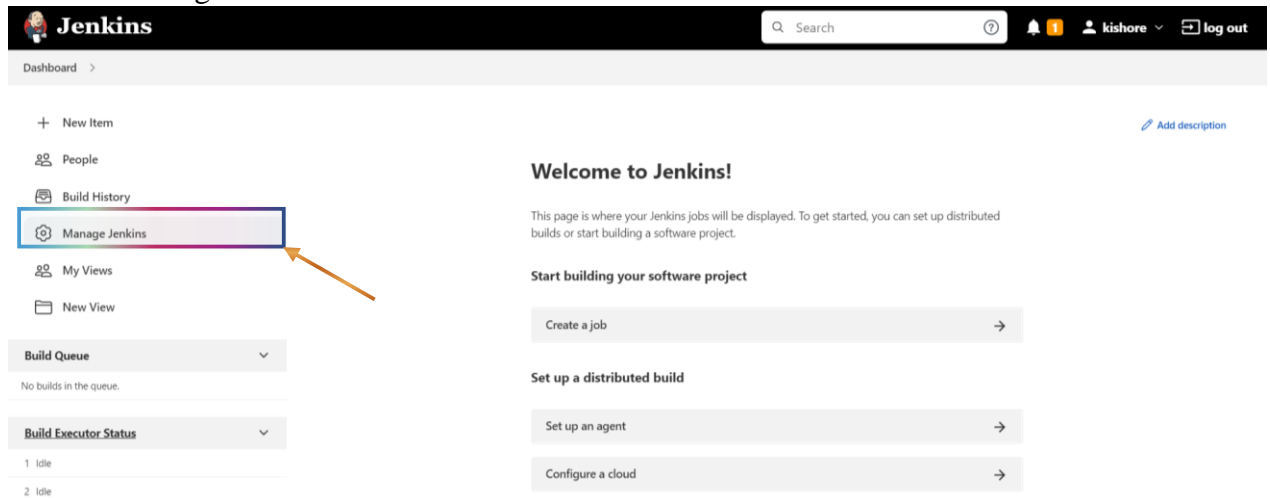


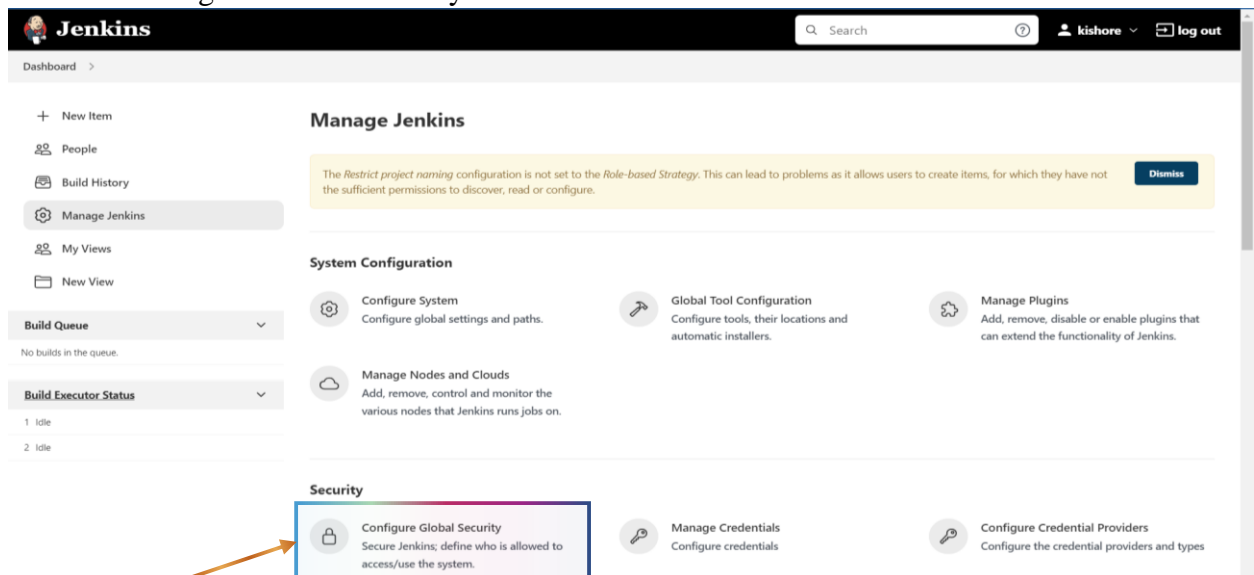
MASTER-SLAVE CONFIGURATION

1. Creating slave node:

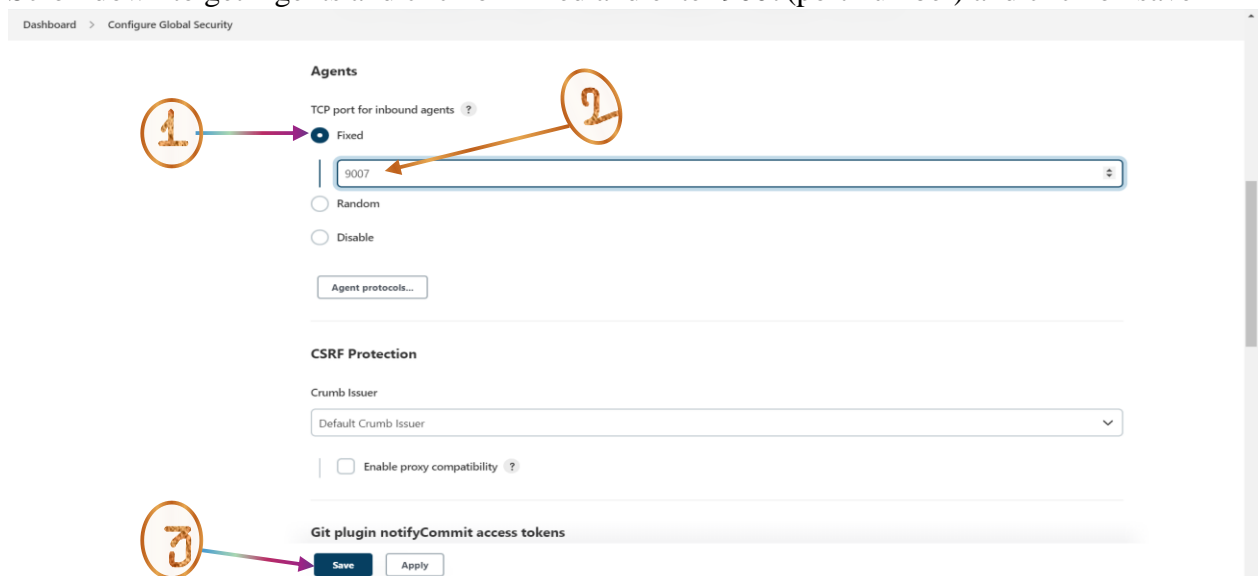
- i) Open and Sign-in to Jenkins
- ii) Click on manage Jenkins



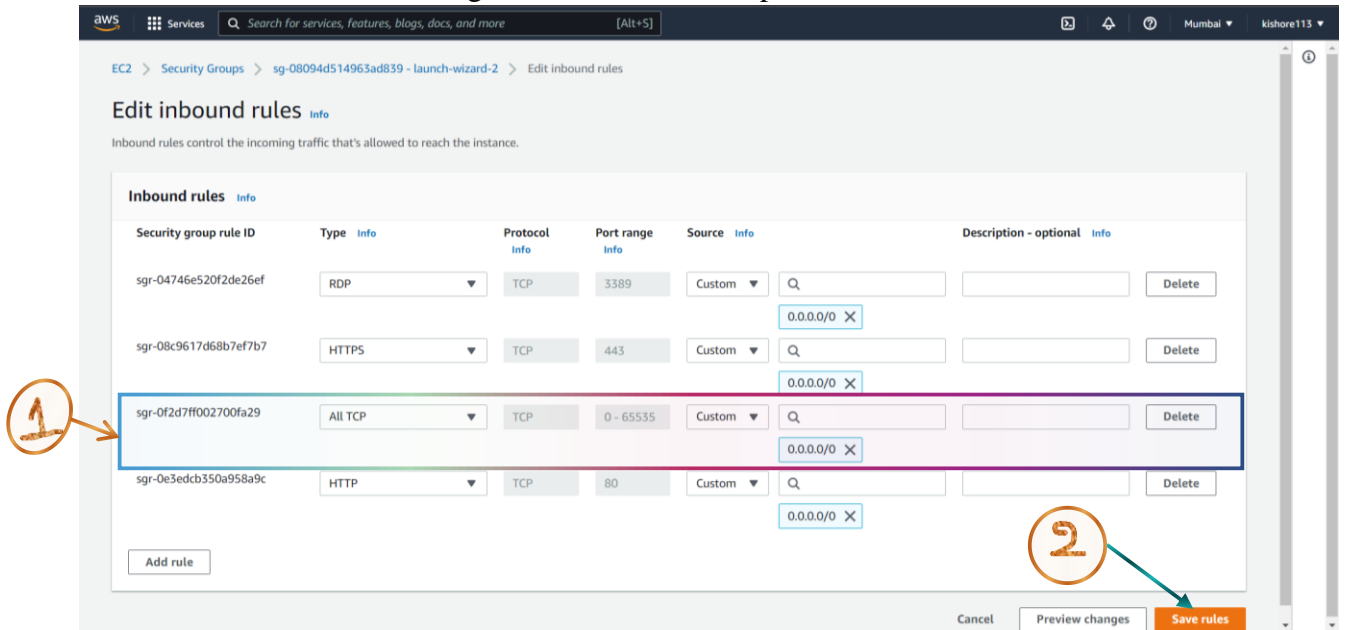
- iii) Click on Configure Global Security



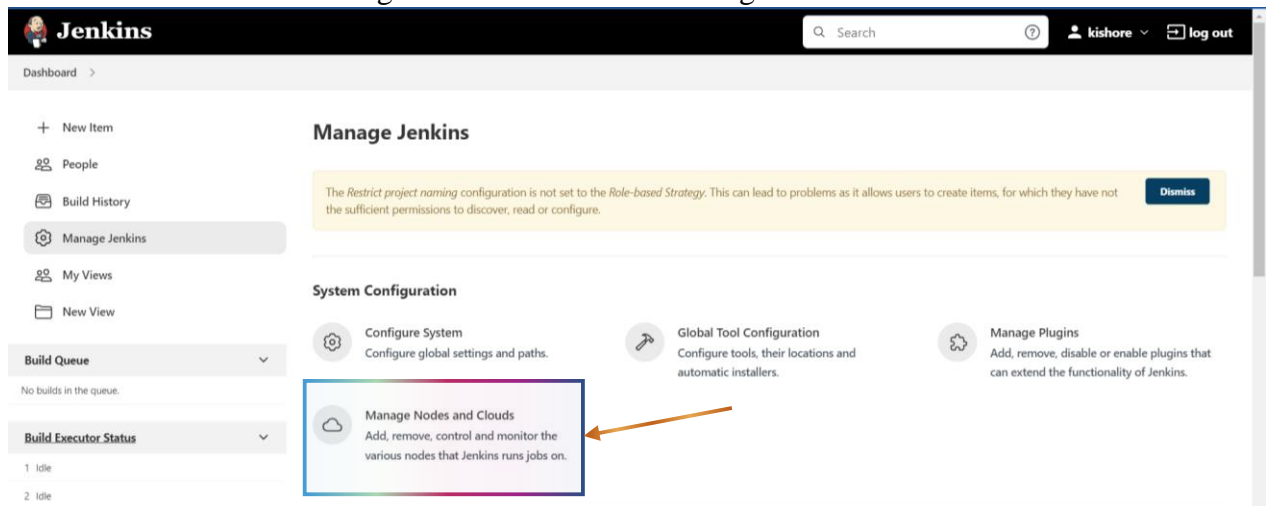
- iv) Scroll down to get Agents and click on Fixed and enter 9007(port number) and click on save



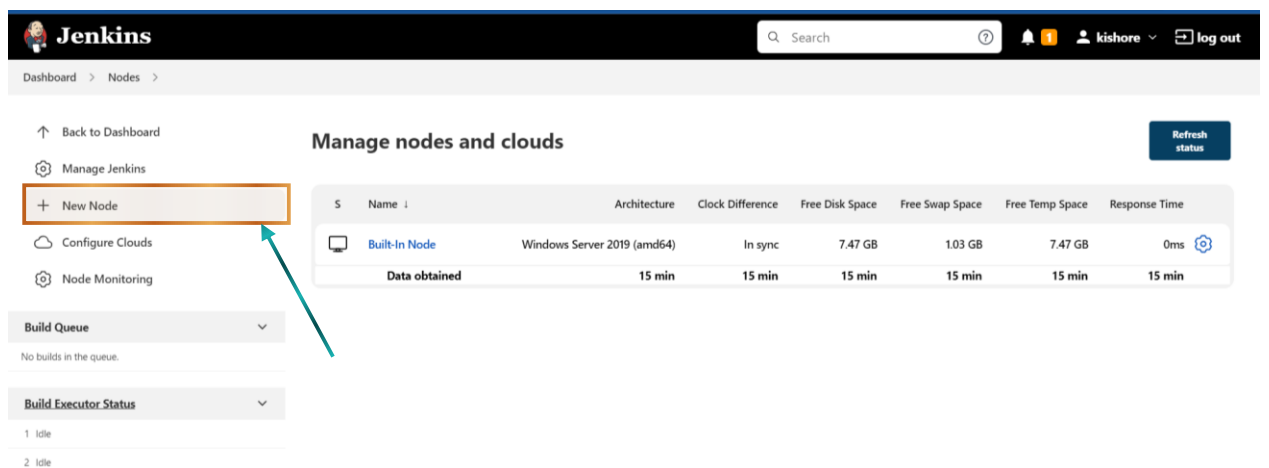
- v) Goto AWS -> EC2 instances and open security group assigned to your instance
- edit the inbound rules and give access to all TCP ports and click on save rules.



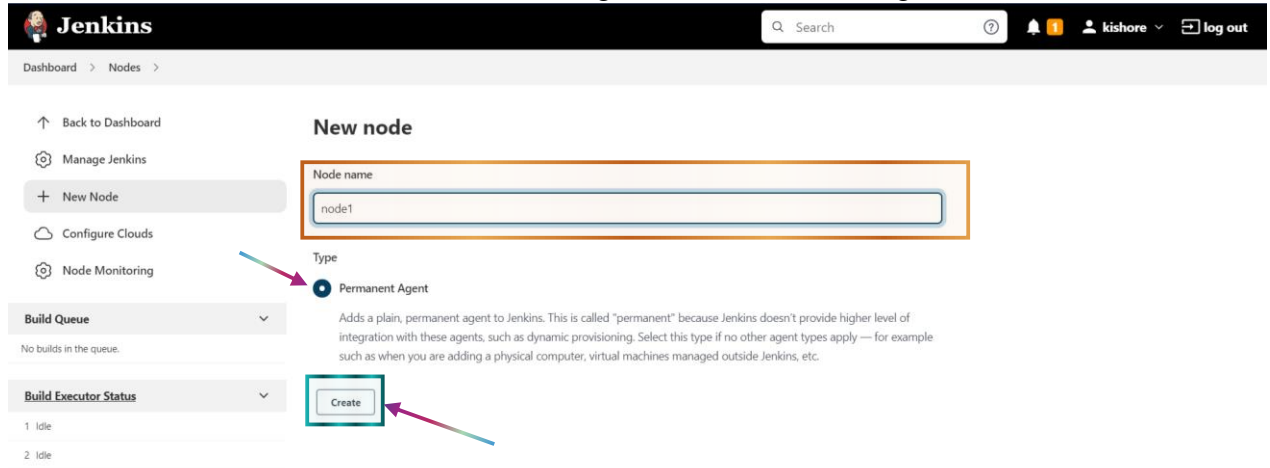
- vi) Get back to Jenkins -> Manage Jenkins -> click on Manage Nodes and Clouds



- vii) Click on New Node

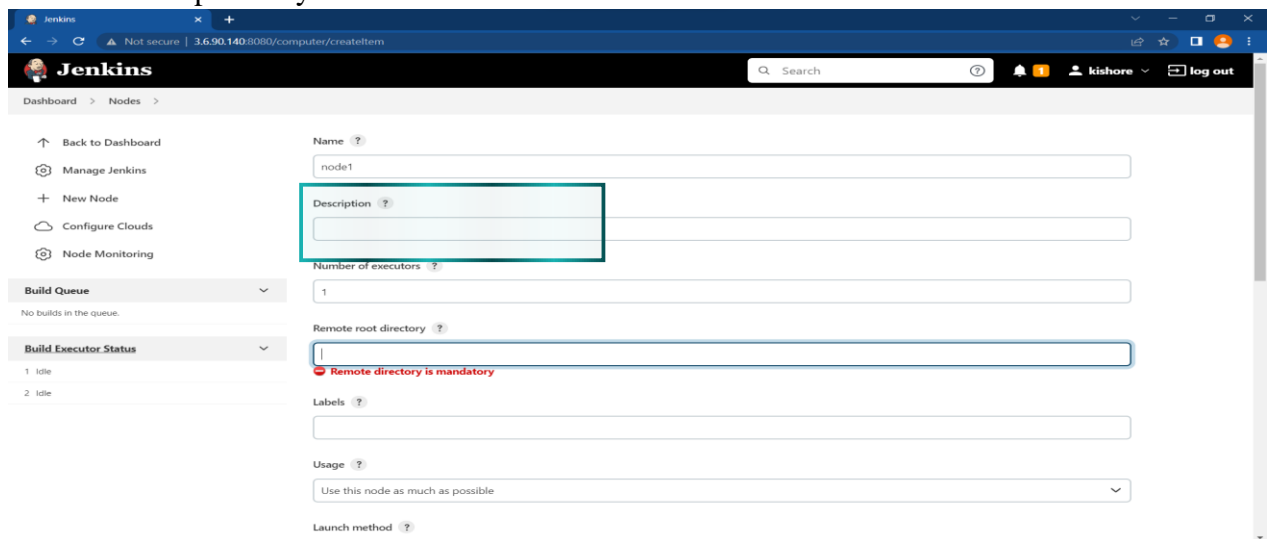


viii) Enter the Node name and check the tick box against the Permanent Agent and click on Create



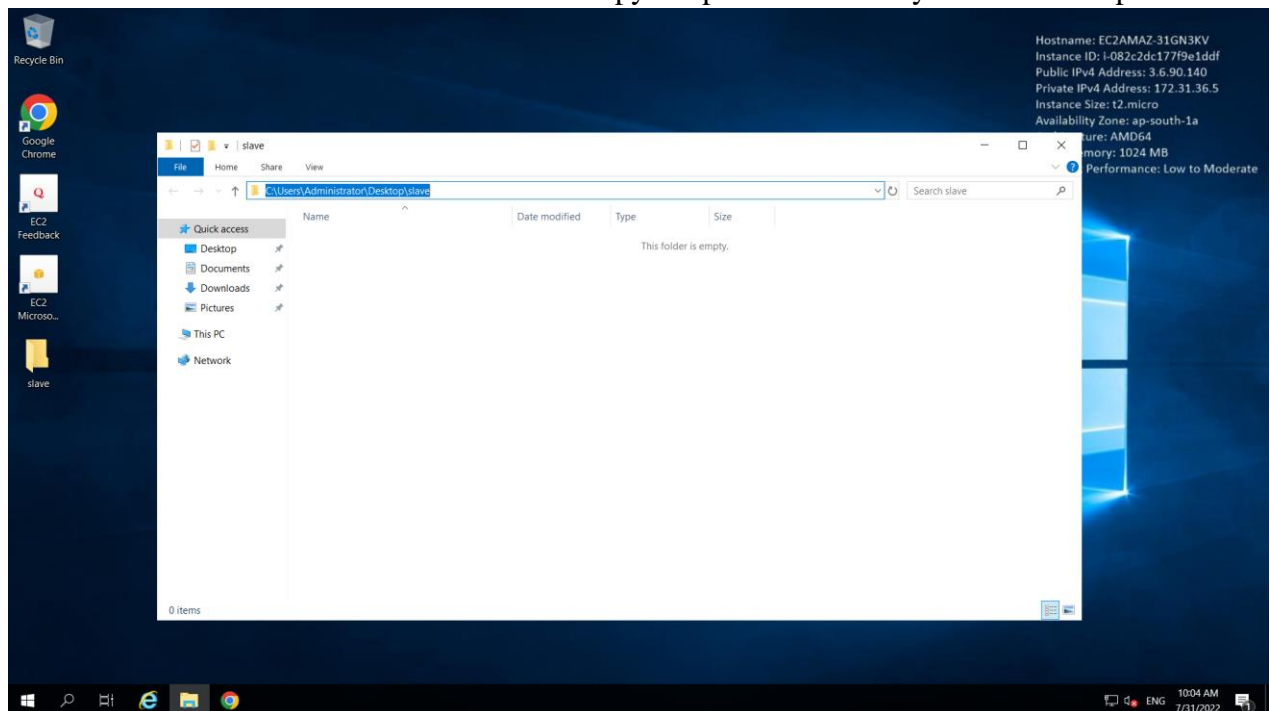
The screenshot shows the Jenkins 'New node' configuration page. The 'Node name' field is set to 'node1'. The 'Type' is set to 'Permanent Agent'. The 'Create' button is highlighted with a red box and a red arrow. The left sidebar shows the 'New Node' button and the 'Build Queue' and 'Build Executor Status' sections.

ix) Add the description if you wanna add it

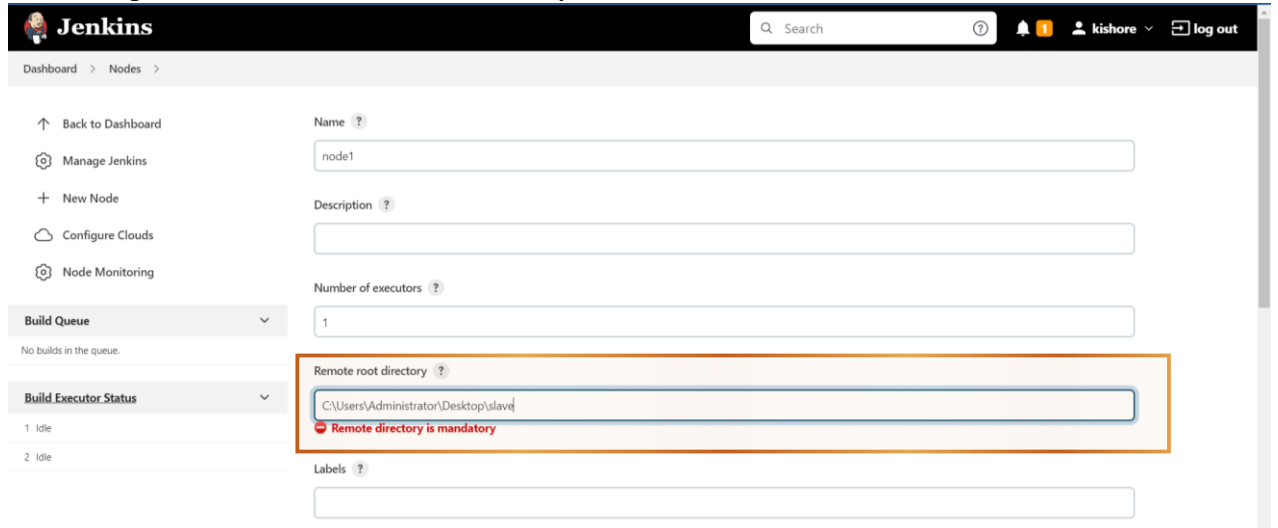


The screenshot shows the Jenkins 'New node' configuration page with the 'Description' field highlighted by a red box. The 'Name' field is 'node1', 'Number of executors' is '1', 'Remote root directory' is empty, 'Labels' is empty, 'Usage' is 'Use this node as much as possible', and 'Launch method' is 'Use this node as much as possible'. A red error message 'Remote directory is mandatory' is visible below the 'Remote root directory' field.

x) Next create a folder in the EC2 machine and copy the path of it where you want to keep the slave

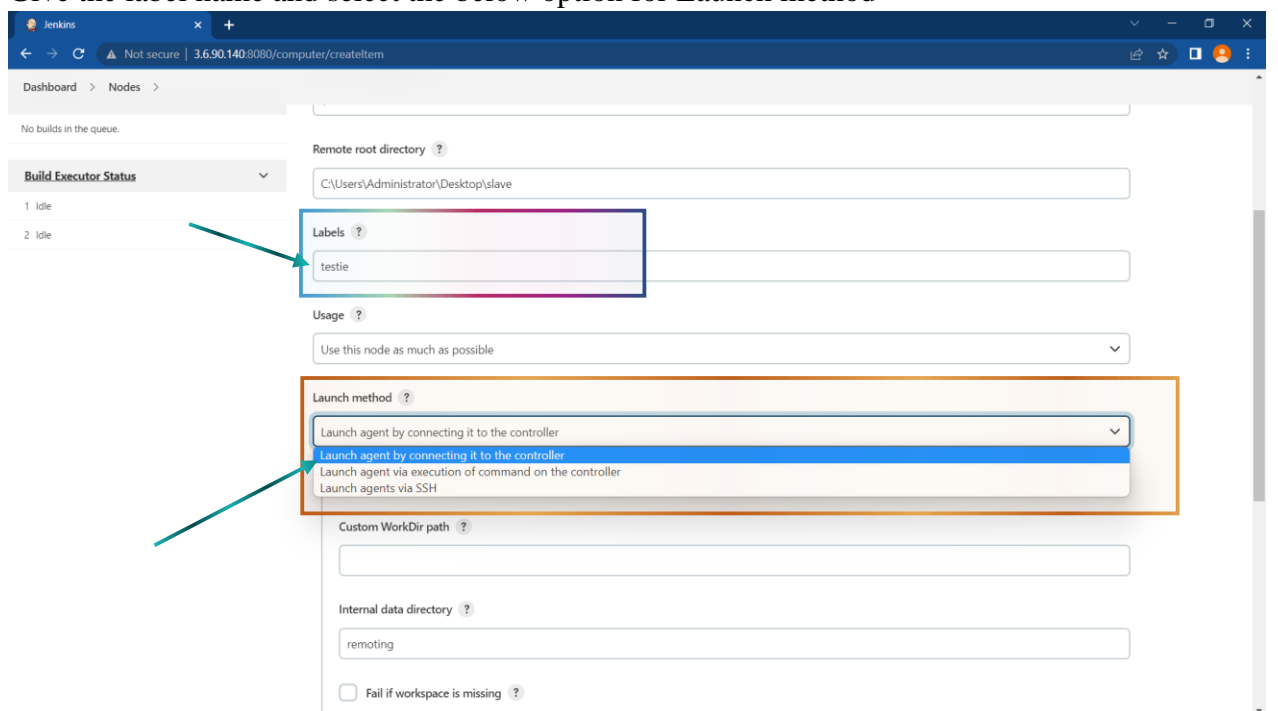


xi) Paste the path in the Remote Root Directory field



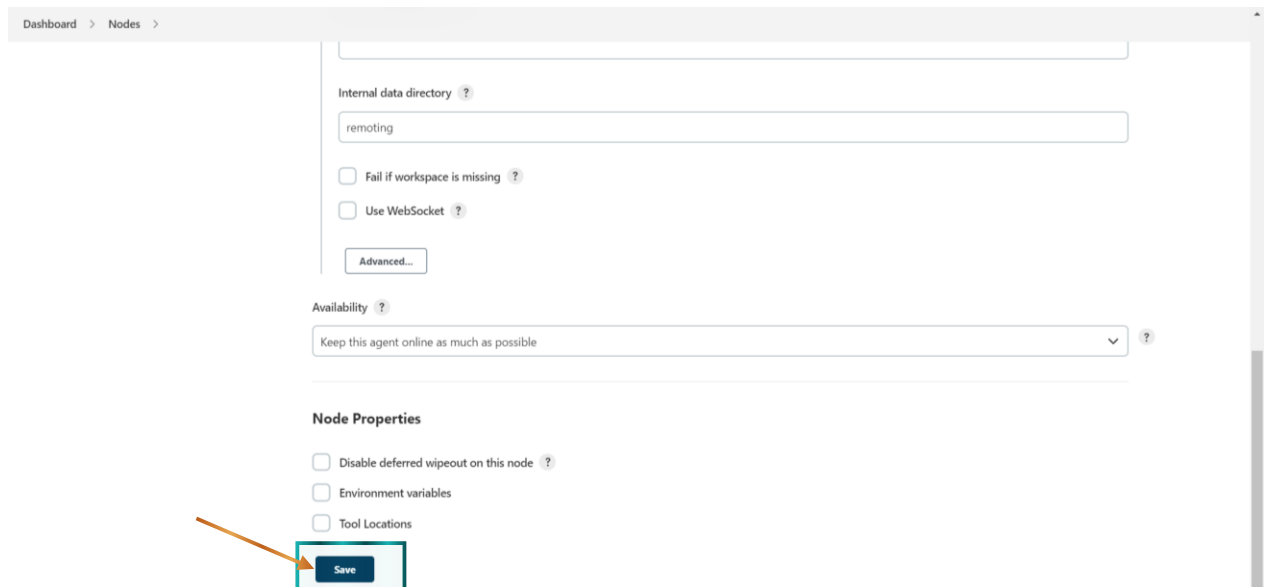
The screenshot shows the Jenkins 'New Node' configuration page. The 'Name' field is 'node1'. The 'Remote root directory' field contains 'C:\Users\Administrator\Desktop\slave' and is highlighted with an orange box. A red error message 'Remote directory is mandatory' is displayed below it. The 'Number of executors' is set to '1'. The 'Labels' field is empty. The left sidebar shows the 'Build Queue' and 'Build Executor Status' sections.

xii) Give the label name and select the below option for Launch method



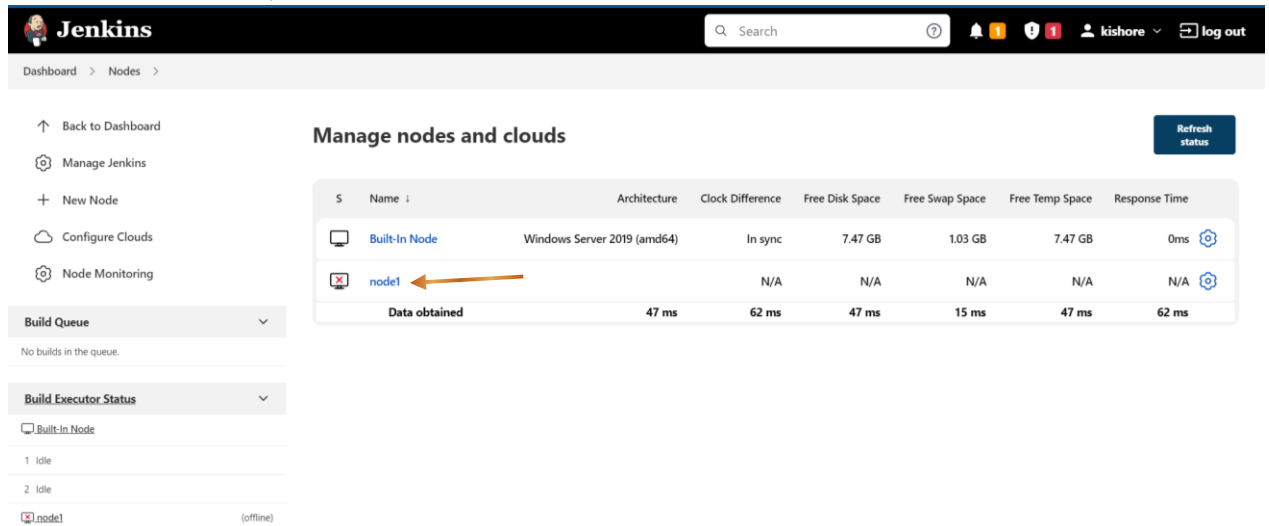
The screenshot shows the Jenkins 'New Node' configuration page. The 'Remote root directory' field contains 'C:\Users\Administrator\Desktop\slave'. The 'Labels' field contains 'testie' and is highlighted with a blue box. The 'Usage' dropdown is set to 'Use this node as much as possible'. The 'Launch method' dropdown is open, showing options: 'Launch agent by connecting it to the controller' (selected), 'Launch agent via execution of command on the controller', and 'Launch agents via SSH'. This dropdown is highlighted with an orange box. The 'Custom WorkDir path' field is empty. The 'Internal data directory' field contains 'remoting'. The 'Fail if workspace is missing' checkbox is unchecked.

xiii) Scroll down and click on save



The screenshot shows the bottom part of the Jenkins 'New Node' configuration page. The 'Internal data directory' field contains 'remoting'. The 'Fail if workspace is missing' and 'Use WebSocket' checkboxes are unchecked. The 'Availability' dropdown is set to 'Keep this agent online as much as possible'. The 'Node Properties' section is visible, with checkboxes for 'Disable deferred wipeout on this node', 'Environment variables', and 'Tool Locations'. The 'Save' button is highlighted with a blue box and an orange arrow points to it.

xiv) To activate the node, click on the node1

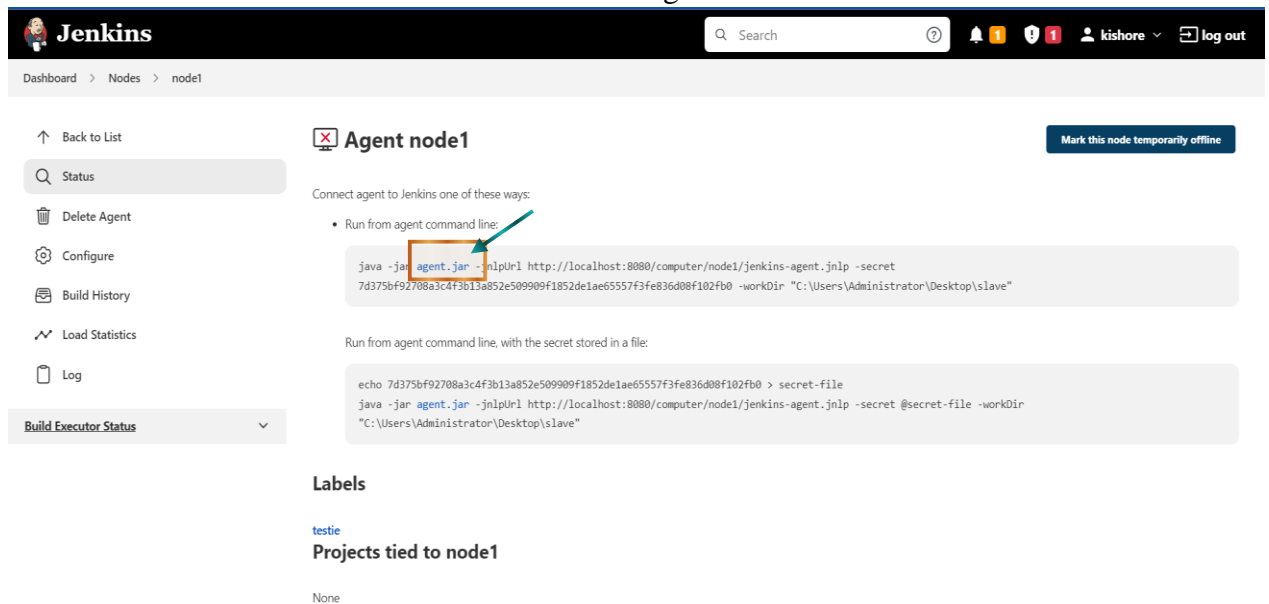


The screenshot shows the Jenkins 'Manage nodes and clouds' page. On the left sidebar, there are links for 'Back to Dashboard', 'Manage Jenkins', 'New Node', 'Configure Clouds', and 'Node Monitoring'. Below these are sections for 'Build Queue' (showing 'No builds in the queue') and 'Build Executor Status' (showing 'Built-In Node' with 1 idle and 2 offline executors, and 'node1' as offline). The main table lists nodes:

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-In Node	Windows Server 2019 (amd64)	In sync	7.47 GB	1.03 GB	7.47 GB	0ms
	node1		N/A	N/A	N/A	N/A	N/A
	Data obtained		47 ms	62 ms	47 ms	15 ms	47 ms

An orange arrow points to the 'node1' row. A 'Refresh status' button is in the top right.

xv) Click on the Link and Jar file will start downloading



The screenshot shows the Jenkins 'Agent node1' page. It includes a 'Status' section with links for 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The 'Build Executor Status' section shows 'node1' as offline. The main content area provides instructions to connect the agent to Jenkins, including a command line and a secret file. The 'Labels' section shows 'testie' and 'Projects tied to node1'.

Connect agent to Jenkins one of these ways:

- Run from agent command line:

```
java -jar agent.jar -jnlpUrl http://localhost:8080/computer/node1/jenkins-agent.jnlp -secret 7d375bf92708a3c4f3b13a852e509909f1852de1ae65557f3fe836d08f102fb0 -workDir "C:\Users\Administrator\Desktop\slave"
```

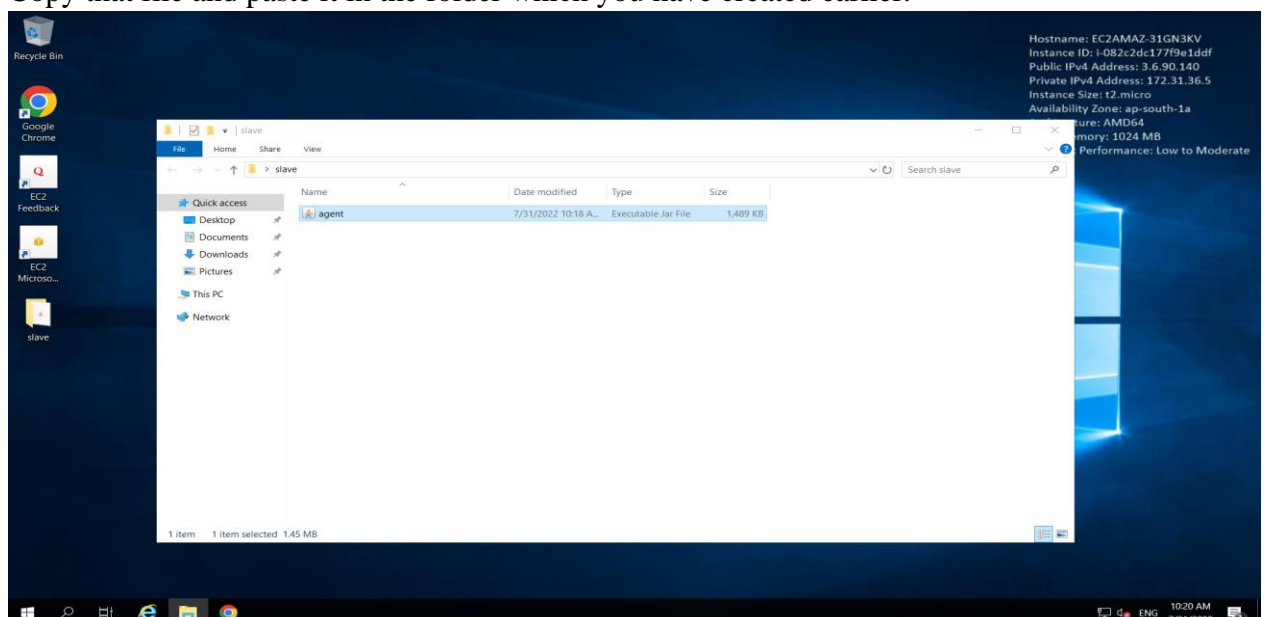
Run from agent command line, with the secret stored in a file:

```
echo 7d375bf92708a3c4f3b13a852e509909f1852de1ae65557f3fe836d08f102fb0 > secret-file
java -jar agent.jar -jnlpUrl http://localhost:8080/computer/node1/jenkins-agent.jnlp -secret @secret-file -workDir "C:\Users\Administrator\Desktop\slave"
```

Labels: testie

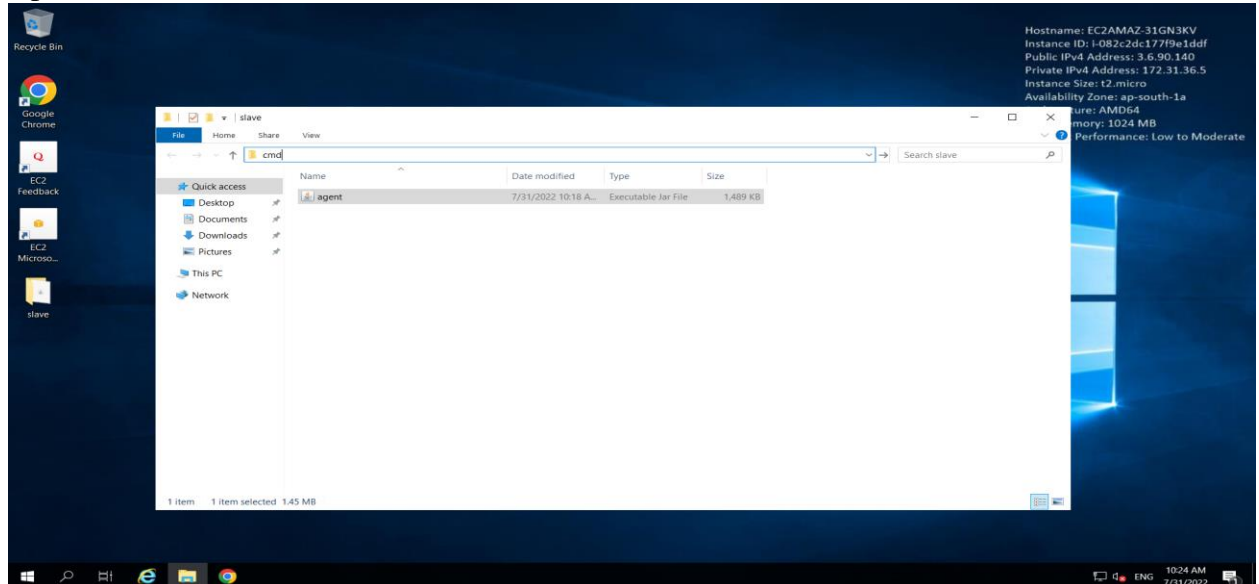
Projects tied to node1: None

xvi) Copy that file and paste it in the folder which you have created earlier.

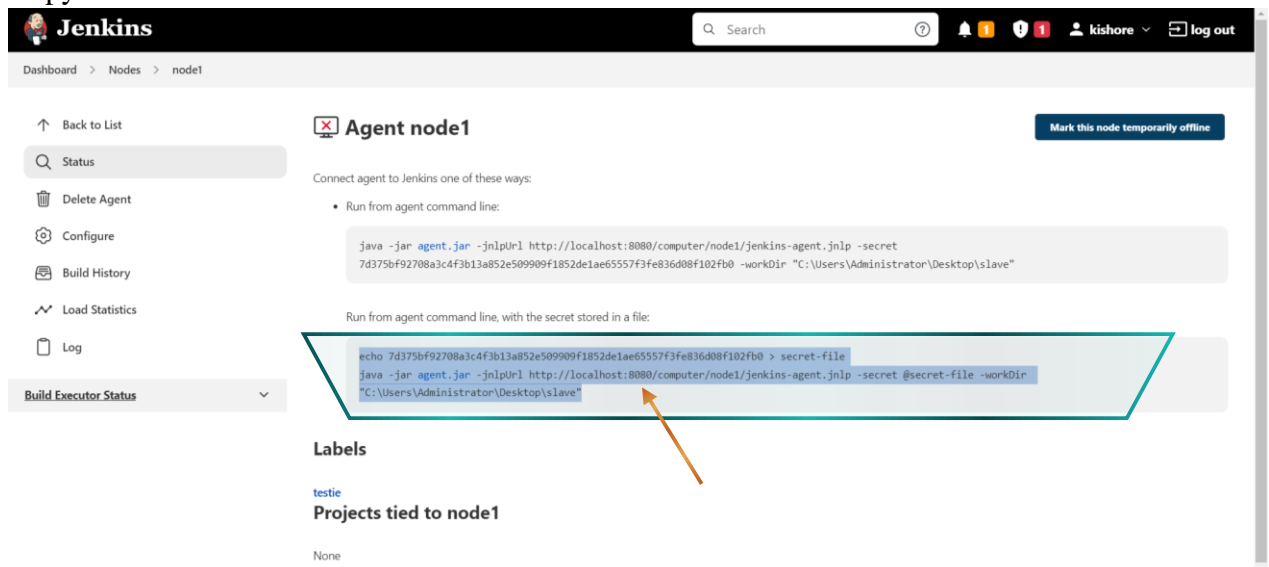


The screenshot shows a Windows File Explorer window open to the 'slave' folder. The 'agent' file (Executable Jar File, 1,489 KB) is selected. The desktop background shows an AWS EC2 instance with details like 'Hostname: EC2AMAZ-31GN3KV', 'Instance ID: I-082c2dc17779e1ddf', and 'Public IPv4 Address: 3.6.90.140'.

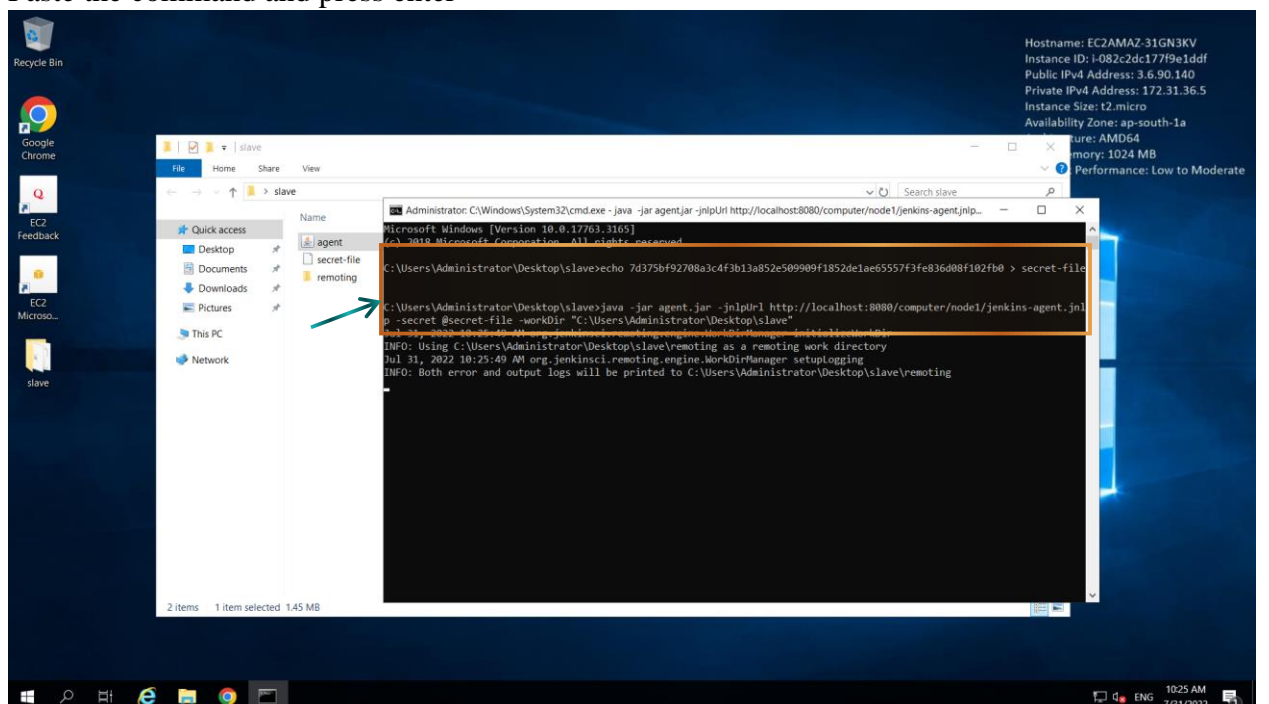
xvii) Open the cmd from the folder

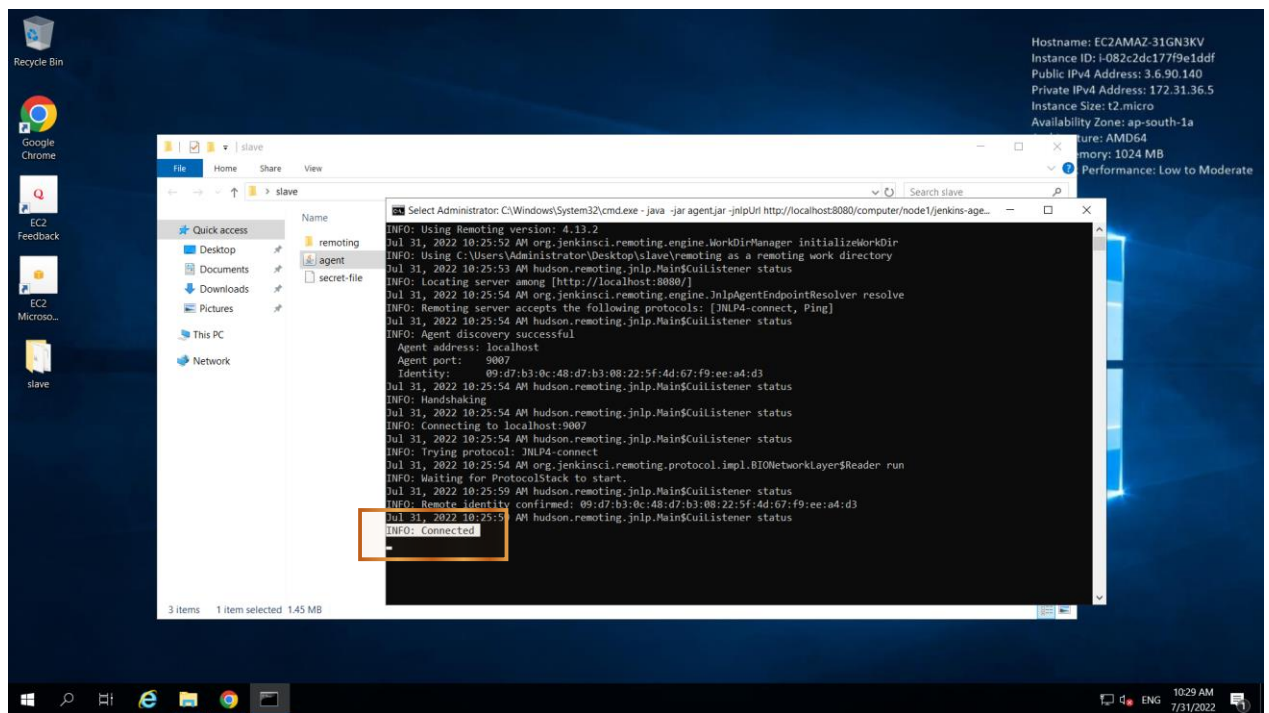


xviii) Copy the command

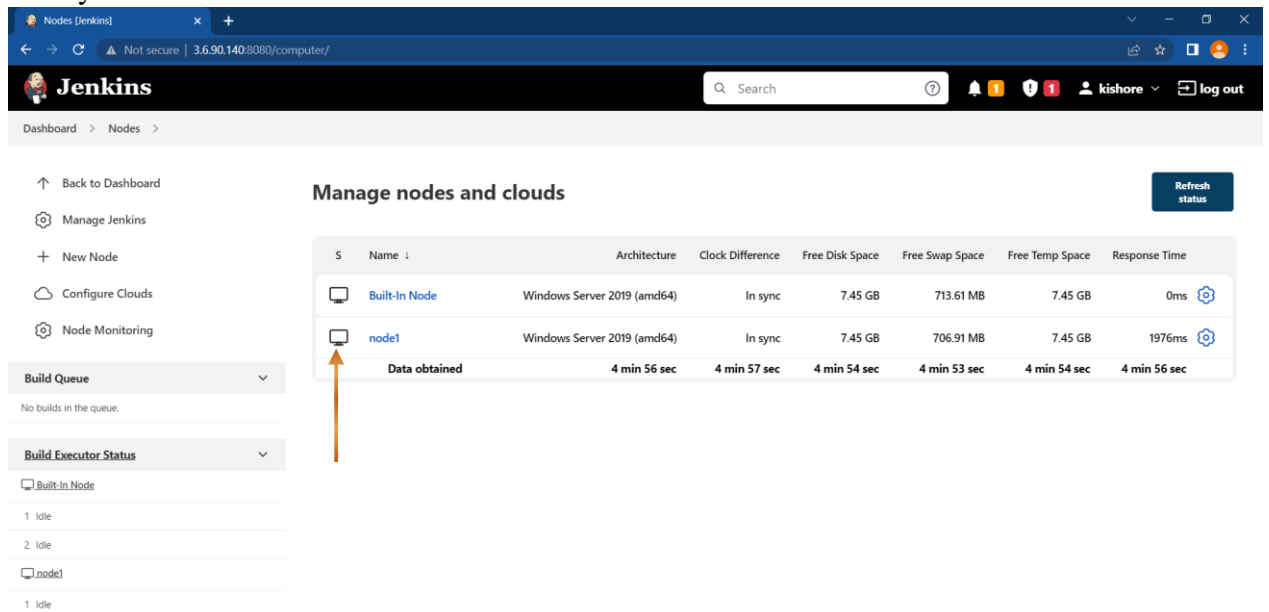


xix) Paste the command and press enter



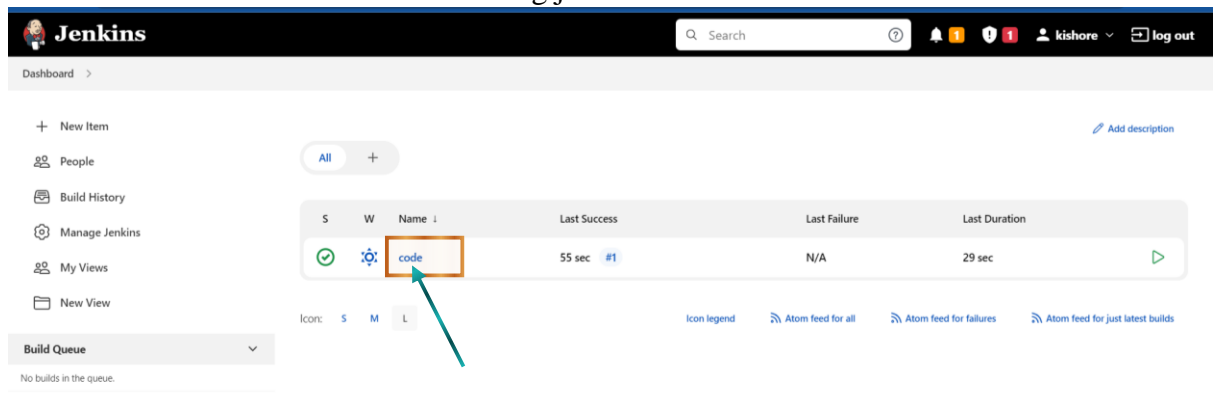


xx) Now you can see the node will be activated

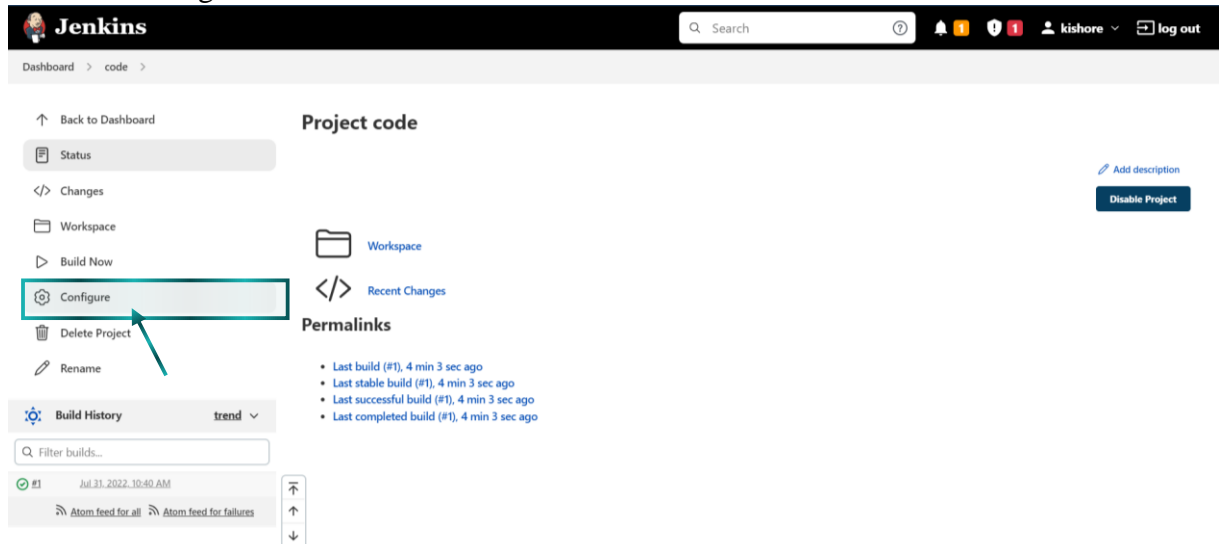


2. Assigning the existing job to the created slave node:

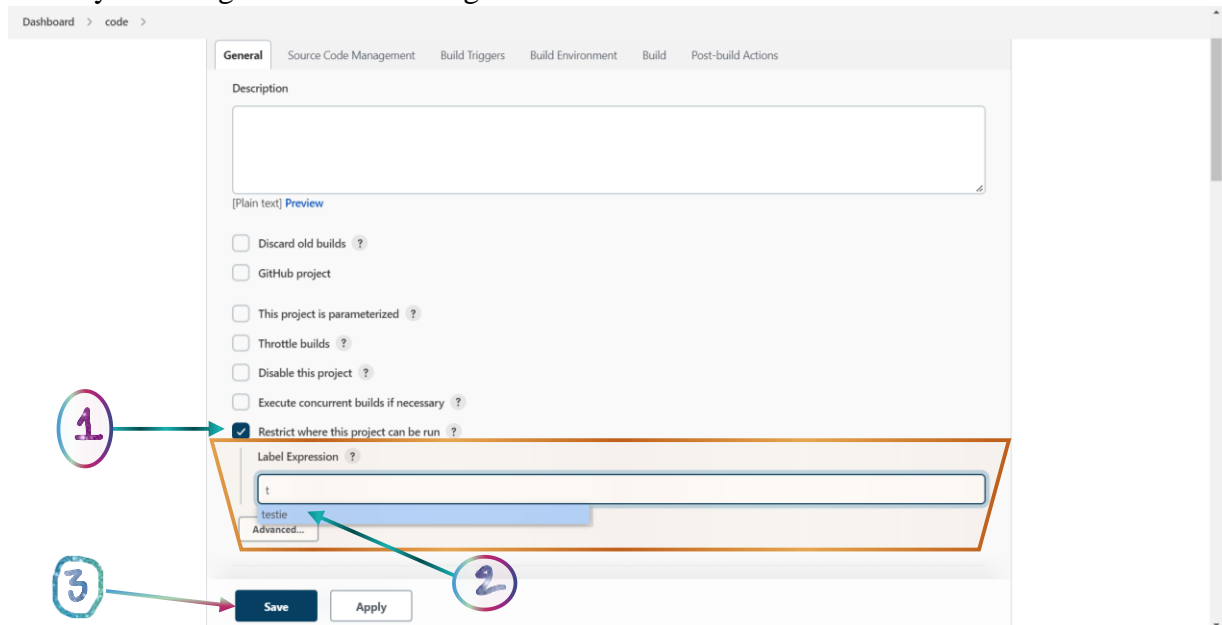
- Go to Dashboard and click on the existing job



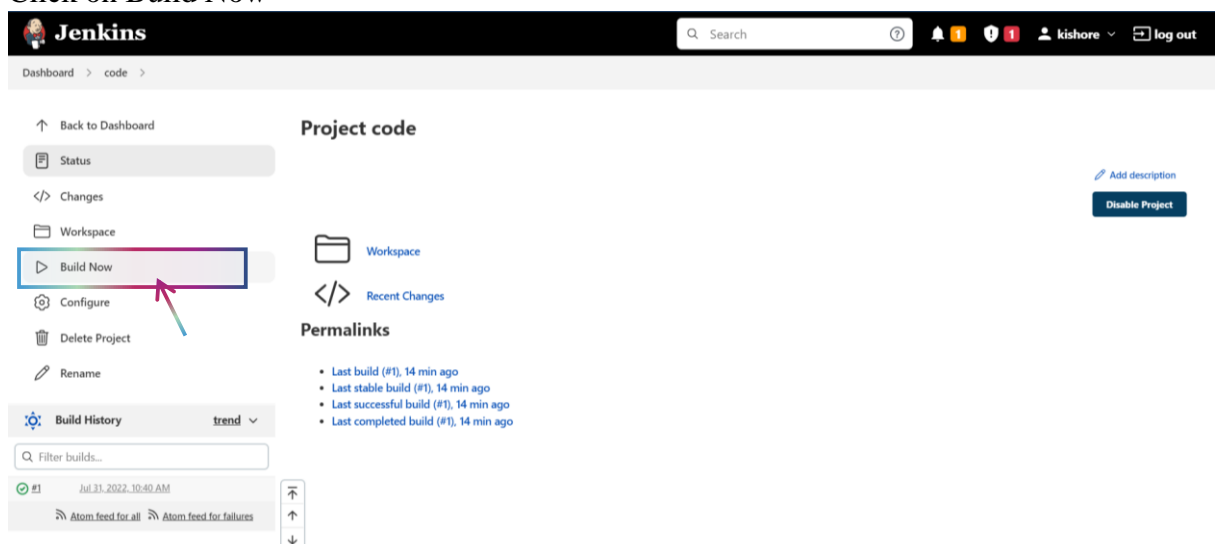
ii. Click on Configure



iii. Check the tickbox against 'Restrict where this project can be run' next give the label name which you have given while creating the slave node and click on Save.



iv. Click on Build Now



v. Click on the link

The screenshot shows the Jenkins 'code' page for a project named 'code'. The left sidebar contains navigation links: Back to Dashboard, Status, Changes, Workspace, Build Now, Configure, Delete Project, and Rename. The main content area is titled 'Project code' and includes links for Workspace, Recent Changes, and Permalinks. The Permalinks section lists four build links: 'Last build (#2), 2 min 2 sec ago', 'Last stable build (#1), 18 min ago', 'Last successful build (#1), 18 min ago', and 'Last completed build (#1), 18 min ago'. A red box highlights the 'Last build (#2)' link, and a red arrow points to it. Below the Permalinks section is a 'Build History' table with columns for build number, time, and status. The table shows two builds: #2 (Jul 31, 2022, 10:57 AM) and #1 (Jul 31, 2022, 10:40 AM). The bottom of the page shows the REST API and Jenkins version 2.346.2.

- Here we can see this is build by the node1 which we are created earlier.

The screenshot shows the Jenkins 'Build #2' page for the 'code' project. The left sidebar contains navigation links: Back to Project, Status, Changes, Console Output, View as plain text, Edit Build Information, Delete build '#2', Git Build Data, and Previous Build. The main content area is titled 'Build #2 (Jul 31, 2022, 10:57:03 AM)' and includes a 'Keep this build forever' button. The build details show 'No changes', 'Started by user kishore', and 'Revision: 0db15700d7a742c6a96390223129d328641544fc'. The repository is 'https://github.com/kishore1133/DevOpsClassCodes.git'. A red box highlights the 'Keep this build forever' button, and a red arrow points to it. The bottom of the page shows the REST API and Jenkins version 2.346.2.