

Jenkins Assignment - Case study

Email: harshalgite888@gmail.com

Tasks:

You have been Hired as a Devops Engineer in xyz software company. They want to implement CI/CD pipeline in their company. You have been asked to implement this lifecycle as fast as possible. As this is a product-based company, their product is available on this GitHub link.

<https://github.com/hshar/website.git>

Following are the specifications of the Continuous integration:

1. Git Workflow has to be implemented
2. Code Build should automatically be triggered once commit is made to master branch or develop branch.
If commit is made to master branch, build and publish website on port 82. If commit is made to develop branch, just build the product, do not publish.
3. Create a pipeline for the above tasks.
4. Create a container with Ubuntu and apache installed in it and use that container to build the code and the code should be on '/var/www/html'.

Solution:

Steps:

1. Update the instance and install jdk.

```
Get:28 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.4 kB]
Get:29 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [640 B]
Get:30 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [856 kB]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [175 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [11.4 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [953 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [154 kB]
Get:36 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [532 B]
Get:37 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [788 kB]
Get:38 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [144 kB]
Get:39 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.7 kB]
Get:40 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [36.5 kB]
Get:41 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7060 B]
Get:42 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Fetched 27.7 MB in 5s (5587 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
129 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-43-20:~$ sudo apt install openjdk-11-jdk
```

i-01a7d2729641b12f0 (Jenkins-M)

PublicIPs: 18.225.10.247 PrivateIPs: 172.31.43.20

Activate Windows
Go to Settings to activate Windows.

2. Install jenkins with commands

```
No containers need to be restarted.

No user sessions are running outdated binaries.

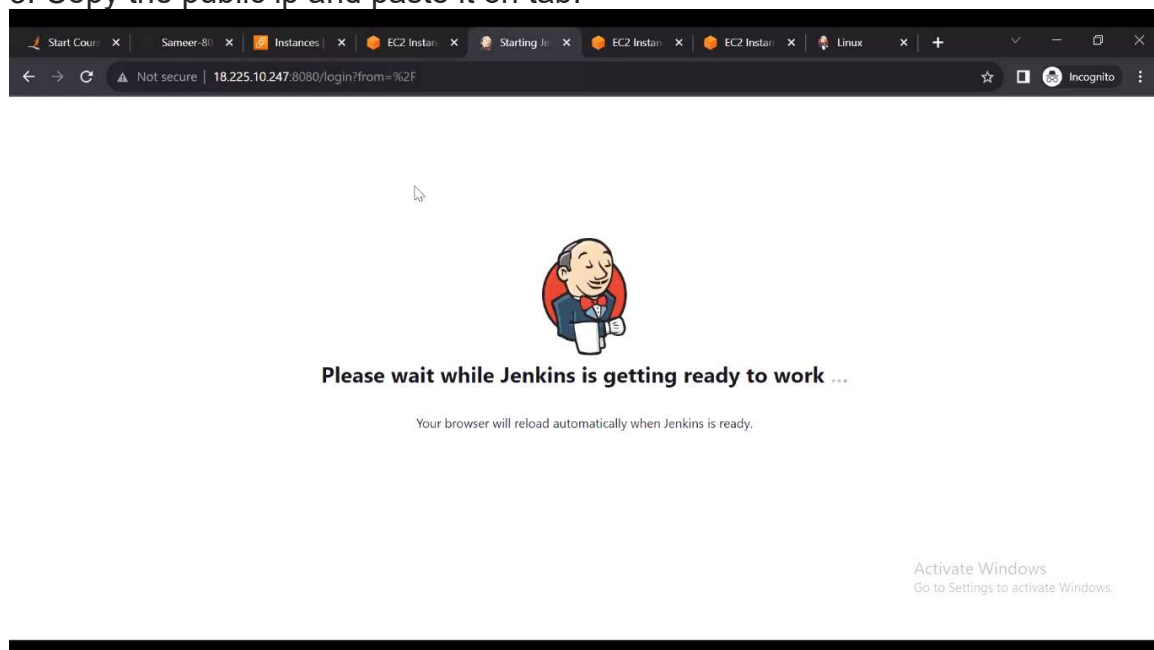
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-43-20:~$ curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee \
/usr/share/keyrings/jenkins-keyring.asc > /dev/null
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 https://pkg.jenkins.io/debian binary/ InRelease
Get:5 https://pkg.jenkins.io/debian binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu jammy-security InRelease

i-01a7d2729641b12f0 (Jenkins-M)
PublicIPs: 18.225.10.247 PrivateIPs: 172.31.43.20

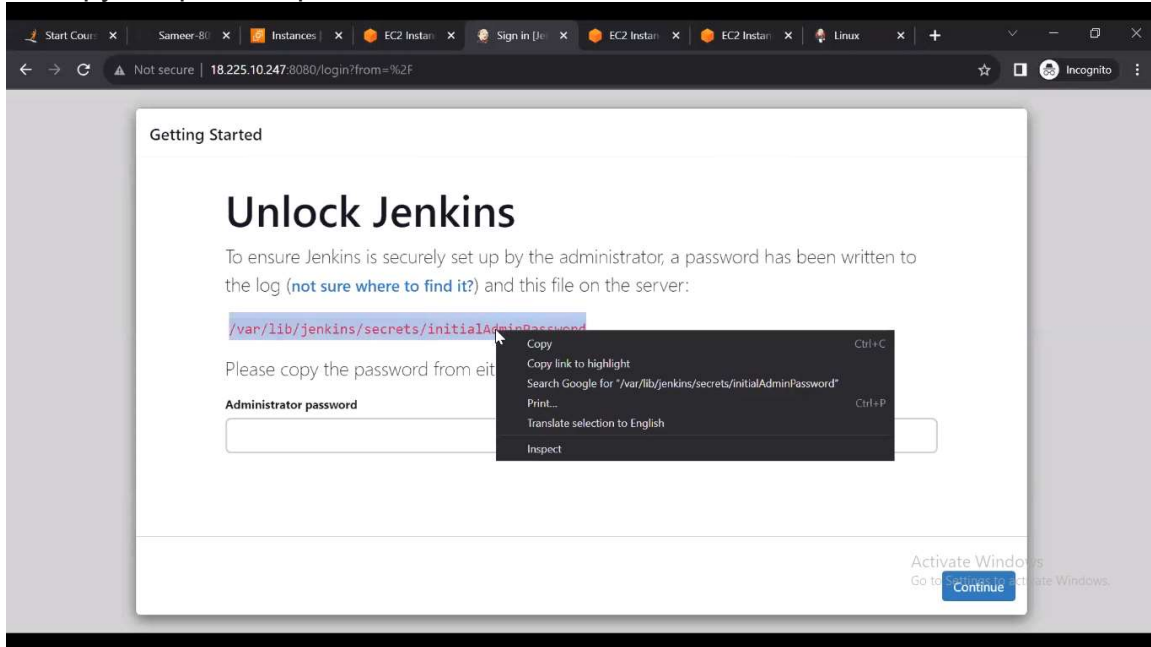
net-tools
The following NEW packages will be installed:
jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 128 not upgraded.
Need to get 89.1 MB of archives.
After this operation, 90.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-lubuntu5 [204 kB]
Get:2 https://pkg.jenkins.io/debian binary/ jenkins 2.426 [88.9 MB]
Fetched 89.1 MB in 8s (11.7 MB/s)
Selecting previously unselected package net-tools.
(Reading database ... 66700 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-lubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-lubuntu5) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../archives/jenkins_2.426_all.deb ...
Unpacking jenkins (2.426) ...
Setting up net-tools (1.60+git20181103.0eebece-lubuntu5) ...
Setting up jenkins (2.426) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /lib/systemd/system/jenkins.service.

i-01a7d2729641b12f0 (Jenkins-M)
PublicIPs: 18.225.10.247 PrivateIPs: 172.31.43.20
```

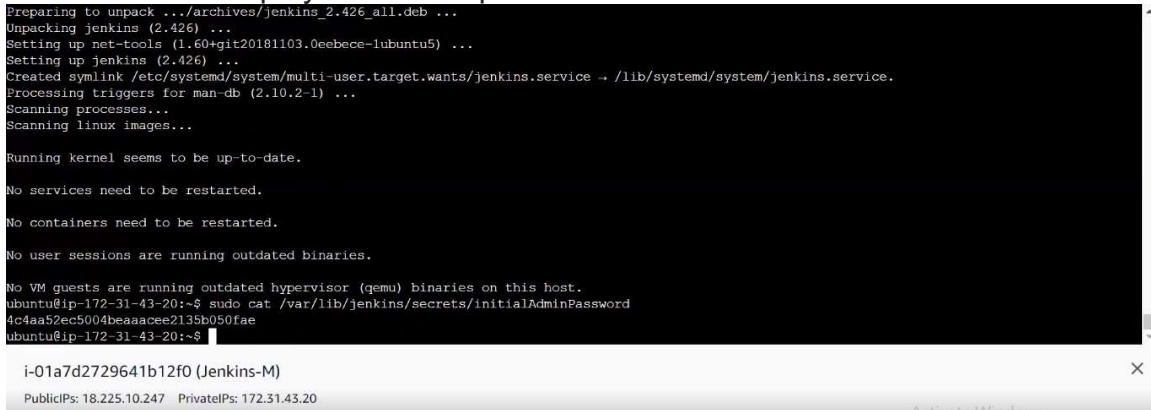
3. Copy the public ip and paste it on tab.



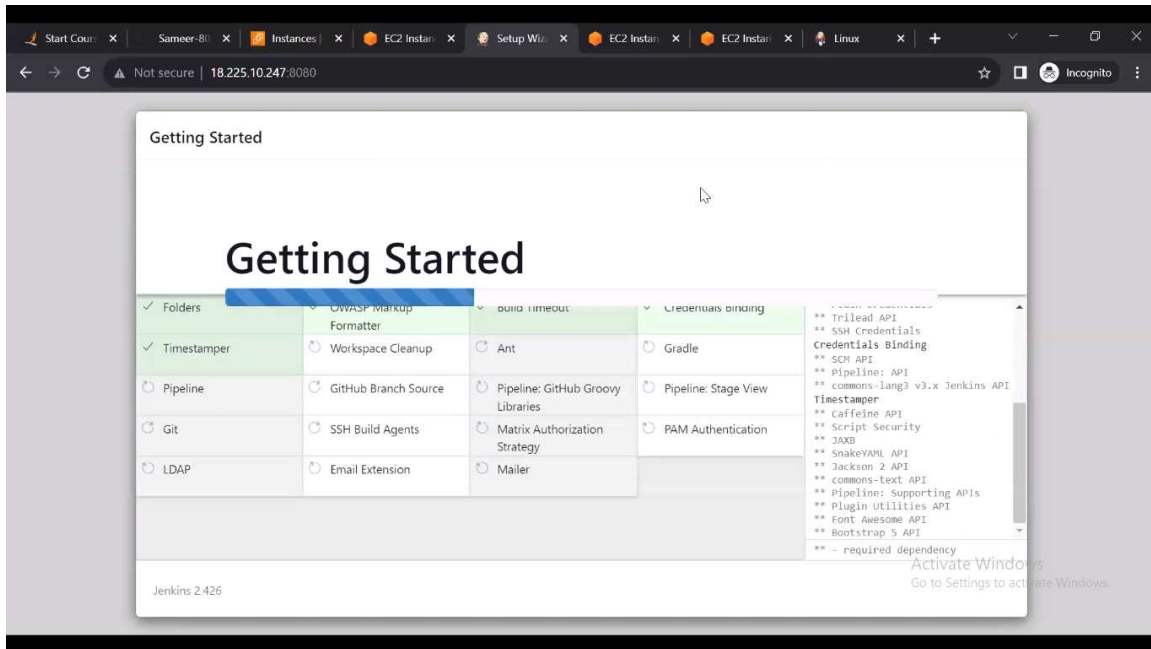
4. Copy the path for password.



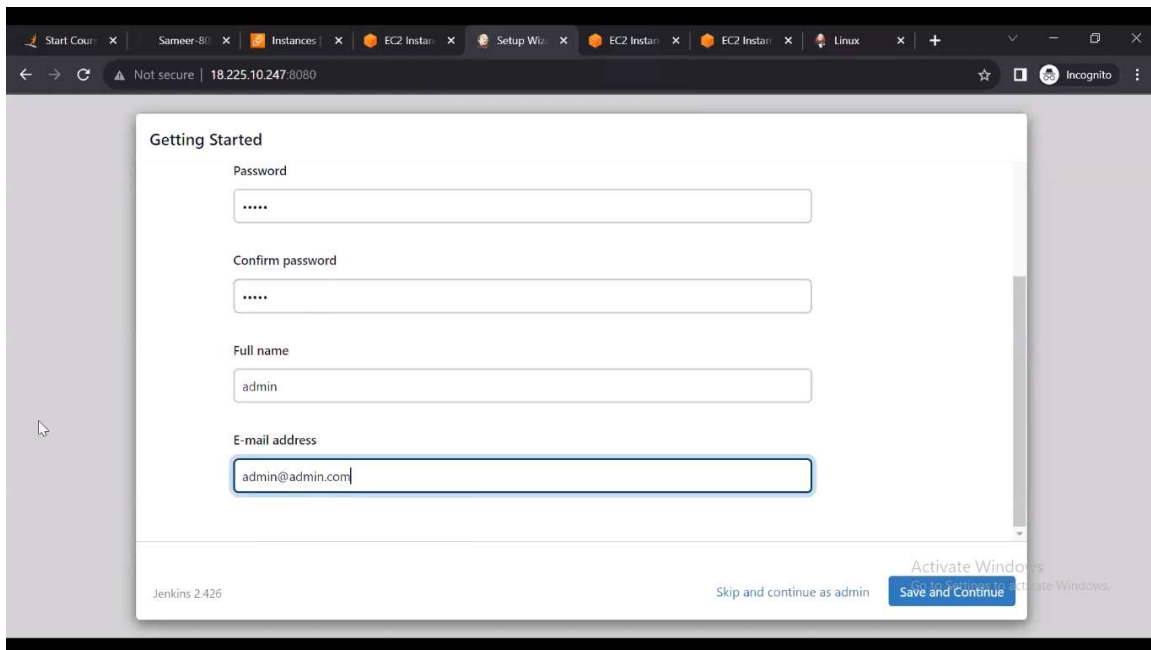
5. command to display content or password

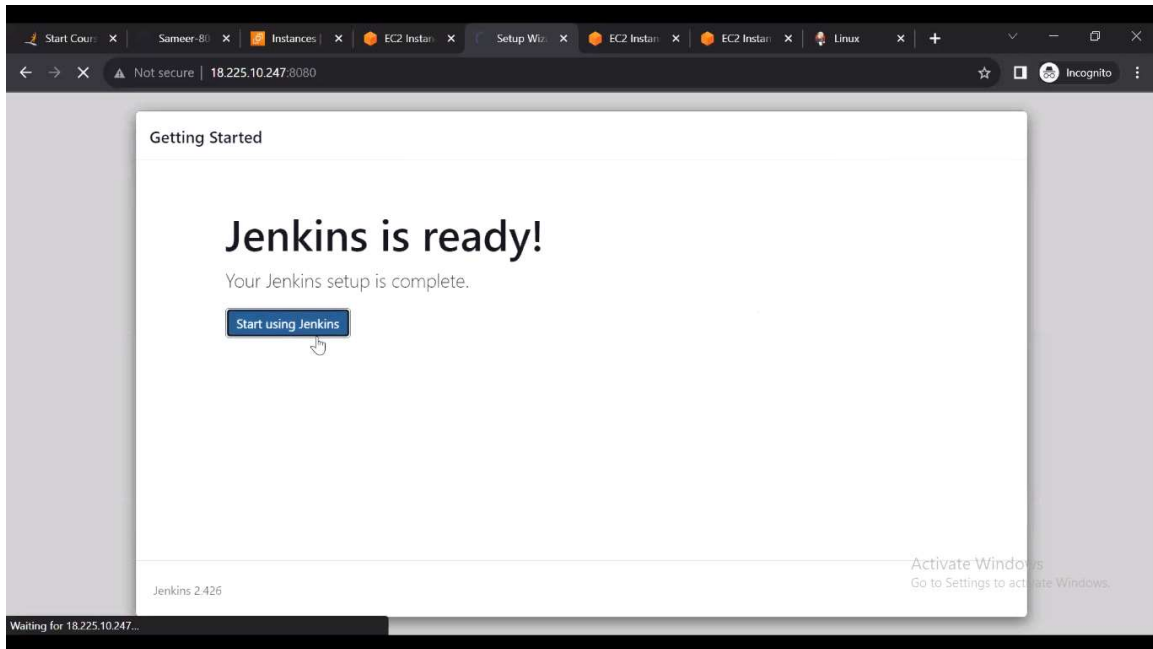


6. Copy the password and paste in browser window. Click on continue.

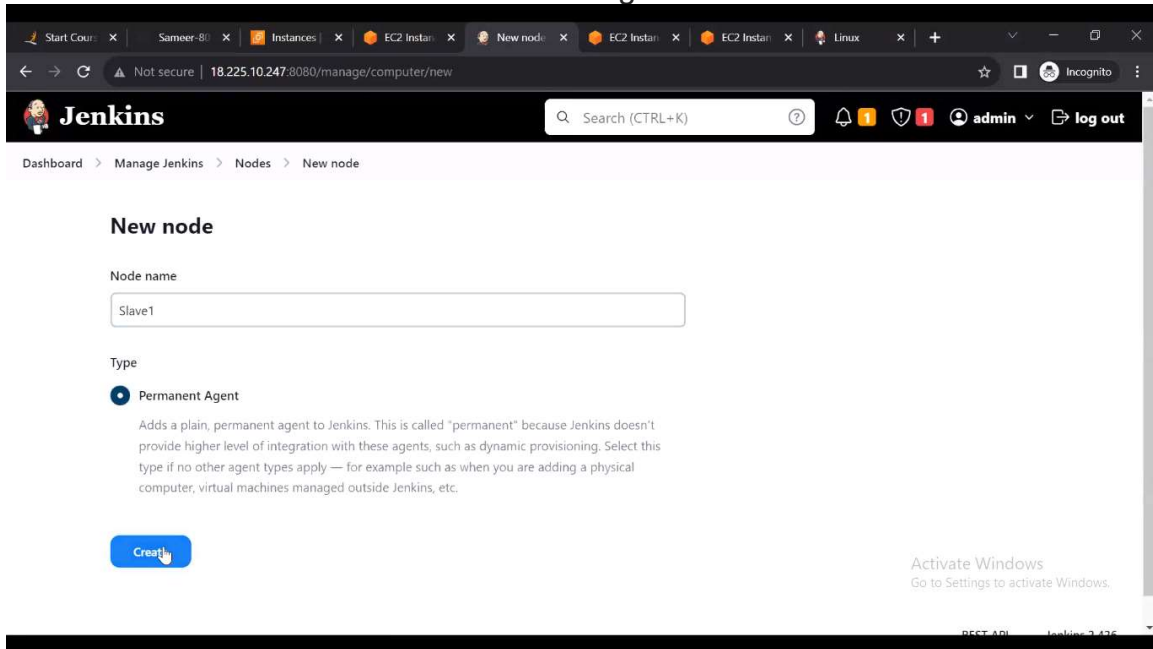


7. select install suggested plugins. Give or create the credentials ...click save and continue.





8.click on new node to create the node andgive the name and create it.



9. Give remote root directory to create folders

Dashboard > Manage Jenkins > Nodes >

Plain text [Preview](#)

Number of executors ?
1

Remote root directory ?
/home/ubuntu/jenkins

Labels ?

Usage ?
Use this node as much as possible

Launch method ?
Launch agent by connecting it to the controller

[Save](#)

Activate Windows
Go to Settings to activate Windows.

10. select launch with ssh

-Copy private ip from instance connect and paste it in host section

Dashboard > Manage Jenkins > Nodes >

Launch method ?
Launch agents via SSH

Host ?
172.31.38.128

Credentials ?
- none -
[+ Add](#)

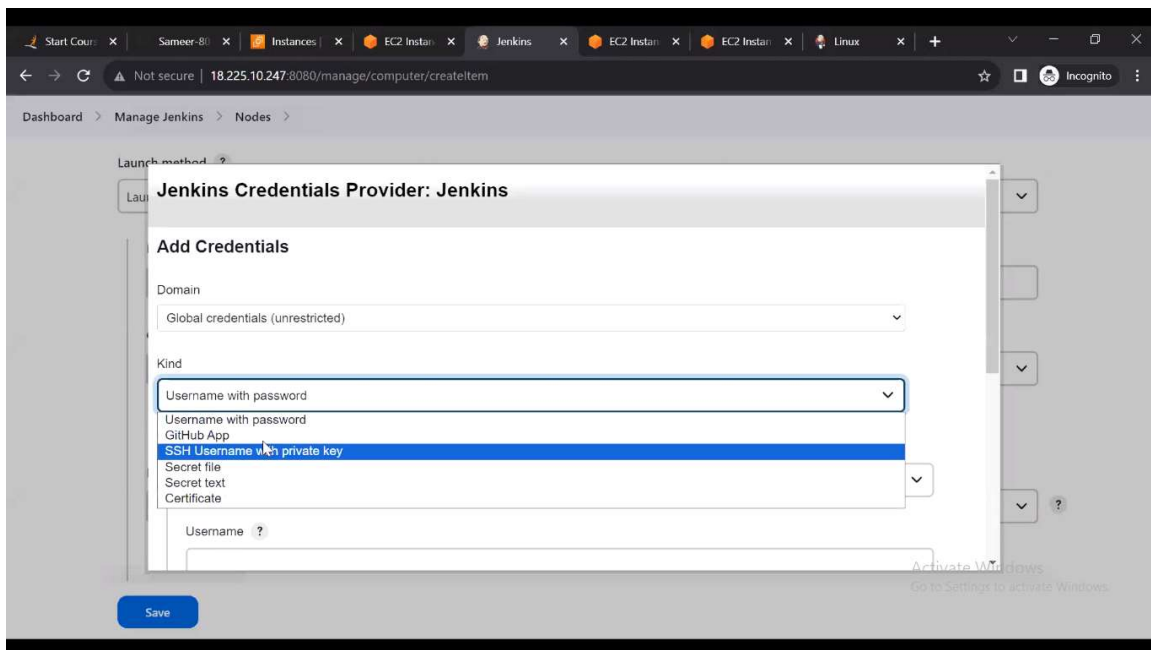
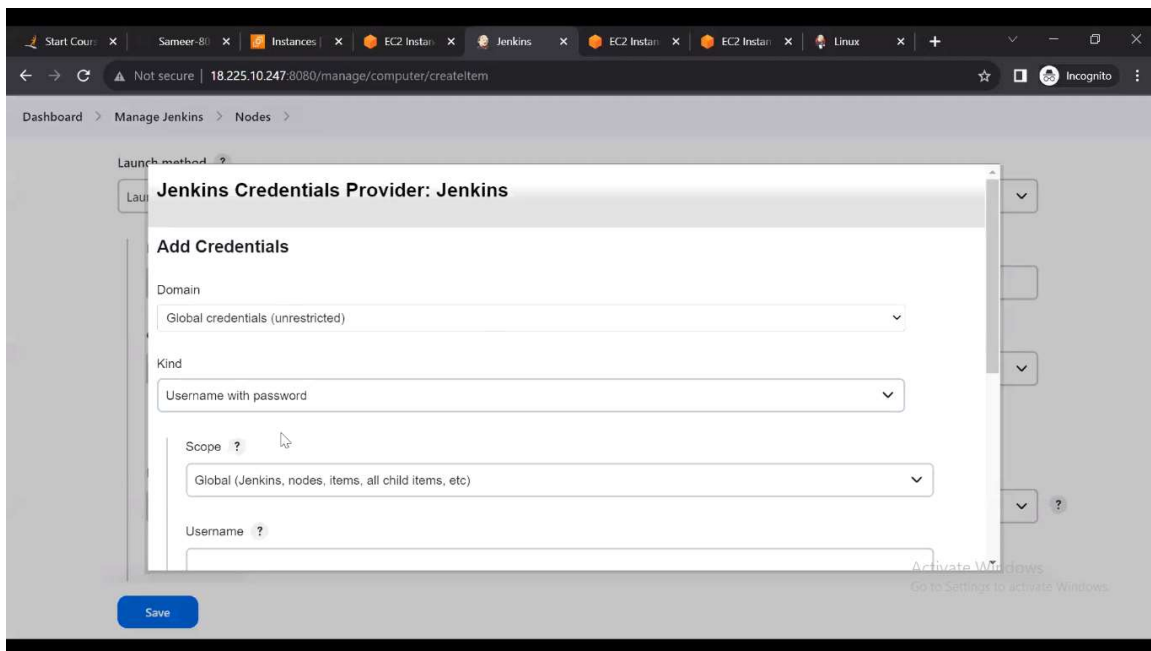
Host Key Verification Strategy ?
Known hosts file Verification Strategy

[Advanced](#)

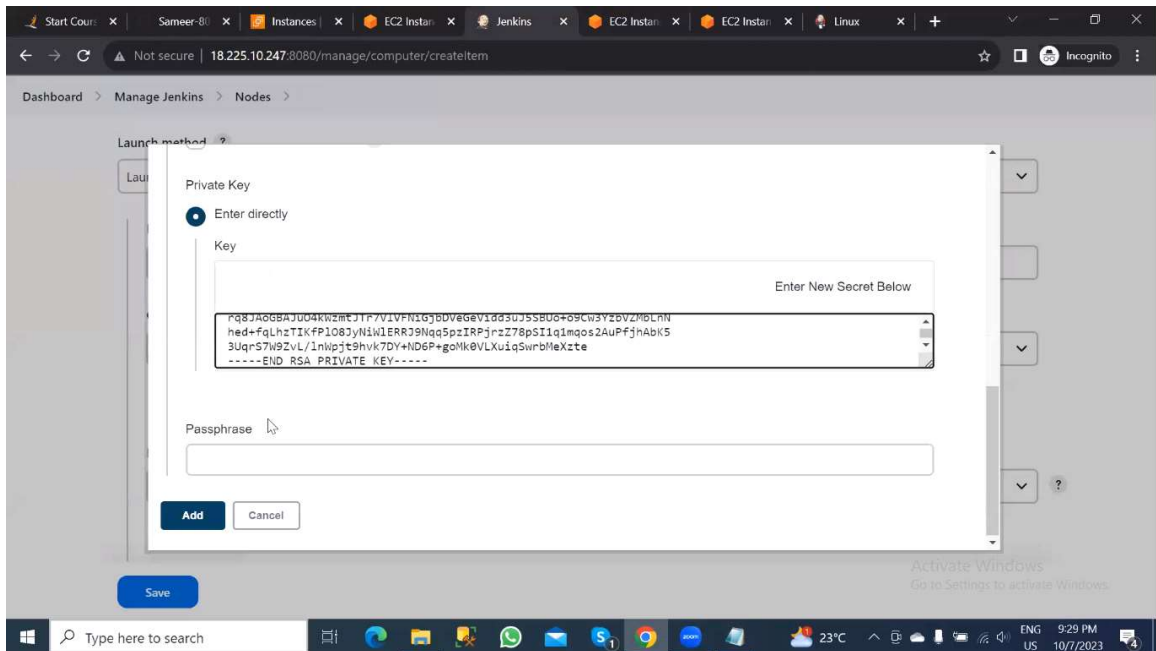
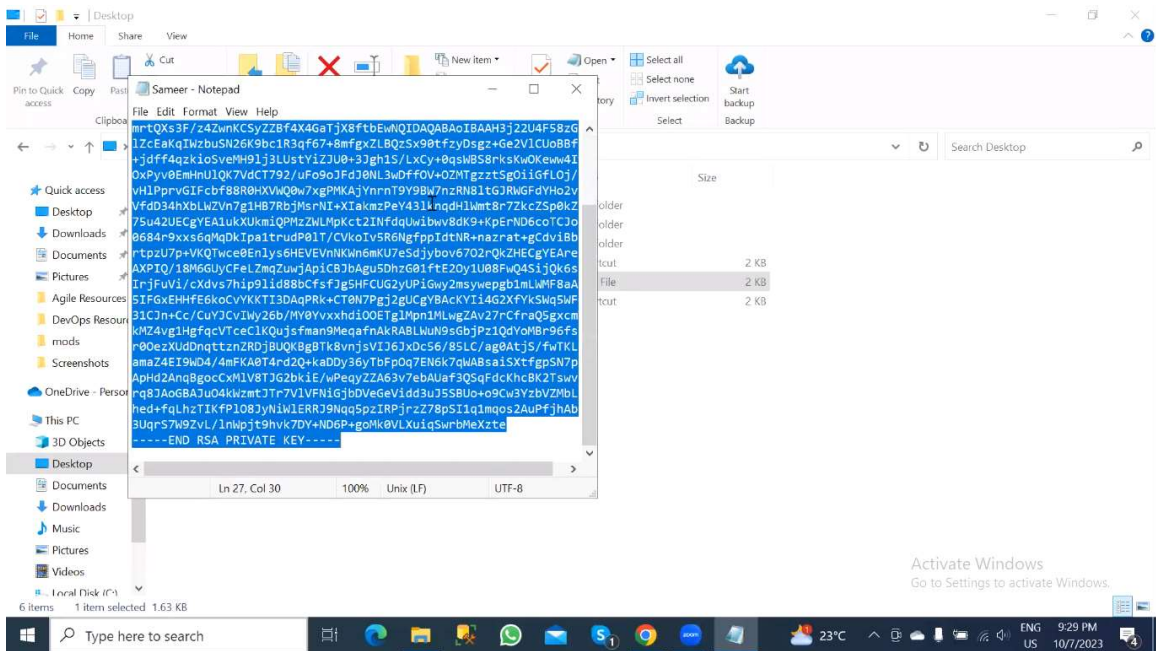
[Save](#)

Activate Windows
Go to Settings to activate Windows.

11. click on add credentials select ssh with private key give username. select enter directly.



12. open the pem key copy the content and paste in the private key box and click on add.



13. Select the *non verifying strategy* host key verification strategy and click on save

Dashboard > Manage Jenkins > Nodes

Launch method ?
Launch agents via SSH

Host ?
172.31.38.128

Credentials ?
ubuntu
+ Add

Host Key Verification Strategy ?
Non verifying Verification Strategy

Advanced

Save

Activate Windows
Go to Settings to activate Windows.

14.slave node created succesfully

Dashboard > Manage Jenkins > Nodes

Nodes

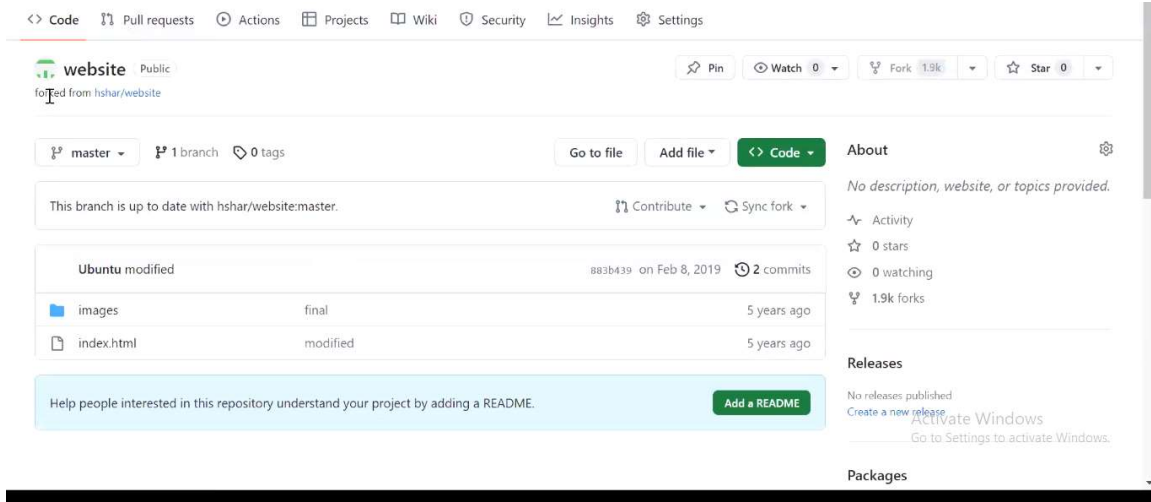
+ New Node Node Monitoring

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	1.0 B	4.85 GB	0ms
	Slave1	Linux (amd64)	In sync	5.32 GB	1.0 B	5.32 GB	80ms
Data obtained		0.29 sec	0.29 sec	0.24 sec	0.23 sec	0.25 sec	0.23 sec

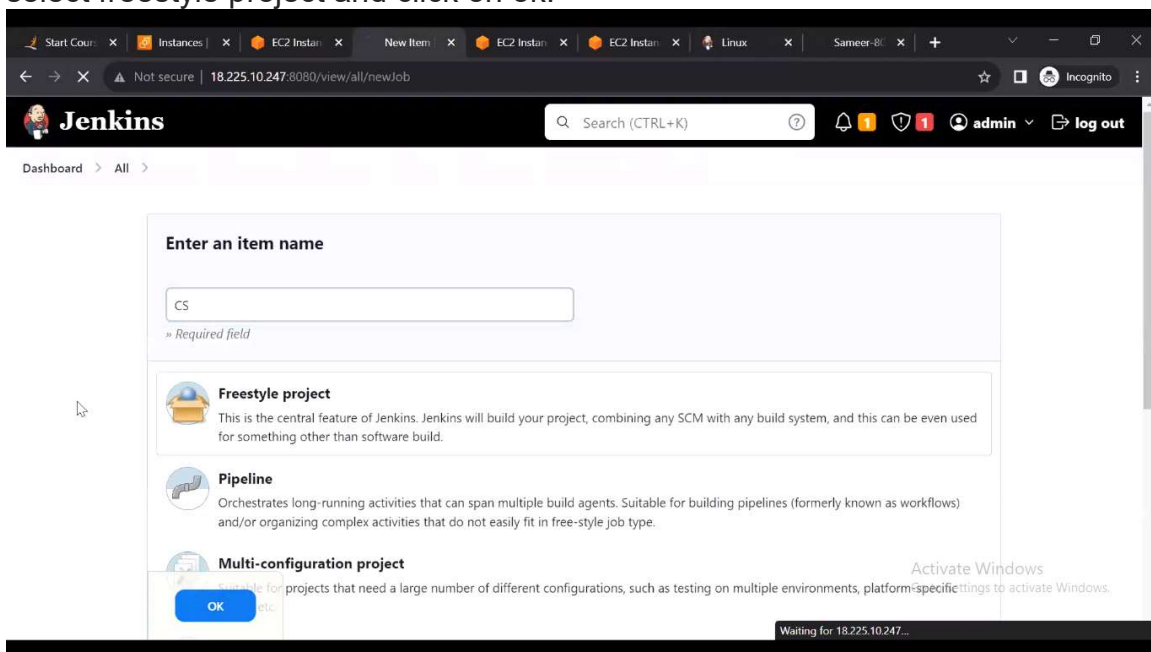
Activate Windows
Go to Settings to activate Windows.

REST API Jenkins 2.426

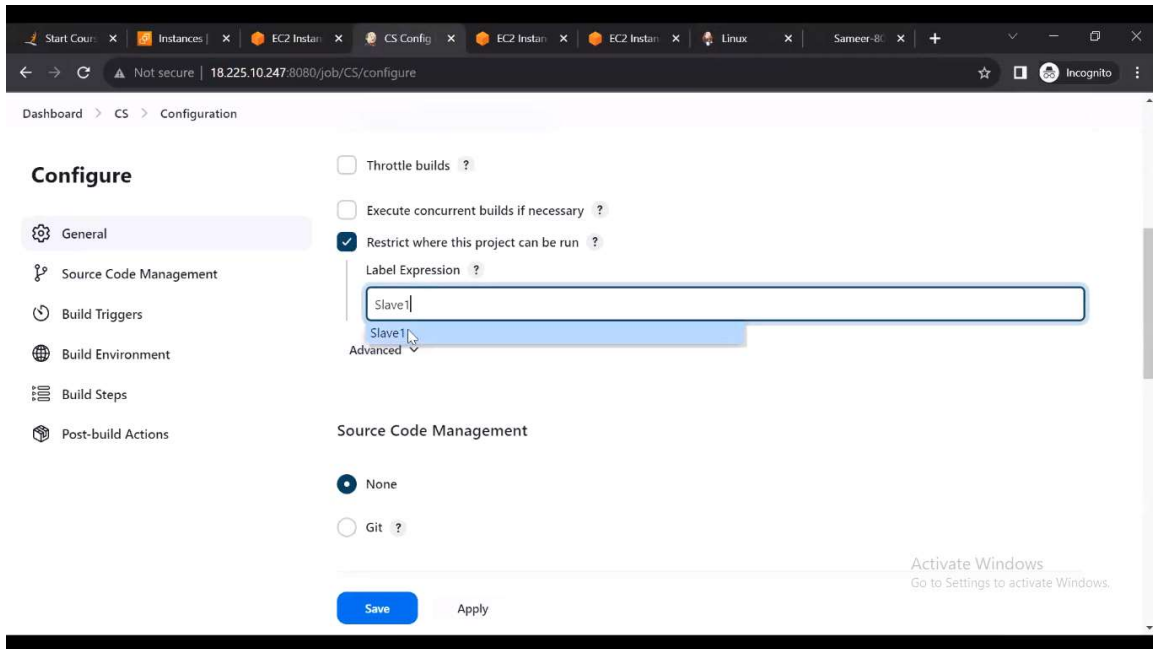
15. Open the repository that provided



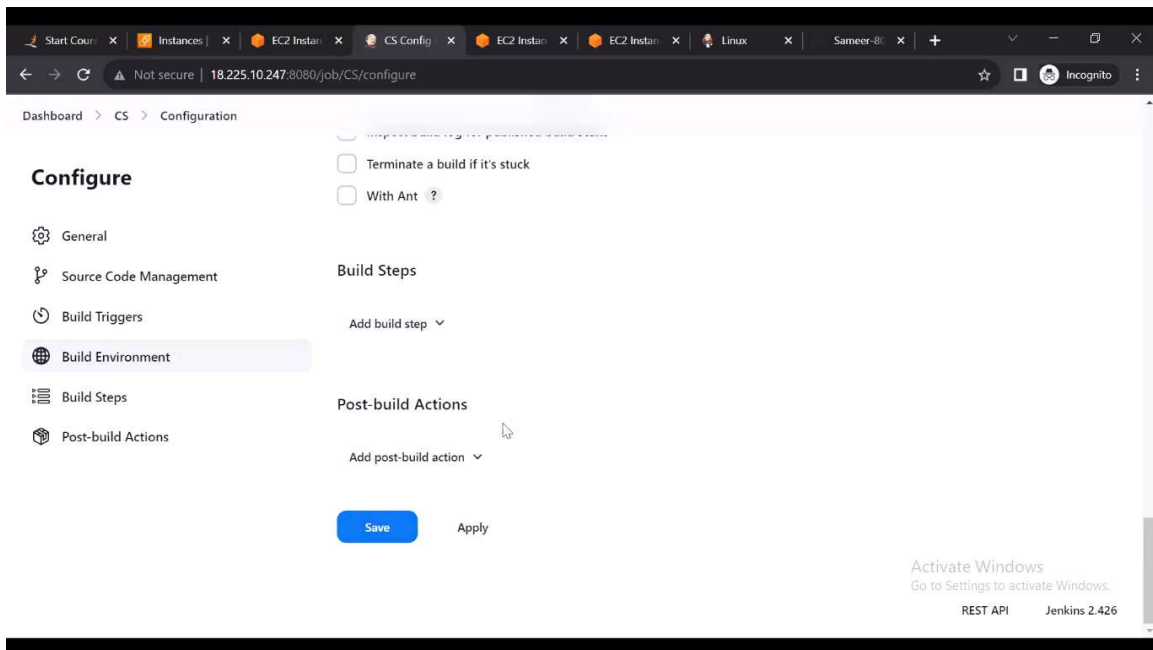
16. Click on fork to create fork and then click on new item and name the item select freestyle project and click on ok.



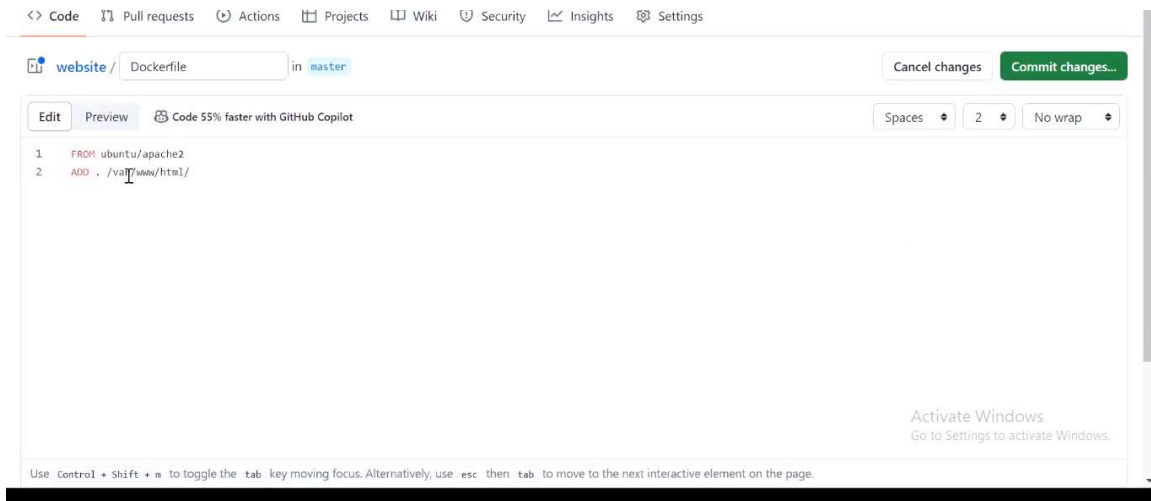
17. Select restrict where give label as slave1



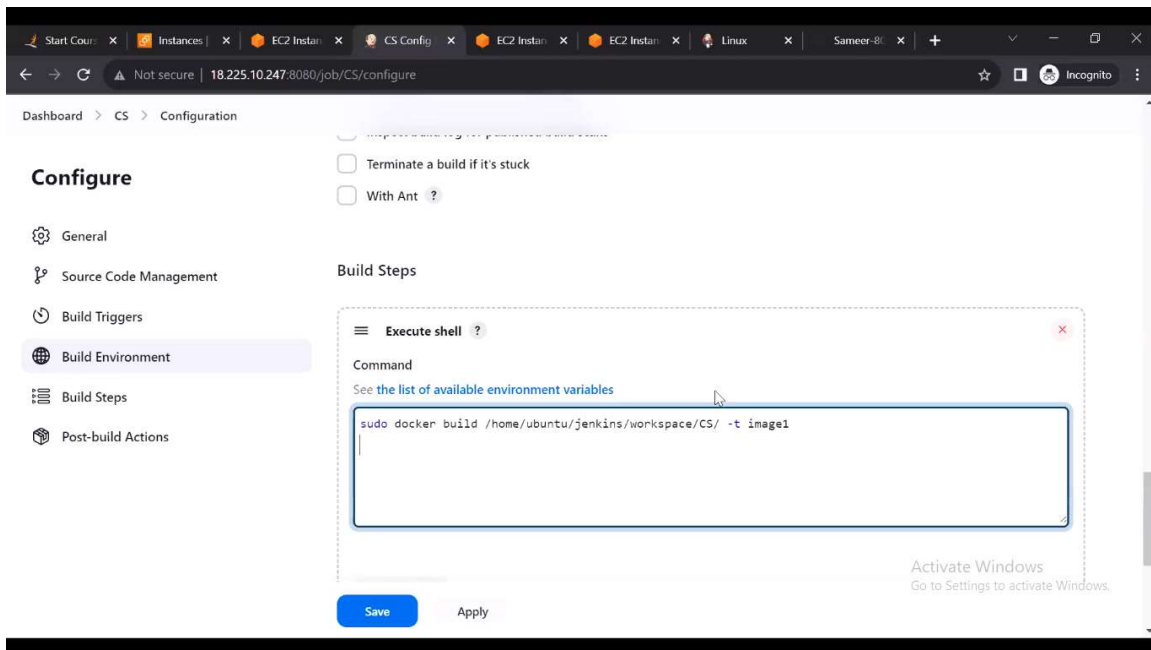
18. Select git and give repository url and click on save.

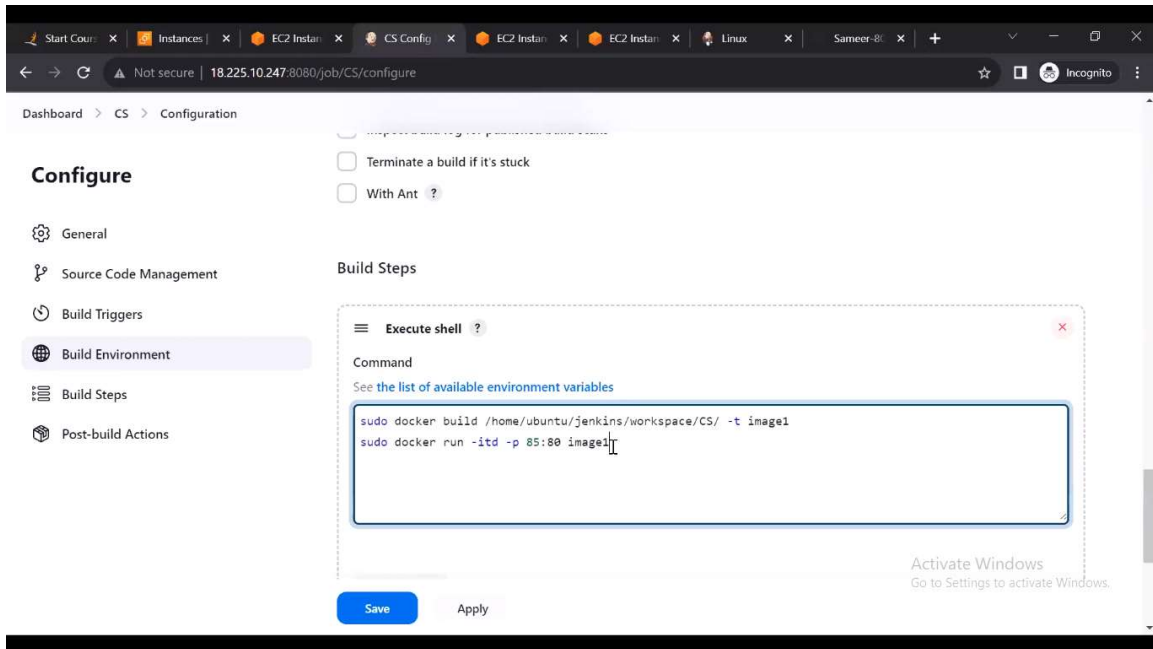


19. Create a new file ie docker file.

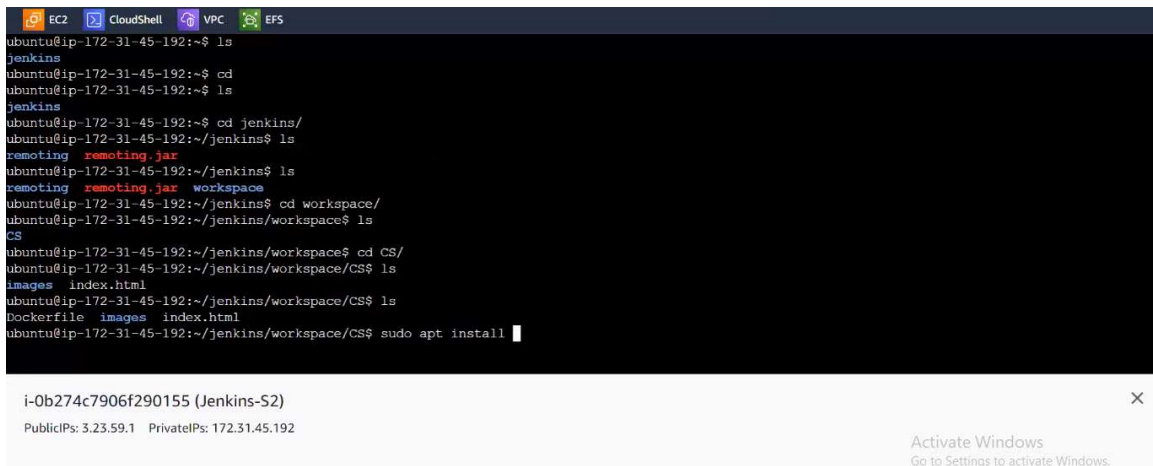


20. Configuration of the job. And click on save.

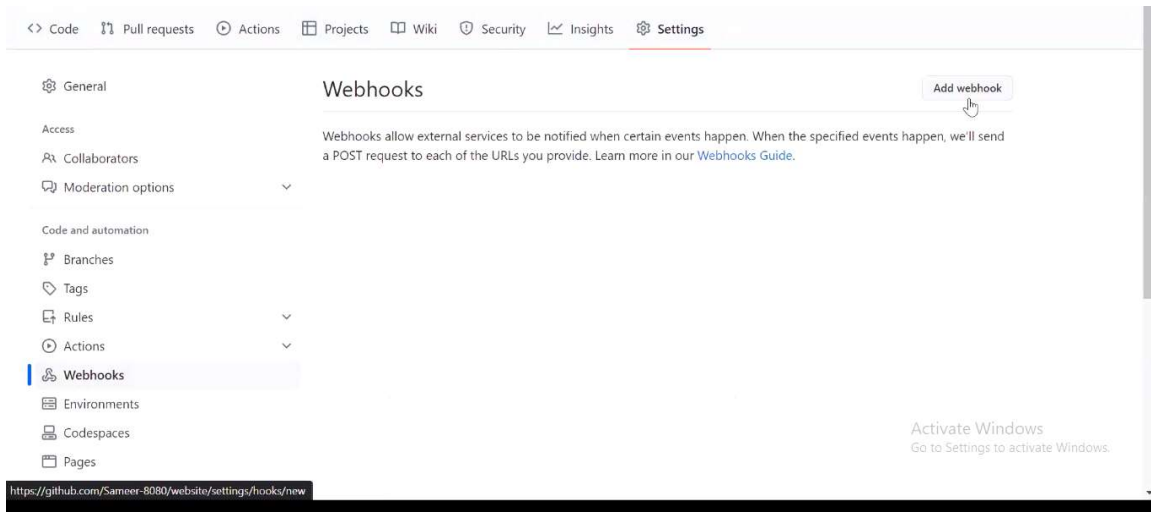




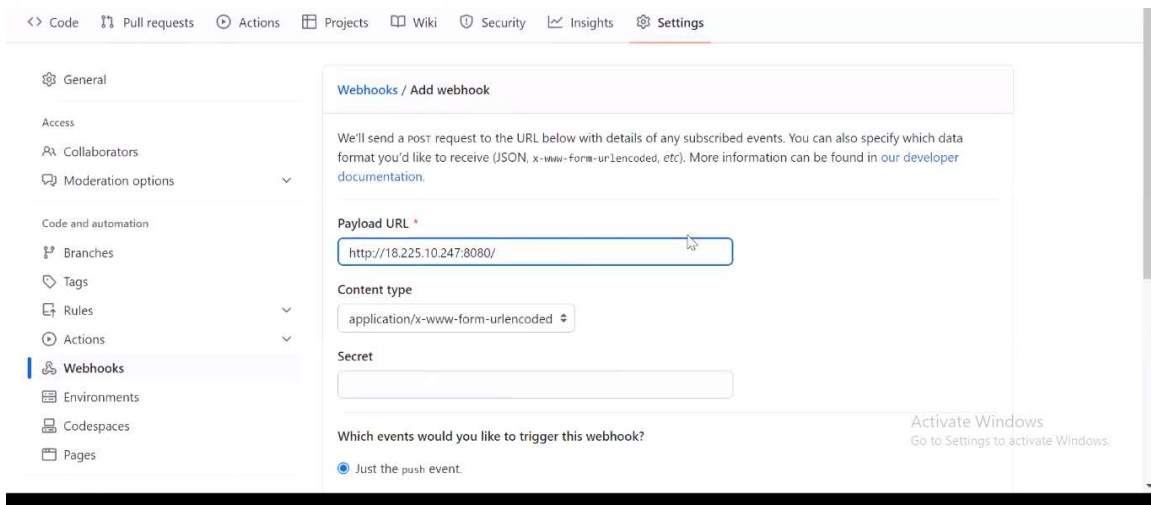
21. Install docker and check output



22. Create the webhook for automatically triggering function



23. give genkins url in payload url abd click on add webhook.



24. configure the job select the github hook trigger click on save.
-Make the changes in the file in repository and check for trigger function

