a minute ago	Connect	Instance state ▾	Actions ▾	Launch instances	▼
<input type="text"/> Find Instance by attribute or tag (case-sensitive)				All states ▾		◀ 1 ▶	⚙
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	▼
<input checked="" type="checkbox"/>	Master	i-07161449111b2e43d	Running Q Q	t3.micro	...	View alarms +	▼
<input type="checkbox"/>	Slave_1	i-0b4afc67534583d39	Running Q Q	t3.micro	...	View alarms +	▼
<input type="checkbox"/>	Slave_2	i-0bf9349753c6adc68	Running Q Q	t3.micro	...	View alarms +	▼

Connect all three instances using SSH Client



Ansible Master Node

Installation on masternode:

sudo apt update-> update package lists

sudo apt install software-properties-common-> Install Required Dependencies

sudo add-apt-repository --yes --update ppa:ansible/ansible -> Add the Official Ansible PPA

sudo apt install ansible -> Install Ansible

ansible –version -> verify the installation

```
ubuntu@ip-172-31-39-102:~$ ansible -v
ansible [core 2.17.9]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['~/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  python version = 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-39-102:~$
```

Generates a new **Ansible configuration file (ansible.cfg)** with all available settings, but **commented out (disabled)**:

ansible-config init --disabled > ansible.cfg

sudo adduser ansibleadministrator → create ansible user

```
ubuntu@ip-172-31-39-102:~$ ansible-config init --disabled > ansible.cfg
ubuntu@ip-172-31-39-102:~$ ls
ABC_project1  ansible.cfg  apache-maven-3.9.7-bin.tar.gz
ubuntu@ip-172-31-39-102:~$ sudo adduser ansibleadministrator
info: Adding user 'ansibleadministrator' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group 'ansibleadministrator' (1001) ...
info: Adding new user 'ansibleadministrator' (1001) with group `ansibleadministrator (1001)' ...
info: Creating home directory '/home/ansibleadministrator' ...
info: Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for ansibleadministrator
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
info: Adding new user 'ansibleadministrator' to supplemental / extra groups 'users' ...
info: Adding user 'ansibleadministrator' to group 'users' ...
ubuntu@ip-172-31-39-102:~$ ls
ABC_project1  ansible.cfg  apache-maven-3.9.7-bin.tar.gz
ubuntu@ip-172-31-39-102:~$ su ansibleadministrator
Password:
ansibleadministrator@ip-172-31-39-102:~/home/ubuntu$
```

```
sudo chown -R ansibleadmin:ansibleadmin /etc/ansible
```

```
ansibleadmin@ip-172-31-39-102:/home/ubuntu$ sudo chown -R ansibleadmin:ansibleadmin /etc/ansible
[sudo] password for ansibleadmin:
ansibleadmin is not in the sudoers file.
ansibleadmin@ip-172-31-39-102:/home/ubuntu$ cd
ansibleadmin@ip-172-31-39-102:~$ exit
ubuntu@ip-172-31-39-102:~$ sudo chown -R ansibleadmin:ansibleadmin /etc/ansible
ubuntu: invalid user: ansibleadmin:ansibleadmin
ubuntu@ip-172-31-39-102:~$ sudo chown -R ansibleadmin:ansibleadmin /etc/ansible
ubuntu@ip-172-31-39-102:~$ cd /etc/ansible/
ubuntu@ip-172-31-39-102:/etc/ansible$ ls
ansible.cfg hosts roles
ubuntu@ip-172-31-39-102:/etc/ansible$
```

```
cat /etc/ansible/ansible.cfg
```

```
ls -al
```

```
ubuntu@ip-172-31-39-102:/etc/ansible$ ls
ansible.cfg hosts roles
ubuntu@ip-172-31-39-102:/etc/ansible$ sudo chown -R ansibleadmin:ansibleadmin /etc/ansible
ubuntu@ip-172-31-39-102:/etc/ansible$ ls -l
total 12
-rw-r--r-- 1 ansibleadmin ansibleadmin 614 Feb 25 16:59 ansible.cfg
-rw-r--r-- 1 ansibleadmin ansibleadmin 1175 Feb 25 16:59 hosts
drwxr-xr-x 2 ansibleadmin ansibleadmin 4096 Feb 25 16:59 roles
ubuntu@ip-172-31-39-102:/etc/ansible$ cat /etc/ansible/ansible.cfg
# Since Ansible 2.12 (core):
# To generate an example config file (a "disabled" one with all default settings, commented out):
#         $ ansible-config init --disabled > ansible.cfg
#
# Also you can now have a more complete file by including existing plugins:
# ansible-config init --disabled -t all > ansible.cfg
#
# For previous versions of Ansible you can check for examples in the 'stable' branches of each version
# Note that this file was always incomplete and lagging changes to configuration settings
#
# for example, for 2.9: https://github.com/ansible/ansible/blob/stable-2.9/examples/ansible.cfg
ubuntu@ip-172-31-39-102:/etc/ansible$ ls -l
total 12
-rw-r--r-- 1 ansibleadmin ansibleadmin 614 Feb 25 16:59 ansible.cfg
-rw-r--r-- 1 ansibleadmin ansibleadmin 1175 Feb 25 16:59 hosts
drwxr-xr-x 2 ansibleadmin ansibleadmin 4096 Feb 25 16:59 roles
ubuntu@ip-172-31-39-102:/etc/ansible$ ls -al
total 20
drwxr-xr-x  3 ansibleadmin ansibleadmin 4096 Mar 25 19:57 .
drwxr-xr-x 114 root      root        4096 Mar 25 20:03 ..
-rw-r--r--  1 ansibleadmin ansibleadmin 614 Feb 25 16:59 ansible.cfg
-rw-r--r--  1 ansibleadmin ansibleadmin 1175 Feb 25 16:59 hosts
drwxr-xr-x  2 ansibleadmin ansibleadmin 4096 Feb 25 16:59 roles
ubuntu@ip-172-31-39-102:/etc/ansible$
```

```
su ansibleadmin-> ansible admin user
```

```
drwxr-xr-x  2 ansibleadmin ansibleadmin 4096 Feb 25 16:59 roles
ubuntu@ip-172-31-39-102:/etc/ansible$ cd
ubuntu@ip-172-31-39-102:~$ su ansibleadmin
Password:
ansibleadmin@ip-172-31-39-102:/home/ubuntu$ pwd
/home/ubuntu
ansibleadmin@ip-172-31-39-102:/home/ubuntu$ cd ~
ansibleadmin@ip-172-31-39-102:~$ pwd
/home/ansibleadmin
ansibleadmin@ip-172-31-39-102:~$
```

Copy paste Ansible Managed Node(Slave_Node1) public ip adres on Master Node:

```
vi /etc/ansible/hosts
```

```

# If you have multiple hosts following a pattern, you can specify
# them like this:
## www[001:006].example.com
# You can also use ranges for multiple hosts:
## db-[99:101]-node.example.com
# Ex 3: A collection of database servers in the 'dbservers' group:
## [dbservers]
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57
# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':
## [Debian]
## alpha.example.org
## beta.example.org
## [openSUSE]
## green.example.com
## blue.example.com
[test]
13.60.44.140
13.60.77.237
"/etc/ansible/hosts" 56L, 1207B

```

56,12 Bot

Save and close after adding.

```

ansibleleadadministrator@ip-172-31-39-102:~/.ssh$ vi /etc/ansible/hosts
ansibleleadadministrator@ip-172-31-39-102:~/.ssh$ ls -al
total 24
drwxr-x--- 2 ansibleleadadministrator ansibleleadadministrator 4096 Mar 26 11:38 .
drwxr-x--- 5 ansibleleadadministrator ansibleleadadministrator 4096 Mar 26 16:16 ..
-rw-r--r-- 1 ansibleleadadministrator ansibleleadadministrator 432 Mar 26 11:19 id_ed25519
-rw-r--r-- 1 ansibleleadadministrator ansibleleadadministrator 119 Mar 26 11:19 id_ed25519.pub
-rw-r--r-- 1 ansibleleadadministrator ansibleleadadministrator 1956 Mar 26 11:38 known_hosts
-rw-r--r-- 1 ansibleleadadministrator ansibleleadadministrator 1120 Mar 26 11:38 known_hosts.old
ansibleleadadministrator@ip-172-31-39-102:~/.ssh$ ls
id_ed25519 id_ed25519.pub known_hosts known_hosts.old
ansibleleadadministrator@ip-172-31-39-102:~/.ssh$ 

```

Adding Public Key to authorized_keys File on Master Node:

```

ubuntu@ip-172-31-39-102:~$ cd .ssh/
sudo cp authorized_keys /home/ansibleleadadministrator/.ssh/

```

Used to give ownership of the file to the ansible administrator user and group:

```

sudo chown ansibleleadadministrator:ansibleleadadministrator
/home/ansibleleadadministrator/.ssh/authorized_keys

```

SSH expects the .ssh folder and files within it to be owned only by the owning user. Incorrect authorization can cause connection problems.

```

ansibleAdministrator@ip-172-31-39-102:~/ssh$ ls -al
total 9460
drwxr-x--- 6 ubuntu ubuntu 4096 Mar 25 20:22 .
drwxr-xr-x 4 root root 4096 Mar 25 20:02 ..
-rw----- 1 ubuntu ubuntu 558 Mar 25 20:22 .Xauthority
drwxrwxr-x 3 ubuntu ubuntu 4096 Mar 25 19:58 .ansible
-rw----- 1 ubuntu ubuntu 7521 Mar 25 18:17 .bash_history
-rw-r--r-- 1 ubuntu ubuntu 220 Mar 31 2024 .bash_logout
-rw-r--r-- 1 ubuntu ubuntu 3823 Mar 24 20:32 .bashrc
drwxr---- 2 ubuntu ubuntu 4096 Mar 24 18:01 .cache
-rw-rw-r- 1 ubuntu ubuntu 50 Mar 25 10:43 .gitconfig
-rw----- 1 ubuntu ubuntu 20 Mar 25 16:05 .lesshst
-rw-r--r- 1 ubuntu ubuntu 807 Mar 31 2024 .profile
drwxr----- 2 ubuntu ubuntu 4096 Mar 24 18:14 .ssh
-rw-r--r- 1 ubuntu ubuntu 0 Mar 24 20:28 .sudo_as_admin_successful
-rw-rw-r- 1 ubuntu ubuntu 164 Mar 25 11:03 .wget-hsts
drwxrwxr-x 5 ubuntu ubuntu 4096 Mar 25 10:44 ABC_project1
-rw-rw-r- 1 ubuntu ubuntu 40810 Mar 25 20:01 ansible.cfg
-rw-rw-r- 1 ubuntu ubuntu 9581488 Mar 25 10:39 apache-maven-3.9.7-bin.tar.gz
ubuntu@ip-172-31-39-102:~/ssh$ cd .ssh/
ubuntu@ip-172-31-39-102:~/ssh$ ls
authorized_keys known_hosts
ubuntu@ip-172-31-39-102:~/ssh$ cp authorized_keys /home/ansibleAdministrator/.ssh/
cp: cannot stat '/home/ansibleAdministrator/.ssh': Permission denied
ubuntu@ip-172-31-39-102:~/ssh$ sudo cp authorized_keys /home/ansibleAdministrator/.ssh/
ubuntu@ip-172-31-39-102:~/ssh$ 

```

```

ubuntu@ip-172-31-39-102:~/ssh$ sudo cp authorized_keys /home/ansibleAdministrator/.ssh/
cp: cannot stat '/home/ansibleAdministrator/.ssh': Permission denied
ubuntu@ip-172-31-39-102:~/ssh$ sudo chown ansibleAdministrator:ansibleAdministrator /home/ansibleAdministrator/.ssh/authorized_keys
ubuntu@ip-172-31-39-102:~/ssh$ 

```

Ansible Managed Node(Slave_Node)

sudo apt update

I create same password and username with Master Node.

sudo adduser ansibleAdministrator

su ansibleAdministrator

Created public ve private key for SSH (Secure Shell):

ssh-keygen

```

Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ansibleadministrator/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ansibleadministrator/.ssh/id_ed25519
Your public key has been saved in /home/ansibleadministrator/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:R18LV380docDuR5Nc2Kr0MbG7usnTCHSGy/lqDGXNm8 ansibleadministrator@ip-172-31-39-102
The key's randomart image is:
++-[ED25519 256]--+
|   oo.+ |
|   .+..B=.. |
|   ..++.=.*o |
|   .+=(@ o |
|   S/ + |
|   o O B |
|   * B |
|   . E . |
|   oo+ |
+---[SHA256]-----+
ansibleadministrator@ip-172-31-39-102:~$ ls -al
total 36
drwxr-x--- 4 ansibleadministrator ansibleadministrator 4096 Mar 25 20:29 .
drwxr-xr-x  4 root          root        4096 Mar 25 20:28 ..
drwxrwxr-x  4 ansibleadministrator ansibleadministrator 4096 Mar 25 20:28 .ansible
-rw-r--r--  1 ansibleadministrator ansibleadministrator 158 Mar 25 20:39 .bash_history
-rw-r--r--  1 ansibleadministrator ansibleadministrator 220 Mar 25 20:02 .bash_logout
-rw-r--r--  1 ansibleadministrator ansibleadministrator 3771 Mar 25 20:02 .bashrc
-rw-r--r--  1 ansibleadministrator ansibleadministrator 807 Mar 25 20:02 .profile
drwxr----- 2 ansibleadministrator ansibleadministrator 4096 Mar 25 20:40 .ssh
-rw-r--r--  1 ansibleadministrator ansibleadministrator 853 Mar 25 20:28 viminfo
ansibleadministrator@ip-172-31-39-102:~$ cd .ssh
ansibleadministrator@ip-172-31-39-102:~/ssh$ ls

```



```

ansibleAdministrator@ip-172-31-37-44:~$ ls -al
total 44
drwxr-x--- 5 ansibleAdministrator ansibleAdministrator 4096 Mar 26 11:31 .
drwxr-xr-x  4 root          root        4096 Mar 26 09:32 ..
drwxrwxr-x  3 ansibleAdministrator ansibleAdministrator 4096 Mar 26 09:40 .ansible
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 824 Mar 26 14:16 .bash_history
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 220 Mar 26 09:32 .bash_logout
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 3771 Mar 26 09:32 .bashrc
drwxr----- 2 ansibleAdministrator ansibleAdministrator 4096 Mar 26 11:31 .cache
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 807 Mar 26 09:32 .profile
drwxr----- 2 ansibleAdministrator ansibleAdministrator 4096 Mar 26 11:31 .ssh
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 4393 Mar 26 11:31 viminfo
ansibleAdministrator@ip-172-31-37-44:~$ 
ansibleAdministrator@ip-172-31-37-44:~$ cd .ssh
ansibleAdministrator@ip-172-31-37-44:~/ssh$ ls -al
total 20
drwxr----- 2 ansibleAdministrator ansibleAdministrator 4096 Mar 26 11:31 .
drwxr-x---  5 ansibleAdministrator ansibleAdministrator 4096 Mar 26 11:31 ..
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 119 Mar 26 11:31 authorized_keys
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 978 Mar 26 11:25 known_hosts
-rw-r--r--  1 ansibleAdministrator ansibleAdministrator 142 Mar 26 11:25 known_hosts.old
ansibleAdministrator@ip-172-31-37-44:~/ssh$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Master Node

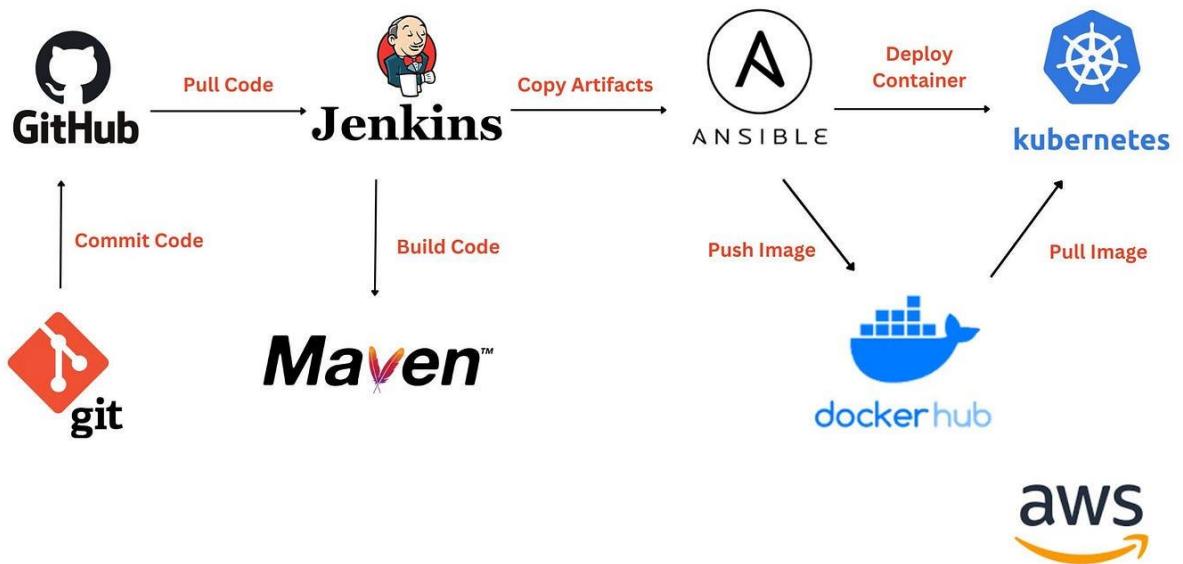
On master node check ansible is work or not with ansible username:
ansible all -m ping

```

yes
[WARNING]: Platform linux on host 13.60.77.237 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
13.60.77.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
ansibleadministrator@ip-172-31-39-102:~$ ansible all -m ping
[WARNING]: Platform linux on host 13.60.77.237 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
13.60.77.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Platform linux on host 13.60.44.140 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
13.60.44.140 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
ansibleadministrator@ip-172-31-39-102:~$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

The structure to be established:



We worked with ansible playbooks yaml file. I created ansible_playbooks folder and also jenkins-install.yml

The screenshot shows a MobaXterm window with two tabs open: "3. Master" and "9. Slave_1". The "3. Master" tab displays an Ansible playbook named "jenkins-install.yml" which installs Jenkins on an Ubuntu 24.04 system. The "9. Slave_1" tab shows the execution of this playbook, displaying command-line output for each task. A file explorer sidebar on the left shows the local directory structure of the master node.

```

name: Install Jenkins on Ubuntu 24.04
hosts: all
become: yes

tasks:
- name: Update and upgrade system packages
  apt:
    update_cache: yes
    upgrade: yes

- name: Install Java (Jenkins dependency)
  apt:
    name: openjdk-21-jdk
    state: present

- name: Add Jenkins repository key
  apt_key:
    url: https://pkg.jenkins.io/debian/jenkins.io-2023.key
    state: present

- name: Add Jenkins APT repository
  apt_repository:
    repo: "deb https://pkg.jenkins.io/debian-stable binary/"
    state: present
    filename: jenkins

- name: Install Jenkins
  apt:
    name: jenkins
    state: present
    update_cache: yes

```

1,1

```

ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ cd ansible_playbooks
ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ ls
jenkins-install.yml
ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ ^C
ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ vi jenkins-install.yml
ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ ansible all -m ping
[WARNING]: Platform linux on host 13.60.44.140 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
13.60.44.140 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Platform linux on host 13.60.77.237 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
13.60.77.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
ansibleAdministrator@ip-172-31-39-102:~/ansible_playbooks$ █

```

INDETERMINATE VERDICT. Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Running *ansible-playbook* on Master Node:

ansible-playbook jenkins-install.yml

The screenshot shows a MobaXterm window titled "Master" with several tabs open: "3. Master", "9. Slave_1", and "8. Slave_2". The "3. Master" tab displays the following Ansible playbook output:

```
TASK [Add Jenkins APT repository] ****
ok: [13.60.44.140]
ok: [13.60.77.237]

TASK [Install Jenkins] ****
ok: [13.60.44.140]
ok: [13.60.77.237]

TASK [Start and enable Jenkins service] ****
ok: [13.60.44.140]
ok: [13.60.77.237]

TASK [Allow Jenkins through UFW (port 8080)] ****
ok: [13.60.44.140]
ok: [13.60.77.237]

TASK [Get initial Jenkins admin password] ****
ok: [13.60.44.140]
ok: [13.60.77.237]

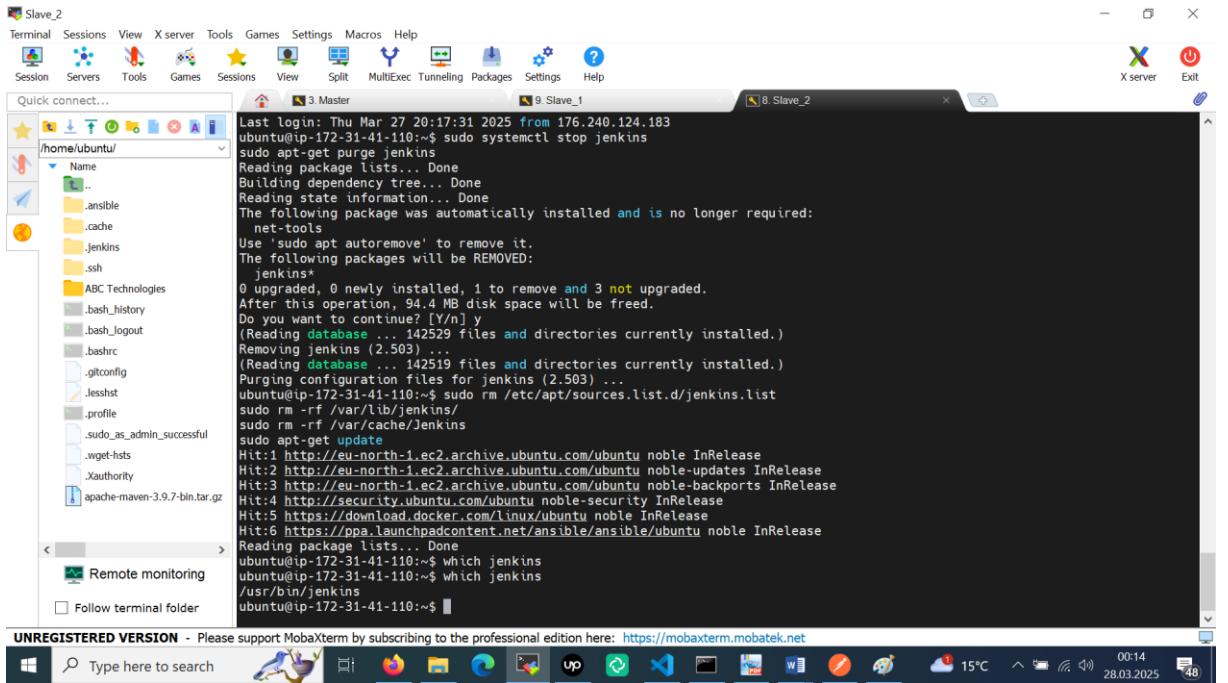
TASK [Show Jenkins initial admin password] ****
ok: [13.60.44.140] => {
    msg: "Jenkins Initial Admin Password: 268fafaae73842c3ab03b00e8dacc2a9"
}
ok: [13.60.77.237] => {
    msg: "Jenkins Initial Admin Password: f9db8ccc12374b919e81f9a9530447d5"
}

PLAY RECAP ****
13.60.44.140 : ok=10  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
13.60.77.237 : ok=10  changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

The terminal prompt at the bottom is "ansibleadministrator@ip-172-31-39-102:~/ansible_playbooks\$".

Jenkins is installed on Slave Node.

The screenshot shows the MobaXterm interface. At the top, there's a menu bar with options like Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help, and a quick connect search bar. Below the menu is a toolbar with icons for Session, Servers, Tools, Games, Sessions, View, Split, MultiExec, Tunneling, Packages, Settings, and Help. The main window has two terminal tabs: '3. Master' and '8. Slave_1'. The 'Slave_1' tab displays a command-line session where Jenkins is being uninstalled via Ansible. The session shows commands like `sudo systemctl stop jenkins`, `sudo apt-get purge jenkins`, and `sudo apt autoremove`. It also lists packages to be removed and asks for confirmation. The terminal output continues with package removal details and ends with a warning about purging configuration files. The bottom of the terminal window shows the command `sudo apt-get update` followed by a list of repositories. On the left side of the interface, there's a file explorer sidebar with a tree view showing local files and folders, including '.ansible', '.cache', 'jenkins', '.ssh', 'ABC Technologies', '.bash_history', '.bash_logout', '.bashrc', '.gitconfig', '.gitsht', '.profile', '.sudo_as_admin_successful', '.wget-hsts', '.Xauthority', and 'apache-maven-3.9.7-bin.tar.gz'. There are also buttons for 'Remote monitoring' and 'Follow terminal folder'. The bottom of the screen features a taskbar with various application icons and system status indicators.



```
ansible-playbook -i inventory.ini kubernetes_service.yml
```

7. Kubernetes

```
# Update and Upgrade Ubuntu (all nodes)
sudo apt update && sudo apt upgrade -y
sudo hostnamectl set-hostname "master.example.net"
exec bash

# On the worker nodes, run
sudo hostnamectl set-hostname "worker1.example.net"      // 1st worker node
sudo hostnamectl set-hostname "worker2.example.net"      // 2nd worker node
exec bash

MASTER NODE:
sudo nano /etc/hosts
#####Edit ip address#####
# Master Node
16.171.37.119    master-node master.example.net
16.171.37.120# First Worker Node
16.171.37.12113.60.44.140    worker1-node worker1.example.net
16.171.37.122# Second Worker Node
16.171.37.12313.60.77.237    worker2-node worker2.example.net
16.171.37.124## Disable Swap (all nodes)
sudo swapoff -a
sudo sed -i '/ swap / s/^(\.*\$/#\1/g' /etc/fstab

# Load the required kernel modules on all nodes:
sudo tee /etc/modules-load.d/containerd.conf <<EOF
overlay
br_netfilter
EOF
```

```

EOF
sudo modprobe overlay
sudo modprobe br_netfilter

# To load the above kernel parameters, run
sudo sysctl -system

## Install Containerd Runtime (all nodes)
sudo apt install -y curl gnupg2 software-properties-common apt-transport-
https ca-certificates

# Enable the Docker repository:
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmour -o /etc/apt/trusted.gpg.d/docker.gpg
sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"

# Update the package list and install containerd:
sudo apt update
sudo apt install -y containerd.io

# Configure containerd to start using systemd as cgroup:
containerd config default | sudo tee /etc/containerd/config.toml >/dev/null
2>&1
sudo sed -i 's/SystemdCgroup \= false/SystemdCgroup \= true/g'
/etc/containerd/config.toml
sudo systemctl restart containerd
sudo systemctl enable containerd

# Add Apt Repository for Kubernetes (all nodes)
echo "deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg]
https://pkgs.k8s.io/core:/stable:/v1.30/deb/ /" | sudo tee
/etc/apt/sources.list.d/kubernetes.list
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.30/deb/Release.key | sudo
gpg --dearmor -o /etc/apt/keyrings/kubernetes-apt-keyring.gpg

#Install Kubectl, Kubeadm, and Kubelet (all nodes)
sudo apt update
sudo apt install -y kubelet kubeadm kubectl
sudo apt-mark hold kubelet kubeadm kubectl

# Initialize Kubernetes Cluster with Kubeadm (master node)
sudo kubeadm init --ignore-preflight-errors=Mem

```

With all the prerequisites in place, initialize the Kubernetes cluster on the master node using the following Kubeadm command:

Initialize Kubernetes Cluster with Kubeadm (master node):

```
sudo kubeadm init --ignore-preflight-errors=Mem
```

```

[10 master] Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: Kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.39.102:6443 --token h57bhe.c4jnt9x5jfv508fk \
--discovery-token-ca-cert-hash sha256:85ef589a0ec5098381e8f68909cf88eca2d30ba0c2d1186c5102117ef5484234
ubuntu@master:~$ mkdir -p $HOME/.kube
ubuntu@master:~$ sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
ubuntu@master:~$ sudo chown $(id -u):$(id -g) $HOME/.kube/config
ubuntu@master:~$ kubectl get nodes
NAME           STATUS    ROLES     AGE   VERSION
master.example.net   NotReady   control-plane   32s   v1.30.11
ubuntu@master:~$ kubectl get nodes
NAME           STATUS    ROLES     AGE   VERSION
master.example.net   NotReady   control-plane   41s   v1.30.11

```

After the initialization is complete make a note of the kubeadm join command for future reference.

Run the following commands on the master node:

```

mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config

```

Add Worker Nodes1 and 2 to the Cluster (worker nodes1 and 2):

```

sudo kubeadm join 172.31.39.102:6443 --token h57bhe.c4jnt9x5jfv508fk \
--discovery-token-ca-cert-hash
sha256:85ef589a0ec5098381e8f68909cf88eca2d30ba0c2d1186c5102117ef5484234

```

Verify the cluster and test (master node) :

```

kubectl get pods -n kube-system
kubectl get nodes

```

```

ubuntu@master:~$ kubectl get pods -n kube-system
NAME                                         READY   STATUS    RESTARTS   AGE
calico-kube-controllers-5b9b456c66-xjr97   1/1    Running   0          21m
calico-node-b7dk5                           0/1    Running   0          21m
calico-node-c6php                          0/1    Running   0          17m
calico-node-vw64q                           0/1    Running   0          19m
coredns-55cb58b774-66s6h                   1/1    Running   0          25m
coredns-55cb58b774-xtnn9                   1/1    Running   0          25m
etcd-master.example.net                    1/1    Running   2          25m
kube-apiserver-master.example.net          1/1    Running   3          25m
kube-controller-manager-master.example.net  1/1    Running   4          25m
kube-proxy-ngwzc                           1/1    Running   0          25m
kube-proxy-q44bs                           1/1    Running   0          19m
kube-proxy-xclk9                           1/1    Running   0          17m
kube-scheduler-master.example.net          1/1    Running   5          25m
NAME           STATUS   ROLES      AGE   VERSION
master.example.net   Ready    control-plane   25m   v1.30.11
worker1.example.net  Ready    <none>     19m   v1.30.11
worker2.example.net  Ready    <none>     17m   v1.30.11
ubuntu@master:~$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Deploy test application on cluster (master node):
`kubectl run nginx --image=nginx`

```

ubuntu@master:~$ kubectl run nginx --image=nginx
pod/nginx created
ubuntu@master:~$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Copying Kubeconfig File to Worker Node:

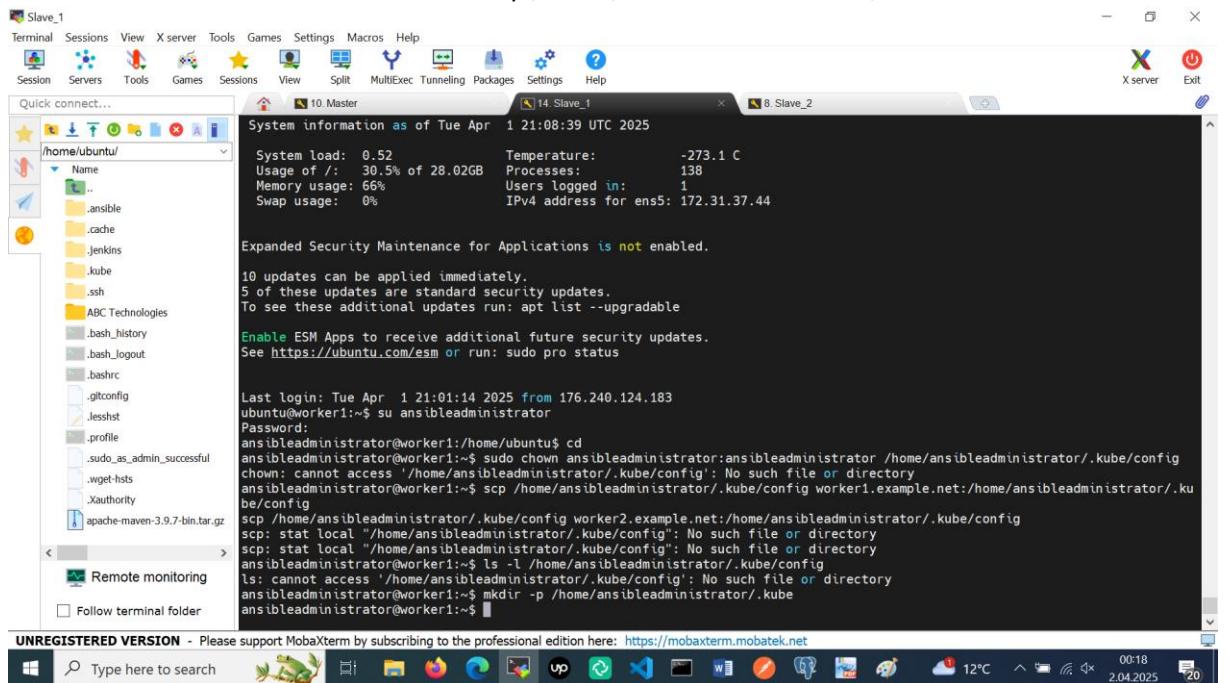
```

scp /home/ansibleadministrator/.kube/config
worker1.example.net:/home/ansibleadministrator/.kube/config
scp /home/ansibleadministrator/.kube/config
worker2.example.net:/home/ansibleadministrator/.kube/config

```

On worker node:

```
ansibleadministrator@worker2:~$ mkdir -p /home/ansibleadministrator/.kube
```



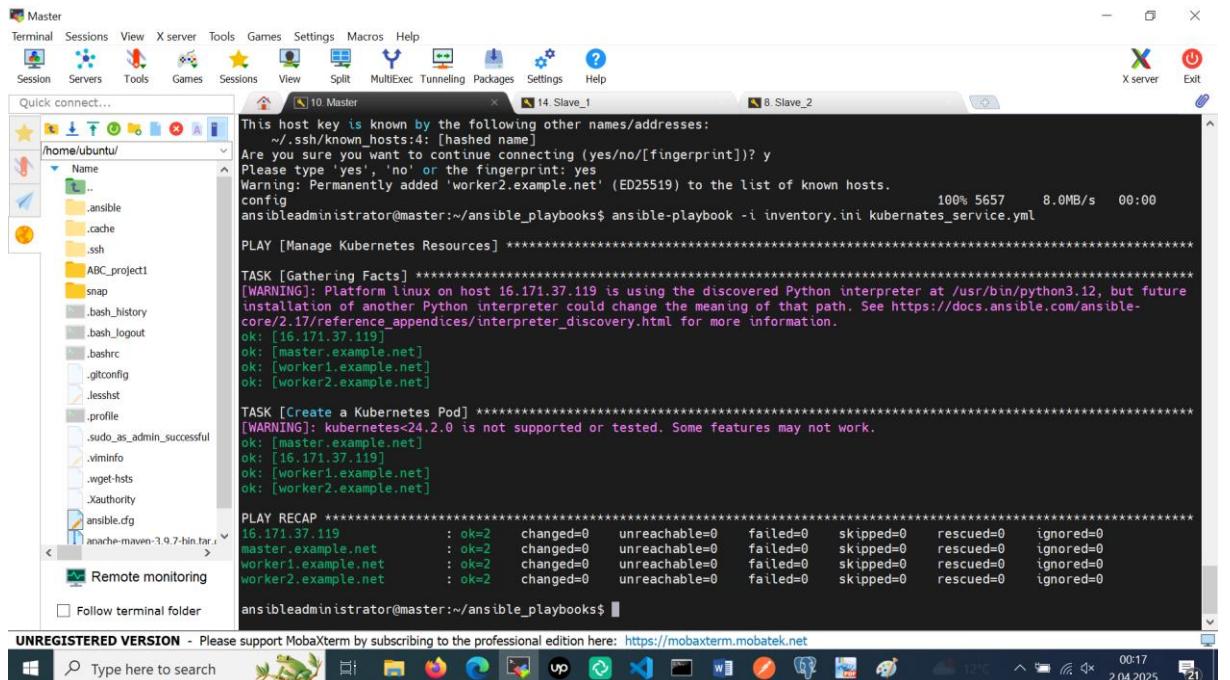
```
System information as of Tue Apr 1 21:08:39 UTC 2025
System load: 0.52 Temperature: -273.1 C
Usage of /: 30.5% of 28.02GB Processes: 138
Memory usage: 66% Users logged in: 1
Swap usage: 0% IPv4 address for ens5: 172.31.37.44

Expanded Security Maintenance for Applications is not enabled.
10 updates can be applied immediately.
5 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

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

Last login: Tue Apr 1 21:01:14 2025 from 176.240.124.183
ubuntu@worker1:~$ su ansibleadministrator
Password:
ansibleadministrator@worker1:/home/ubuntu$ cd
ansibleadministrator@worker1:/home/ubuntu$ sudo chown ansibleadministrator:ansibleadministrator /home/ansibleadministrator/.kube/config
chown: cannot access '/home/ansibleadministrator/.kube/config': No such file or directory
ansibleadministrator@worker1:/home/ubuntu$ scp /home/ansibleadministrator/.kube/config worker1.example.net:/home/ansibleadministrator/.kube/config
scp: stat local "/home/ansibleadministrator/.kube/config": No such file or directory
scp: stat local "/home/ansibleadministrator/.kube/config": No such file or directory
ansibleadministrator@worker1:/home/ubuntu$ ls -l /home/ansibleadministrator/.kube/config
ls: cannot access '/home/ansibleadministrator/.kube/config': No such file or directory
ansibleadministrator@worker1:/home/ubuntu$ mkdir -p /home/ansibleadministrator/.kube
ansibleadministrator@worker1:/home/ubuntu$
```

I used it to install with ansible.



```
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:4: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
^Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'worker2.example.net' (ED25519) to the list of known hosts.
config
ansibleadministrator@master:~/ansible_playbooks$ ansible-playbook -i inventory.ini kubernetes_service.yml
  100% 5657     8.0MB/s  00:00
PLAY [Manage Kubernetes Resources] ****
TASK [Gathering Facts] ****
[WARNING]: Platform linux on host 16.171.37.119 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [16.171.37.119]
ok: [master.example.net]
ok: [worker1.example.net]
ok: [worker2.example.net]

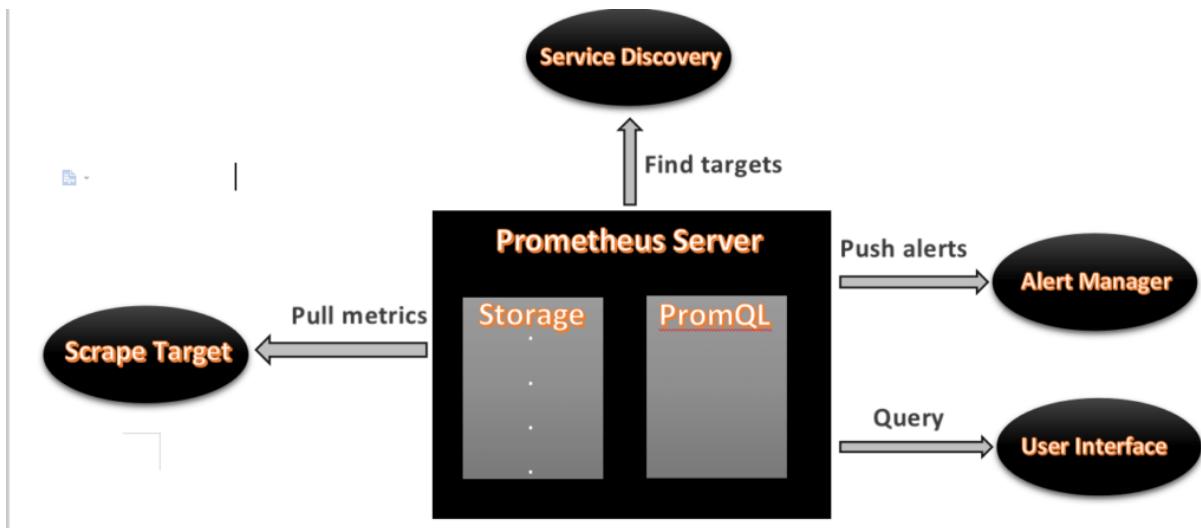
TASK [Create a Kubernetes Pod] ****
[WARNING]: kubernetes<24.2.0 is not supported or tested. Some features may not work.
ok: [master.example.net]
ok: [16.171.37.119]
ok: [worker1.example.net]
ok: [worker2.example.net]

PLAY RECAP ****
16.171.37.119 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
master.example.net : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker1.example.net : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker2.example.net : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansibleadministrator@master:~/ansible_playbooks$
```

8. Prometheus

Prometheus is an **open-source monitoring and alerting system** designed to collect, store, and analyze time-series data. It is widely used for monitoring **servers, applications, containers (Kubernetes, Docker), and cloud environments (AWS, GCP, Azure)**. Prometheus is a **pull-based system**, meaning it actively **scrapes metrics from predefined targets** at specified intervals.



```

#Create a system user for Prometheus using below commands:
sudo useradd --no-create-home --shell /bin/false prometheus

#Create the directories in which we will be storing our configuration files
#and libraries:
sudo mkdir /etc/prometheus
sudo mkdir /var/lib/prometheus

# Set the ownership of the /var/lib/prometheus directory with below command:
sudo chown prometheus:prometheus /var/lib/prometheus

# Download Prometheus Binary File
cd /tmp/

#Download the Prometheus setup using wget
wget
https://github.com/prometheus/prometheus/releases/download/v2.46.0/prometheus-
2.46.0.linux-amd64.tar.gz

# Extract the downloaded file using below command:
tar xvf prometheus-2.46.0.linux-amd64.tar.gz

# Move the configuration file and set the owner to the prometheus user:
cd prometheus-2.46.0.linux-amd64
sudo mv console* /etc/prometheus
sudo mv prometheus.yml /etc/prometheus
sudo chown -R prometheus:prometheus /etc/prometheus

# Move the binaries and set the owner:
sudo mv prometheus /usr/local/bin/
sudo chown prometheus:prometheus /usr/local/bin/prometheus

```

```
#Prometheus configuration file
sudo nano /etc/prometheus/prometheus.yml

#Add the following content to the file:
global:
  scrape_interval: 15s

scrape_configs:
  - job_name: 'prometheus'
    static_configs:
      - targets: ['localhost:9090']
  - job_name: 'worker-nodes'
    static_configs:
      - targets: ['13.60.44.140:9100', '13.60.77.237:9100']
  - job_name: 'kubernetes-nodes'
    kubernetes_sd_configs:
      - role: node

#Create a systemd unit file for Prometheus
sudo nano /etc/systemd/system/prometheus.service

[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
  --config.file /etc/prometheus/prometheus.yml \
  --storage.tsdb.path /var/lib/prometheus/ \
  --web.listen-address=:9090 \
  --web.enable-lifecycle
[Install]
WantedBy=multi-user.target

#reload the systemd daemon
sudo systemctl daemon-reload

#Start the Prometheus service and enable it to start on boot:
sudo systemctl start prometheus
sudo systemctl enable prometheus
sudo systemctl status prometheus
```

```
#Use below command to enable prometheus service in firewall
sudo ufw allow 9090/tcp

# Prometheus service is ready to run and we can access it from any web
browser.

http://server-IP-or-Hostname:9090
```

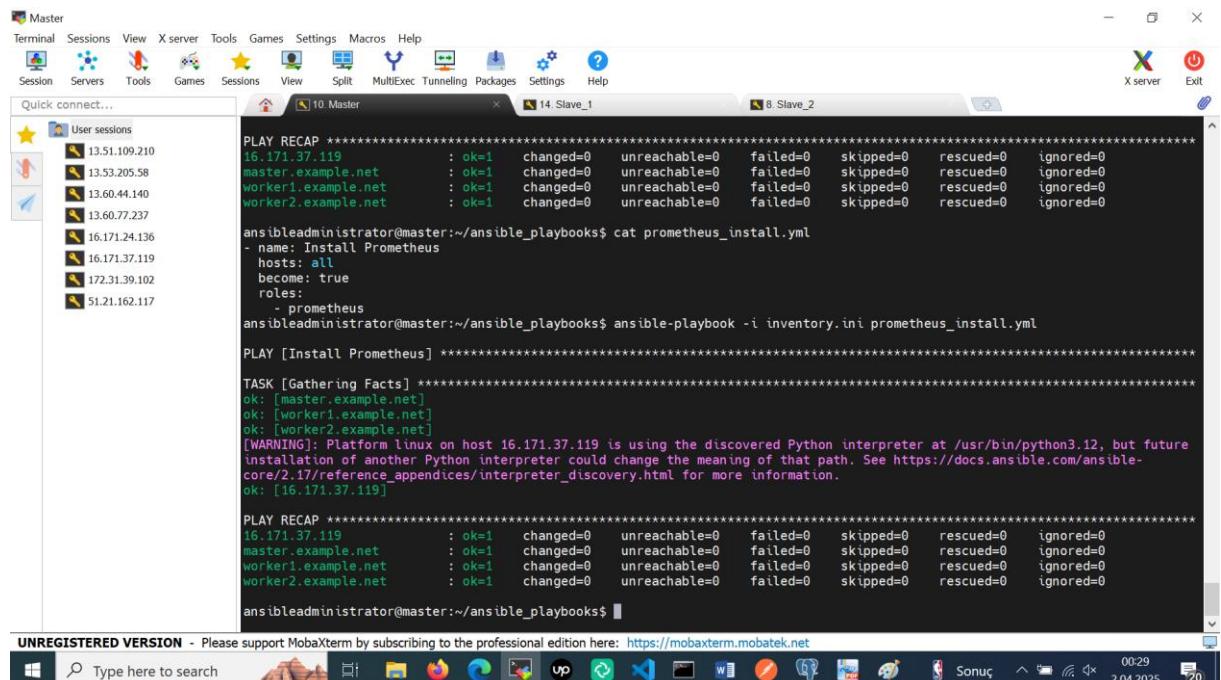
I integrate the prometus with ansible. I run the following script on master node.

```
ansibleadministrator@master:~/ansible_playbooks$ ansible-galaxy init prometheus
- Role prometheus was created successfully
```

```
ansibleadministrator@master:~/ansible_playbooks$ cat prometheus_install.yml
- name: Install Prometheus
  hosts: all
  become: true
  roles:
    - prometheus
ansibleadministrator@master:~/ansible_playbooks$
```

To Install Prometheus on All Nodes:

```
ansible-playbook -i inventory.ini prometheus_install.yml
```



```

- prometheus
ansibleadministrator@master:~/ansible_playbooks$ ansible-playbook -i inventory.ini prometheus_install.yml
PLAY [Install Prometheus] *****
TASK [Gathering Facts] *****
ok: [master.example.net]
ok: [worker1.example.net]
ok: [worker2.example.net]
[WARNING]: Platform linux on host 16.171.37.119 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [16.171.37.119]

PLAY RECAP *****
16.171.37.119 : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
master.example.net : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker1.example.net : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker2.example.net : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansibleadministrator@master:~/ansible_playbooks$ systemctl status prometheus
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-03-31 22:08:48 UTC; 23h ago
     Main PID: 3360897 (prometheus)
        Tasks: 9 (limit: 9365)
       Memory: 99.3M (peak: 127.7M)
          CPU: 1m1n 42.070s
         CGroup: /system.slice/prometheus.service
                  └─3360897 /usr/local/bin/prometheus --config.file /etc/prometheus/prometheus.yml --storage.tsdb.path /var/lib/prometheus

Warning: some journal files were not opened due to insufficient permissions.
lines 1-11/11 (END)

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

<http://16.171.37.119:9090> -> I access to Prometheus interface



Download Node Exporter:

Go to the official release page of Prometheus Node Exporter and copy the link of the latest version of the Node Exporter package according to your OS type.

```
####Install Node Exporter#####
#Create a system user for Node Exporter using below commands:
sudo useradd --no-create-home --shell /bin/false node_exporter
```

```
cd /tmp/  
  
#Download the Node Exporter setup using wget  
wget  
https://github.com/prometheus/node_exporter/releases/download/v1.6.1/node_expo  
rter-1.6.1.linux-amd64.tar.gz  
  
# Extract the downloaded file using below command:  
sudo tar xvfz node_exporter-*.*-amd64.tar.gz  
  
# Move the binary file of node exporter to /usr/local/bin location  
sudo mv node_exporter-*.*-amd64/node_exporter /usr/local/bin/  
  
# Create a node_exporter user to run the node exporter service  
sudo useradd -rs /bin/false node_exporter  
  
# Create a systemd unit file for Node Exporter  
sudo nano /etc/systemd/system/node_exporter.service  
  
[Unit]  
  
Description=Node Exporter  
  
After=network.target  
  
  
[Service]  
  
User=node_exporter  
  
Group=node_exporter  
  
Type=simple  
  
ExecStart=/usr/local/bin/node_exporter  
  
  
[Install]  
  
WantedBy=multi-user.target  
  
# Reload the systemd daemon  
sudo systemctl daemon-reload  
sudo systemctl start node_exporter  
sudo systemctl enable node_exporter  
sudo systemctl status node_exporter
```

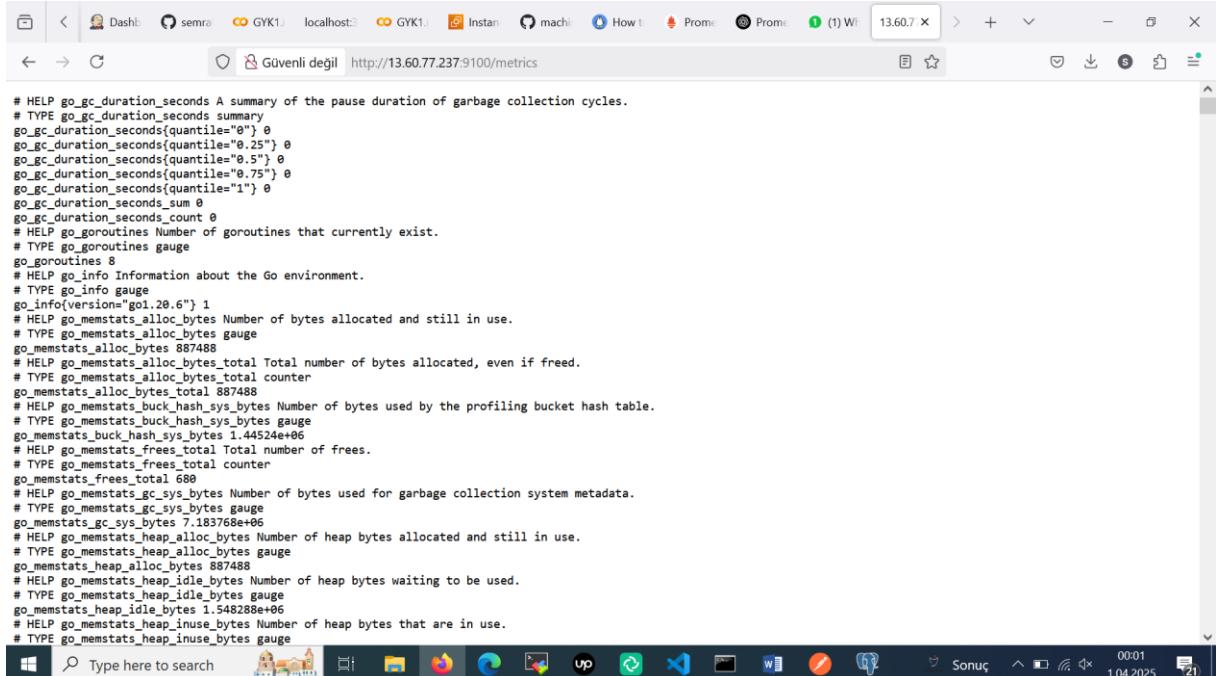
```
ubuntu@worker1:/tmp$ sudo mv node_exporter-*amd64/node_exporter /usr/local/bin/
ubuntu@worker1:/tmp$ sudo useradd -rs /bin/false node_exporter
useradd: user 'node_exporter' already exists
ubuntu@worker1:/tmp$ sudo useradd -rs /bin/false node_exporter
useradd: user 'node_exporter' already exists
ubuntu@worker1:/tmp$ sudo nano /etc/systemd/system/node_exporter.service
ubuntu@worker1:/tmp$ sudo systemctl daemon-reload
sudo systemctl start node_exporter
sudo systemctl enable node_exporter
sudo systemctl status node_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/node_exporter.service → /etc/systemd/system/node_exporter.service.
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-03-31 20:48:47 UTC; 384ms ago
       Main PID: 5936 (node_exporter)
          Tasks: 4 (limit: 1077)
            Memory: 14.7M (peak: 15.0M)
              CPU: 28ms
            CGroup: /system.slice/node_exporter.service
                    └─5936 /usr/local/bin/node_exporter

Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=node_exporter.go:117 level=info c>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.592Z caller=tls_config.go:274 level=info msg=>
Mar 31 20:48:47 worker1.example.net node_exporter[5936]: ts=2025-03-31T20:48:47.594Z caller=tls_config.go:277 level=info msg=>
lines 1-20/20 (END)
ubuntu@worker1:/tmp$
```

```
ubuntu@worker2:/tmp$ sudo systemctl daemon-reload
sudo systemctl start node_exporter
sudo systemctl enable node_exporter
sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-03-31 20:53:10 UTC; 503ms ago
       Main PID: 6874 (node_exporter)
          Tasks: 4 (limit: 1077)
            Memory: 14.9M (peak: 15.1M)
              CPU: 32ms
            CGroup: /system.slice/node_exporter.service
                    └─6874 /usr/local/bin/node_exporter

Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=node_exporter.go:117 level=info c>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.078Z caller=tls_config.go:274 level=info msg=>
Mar 31 20:53:11 worker2.example.net node_exporter[6874]: ts=2025-03-31T20:53:11.080Z caller=tls_config.go:277 level=info msg=>
lines 1-20/20 (END)
ubuntu@worker2:/tmp$
```

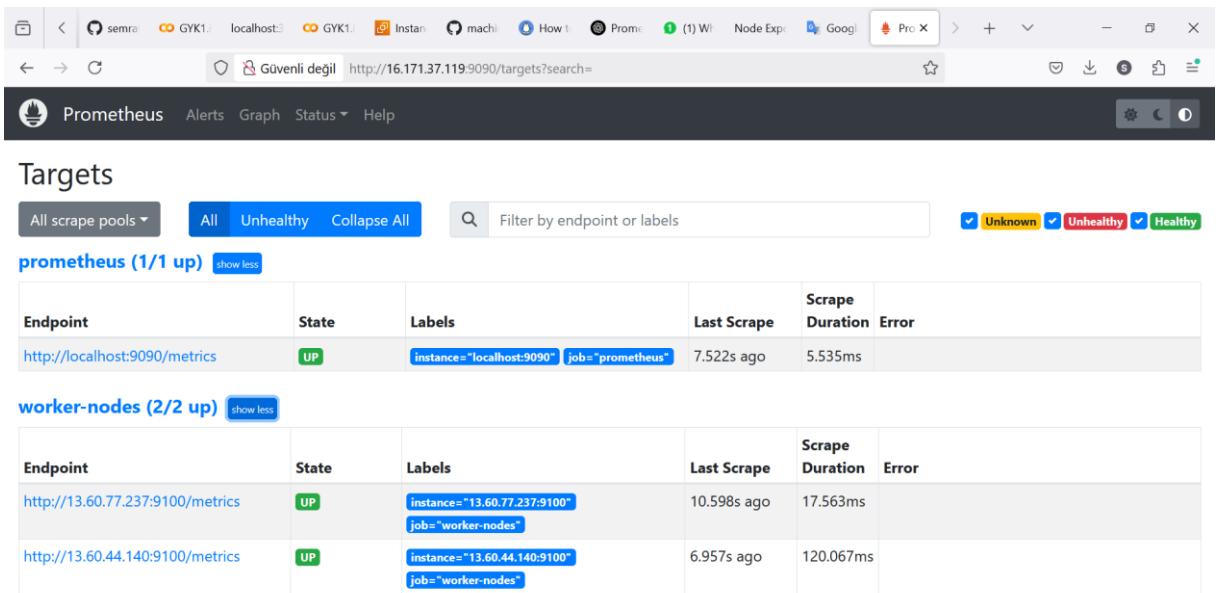
Verify Access



```
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 0
go_gc_duration_seconds{quantile="0.25"} 0
go_gc_duration_seconds{quantile="0.5"} 0
go_gc_duration_seconds{quantile="0.75"} 0
go_gc_duration_seconds{quantile="1"} 0
go_gc_duration_seconds_sum 0
go_gc_duration_seconds_count 0
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 8
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.26.6"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 887488
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 887488
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 1.445248e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 688
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 7.183768e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 887488
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 1.548288e+06
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
```

Configure the Node Exporter as a Prometheus target:

Now to scrape the node_exporter lets instruct the Prometheus by making a minor change in prometheus.yml file. So go to etc/prometheus and open prometheus.yml



Targets

All scrape pools ▾ All Unhealthy Collapse All Filter by endpoint or labels Unknown Unhealthy Healthy

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	7.522s ago	5.535ms	

worker-nodes (2/2 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://13.60.77.237:9100/metrics	UP	instance="13.60.77.237:9100" job="worker-nodes"	10.598s ago	17.563ms	
http://13.60.44.140:9100/metrics	UP	instance="13.60.44.140:9100" job="worker-nodes"	6.957s ago	120.067ms	



I can create a new playbook that automates the Node Exporter installation process.

```
node_exporter.yml Playbook
```

```
---
- name: Install Node Exporter on Ubuntu
  hosts: all # Target all nodes or specify a group like 'workers' or 'masters'
  become: true
  tasks:
    - name: Download Node Exporter tarball
      get_url:
        url:
          "https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz"
        dest: "/tmp/node_exporter-1.3.1.linux-amd64.tar.gz"

    - name: Extract Node Exporter tarball
      unarchive:
        src: "/tmp/node_exporter-1.3.1.linux-amd64.tar.gz"
        dest: "/opt/"
        remote_src: yes

    - name: Create Node Exporter systemd service
      copy:
        dest: "/etc/systemd/system/node_exporter.service"
        content: |
          [Unit]
          Description=Prometheus Node Exporter
          Documentation=https://prometheus.io/docs/instrumenting/exporters/
          After=network.target

          [Service]
          User=nobody
          Group=nogroup
          ExecStart=/opt/node_exporter-1.3.1.linux-amd64/node_exporter

          [Install]
          WantedBy=multi-user.target

    - name: Reload systemd to register the Node Exporter service
      systemd:
        daemon_reload: yes

    - name: Start Node Exporter service
      systemd:
        name: node_exporter
        state: started
        enabled: yes
```

```

- name: Verify Node Exporter is running
  command: systemctl status node_exporter
  register: node_exporter_status
  failed_when: "'active (running)' not in node_exporter_status.stdout"
  changed_when: false

```

Run Playbook:

```
ansible-playbook -i inventory.ini node_exporter.yml
```

```

Master
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Session 10 Master 14 Slave_1 8 Slave_2
X server Exit
Quick connect...
changed: [master.example.net]
TASK [Create Node Exporter systemd service] *****
changed: [worker1.example.net]
changed: [worker2.example.net]
changed: [16.171.37.119]
ok: [master.example.net]

TASK [Reload systemd to register the Node Exporter service] *****
ok: [master.example.net]
ok: [worker1.example.net]
ok: [worker2.example.net]
ok: [16.171.37.119]

TASK [Start Node Exporter service] *****
ok: [worker1.example.net]
ok: [worker2.example.net]
changed: [16.171.37.119]
changed: [master.example.net]

TASK [Verify Node Exporter is running] *****
ok: [master.example.net]
ok: [16.171.37.119]
ok: [worker1.example.net]
ok: [worker2.example.net]

PLAY RECAP *****
16.171.37.119 : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
master.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker1.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker2.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ansibleleadadministrator@master:~/ansible_playbooks$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Verify Installation with ansible:

```

Master
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Session 10 Master 14 Slave_1 8 Slave_2
X server Exit
Quick connect...
ok: [worker2.example.net]
ok: [16.171.37.119]

TASK [Start Node Exporter service] *****
ok: [worker1.example.net]
ok: [worker2.example.net]
changed: [16.171.37.119]
changed: [master.example.net]

TASK [Verify Node Exporter is running] *****
ok: [master.example.net]
ok: [16.171.37.119]
ok: [worker1.example.net]
ok: [worker2.example.net]

PLAY RECAP *****
16.171.37.119 : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
master.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker1.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker2.example.net : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ansibleleadadministrator@master:~/ansible_playbooks$ systemctl status node_exporter
● node_exporter.service - Prometheus Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-04-01 21:33:10 UTC; 3min 8s ago
       Docs: https://prometheus.io/docs/instrumenting/exporters/
     Main PID: 237498 (node_exporter)
        Tasks: 5 (limit: 9365)
      Memory: 2.3M (peak: 2.6M)
        CPU: 15ms
       CGroup: /system.slice/node_exporter.service
               └─237498 /opt/node_exporter-1.3.1.linux-amd64/node_exporter
ansibleleadadministrator@master:~/ansible_playbooks$ 

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

9. Grafana

- Grafana is a free and open source visualization tool mostly used with Prometheus to which monitor metrics.
- Grafana provides various dashboards, charts, graphs, alerts for the particular data source.
- Grafana allows us to query, visualize, explore metrics and set alerts for the data source which can be a system, server, nodes, cluster, etc.
- We can also create our own dynamic dashboard for visualization and monitoring.
- We can save the dashboard and can even share with our team members which is one of the main advantage of Grafana.

Now lets Install Grafana for wonderful dashboards and data visualization for monitoring systems, servers, services, etc. We can install grafana on Ubuntu either by downloading .deb package from Grafana Download page or using APT which is more easier.Add the Grafana GPG key in Ubuntu using wget

Install Grafana:

```
#### Install Grafana ####
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -

##Next, add the Grafana repository to your APT sources:
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"

#Refresh your APT cache to update your package lists:
sudo apt update

#Install Grafana using the following command:
sudo apt install grafana

#Start the Grafana service and enable it to start on boot:
sudo systemctl start grafana-server

#Enable the Grafana service to start on boot:
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@master:/tmp$ sudo systemctl start grafana-server
ubuntu@master:/tmp$ sudo systemctl enable grafana-server
sudo systemctl status grafana-server
Synchronizing state of grafana-server.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /usr/lib/systemd/system/grafana-server.service.

● grafana-server.service - Grafana instance
    Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; preset: enabled)
    Active: active (running) since Mon 2025-03-31 22:20:11 UTC; 9s ago
      Docs: http://docs.grafana.org
        Main PID: 3370761 (grafana)
          Tasks: 18 (limit: 9365)
            Memory: 95.9M (peak: 96.6M)
              CPU: 5.312s
            CGroup: /system.slice/grafana-server.service
                    └─3370761 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=/run/grafana/grafana.pid

Mar 31 22:20:20 master.example.net grafana[3370761]: logger=grafana-apiserver t=2025-03-31T22:20:20.882392018Z level=info msg="Starting Grafana API server"
Mar 31 22:20:20 master.example.net grafana[3370761]: logger=grafana-apiserver t=2025-03-31T22:20:20.894257319Z level=info msg="Starting Grafana API server"
Mar 31 22:20:20 master.example.net grafana[3370761]: logger=grafana-apiserver t=2025-03-31T22:20:20.904338367Z level=info msg="Starting Grafana API server"
Mar 31 22:20:20 master.example.net grafana[3370761]: logger=grafana-apiserver t=2025-03-31T22:20:20.909217255Z level=info msg="Starting Grafana API server"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=uninstaller.fs t=2025-03-31T22:20:21.042640259Z level=info msg="Downloaded Grafana API server configuration file"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=app-registry t=2025-03-31T22:20:21.098617637Z level=info msg="App registry initialized"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=plugins.registration t=2025-03-31T22:20:21.184855306Z level=info msg="Plugin registration initialized"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=plugin.backgroundinstaller t=2025-03-31T22:20:21.185223242Z level=info msg="Background plugin installer initialized"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=plugin.backgroundinstaller t=2025-03-31T22:20:21.185336367Z level=info msg="Background plugin installer initialized"
Mar 31 22:20:21 master.example.net grafana[3370761]: logger=plugin.installer t=2025-03-31T22:20:21.335558968Z level=info msg="Background plugin installer initialized"

ubuntu@master:/tmp$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Grafana Dashboard open your favorite browser, type server IP or Name followed by grafana default port 3000.

Create a Playbook to Install Grafana:

Nano grafana_install.yml

```
GNU nano 7.2                                     grafana_install.yml *
```

```
---
- name: Install Grafana on Ubuntu
  hosts: all # Can be adjusted to specific groups like 'workers' or 'masters'
  become: true
  tasks:
    - name: Add Grafana APT repository
      apt_repository:
        repo: "deb https://packages.grafana.com/oss/deb stable main"
        state: present
        filename: "grafana"

    - name: Install Grafana package
      apt:
        name: grafana
        state: latest
        update_cache: yes

    - name: Enable and start Grafana service
      systemd:
        name: grafana-server
        state: started
        enabled: yes

    - name: Verify Grafana is running
      command: systemctl status grafana-server
      register: grafana_status
      failed_when: "'active (running)' not in grafana_status.stdout"
      changed_when: false
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

```

^C [ERROR]: User interrupted execution
ansibleAdministrator@master:~/ansible_playbooks$ ansible-playbook -i inventory.ini grafana_install.yml

PLAY [Install Grafana on Ubuntu] ****
TASK [Gathering Facts] ****
ok: [worker2.example.net]
ok: [worker1.example.net]
ok: [master.example.net]
[WARNING]: Platform linux on host 16.171.37.119 is using the discovered Python interpreter at /usr/bin/python3.12, but future
installation of another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [16.171.37.119]

TASK [Add Grafana APT repository] ****
ok: [worker2.example.net]
ok: [master.example.net]
ok: [16.171.37.119]
ok: [worker1.example.net]

TASK [Install Grafana package] ****
ok: [16.171.37.119]
ok: [master.example.net]
changed: [worker2.example.net]
ok: [worker1.example.net]

TASK [Enable and start Grafana service] ****
ok: [master.example.net]
ok: [16.171.37.119]
changed: [worker1.example.net]
changed: [worker2.example.net]

TASK [Verify Grafana is running] ****

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Check prometheus status on ansible :

sudo systemctl status prometheus

```

ok: [master.example.net]
ok: [16.171.37.119]
ok: [worker2.example.net]
ok: [worker1.example.net]

PLAY RECAP ****
16.171.37.119 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
master.example.net : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker1.example.net : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
worker2.example.net : ok=5    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ansibleAdministrator@master:~/ansible_playbooks$ sudo systemctl status prometheus
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-03-31 22:08:48 UTC; 23h ago
       Main PID: 3360897 (prometheus)
          Tasks: 9 (limit: 9365)
         Memory: 110.0M (peak: 127.7M)
            CPU: 1m1n 44.734s
           CGroup: /system.slice/prometheus.service
                   └─3360897 /usr/local/bin/prometheus --config.file /etc/prometheus/prometheus.yml --storage.tsdb.path /var/lib/pr

Apr 01 17:00:07 master.example.net prometheus[3360897]: ts=2025-04-01T17:00:07.390Z caller=compact.go:464 level=info component=prometheus
Apr 01 17:00:07 master.example.net prometheus[3360897]: ts=2025-04-01T17:00:07.396Z caller=db.go:1617 level=info component=tsdb
Apr 01 17:00:07 master.example.net prometheus[3360897]: ts=2025-04-01T17:00:07.400Z caller=db.go:1617 level=info component=tsdb
Apr 01 17:00:07 master.example.net prometheus[3360897]: ts=2025-04-01T17:00:07.403Z caller=db.go:1617 level=info component=tsdb
Apr 01 19:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T19:00:10.192Z caller=compact.go:523 level=info component=compact
Apr 01 19:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T19:00:10.197Z caller=head.go:1293 level=info component=compact
Apr 01 21:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T21:00:10.187Z caller=compact.go:523 level=info component=compact
Apr 01 21:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T21:00:10.203Z caller=head.go:1293 level=info component=compact
Apr 01 21:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T21:00:10.204Z caller=checkpoint.go:100 level=info component=checkpoint
Apr 01 21:00:10 master.example.net prometheus[3360897]: ts=2025-04-01T21:00:10.332Z caller=head.go:1261 level=info component=compact

```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

```

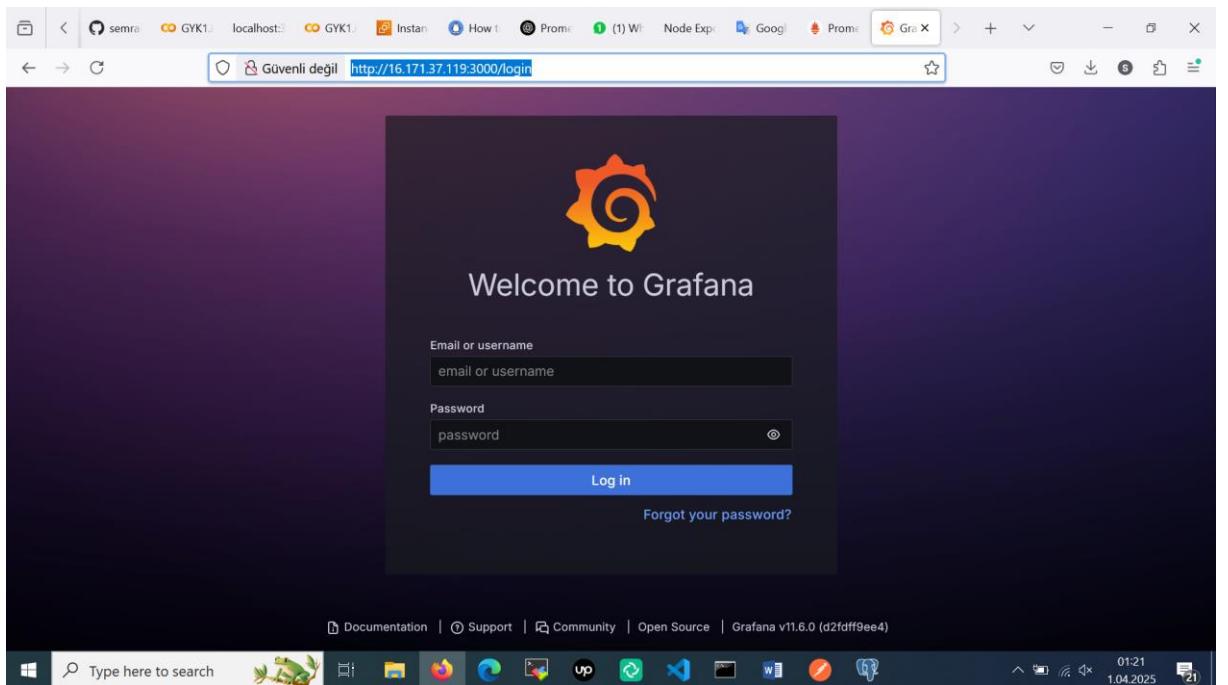
sudo ufw allow 3000/tcp
http://16.171.37.119:3000/login

```

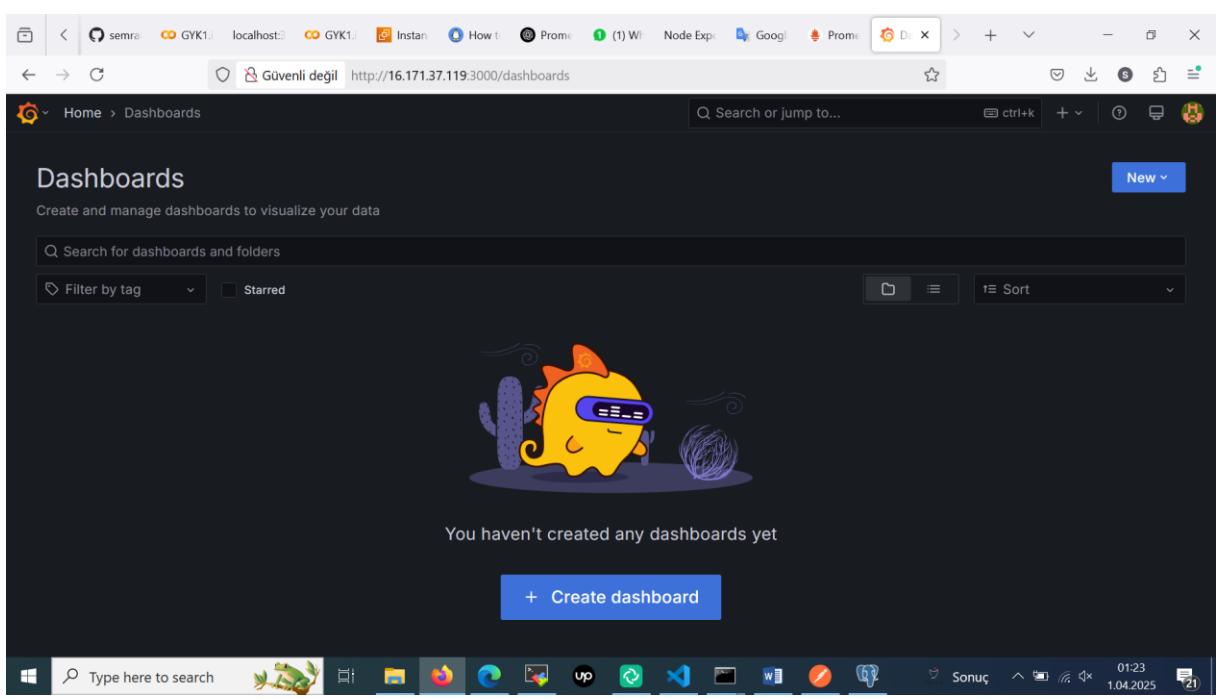
```

#Username - Admin
#Password - Admin

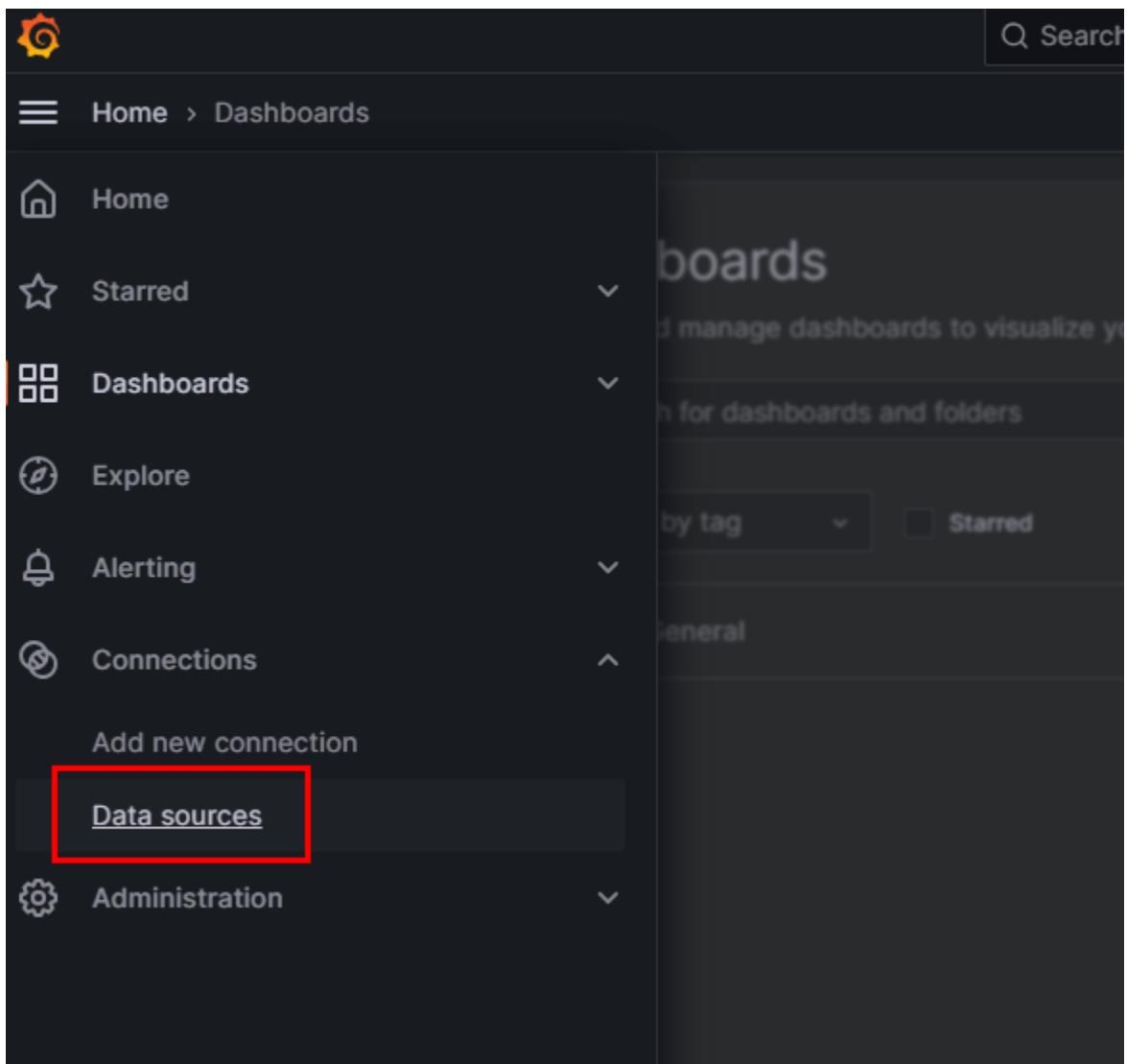
```



Now here you can see Home Dashboard page of Grafana



Configure Prometheus as Grafana DataSource:



Now lets click on **Add Data sources** and select **Prometheus**

Now configure Prometheus data source by providing Prometheus URL

Configure your Prometheus data source below
Or skip the effort and get Prometheus (and Loki) as fully-managed, scalable, and hosted data sources from Grafana Labs with the [free-forever Grafana Cloud plan](#).

Name: prometheus Default:

Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, [view the documentation](#).

Fields marked with * are required

Connection

Prometheus server URL *: http://16.171.37.119:9090/

Now click on **Save & test** so it will prompt a message **Data Source is working**.

Now provide the Grafana.com Dashboard ID which is **1860** and click on **Load**

Import dashboard

Import dashboard from file or Grafana.com

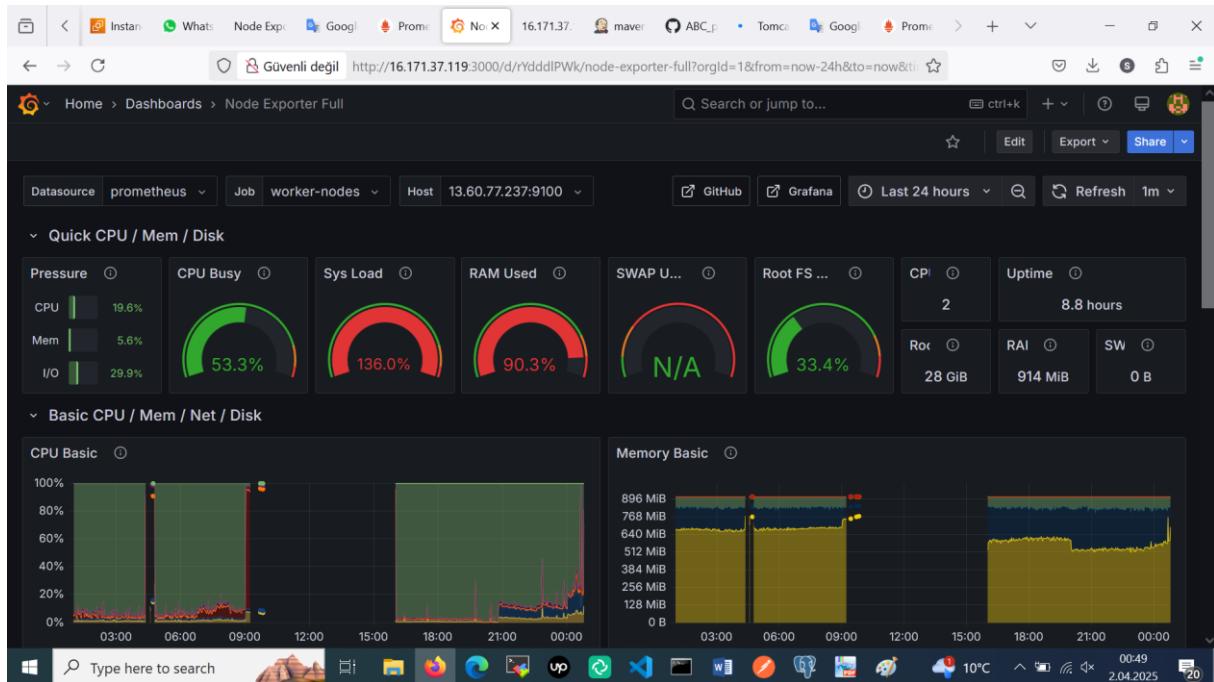
Upload dashboard JSON file
Drag and drop here or click to browse
Accepted file types: json, .txt

Find and import dashboards for common applications at [grafana.com/dashboards](#)

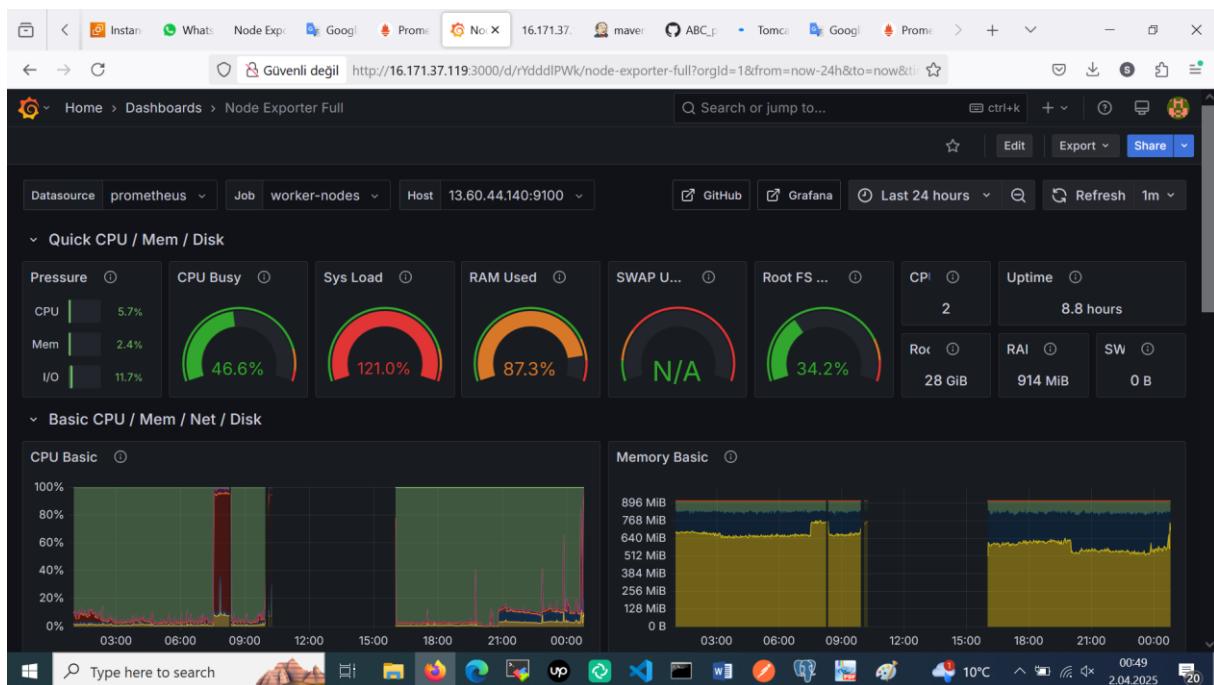
1860

Import via dashboard JSON model

```
{
  "title": "Example - Repeating Dictionary variables",
  "uid": "_OHnEoN4z",
  "panels": [...]
}
```



I changed host.



13.60.44.140 and 13.60.77.237 ip address belongs to worker nodes.

