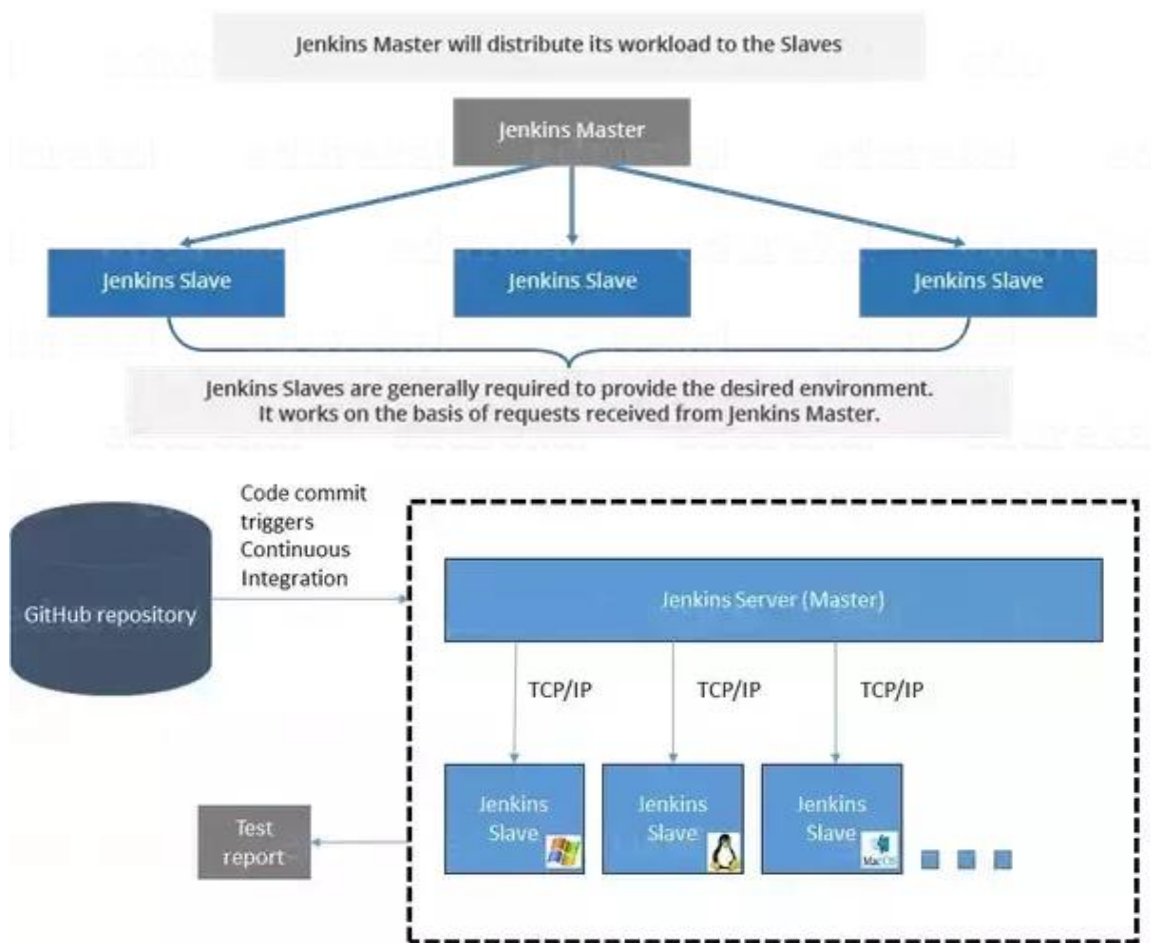


use of master and slave in Jenkins

- Jenkins checks the Git repository at periodic intervals for any changes made in the source code.
- Each builds requires a different testing environment which is not possible for a single Jenkins server. In order to perform testing in different environments Jenkins uses various Slaves as shown in the diagram.
- Jenkins Master requests these Slaves to perform testing and to generate test reports.



Jenkins distributed Architecture

- Jenkins uses a Master Slave architecture to manage the distributed builds
- The Master and slave communicate through Tcp/IP Protocol.

Jenkins Master: Your main Jenkins server is the Master.

- Scheduling build jobs.
- Dispatching builds to the slaves for the actual execution.
- Monitor the slaves (possibly taking them online and offline as required).
- Recording and presenting the build results.
- A Master instance of Jenkins can also execute build jobs directly.

Jenkins Slave: A Slave is a Java executable that runs on a remote machine

- It hears requests from the Jenkins Master instance.
- Slaves can run on a variety of operating systems.
- The job of a Slave is to do as they are told to, which involves executing build jobs dispatched by the Master.

How to configure Jenkins Master to Slaves/Nodes/Clients

Create 3 instances in AWS which ever you want

I am taking 3 ubuntu instances i.e

- 1) Jenkins_Master → Private Ip: 172.31.95.197
- 2) Jenkins_Slave1
- 3) Jenkins_Slave2

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
<input checked="" type="checkbox"/>	Jenkins_Master	i-0d2dc45da8c4c1c7a	t2.micro	us-east-1a	running	Initializing	None	ec2-34-205-16-203.co...
<input type="checkbox"/>	Jenkins_Slave1	i-09100aae5e6da1b...	t2.micro	us-east-1a	running	Initializing	None	ec2-34-238-121-109.co...
<input type="checkbox"/>	Jenkins_Slave2	i-0160dca4646a413a5	t2.micro	us-east-1a	running	Initializing	None	ec2-54-208-63-124.co...

Install Jenkins on Master -> **Open the Terminal of Jenkins_Master**

- switch user to root → sudo su -
- switch user to Jenkins → sudo su - jenkins
- now generate sshkey in Master

```
root@ip-172-31-95-197: ~
root@ip-172-31-95-197:~# sudo su - jenkins
jenkins@ip-172-31-95-197:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/var/lib/jenkins/.ssh/id_rsa):
Created directory '/var/lib/jenkins/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /var/lib/jenkins/.ssh/id_rsa.
Your public key has been saved in /var/lib/jenkins/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:4cWEn7ZwugnLFCt7KVf1wK9G4FuMZRM5WiV/tswyqHY jenkins@ip-172-31-95-197
The key's randomart image is:
+---[RSA 2048]-----+
|      .oo.      |
|      .o=o      |
|      .o+o. o   |
|      ..+o%. = . |
|      +S%.*o +   |
|      . + =. = oo |
|      = o*E.     |
|      o *.+.o    |
|      + .       |
+---[SHA256]-----+

+---[SHA256]-----+
jenkins@ip-172-31-95-197:~$ cat /var/lib/jenkins/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAQDNxJt5nR9k012NLjbXC2EzqfEH6oRRTF11CG2rgu+3a0U7DGI2E1JFTXuiZivADeO/6QzoDfhhEOKmp3oiCHCV8nAdfbei/ks6z7+qpAjFzbslh9YceL+QyUp5bFy46UHX
SahfubWqb+YL9cmQb+Ati2PMHWvAMaAKR52+riBjEYV+5dgJ5F6mMEOXZnBcfAcKSL2O5mmec2ULIVhQ47fFA8eSzU4baJbMbUgP5sWszV0m22iFh/XtPiYl25McmlekrZ5t0iUjzQ1TuweretkrnzTRARu01MeNspMBMDH+
iJlTIC8ryD67MzoY7hgE02Q2JbMz3+3D7i69MsMI0nNP jenkins@ip-172-31-95-197
jenkins@ip-172-31-95-197:~$
```

- copy the public key and paste into the Jenkins_Slave1 and Jenkins_Slave2

Now open Jenkins_Slave1 terminal

vi .ssh/authorized_keys → esc I, esc o → paste the master public key into this and close esc :wq!

Check is correctly pasted or not → cat .ssh/authorized_keys

```
ubuntu@ip-172-31-83-83:~$ vi .ssh/authorized_keys
ubuntu@ip-172-31-83-83:~$ cat .ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAQDNxJt5nR9k012NLjbXC2EzqfEH6oRRTF11CG2rgu+3a0U7DGI2E1JFTXuiZivADeO/6QzoDfhhEOKmp3oiCHCV8nAdfbei/ks6z7+qpAjFzbslh9YceL+QyUp5bFy46UHX
SahfubWqb+YL9cmQb+Ati2PMHWvAMaAKR52+riBjEYV+5dgJ5F6mMEOXZnBcfAcKSL2O5mmec2ULIVhQ47fFA8eSzU4baJbMbUgP5sWszV0m22iFh/XtPiYl25McmlekrZ5t0iUjzQ1TuweretkrnzTRARu01MeNspMBMDH+
iJlTIC8ryD67MzoY7hgE02Q2JbMz3+3D7i69MsMI0nNP jenkins@ip-172-31-95-197
ubuntu@ip-172-31-83-83:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 12:5c:b0:01:b1:ce
          inet addr:172.31.83.83  Bcast:172.31.95.255  Mask:255.255.240.0
          inet6 addr: fe80::105c:b0ff:fe01:b1ce/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:821 errors:0 dropped:0 overruns:0 frame:0
          TX packets:734 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:330819 (330.8 KB)  TX bytes:107919 (107.9 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:192 errors:0 dropped:0 overruns:0 frame:0
          TX packets:192 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:14456 (14.4 KB)  TX bytes:14456 (14.4 KB)

ubuntu@ip-172-31-83-83:~$
```

Copy the private ip of Jenkins_Slave1 to master for the communication purpose
ssh ubuntu@172.31.83.83

```
jenkins@ip-172-31-95-197:~$ ssh ubuntu@172.31.83.83
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-1049-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.

Last login: Sat Mar 24 10:22:40 2018 from 106.77.190.213
ubuntu@ip-172-31-83-83:~$
```

Master

Exit from that to master

- **Jenkins is written in Java so if want to connect the Jenkins master java is prerequisite so install java in Jenkins_slave1 and also update apt Repo**

sudo apt-get update

sudo apt-get install openjdk-8-jdk

after that create one directory name as Jenkins

mkdir Jenkins → ls → cd Jenkins → pwd

```
ubuntu@ip-172-31-83-83:~$ history
 1  exit
 2  clear
 3  apt-get install openjdk-8-jdk
 4  clear
 5  sudo apt-get update
 6  sudo apt-get install openjdk-8-jdk
 7  history

ubuntu@ip-172-31-83-83:~$ mkdir jenkins
ubuntu@ip-172-31-83-83:~$ ls
jenkins
ubuntu@ip-172-31-83-83:~$ cd jenkins/
ubuntu@ip-172-31-83-83:~/jenkins$ pwd
/home/ubuntu/jenkins
ubuntu@ip-172-31-83-83:~/jenkins$
```

Jenkins_Slave1

Now go to Jenkins Dashboard → <http://54.89.229.86:8080/>

Manage Jenkins → Manage Nodes →



Manage Nodes

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

search

vikranth | log out

Jenkins > Nodes > [ENABLE AUTO REFRESH](#)

[Back to Dashboard](#)
[Manage Jenkins](#)
[New Node](#)
[Configure](#)

Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	master	Linux (amd64)	In sync	6.31 GB	0 B	6.31 GB	0ms
	Data obtained	32 min	32 min	32 min	32 min	32 min	32 min

Refresh status

Click on New Node → give the node Name and select Permanent → ok

search

vikranth | log out

Jenkins > Nodes >

[Back to Dashboard](#)
[Manage Jenkins](#)
[New Node](#)
[Configure](#)

Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

Node name

Jenkins_slave1

☒ 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.

OK

Jenkins > Nodes >

[Back to Dashboard](#)
[Manage Jenkins](#)
[New Node](#)
[Configure](#)

Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

Name

Jenkins_slave1

Description

of executors

1

Remote root directory

Remote directory is mandatory

Labels

Usage

Use this node as much as possible

Launch method

Launch agent via execution of command on the master

Launch command

No launch command specified

Availability

Keep this agent online as much as possible

Node Properties

☐ Environment variables
☐ Tool Locations

Save

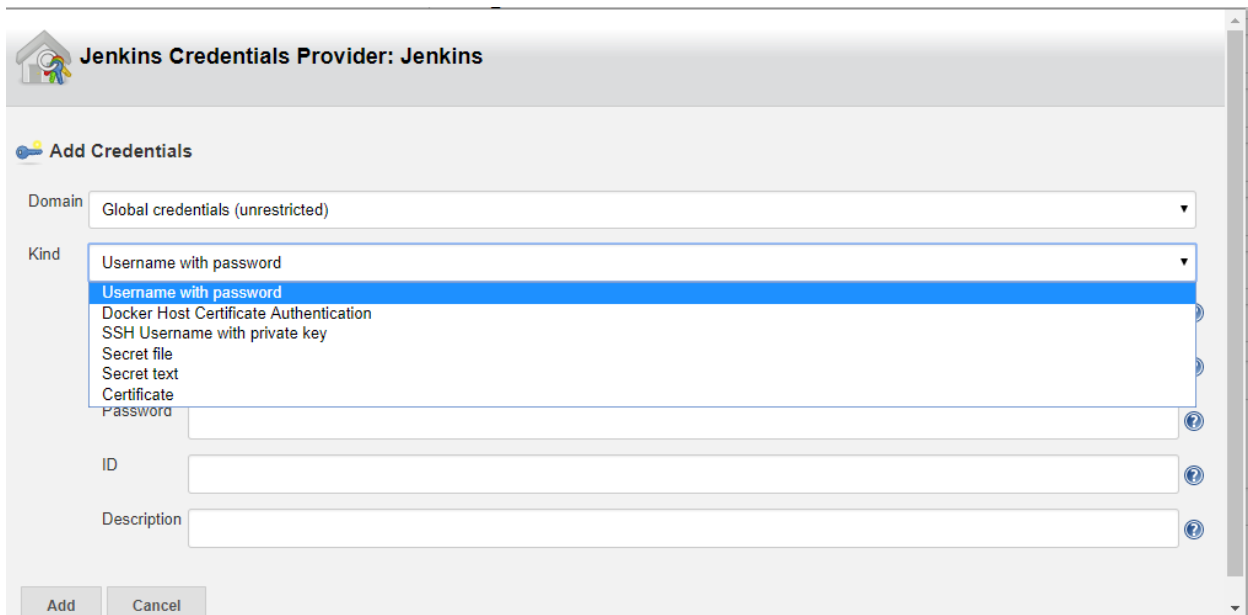
Under Remote root Directory-→ give slave1 home Directory path i.e

/home/ubuntu/Jenkins

Give the labels as → **linux** whatever it is

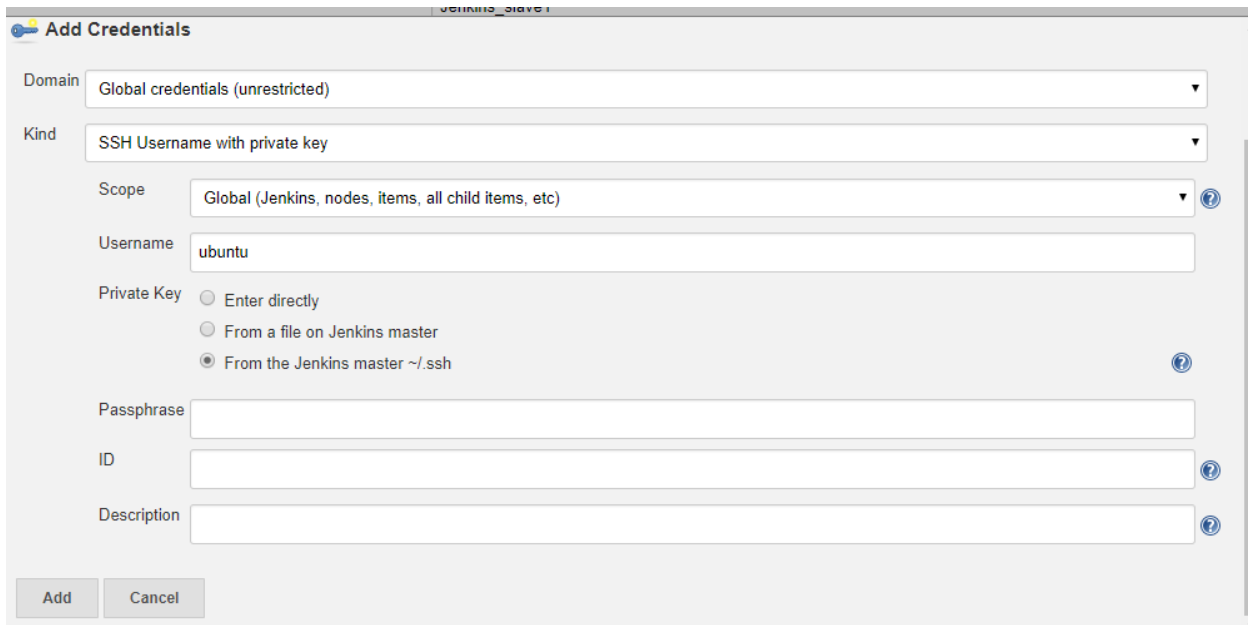
Launch method select the → **Launch ssh agents via SSH**

Host → give the private ip of slave1 → 172.31.83.83




The screenshot shows the 'Jenkins Credentials Provider: Jenkins' interface. Under the 'Add Credentials' section, the 'Domain' is set to 'Global credentials (unrestricted)'. The 'Kind' dropdown menu is open, showing options: 'Username with password', 'Docker Host Certificate Authentication', 'SSH Username with private key', 'Secret file', 'Secret text', 'Certificate', and 'Password'. The 'Username with password' option is highlighted. Below the dropdown, there are input fields for 'ID' and 'Description', each with a help icon. At the bottom are 'Add' and 'Cancel' buttons.

Kind→ username with password

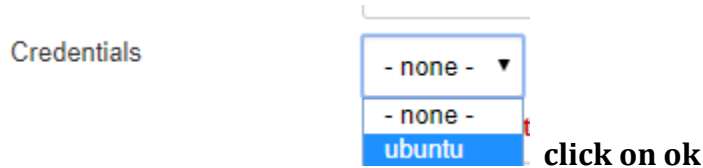


The screenshot shows the 'Jenkins Credentials Provider: Jenkins' interface. Under the 'Add Credentials' section, the 'Domain' is set to 'Global credentials (unrestricted)'. The 'Kind' dropdown menu is set to 'SSH Username with private key'. The 'Scope' dropdown menu is set to 'Global (Jenkins, nodes, items, all child items, etc)'. The 'Username' field contains 'ubuntu'. The 'Private Key' section has three radio buttons: 'Enter directly', 'From a file on Jenkins master', and 'From the Jenkins master ~/.ssh', with the last one selected. There are input fields for 'Passphrase', 'ID', and 'Description', each with a help icon. At the bottom are 'Add' and 'Cancel' buttons.

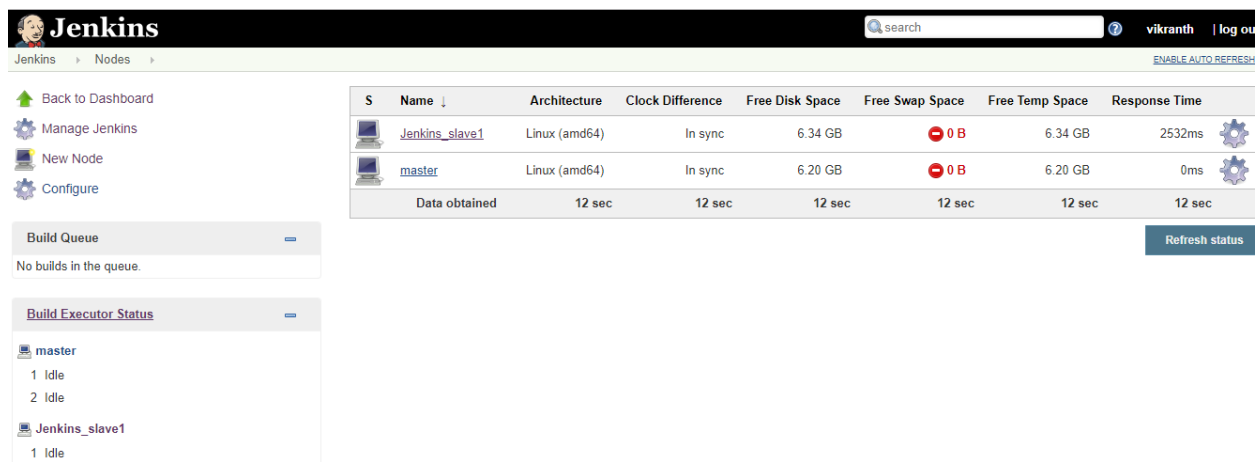
User name as →ubuntu

Private key→  From the Jenkins master ~/.ssh → add

Under credentials → take the user name



Now click on Jenkins_Slave1→

A screenshot of the Jenkins dashboard. The top bar shows the Jenkins logo, a search bar, and the user 'vikranth' with a 'log out' link. Below the top bar, there's a navigation menu with 'Jenkins' and 'Nodes'. The main content area shows a table of nodes. The table has columns: 'S', 'Name', 'Architecture', 'Clock Difference', 'Free Disk Space', 'Free Swap Space', 'Free Temp Space', and 'Response Time'. There are two nodes listed: 'Jenkins_slave1' and 'master'. The 'Jenkins_slave1' node has a response time of 2532ms, while the 'master' node has a response time of 0ms. Below the table, there's a 'Data obtained' row showing times for each column. On the left side of the dashboard, there are links for 'Back to Dashboard', 'Manage Jenkins', 'New Node', and 'Configure'. Below these links, there are two sections: 'Build Queue' (showing 'No builds in the queue.') and 'Build Executor Status' (showing 'master' with 1 idle executor and 'Jenkins_slave1' with 1 idle executor). A 'Refresh status' button is located at the bottom right of the table.

Now Jenkins_Slave1 added to master successfully

Similarly you can add many number of slaves to ur Master

Now connect slave2 to master

Steps:

- Copy the ssh public key and paste into the Slave2
- Check the communication between master and slave2
- Install the java in slave2
- Create the directory as Jenkins in slave
- **on Jenkins dashboard→**

Manage nodes→

add node→

give remote directory as slave home directory → **/home/ubuntu/jenkins**

labels name -- **linux**→

Take launch method as → **Launch slave agents via SSH**

Host → **give the slave2 private ip**

Click on **add** → **Jenkins**→ kind→ **SSH Username with private key**

User Name → **ubuntu**

Private key → select **From the Jenkins Master ~/.ssh** → **add**

Credentials →

take user name → **ubuntu**

save

click on **launch the slave2**

```
ubuntu@ip-172-31-82-254:~$ vi .ssh/authorized_keys
ubuntu@ip-172-31-82-254:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 12:4e:f9:c7:7f:bc
          inet addr:172.31.82.254  Bcast:172.31.95.255  Mask:255.255.240.0
          inet6 addr: fe80::104e:f9ff:fec7:7fbc/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:563 errors:0 dropped:0 overruns:0 frame:0
          TX packets:525 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:303801 (303.8 KB)  TX bytes:66875 (66.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:192 errors:0 dropped:0 overruns:0 frame:0
          TX packets:192 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:14456 (14.4 KB)  TX bytes:14456 (14.4 KB)
```


Slave_2

sudo apt-get update

sudo apt-get install openjdk-8-jdk

mkdir Jenkins → **/home/ubuntu/Jenkins**

now go to Jenkins dashboard → **Manage Jenkins**→ **Manage Nodes**→ **New Node**

 Jenkins

[vikranth](#) | [log out](#)

[Jenkins](#) > [Nodes](#) > [ENABLE AUTO REFRESH](#)

[Back to Dashboard](#)
[Manage Jenkins](#)
[New Node](#)
[Configure](#)

Build Queue

No builds in the queue.



Build Executor Status

master


1 Idle
2 Idle

Jenkins_slave1

1 Idle

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Jenkins_slave1	Linux (amd64)	In sync	6.34 GB	0 B	6.34 GB	1691ms
	master	Linux (amd64)	In sync	6.20 GB	0 B	6.20 GB	0ms
Data obtained		22 min	22 min	22 min	22 min	22 min	22 min

Refresh status

 Jenkins

[vikranth](#) | [log out](#)

[Jenkins](#) > [Nodes](#) >

[Back to Dashboard](#)
[Manage Jenkins](#)
[New Node](#)
[Configure](#)

Build Queue

No builds in the queue.

Build Executor Status

master

1 Idle
2 Idle

Jenkins_slave1

1 Idle

Node name

Jenkins_Slave2


☐ 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.

☒ Copy Existing Node

Copy from Jenkins_slave1

OK

 Jenkins

[vikranth](#) | [log out](#)

[Jenkins](#) > [Nodes](#) > [Jenkins_Slave2](#)

[Back to List](#)
[Status](#)
[Delete Agent](#)
[Configure](#)
[Build History](#)
[Load Statistics](#)
[Log](#)

Build Executor Status

Name

Jenkins_Slave2

Description

of executors

1

Remote root directory

/home/ubuntu/jenkins

Labels

linux

Usage

Use this node as much as possible

Launch method

Launch slave agents via SSH

Host

172.31.82.254

Credentials

ubuntu Add

Host Key Verification Strategy

Known hosts file Verification Strategy

Advanced...

Save

Give the all details and save

Back to List

Status

Delete Agent

Configure

Build History

Load Statistics

Log

Build Executor Status

Agent Jenkins_Slave2

This node is being launched. [See log for more details](#)

Relaunch agent

Labels

linux

Projects tied to Jenkins_Slave2

None

Mark this node temporarily offline

Click on Relaunch agent

And go to nodes→

Back to Dashboard

Manage Jenkins

New Node

Configure

Build Queue

Build Executor Status

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Jenkins_slave1	Linux (amd64)	In sync	6.34 GB	0 B	6.34 GB	1280ms
	Jenkins_Slave2	Linux (amd64)	In sync	6.34 GB	0 B	6.34 GB	48ms
	master	Linux (amd64)	In sync	6.20 GB	0 B	6.20 GB	0ms
Data obtained		1 min 15 sec	1 min 15 sec	1 min 15 sec	1 min 15 sec	1 min 15 sec	1 min 15 sec

Refresh status

master
 1 Idle
 2 Idle
Jenkins_Slave2
 1 Idle
Jenkins_slave1
 1 Idle

