



INDIAN INSTITUTE OF
INFORMATION
TECHNOLOGY

DevOps(DS457)

Assignment - 2

Jenkins Master-Slave

Submitted to
Dr.Uma.S

Submitted by
Team13

M.P. Bharath(18BCS057)
T.Umesh Anand babu(18BCS105)
M.Venkata Kalyan Babu(18BCS049)
Y.Mokshith ramendra(18BCS112)

Setting up jenkins master-slave architecture for testing and production servers

1. Setting up master and slave servers in aws.

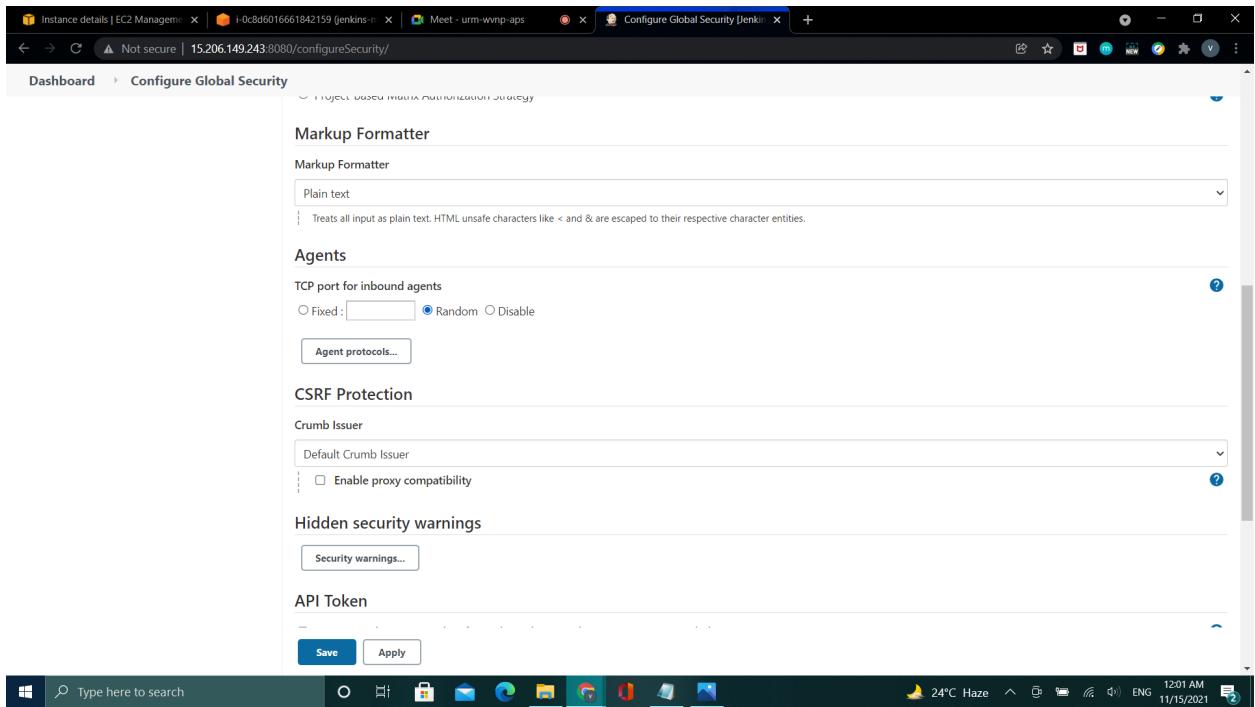
- Setup 3 nodes of type t2.micro of ubuntu 20.04 os with security group instructions to allow all traffic and keep all other details as default.
- NOTE : we will be installing jenkins only on master node and monitor slave nodes using master node.
- On the master node execute these instructions.

```
➤ sudo apt install openjdk-8-jdk
➤ wget -q -O -
  https://pkg.jenkins.io/debian/jenkins.io.key | 
  sudo apt-key add -
➤ sudo sh -c 'echo deb
  http://pkg.jenkins.io/debian-stable binary/ >
  /etc/apt/sources.list.d/jenkins.list'
➤ sudo apt update
➤ sudo apt install jenkins
➤ sudo systemctl start jenkins
➤ sudo systemctl status jenkins
```

- On completion of above instructions access this server on a browser using a public ipv4 address with port number 8080 i.e ipaddress:8080.

2. To configure jenkins

- Goto manage jenkins -> configure global settings -> Agents -> random -> click on save.
- Goto manage jenkins -> manage nodes -> new node.
- Create 2 slave nodes named slave-1 and slave-2 respectively with permanent agent option enabled and with the launch method as configure when master is triggered. Also add custom workdir path to be /home/ubuntu/jenkins and click on save.



- click on respective agents and download their agent.jar executable file.
- Now these agent.jar files are to be sent to the respective slave nodes that were created from above points. For ubuntu os run the following command.

3. Connect to slave nodes from master node.

- Goto manage jenkins -> manage nodes -> click on respective slave and copy paste the code under Run from agent command line.
- Now copy paste the code in slave nodes.

```
> sudo apt-get update  
> sudo apt install openjdk-8-jdk
```

Instance details | EC2 Manager | i-0c8d6016661842159 [jenkins-r] | Meet - urm-wwnp-aps | slave-1 [Jenkins]

← → C Not secure | 15.206.149.243:8080/computer/slave-1/

Jenkins

Dashboard > Nodes > slave-1

Agent slave-1

Connect agent to Jenkins one of these ways:

- Launch agent from browser
- Run from agent command line:

```
java -jar agent.jar -jnlpUrl http://15.206.149.243:8080/computer/slave-1/jenkins-agent.jnlp -secret 1e896bf358e978e59524c9a76a39940bcd1b78e7b6857dea42f816212a56c49f -workDir "/home/ubuntu/jenkins"
```

Run from agent command line, with the secret stored in a file:

```
echo 1e896bf358e978e59524c9a76a39940bcd1b78e7b6857dea42f816212a56c49f > secret-file
java -jar agent.jar -jnlpUrl http://15.206.149.243:8080/computer/slave-1/jenkins-agent.jnlp -secret @secret-file -workDir "/home/ubuntu/jenkins"
```

Projects tied to slave-1

None

REST API Jenkins 2.303.3

Type here to search

24°C Haze ENG 12:04 AM 11/15/2021

INFO: Both error and output log will be printed to /home/ubuntu/jenkins/remoting
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: slave-1
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener <init>
INFO: Jenkins agent is running in headless mode.
Nov 14, 2021 6:46:51 PM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 4.10.1
Nov 14, 2021 6:46:51 PM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Locating server among [http://15.206.149.243:8080]
Nov 14, 2021 6:46:51 PM org.jenkinsci.remoting.engine.JnlpAgentEndpointResolver resolve
INFO: Remoting server accepts the following protocols: [JNLP4-connect, Ping]
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Agent discovery successful
Agent address: 15.206.149.243
Agent port: 42613
Identity: d1:a8:63:b0:f4:f2:2f:49:13:e1:81:31:c1:46:54:e0
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Handshaking
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Connecting to 15.206.149.243:42613
Nov 14, 2021 6:46:51 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Trying protocol: JNLP4-connect
Nov 14, 2021 6:46:51 PM org.jenkinsci.remoting.protocol.impl.BIONetworkLayers\$Reader run
INFO: Waiting for ProtocolStack to start.
Nov 14, 2021 6:46:52 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Remote identity confirmed: d1:a8:63:b0:f4:f2:2f:49:13:e1:81:31:c1:46:54:e0
Nov 14, 2021 6:46:53 PM hudson.remoting.jnlp.Main\$CuiListener status
INFO: Connected

i-088cb38134fbba92 (jenkins-slave1)

Public IPs: 3.110.37.36 Private IPs: 172.31.14.95

agent.jar agent.jar

← → C Not secure | 15.206.149.243:8080/computer/slave-1/

24°C Haze ENG 12:16 AM 11/15/2021

```

INFO: Both error and output log will be printed to /home/ubuntu/jenkins/remoting
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: slave-2
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main$CuiListener <init>
INFO: Jenkins agent is running in headless mode
Nov 14, 2021 6:49:36 PM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 4.10.1
Nov 14, 2021 6:49:36 PM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/jenkins/remoting as a remoting work directory
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Locating server among [http://15.206.149.243:8080/]
Nov 14, 2021 6:49:36 PM org.jenkinsci.remoting.engine.JnlpAgentEndpointResolver resolve
INFO: Remoting server accepts the following protocols: [JNLP4-connect, Ping]
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Agent discovery successful
  Agent address: 15.206.149.243
  Agent port: 42613
  Identity: d1:a8:63:b0:f4:f2:2f:49:13:e1:81:31:c1:46:54:e0
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Handshaking
Nov 14, 2021 6:49:36 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connecting to 15.206.149.243:42613
Nov 14, 2021 6:49:37 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Trying protocol: JNLP4-connect
Nov 14, 2021 6:49:37 PM org.jenkinsci.remoting.protocol.impl.BIONetworkLayer$Reader run
INFO: Waiting for ProtocolStack to start.
Nov 14, 2021 6:49:37 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Remote identity confirmed: d1:a8:63:b0:f4:f2:2f:49:13:e1:81:31:c1:46:54:e0
Nov 14, 2021 6:49:38 PM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connected

```

i-015848314423b6b26 (jenkins-slave2)

Public IPs: 13.232.158.141 Private IPs: 172.31.2.206



- Now verify that the master and slave nodes are connected.

4. Configure Jenkins to build a project on slave-1(Test server) if successful build on slave-2(Production server).

- NOTE : make sure to duplicate sessions in the slave node so that connection doesn't terminate.
- Install Docker on both the slaves.

➤ `sudo apt-get install docker.io`

- verify docker installation using

➤ `docker --version`

- Head Back to jenkins. create new job -> make it new freestyle project -> name it slave-1. Now under configure in the general section click on github project and copy paste the url of github

project. Also enable the Restrict where this project can be run and type in slave-1 under label expression section. Go to source code management and enable git.

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Linux (amd64)	In sync	5.38 GB	0 B	5.38 GB	0ms
	slave-1	Linux (amd64)	In sync	5.73 GB	0 B	5.73 GB	167ms
	slave-2	Linux (amd64)	In sync	5.73 GB	0 B	5.73 GB	54ms

Data obtained 16 sec 16 sec 16 sec 16 sec 16 sec 16 sec

Build Queue
No builds in the queue.

Build Executor Status

- master
 - 1 idle
 - 2 idle
- slave-1
 - 1 idle
- slave-2
 - agent.jar
 - agent.jar

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

Build Environment

- Delete workspace before build starts
- Use secret text(s) or file(s)
- Abort the build if it's stuck
- Add timestamps to the Console Output
- Inspect build log for published Gradle build scans
- With Ant

Build

Execute shell

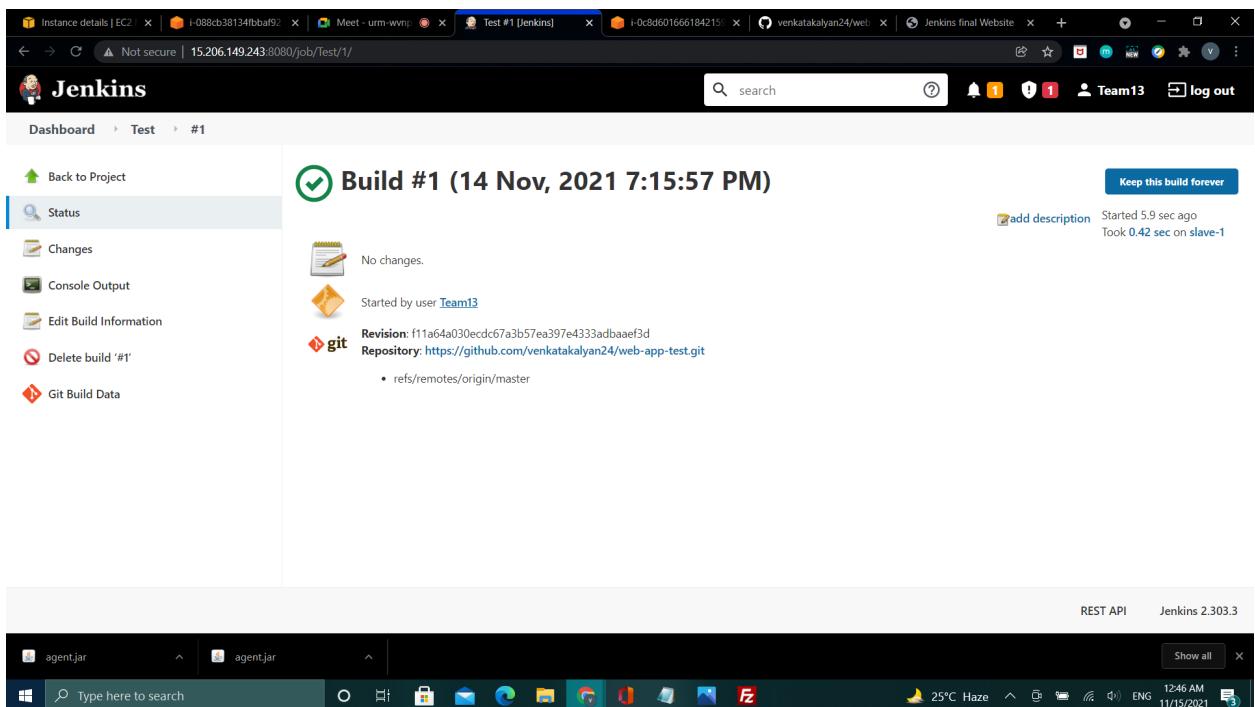
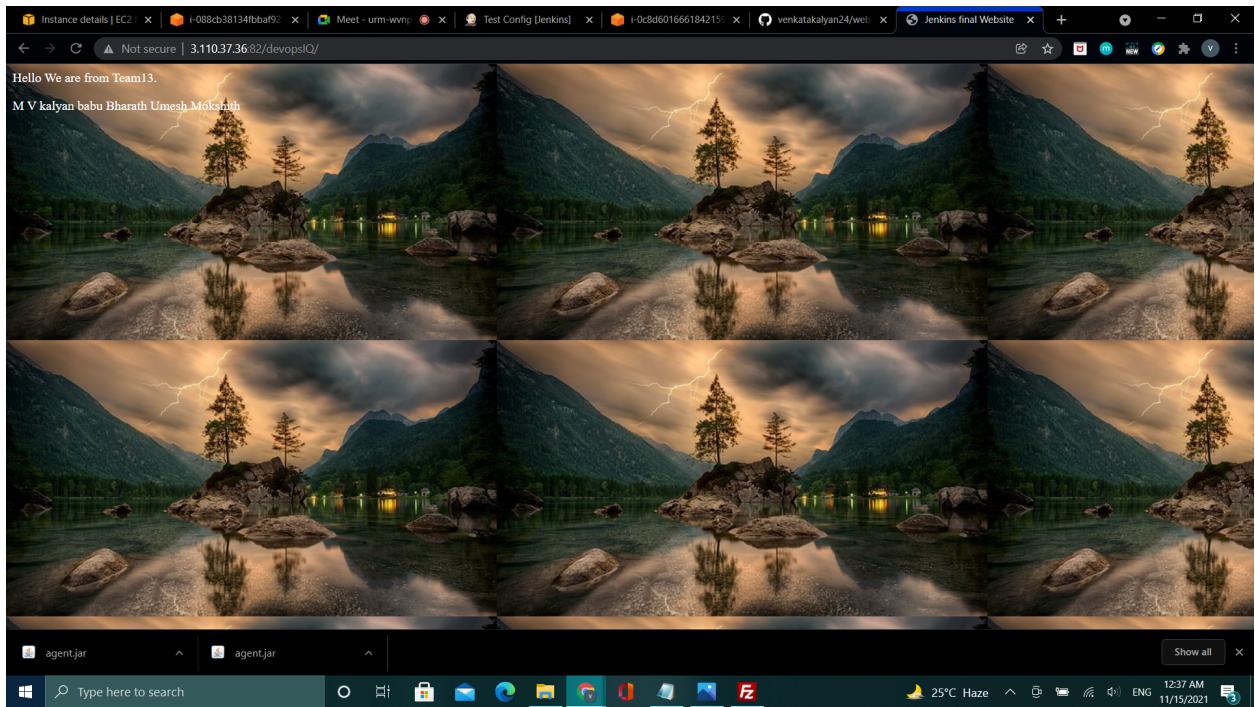
Command

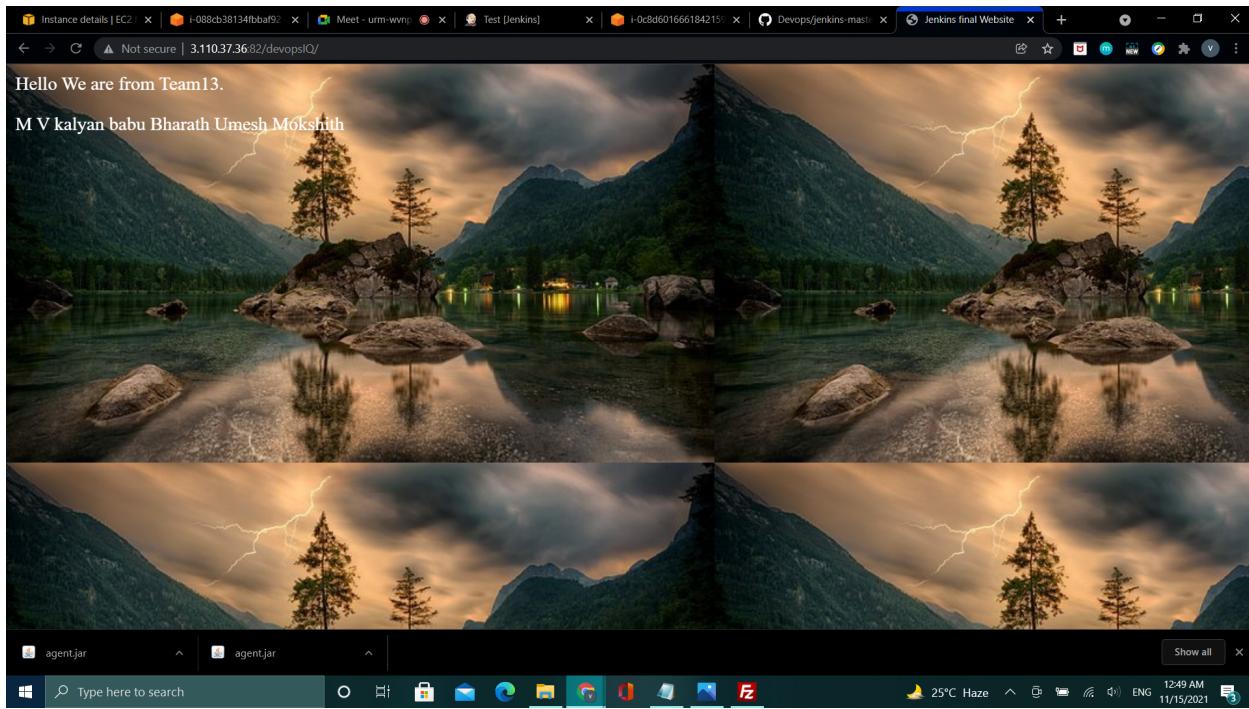
```
sudo docker rm -f $(sudo docker ps -a -q)
sudo docker build /home/ubuntu/workspace/Test -t test
sudo docker run -it -p 82:80 -d test
```

See the list of available environment variables

Add build step **Save** **Apply**

Post-build Actions



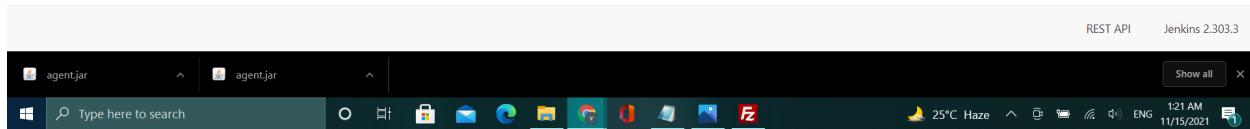
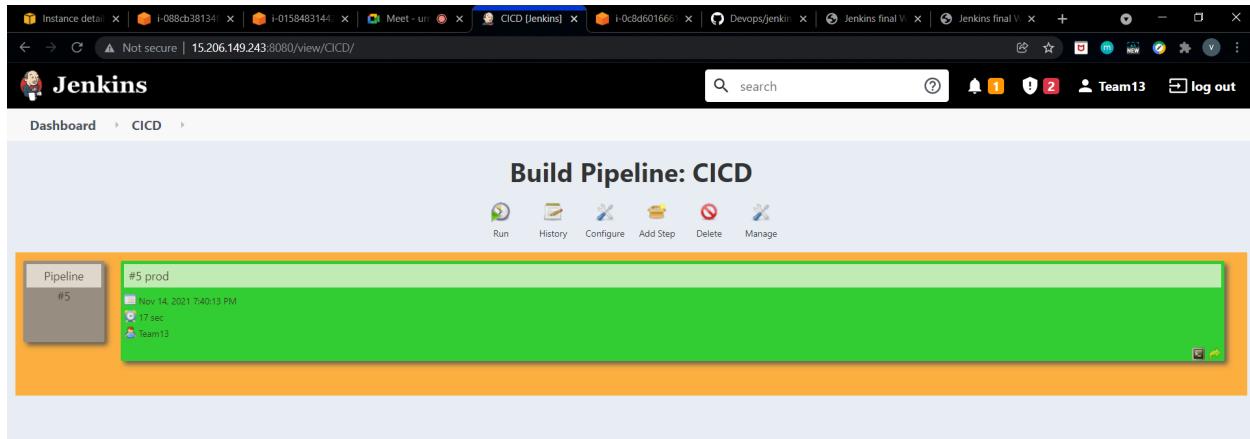


- Under the build section click Add build step -> Execute shell. Now run the below commands.
 - sudo docker rm -f \$(sudo docker ps -a -q)
 - sudo docker build /home/ubuntu/workspace/Test -t test
 - sudo docker run -it -p 82:80 -d test
- NOTE : Make sure to run a custom container on the slave node before executing above commands. The above steps are to be executed for both test and production server.
- Now click on build.
- To check the above build step head over to :/
- Now we need to configure the servers in such a way that after testing production is built. To enable this Goto the test(freestyle project of

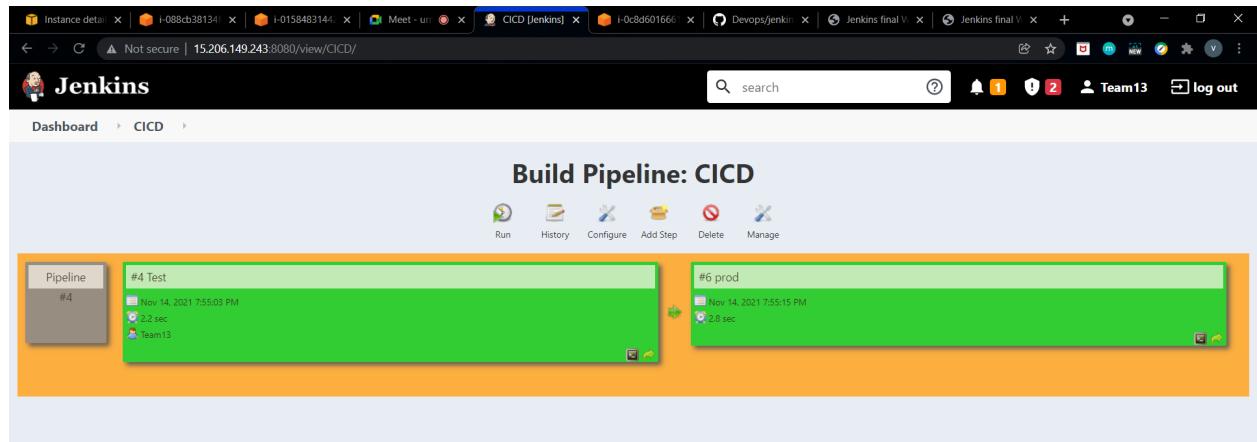
slave-1) and under the post build actions click on build other projects and enter production(freestyle project of slave-2).

5. Build in the CI-CD pipeline for test and production servers.

- Head over to manage jenkins -> manage plugins -> available. Now search for build pipeline, install the plugin.
- In the jenkins home page click on the + sign near the All.
- Now click the Build pipeline view, name the view as CiCd. Also under Build pipeline view title give CiCd and with other options as default and click save.



- Under the Configure section of the CiCd pipeline, make Pipeline flow -> select initial job -> test.
- Now click on Run to run the respective jobs.



- Verify the deployment into production and test servers by heading to (<ipv4addressoftest/production>/)

6. Using Github-Webhook to reflect changes made to github project in the test and production servers.

- Goto jenkins dashboard -> test -> configure -> build triggers. And enable the github hook trigger for GIT-Scm polling option and click save.
- Head over to your github webapp project and goto settings -> Webhooks -> Add webhook. Now copy paste the ip address of your jenkins server(master node) under the payload url section and click addwebhook.
- NOTE : Verify that you get tick mark under the webhook section for the url that you copy pasted.
- Now lets trigger a build by committing port changes to the github webapp project and see the changes made to test and production server.

```
➤ git clone <githuburl>
➤ cd <projectfolder>
➤ ls
➤ nano index.html // make changes to this file
➤ git add .
➤ git commit -m "message"
➤ git push origin master
```

github.com/venkatakalyan24/web-app-test/settings/hooks

Search or jump to... Pull requests Issues Marketplace Explore

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Projects Wiki Security Insights Settings

Webhooks

Add webhook

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://15.206.149.243:8080/github... (push)

Edit Delete

Show all

https://github.com/venkatakalyan24/web-app-test/settings/hooks/328458427

agent.jar agent.jar

Type here to search

25°C Haze 1:30 AM 11/15/2021

```
git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 1 insertion(+)
ubuntu@ip-172-31-12-134:~/web-app-test/devopsIQ$ git push origin master
Username for 'https://github.com': venkatakalyan24
Password for 'https://venkatakalyan24@github.com':
remote: Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.
remote: Please see https://github.blog/2020-12-15-token-authentication-requirements-for-git-operations/ for more information.
fatal: Authentication failed for 'https://github.com/venkatakalyan24/web-app-test.git'
ubuntu@ip-172-31-12-134:~/web-app-test/devopsIQ$ git push origin master
Username for 'https://github.com': venkatakalyan24
Password for 'https://venkatakalyan24@github.com':
remote: Support for password authentication was removed on August 13, 2021. Please use a personal access token instead.
remote: Please see https://github.blog/2020-12-15-token-authentication-requirements-for-git-operations/ for more information.
fatal: Authentication failed for 'https://github.com/venkatakalyan24/web-app-test.git'
ubuntu@ip-172-31-12-134:~/web-app-test/devopsIQ$ git push origin master
Username for 'https://github.com': venkatakalyan24
Password for 'https://venkatakalyan24@github.com':
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 427 bytes | 427.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/venkatakalyan24/web-app-test.git
  f1fa64a..e4f4a48 master -> master
ubuntu@ip-172-31-12-134:~/web-app-test/devopsIQ$
```

i-0c8d6016661842159 (jenkins-master)

Public IPs: 15.206.149.243 Private IPs: 172.31.12.134



