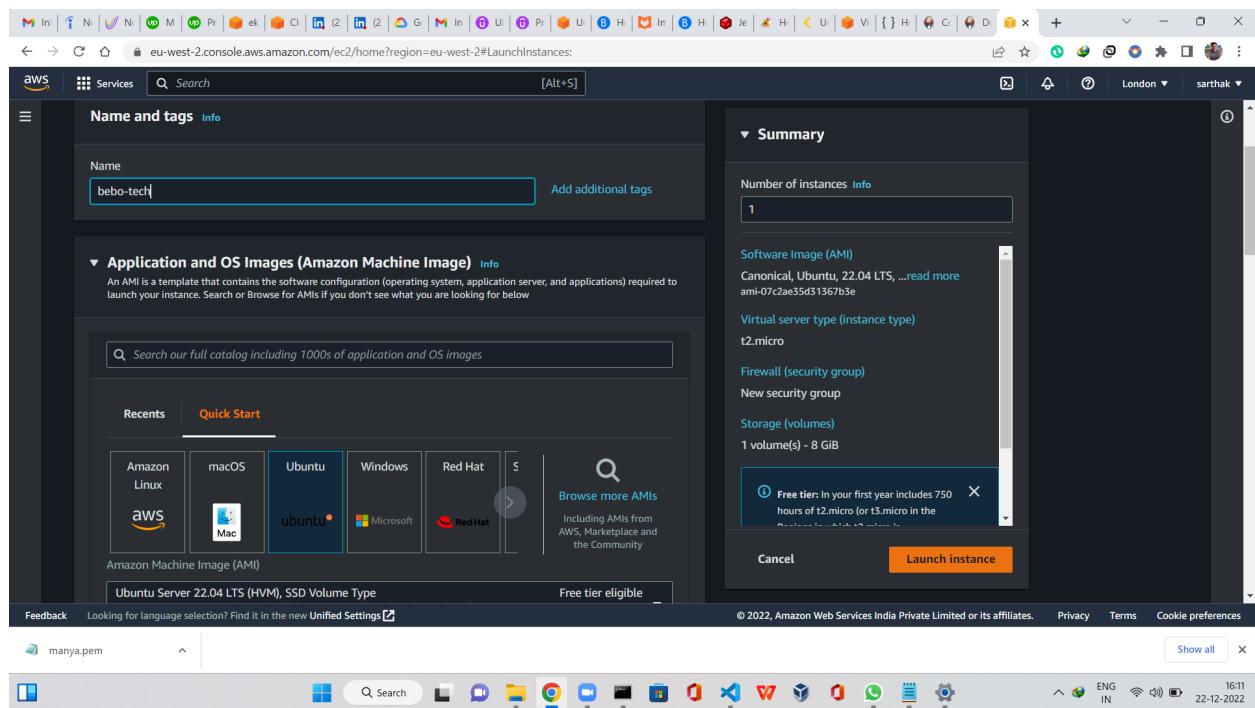
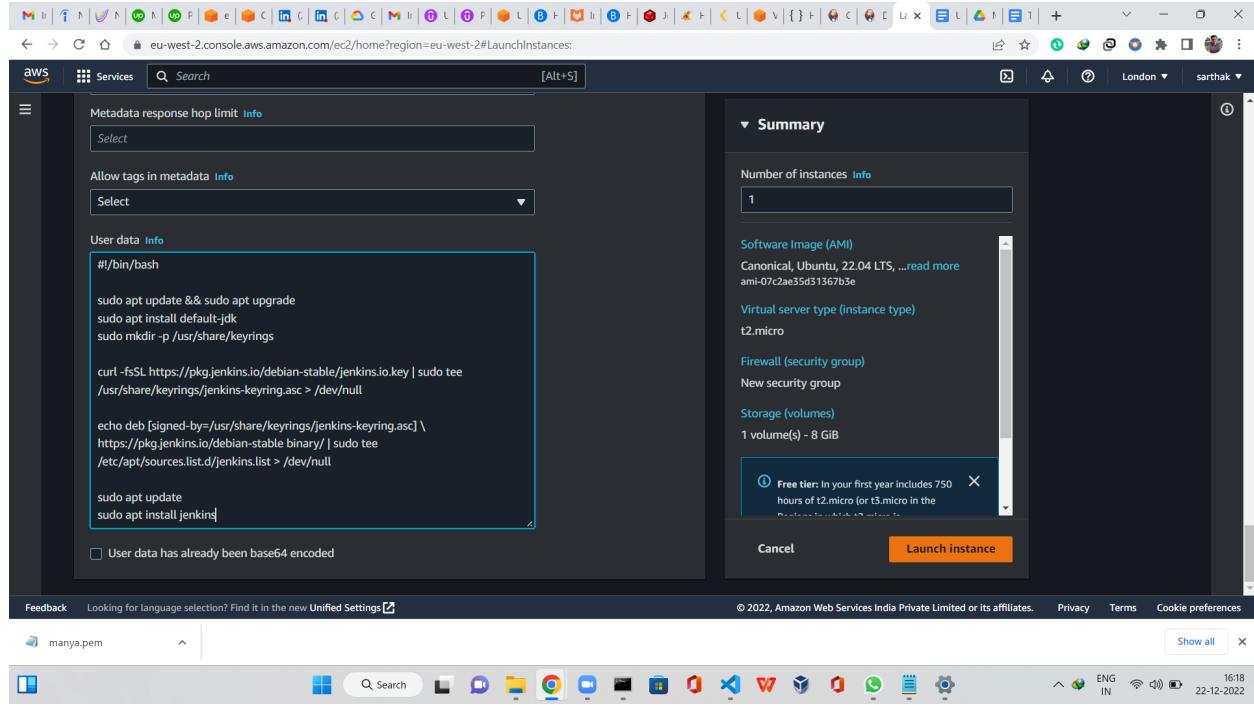


- Jenkins pipeline that should be triggered in case of any commit to the GitHub main/master branch and the execution should happen over the Slave agent. The pipeline should be responsible for installing the latest version of Chrome browser and chromedriver.
 - **Jenkins master:** Ubuntu 20.04 LTS
 - **Slave agent:** Ubuntu 20.04 LTS

I had used AWS ec2 service for the jenkins server deployment.



The configuration and start-up script is listed below, which has installed jenkins in the ubuntu 22.04 machine.

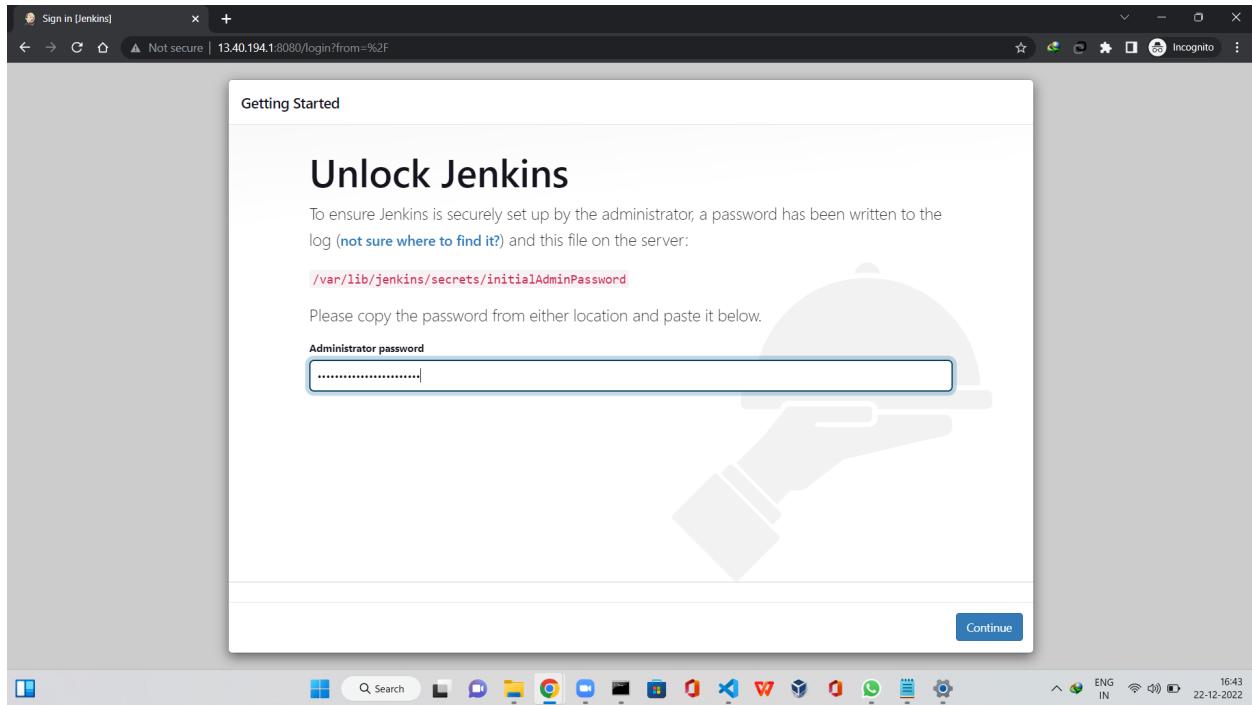


The jenkins status which is actively running after installation.

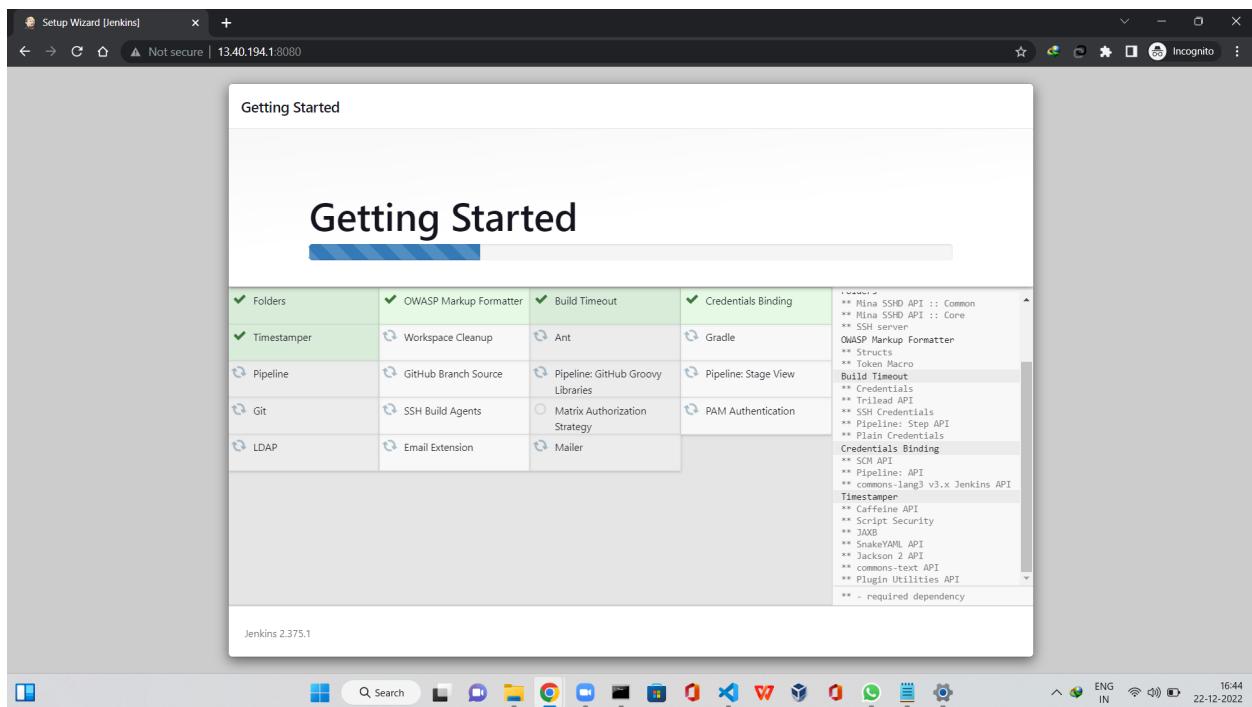
```
root@ip-172-31-34-208:/home/ubuntu
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-34-208:/home/ubuntu# sudo apt install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
jenkins is already the newest version (2.375.1).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ip-172-31-34-208:/home/ubuntu# systemctl status jenkins --no-pager -l
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-12-22 11:09:18 UTC; 32s ago
       Main PID: 15631 (java)
          Tasks: 43 (limit: 1143)
        Memory: 311.8M
         CPU: 41.834s
        CGroup: /system.slice/jenkins.service
                └─15631 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Dec 22 11:08:47 ip-172-31-34-208 jenkins[15631]: 61aF3f8acbebf4963bcf0fc1f5f78e11fd
Dec 22 11:08:47 ip-172-31-34-208 jenkins[15631]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Dec 22 11:08:47 ip-172-31-34-208 jenkins[15631]: ****
Dec 22 11:09:18 ip-172-31-34-208 jenkins[15631]: 2022-12-22 11:09:18.667+0000 [id:29]      INFO      jenkins.InitReactorRunner$1#onAttained: Completed initialization
Dec 22 11:09:18 ip-172-31-34-208 jenkins[15631]: 2022-12-22 11:09:18.694+0000 [id:22]      INFO      hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
Dec 22 11:09:18 ip-172-31-34-208 systemd[1]: Started Jenkins Continuous Integration Server.
Dec 22 11:09:19 ip-172-31-34-208 jenkins[15631]: 2022-12-22 11:09:19.092+0000 [id:44]      INFO      h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.Maven.MavenInstalle
Dec 22 11:09:19 ip-172-31-34-208 jenkins[15631]: 2022-12-22 11:09:19.093+0000 [id:44]      INFO      hudson.util.Retrier#start: Performed the action check updates server successfully at the attempt #1
root@ip-172-31-34-208:/home/ubuntu#
```

The login screen for accessing the jenkins server using <public ip>:8080 on browser.



The initial plugins that are being installed before getting the dashboard console.



After creating the credentials the jenkins dashboard looks like this way with no jobs.

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links for 'New Item', 'People', 'Build History', 'Manage Jenkins' (which is selected), and 'My Views'. Below this are two dropdown menus: 'Build Queue' (empty) and 'Build Executor Status' (showing 1 idle). The main content area is titled 'Manage Jenkins' and contains sections for 'System Configuration' and 'Security'. Under 'System Configuration', there are four items: 'Configure System' (global settings), 'Global Tool Configuration' (configure tools), 'Manage Plugins' (add, remove, disable or enable plugins), and 'Manage Nodes and Clouds' (control and monitor various nodes). Under 'Security', there are three items: 'Configure Global Security' (secure Jenkins), 'Manage Credentials' (configure credentials), and 'Configure Credential Providers' (configure credential providers). The top right of the page has buttons for 'Set up agent', 'Set up cloud', and 'Dismiss'. The bottom right shows the browser status bar with the URL '13.40.194.1:8080/manage/computer' and the date/time '22-12-2022 16:47'.

The webhooks are created down below with payload URL

`http://<public_ip_of_ec2_instance>/jenkins/github-webhook/`

The screenshot shows the GitHub Settings / Webhooks page. The left sidebar has sections for General, Access, Collaborators, Moderation options, Code and automation, Branches, Tags, Actions, Webhooks (which is selected), Environments, Codespaces, and Pages. Under 'Webhooks', there are sections for 'Payload URL' (set to 'http://13.40.194.1/github-webhook/'), 'Content type' (set to 'application/x-www-form-urlencoded'), 'Secret' (empty), and 'Which events would you like to trigger this webhook?'. The options are: 'Just the push event.' (radio button), 'Send me everything.' (radio button, selected), 'Let me select individual events.' (checkbox), and 'Active' (checkbox, checked). The top navigation bar shows the URL 'github.com/Sarthak-Agarwal1410/jenkins/settings/hooks/288717963' and the bottom status bar shows the date/time '22-12-2022 16:56'.

The screenshot shows the Jenkins configuration page for a job named 'job1'. The left sidebar lists several sections: General, Source Code Management, Build Triggers (which is selected), Build Environment, Build Steps, and Post-build Actions. Under 'Build Triggers', the 'GitHub hook trigger for GITScm polling' option is checked. The 'Build Environment' section contains several unchecked checkboxes related to workspace management and timestamps. At the bottom of the page are 'Save' and 'Apply' buttons.

Console output for the very first time after running the job manually.

The screenshot shows the Jenkins console output for the first build of job1. The left sidebar has links for Status, Changes, and Console Output (which is selected). The main area displays the build log, which includes details about cloning the repository from GitHub and performing a successful build. The log ends with 'Finished: SUCCESS'.

```
Started by user jenkins
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/job1
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
  Cloning repository https://github.com/Sarthak-Agarwall1410/jenkins.git
    > git init /var/lib/jenkins/workspace/job1 # timeout=10
  Fetching upstream changes from https://github.com/Sarthak-Agarwall1410/jenkins.git
    > git --version # timeout=10
    > git --version # 'git version 2.34.1'
    > git fetch --tags --progress -- https://github.com/Sarthak-Agarwall1410/jenkins.git +refs/heads/*:refs/remotes/origin/* # timeout=10
    > git config remote.origin.url https://github.com/Sarthak-Agarwall1410/jenkins.git # timeout=10
    > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
  Avoid second fetch
    > git rev-parse refs/remotes/origin/master^(commit) # timeout=10
  Checking out Revision 83c6bf1dbc77c5b46c1995249074d7ded3a226be (refs/remotes/origin/master)
    > git config core.sparsecheckout # timeout=10
    > git checkout -f 83c6bf1dbc77c5b46c1995249074d7ded3a226be # timeout=10
  Commit message: "Update mystet.sh"
  First time build. Skipping changelog.
Finished: SUCCESS
```

Now, Lets configure the slave agent and connect it with the master agent (built in node)

+ New Node

Configure Clouds

Node Monitoring

Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

New node

Node name

Type

Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Create



For the slave node we need another ubuntu instance which is shown below:

The screenshot shows the AWS EC2 'Launch Instances' interface. On the left, there's a sidebar with 'Name and tags' (set to 'slave'), 'Application and OS Images (Amazon Machine Image)' (selected), and a search bar for AMIs. Below these are tabs for 'Recents' and 'Quick Start'. Under 'Quick Start', several AMI icons are displayed: Amazon Linux, macOS, Ubuntu (selected), Windows, Red Hat, and others. A search bar for 'Browse more AMIs' is also present. The main right-hand panel is titled 'Summary' and contains fields for 'Number of instances' (set to 1), 'Software Image (AMI)' (Canonical, Ubuntu, 22.04 LTS), 'Virtual server type (instance type)' (t2.micro), 'Firewall (security group)' (New security group), and 'Storage (volumes)' (1 volume(s) - 8 GiB). A prominent blue button at the bottom right says 'Launch instance'. A tooltip for the free tier is visible, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the future)'. The top navigation bar shows the URL as 'eu-west-2.console.aws.amazon.com/ec2/home?region=eu-west-2#LaunchInstances'.

The instance summary for slave node is shown below.

Dont forget to copy the public IP which will be used as host in slave agent configuration

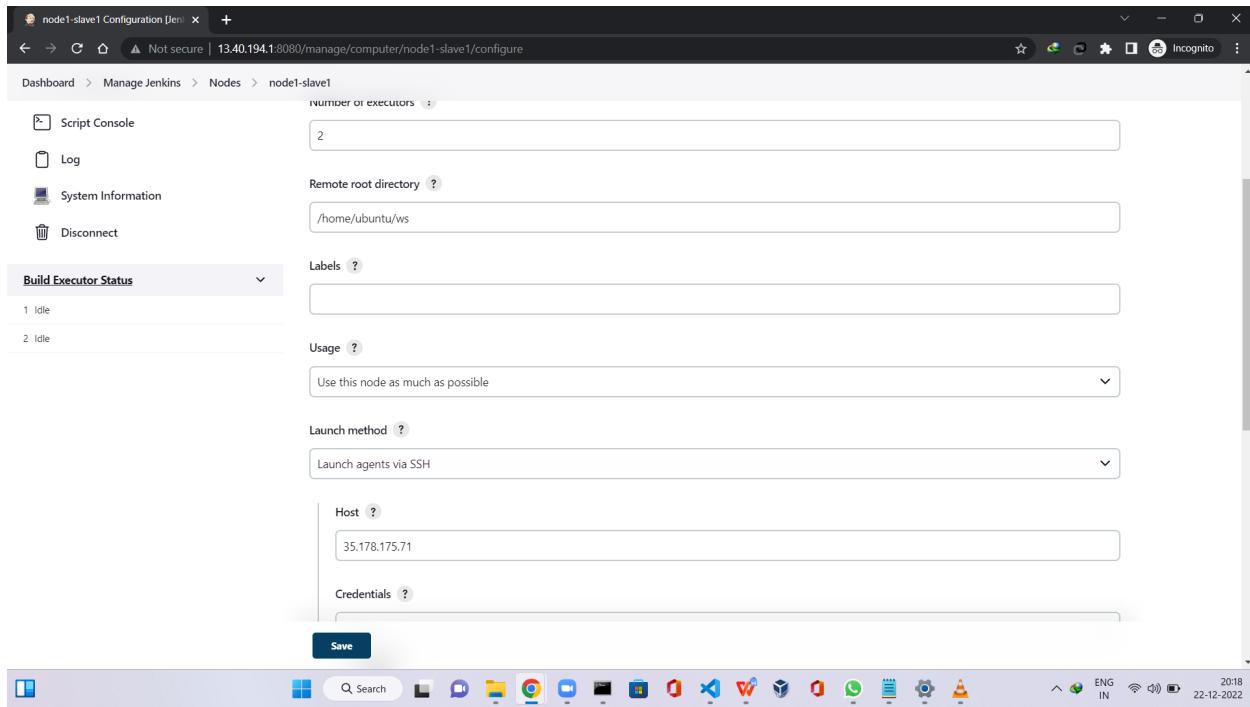
Instance summary for i-00118fa936483999f (slave) [Info](#)

Updated less than a minute ago

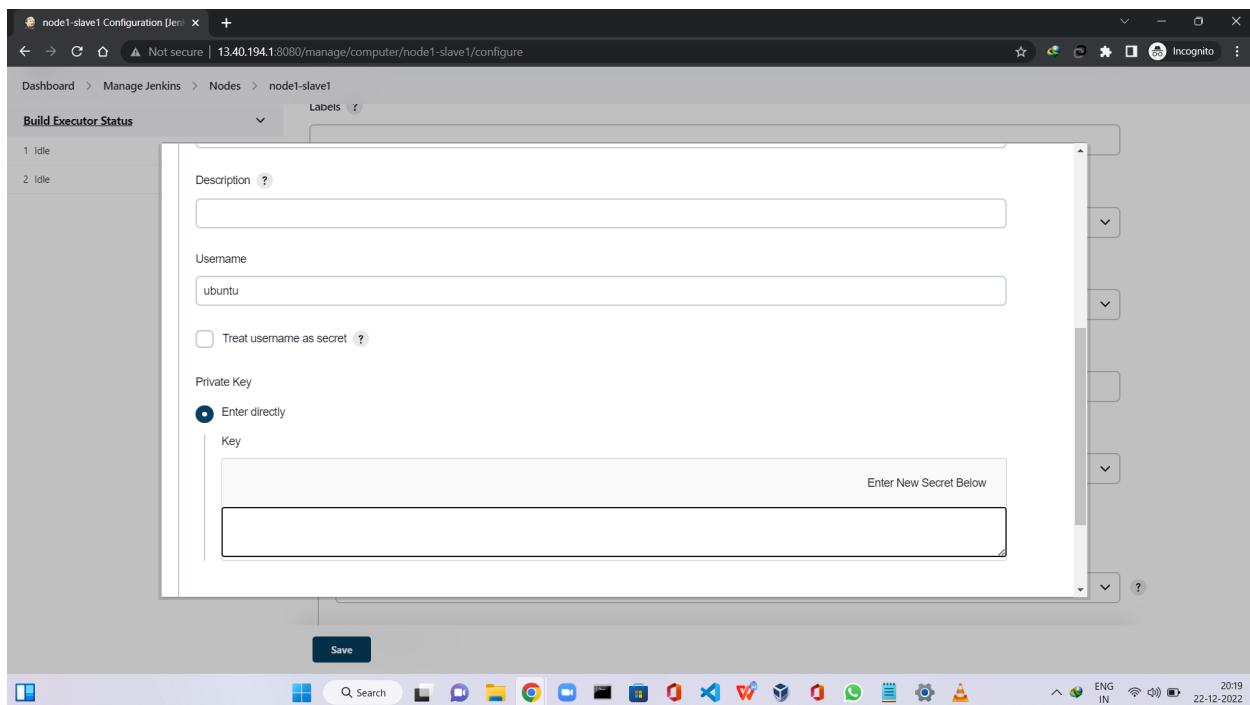
Detail	Value
Instance ID	i-00118fa936483999f (slave)
IPv6 address	-
Hostname type	IP name: ip-172-31-44-197.eu-west-2.compute.internal
Answer private resource DNS name	IPv4 (A)
Auto-assigned IP address	35.178.175.71 [Public IP]
IAM Role	-
Public IPv4 address	35.178.175.71 open address
Instance state	Running
Private IP DNS name (IPv4 only)	ip-172-31-44-197.eu-west-2.compute.internal
Instance type	t2.micro
VPC ID	vpc-00585d2f9ffddfb8
Subnet ID	subnet-0fe02a8d02c561a53
Private IPv4 addresses	172.31.44.197
Public IPv4 DNS	ec2-35-178-175-71.eu-west-2.compute.amazonaws.com open address
Elastic IP addresses	-
AWS Compute Optimizer finding	Opt-in to AWS Compute Optimizer for recommendations.
Auto Scaling Group name	-

Lets attach this slave node to the master agent

Manage Jenkins >> Manage Nodes and Clouds >> New Node



The username and key has to be given here for authentication.



Authentication is successful between master and slave agent using SSH Protocol.

```

[12/22/22 13:05:50] [SSH] Checking java version of /home/ubuntu/ws/jdk/bin/java
Couldn't figure out the Java version of /home/ubuntu/ws/jdk/bin/java
bash: line 1: /home/ubuntu/ws/jdk/bin/java: No such file or directory

[12/22/22 13:05:50] [SSH] Checking java version of java
[12/22/22 13:05:50] [SSH] java -version returned 11.0.17.
[12/22/22 13:05:50] [SSH] Starting sftp client.
[12/22/22 13:05:50] [SSH] Copying latest remoting.jar...
[12/22/22 13:05:50] [SSH] Copied 1,369,460 bytes.
Expanded the channel window size to 4MB
[12/22/22 13:05:50] [SSH] Starting agent process: cd "/home/ubuntu/ws" && java -jar remoting.jar -workDir /home/ubuntu/ws -jar-cache /home/ubuntu/ws/remoting/jarCache
Dec 22, 2022 1:05:50 PM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/ws/remoting as a remoting work directory
Dec 22, 2022 1:05:50 PM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to /home/ubuntu/ws/remoting
<===[JENKINS REMOTING CAPACITY]==>channel started
Remoting version: 3071.7e9b_0dc08466
Launcher: SSHLauncher
Communication Protocol: Standard in/out
This is a Unix agent
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by jenkins.slaves.StandardOutputSwapper$ChannelSwapper to constructor java.io.FileDescriptor(int)
WARNING: Please consider reporting this to the maintainers of jenkins.slaves.StandardOutputSwapper$ChannelSwapper
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Evacuated stdout
Agent successfully connected and online

```



S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	4.85 GB	! 0 B	4.85 GB	0ms
	node1-slave1	Linux (amd64)	In sync	5.08 GB	! 0 B	5.08 GB	7ms
	Data obtained	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Linux (amd64)	In sync	4.85 GB	! 0 B	4.85 GB	0ms
	node1-slave1	Linux (amd64)	In sync	5.08 GB	! 0 B	5.08 GB	7ms
	Data obtained	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec	6 min 26 sec

Lets install Chrome and Chrome Driver on slave agent using the script on Github.

```

(Reading database ... 66415 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-26ubuntu3.1_amd64.deb ...
Unpacking unzip (6.0-26ubuntu3.1) ...
Selecting previously unselected package zip.
Preparing to unpack .../zip_3.0-12build2_amd64.deb ...
Unpacking zip (3.0-12build2) ...
Setting up unzip (6.0-26ubuntu3.1) ...
Setting up zip (3.0-12build2) ...
Processing triggers for man-db (2.10.2-1) ...

Running kernel seems to be up-to-date.

Services to be restarted:

Service restarts being deferred:
systemctl restart networkd-dispatcher.service
systemctl restart unattended-upgrades.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
Archive: chromedriver_linux64.zip
inflating: chromedriver
Finished: SUCCESS

```



The job is running under the slave node section automatically which got success response above.

S	W	Name	Last Success	Last Failure	Last Duration
		job1	1 min 3 sec #5	50 min #4	9.2 sec

Build Queue: No builds in the queue.

Build Executor Status:

- Built-In Node: 1 idle
- node1-slave1: 1 job1 #7, 2 idle

The GitHub code for installation of Chrome and chrome Driver is shown below.

The screenshot shows a GitHub pull request page for a repository named "Sarthak-Agarwal1410/jenkins". The branch is "master". The file "mytest.sh" contains the following script:

```
1 wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
2 sudo apt install ./google-chrome-stable_current_amd64.deb
3 wget https://chromedriver.storage.googleapis.com/2.41/chromedriver_linux64.zip
4 sudo apt-get -y install zip
5 unzip chromedriver_linux64.zip
6 sudo mv chromedriver /usr/bin/chromedriver
7 sudo chmod +x /usr/bin/chromedriver
8 sudo chmod +x /usr/bin/chromedriver
```

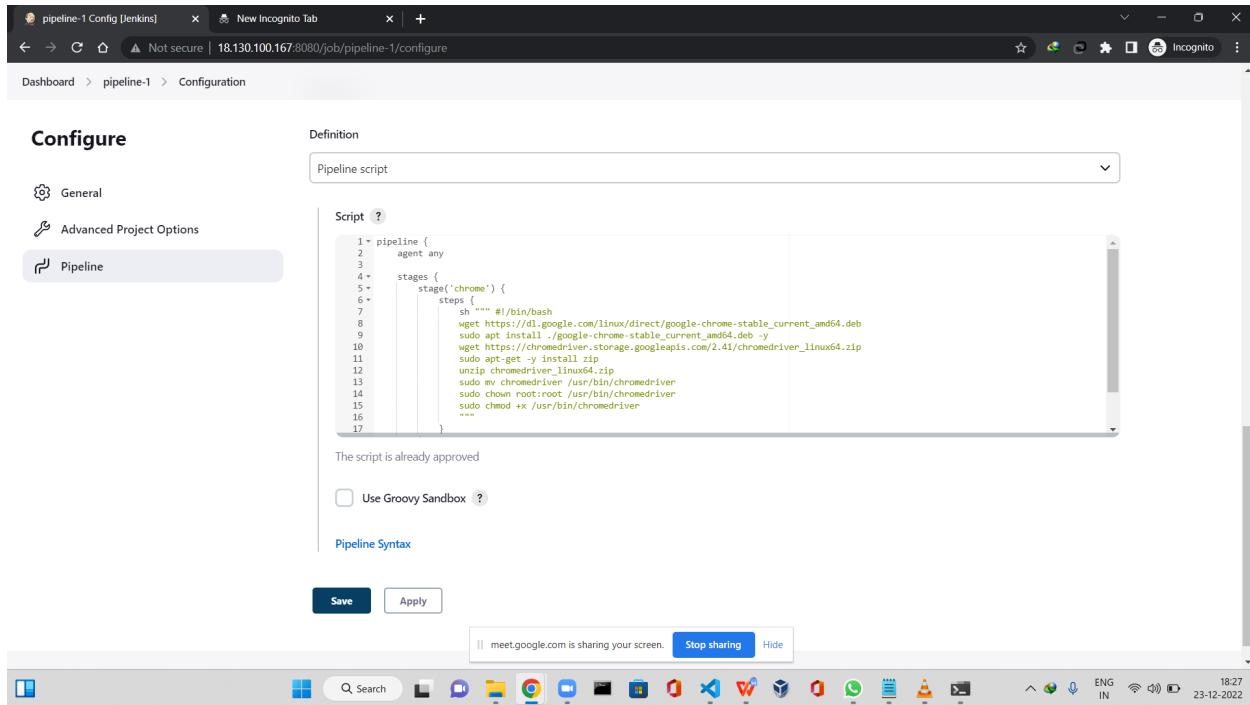
© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About



So, now we will create a new pipeline.

The screenshot shows the Jenkins dashboard at the URL <http://18.130.100.167:8080>. The "Build History" section displays a single pipeline named "pipeline-1" with the last success occurring 44 seconds ago and the last failure 2 minutes ago. The "Build Executor Status" section shows one "Built-In Node" labeled "idle" and one "slave1" labeled "idle".

The Pipeline code is listed below with configuration.



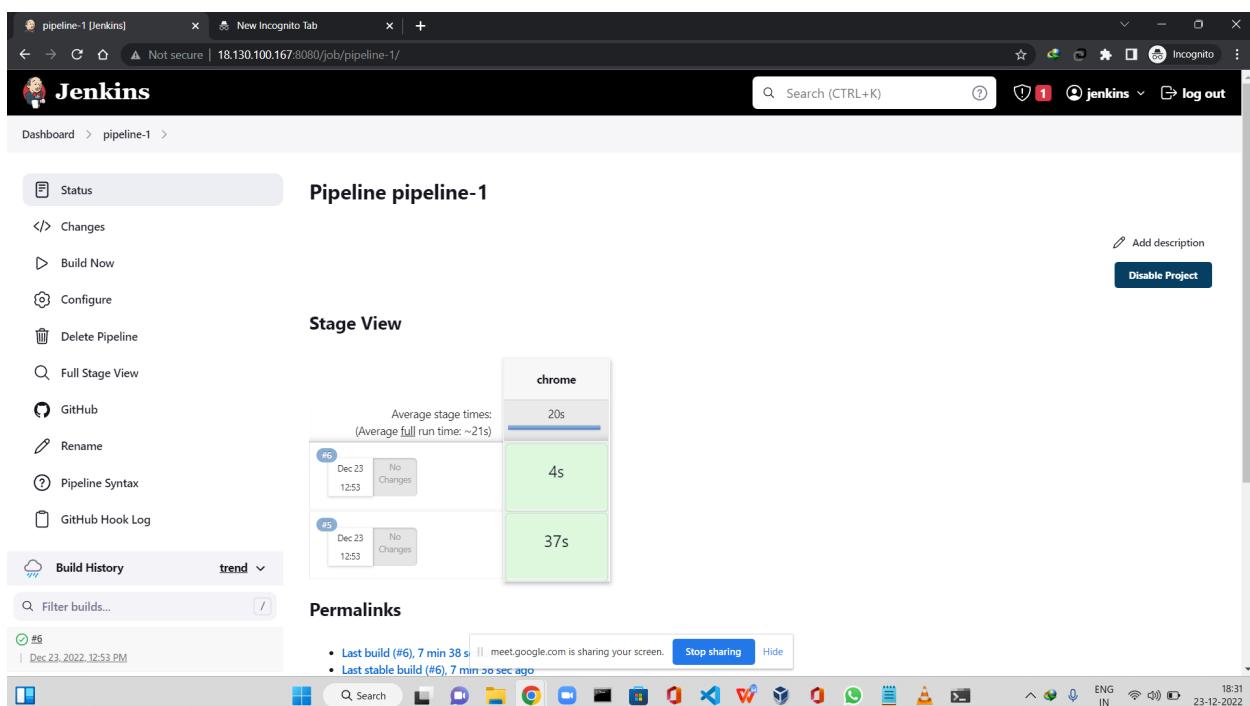
```
1 pipeline {
2     agent any
3
4     stages {
5         stage('chrome') {
6             steps {
7                 script {
8                     wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
9                     sudo apt install ./google-chrome-stable_current_amd64.deb -y
10                    wget https://chromedriver.storage.googleapis.com/2.41/chromedriver_linux64.zip
11                     sudo apt-get -y install zip
12                     unzip chromedriver_linux64.zip
13                     sudo mv chromedriver /usr/bin/chromedriver
14                     sudo chmod root:root /usr/bin/chromedriver
15                     sudo chmod +x /usr/bin/chromedriver
16                 }
17             }
18         }
19     }
20 }
```

The script is already approved

Use Groovy Sandbox

Pipeline Syntax

Save **Apply**



Jenkins

Dashboard > pipeline-1 >

Pipeline pipeline-1

Stage View

stage	duration
chrome	20s
	4s
	37s

Average stage times:
(Average full run time: ~21s)

Build History

trend ▾

Filter builds...

#6 Dec 23, 2022, 12:53 PM 7 min 38 s meet.google.com is sharing your screen. Stop sharing Hide

#5 Dec 23, 2022, 12:53 No Changes 37s

Permalinks

- Last build (#6), 7 min 38 s
- Last stable build (#6), 7 min 30 sec ago

[GitHub Link for Code](#)

Hope the procedure was explained in a good way by using various steps and shown screenshots for proper workflow of the task.

Thank You