**How To Setup Slaves On Jenkins :**

There are two ways for setting up the slaves.

1. Using username and password
2. Using ssh keys

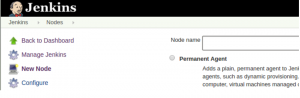
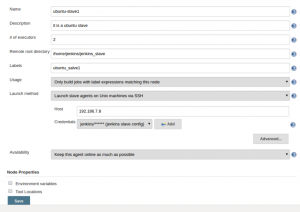
Explaining slave setup using username and password. Preparing slaves using ssh key follow the same steps only difference is that instead of username and password for slave , ssh key is used.

**Slave Prerequisites :**

1. Create a Jenkins user and a password using the following command  
**sudo adduser jenkins –shell /bin/bash**  
2. Now, login as Jenkins user.  
**su jenkins**  
3. Create a “jenkins\_slave” directory under /home/jenkins  
**mkdir /home/jenkins/jenkins\_slave**

**Note : Before Executing above steps JRE (Java Run time Environment ) need to install on slave machine.**

**Setting Up Slaves Using Username And Password :**

1. Go to **Jenkins dashboard** -> **Manage Jenkins** -> **Manage Nodes.**
2. Select **new node** option.  
   
3. Give it a name, select the “**permanent agent**” option and click**OK**.
4. Enter the details as shown in the image below and save it. For credential box, click the add button and enter the slaves Jenkins username and password (For logging into the slave machine) that created during the slave configuration. To know what each option means, click the question mark at the right side of each text box.In above Diagram :**Name** : Name that uniquely identifies an agent within this Jenkins installation.  
   **Description** : Optional ,description for this agent.  
   **# of executors** : The maximum number of concurrent builds that Jenkins may perform on this agent.  
   **Remote root directory** : An agent needs to have a directory dedicated to Jenkins. Specify the path to this directory on the agent. It is best to use an absolute path, such as /var/jenkins or c:\jenkins. This should be a path local to the agent machine. There is no need for this path to be visible from the master.  
   **Labels** : Labels (or tags) are used to group multiple agents into one logical group.**Usage** : Controls how Jenkins schedules builds on this node.Use this node as much as possible only build jobs with label expressions matching this node(select if above label field is set.)

**Launch method** : It Controls how Jenkins starts this agent.  
**Launch agent via Java Web Start** :  
Allows an agent to be launched using Java Web Start. In this case, a JNLP file must be opened on the agent machine, which will establish a TCP connection to the Jenkins master. This means that the agent need not be reachable from the master,the agent just needs to be able to reach the master. By default, the JNLP agent will launch a GUI, but it’s also possible to run a JNLP agent without a GUI, e.g. as a Window service.  
Launch agent via execution of command on the master :  
Starts an agent by having Jenkins execute a command from the master. Use this when the master is capable of remotely executing a process on another machine, e.g. via SSH or RSH**.**  
**Launch slave agents via SSH**:  
Starts a slave by sending commands over a secure SSH connection. The slave needs to be reachable from the master, and master will have to supply an account that can log in on the target machine. No root privileges are required.  
**Let Jenkins control this Windows slave as a Windows service**:  
In case of windows server as slave. Please use this option. Please do it in your own risk as it is not tested in this process.  
Start and stop this node on-demand (Need slave setup plugin):  
Wrapper for other launch methods: executes a script (i.e. to provision a VM) before attempting to connect to the slave machine  then launches the Jenkins slave with a launch method of user choice, and after disconnection executes another script (i.e. for shutting down the VM).  
**Availability** : Controls when Jenkins starts and stops this agent. Keep this agent online as much as possible.  
Take this agent online and offline at specific times :

In this we set “Startup Schedule” and “Scheduled Up time”

**Startup Schedule** : schedule cron job to start and stop node.  
Examples:  
every fifteen minutes : H/15 \* \* \* \*  
**Scheduled Up time** : The number of minutes to keep the node up for. If this is longer than the startup schedule, then the node will remain constantly on-line.Take this agent online when in demand, and offline when idle

1. Click on save button to create new new node  and after that  Jenkins will automatically connect to the slave machine

**How to prepare slave to execute build?**  
To prepare slave setup  and configuration we should  install “Slave Setup” plugin.This plugin prepares slave to execute build . It also allows Jenkins master to start and stop slave on demand from the master node.

**Steps to prepare Slave Server :**

1. Go to  Manage Jenkins->Configure System and check the “Slave Setups” block and click on add button.
2. **pre-launch script**:  
   After configuring the slave on Jenkins master ,slave should be launch from Jenkins master to execute build on slave . Pre launch script is executed before Jenkins-master launch the slave.  
   **Steps to install dependencies on slave using pre-launch script :**Create shell script on $JENKINS\_HOME(i.e. /var/lib/jenkins/). Set that shell script path inpre-launch script. **Example** : ${JENKINS\_HOME}/prelaunch\_dev.sh .  
   Go to Manage Jenkins->Manage Nodes->Click on the node name-> Click on “Launch Agent” button.  
   After clicking on “Launch Agent” Button ,Pre-launch script will be executed
3. **prepare script:**Specify the command to be executed on the master before Jenkins starts using the slave. Follow the steps same as “pre-launch script” and specify the prepare script path in front of “prepare script” in below image.
4. **Label Expression :**In master slave configuration label is used to prepare build on slave and Configure the slave.In slave setup “label expression” is used to identify the slave Node to install dependencies.



**Test The Slave :**

To test the slave, create a sample project and select the option as shown below.In below image  add Node label name in “label Expression “ to provide connection between jobs and slave node.

**Note : Label name in the below image should be same as the Slave label name.**

