Practice for Lesson 3: Assign Roles to User and Build Master-slave configuration in Jenkins Instance

Practices for Lesson 3

Overview

In these practices, you will assign Role based authorization plugins to users created in Jenkins instance. Further, you will learn how to build a master-slave configuration on a Jenkins instances and followed by Backing up the configuration files of Jenkins in the AWS EC2 instance.

Practice 3-2: Distributed Build Master-Slave Configuration

Overview

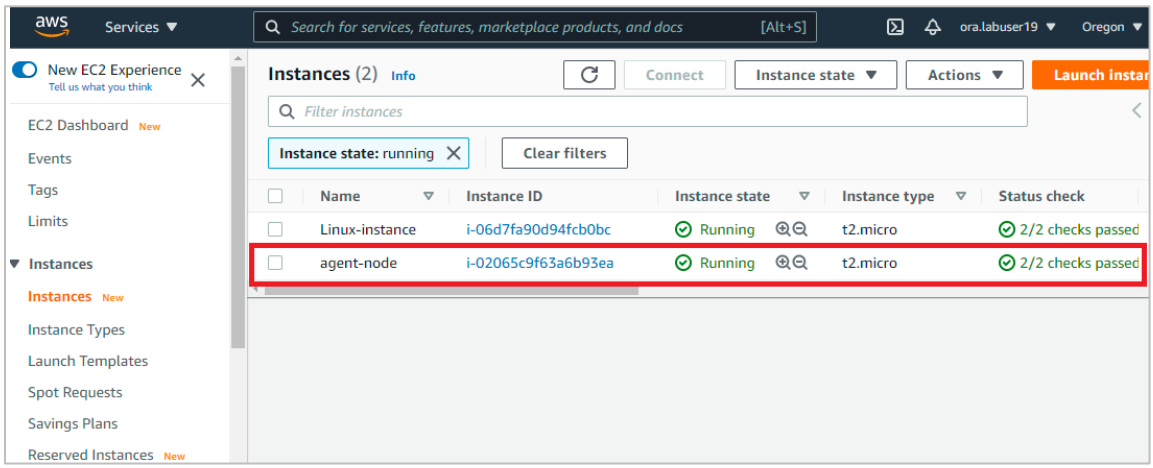
In this practice, you will learn how to build a master-slave configuration on a Jenkins instances and build a Job on Jenkins AWS agent instance.

Assumptions

You should have completed the Practice of Lesson 3-1.

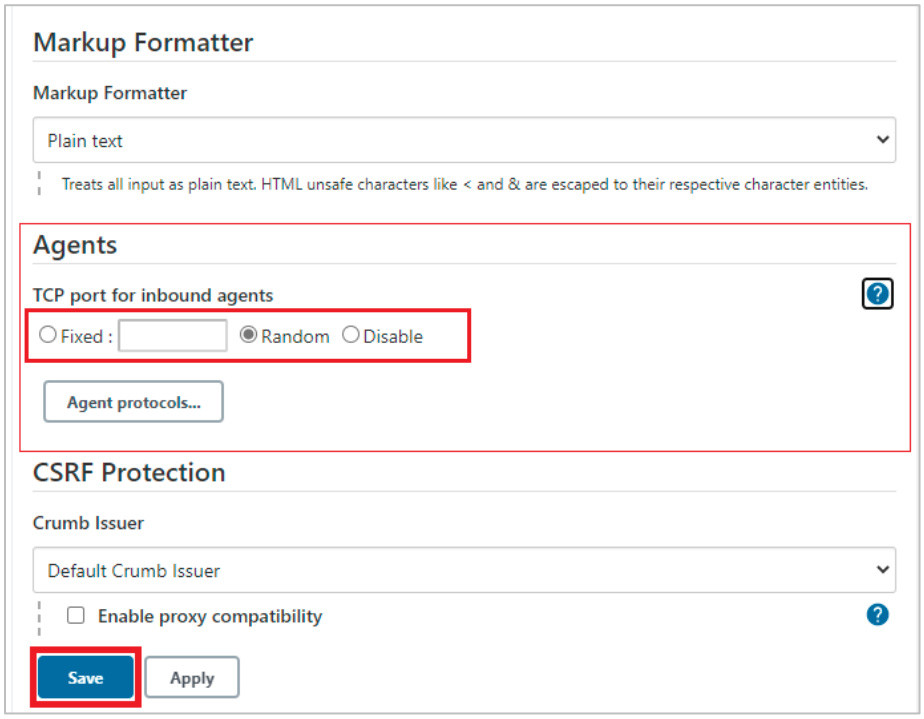
Tasks

1. Create a new instance named agent-node in the AWS EC2 instance.
   a. Follow the steps in practice 1-1 to create the instance in AWS EC2 console.
   b. As shown below, create the agent-node for the master-slave configuration on Jenkins.

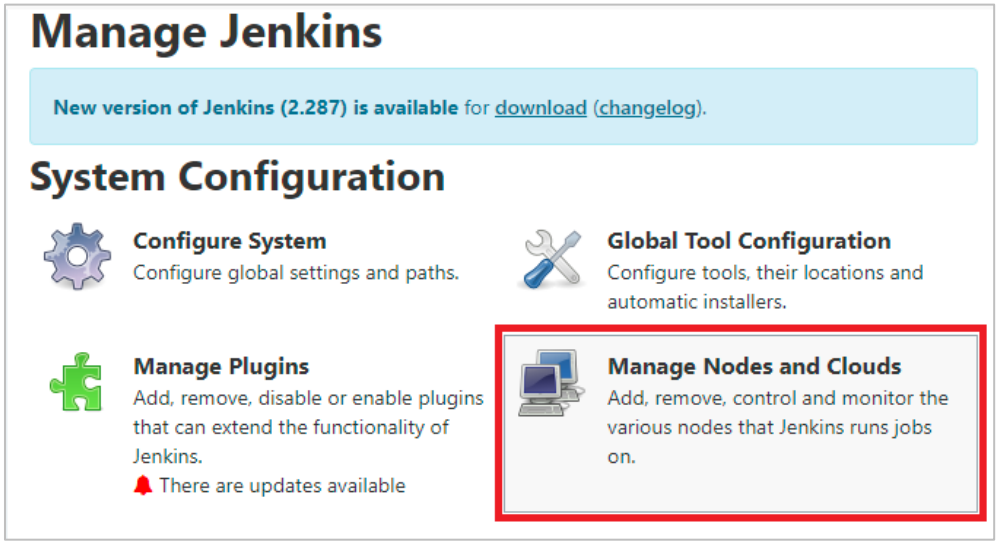


2. Enable Java Network Launch Protocol (JNLP) port in Jenkins instance.
   a. Navigate to **Manage Jenkins** and select **Configure Global Security** under **Security** as shown below.

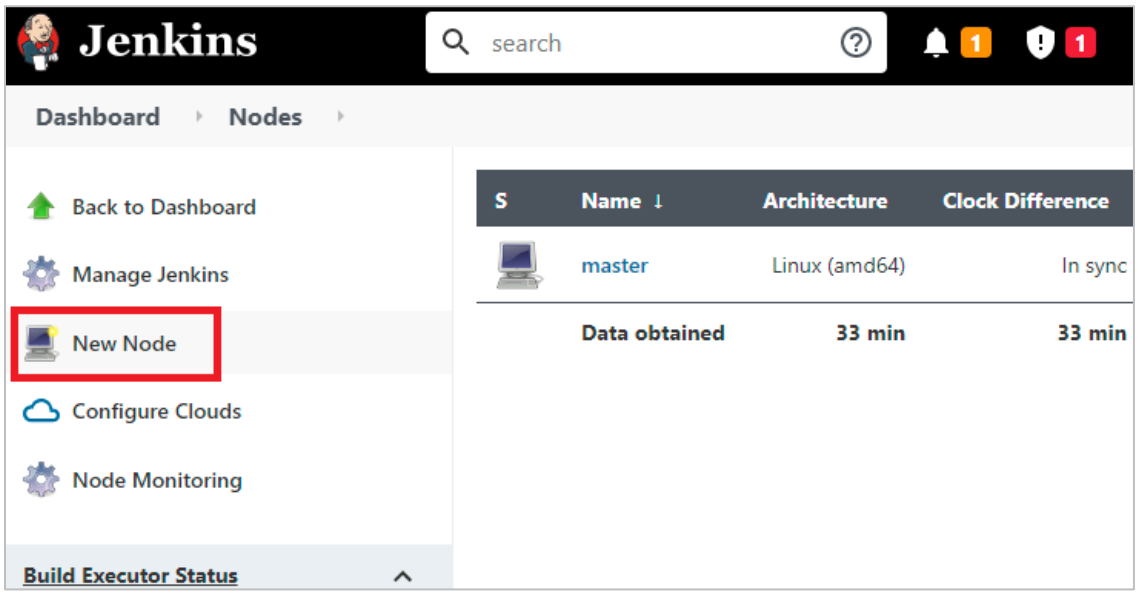
b. Under **Agents,** select **Random** to open the ports of JNLP and click **Save** as shown below.



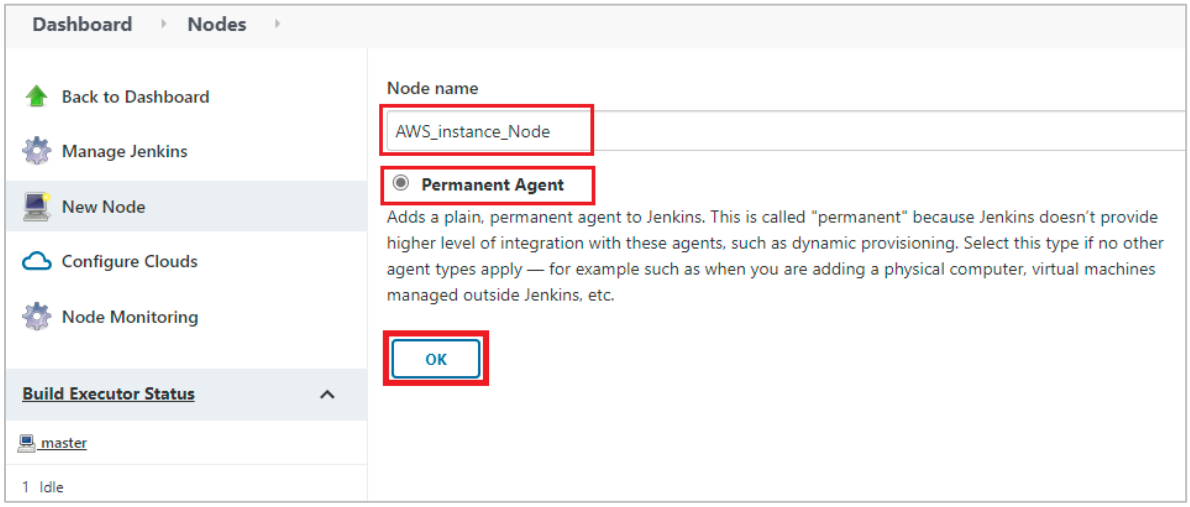
3. Create an agent node (Slave node) in the Jenkins instance.
   a. Navigate to **Manage Jenkins** and select **Manage Nodes and Clouds** under **Security Configuration** as shown below.



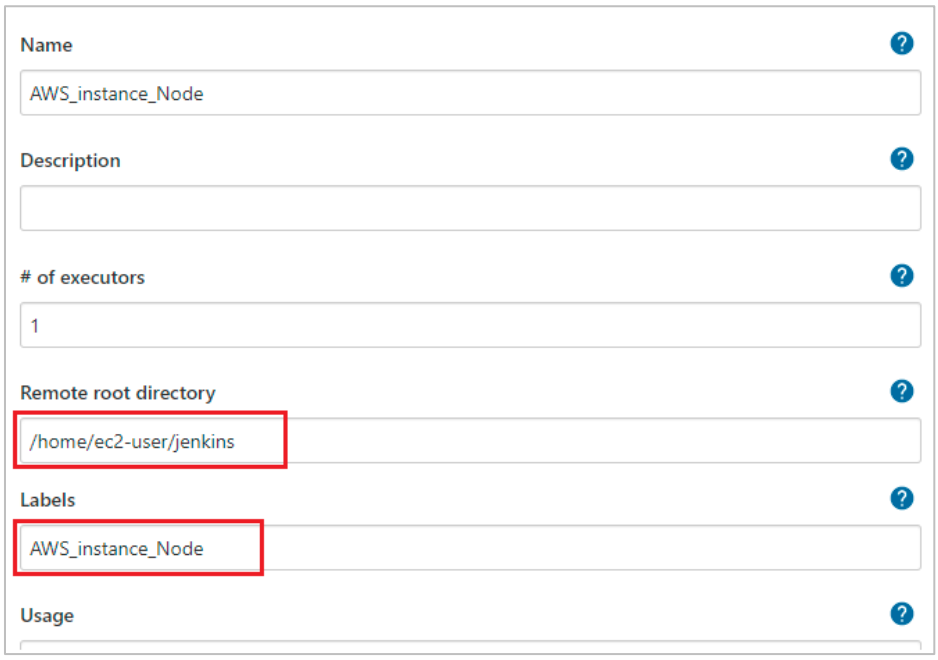
b. Verify that the Jenkins instances consists of only master node. Select **New Node** to add a new node to the Jenkins instance.



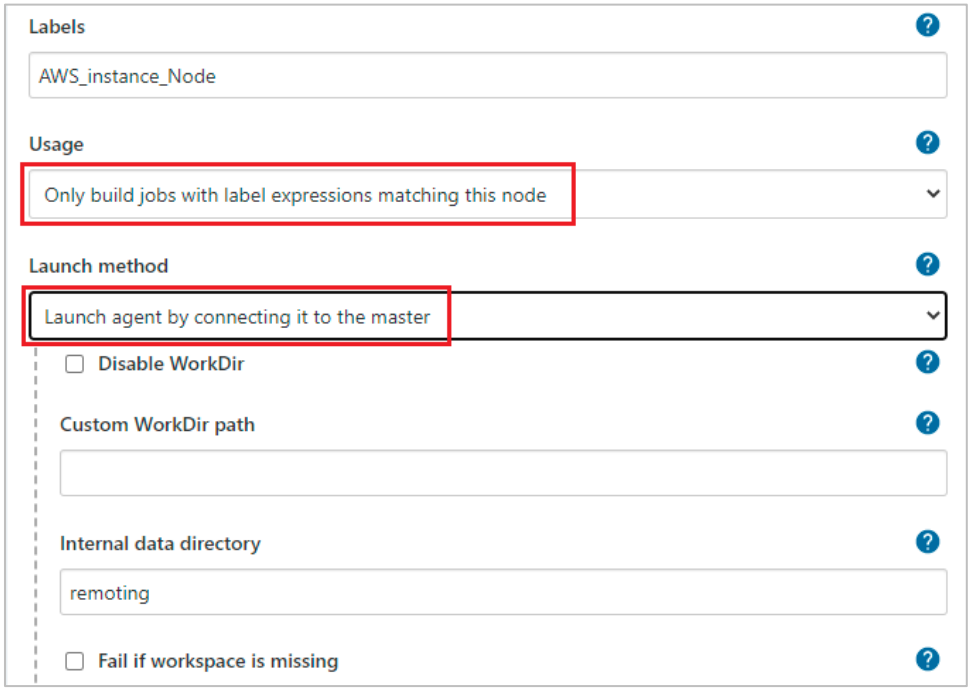
c. Provide the name for the node, select **Permanent Agent** and click **OK** as shown below.



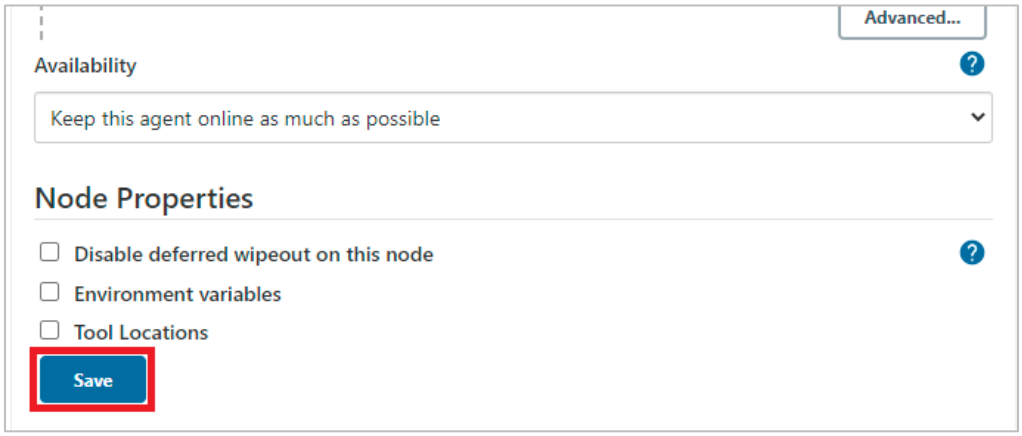
d. Provide a path in **Remote root directory** and **AWS_instance_Node** as label in **Labels** section as shown below.



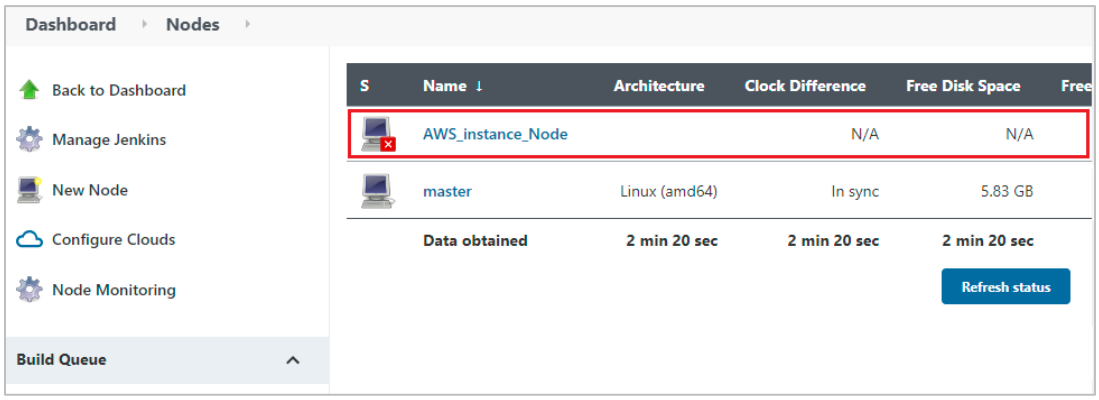
e. Scroll down to **Usage** and select **Only build jobs with label expressions matching this node**, and for **Launch method** select **Launch agent by connecting it to the master** as shown below.



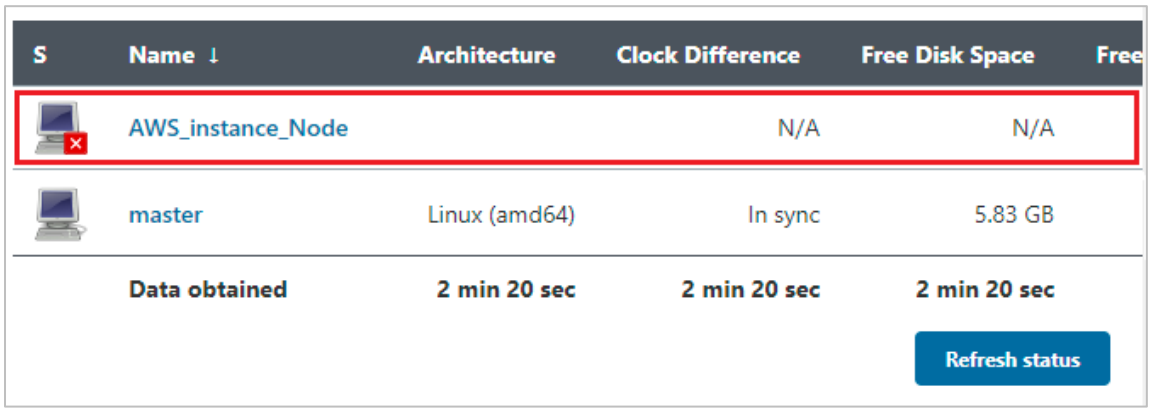
f. Scroll down to end of the page and click **Save.**



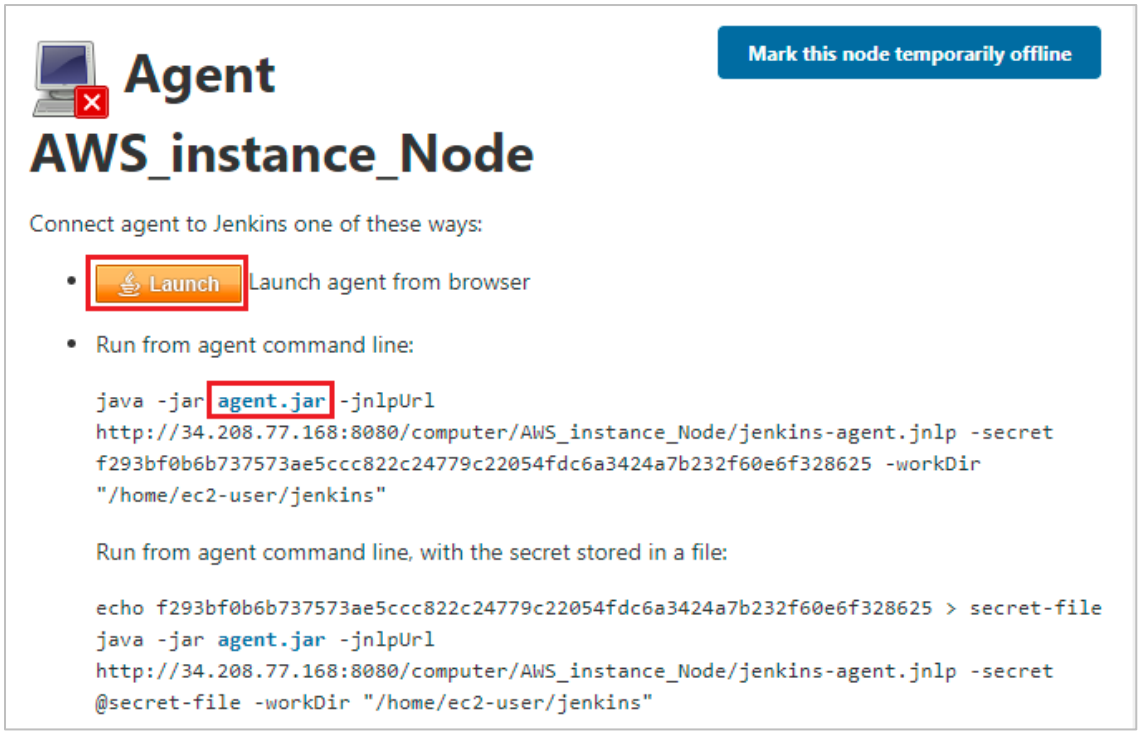
g. Verify the agent node is created as shown below with the red cross mark **(x)**, it indicates that the agent node is not connected to the master node.



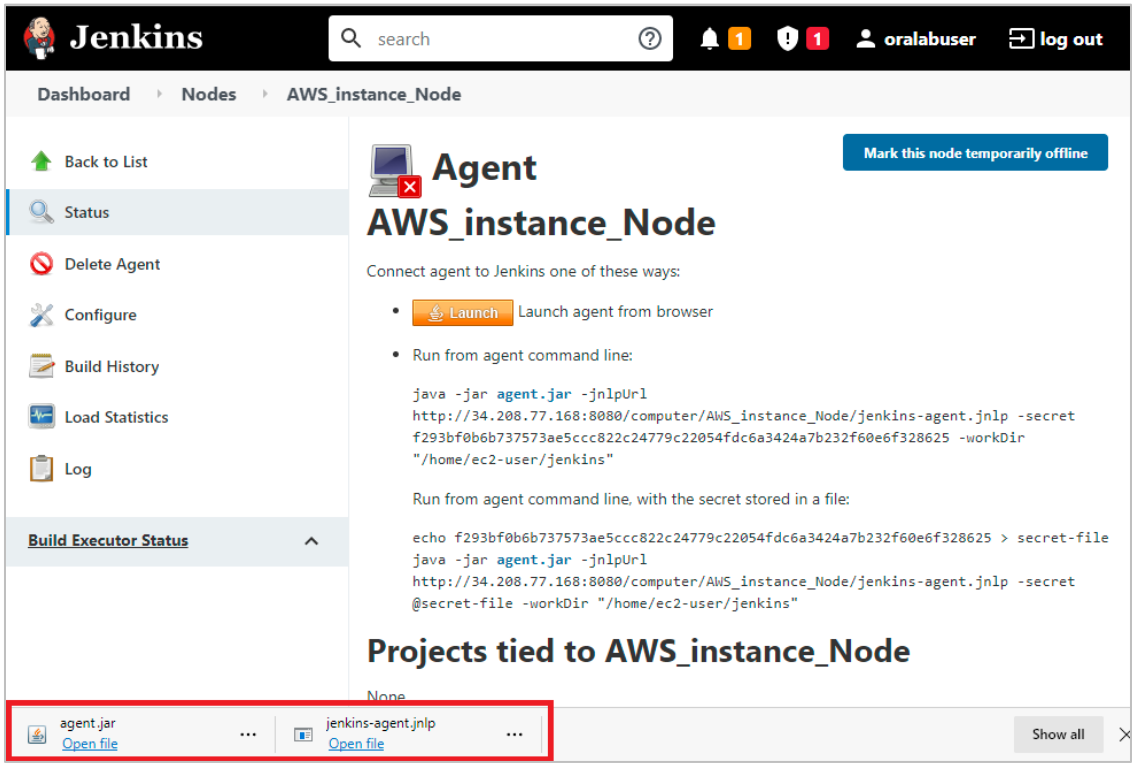
4. Provision the Agent node created in the Jenkins instance.
   a. Click on the new agent node created as shown below.



   b. To connect to the agent node, download the **agent.jar** file and **slave-agent.jnlp** file.

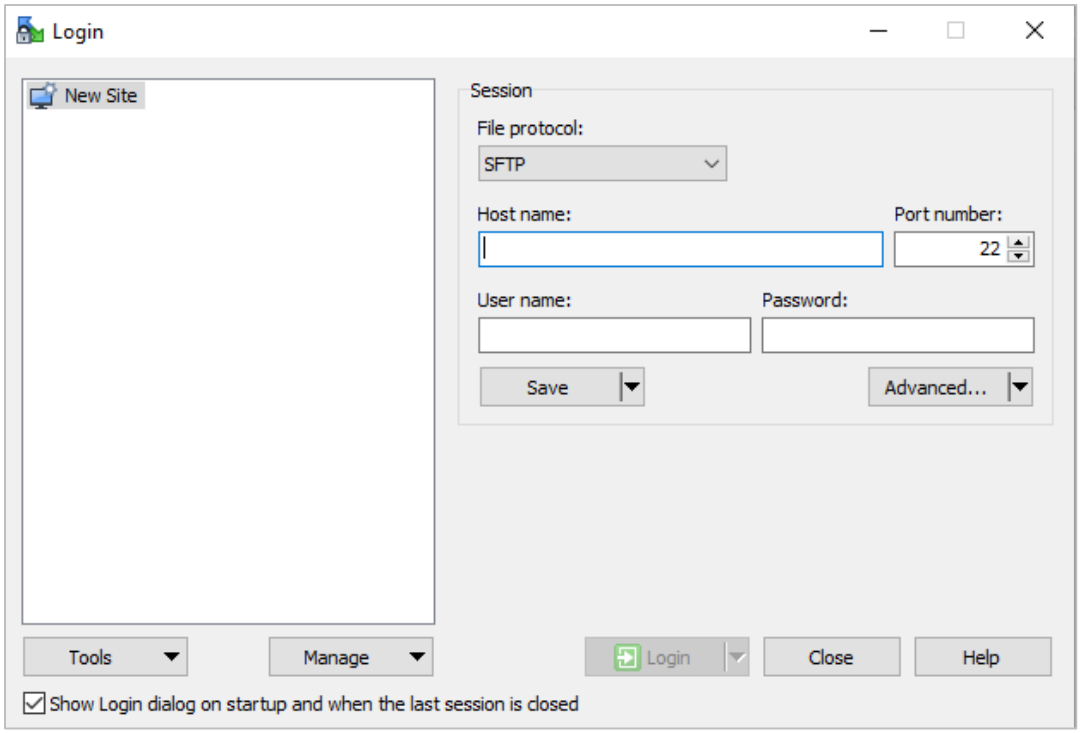


   c. The files are downloaded successfully to the local system. These files have to be placed in the AWS agent-node instance.

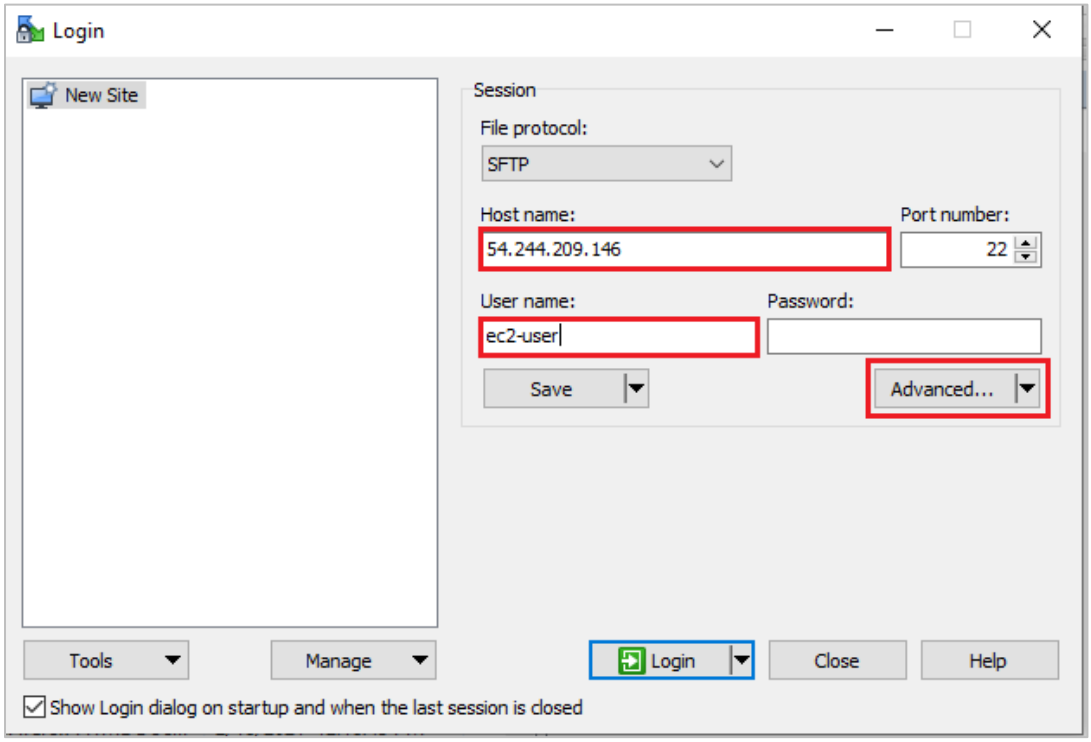


d. Open **WinSCP** to transfer the files to the **agent-node** instance.

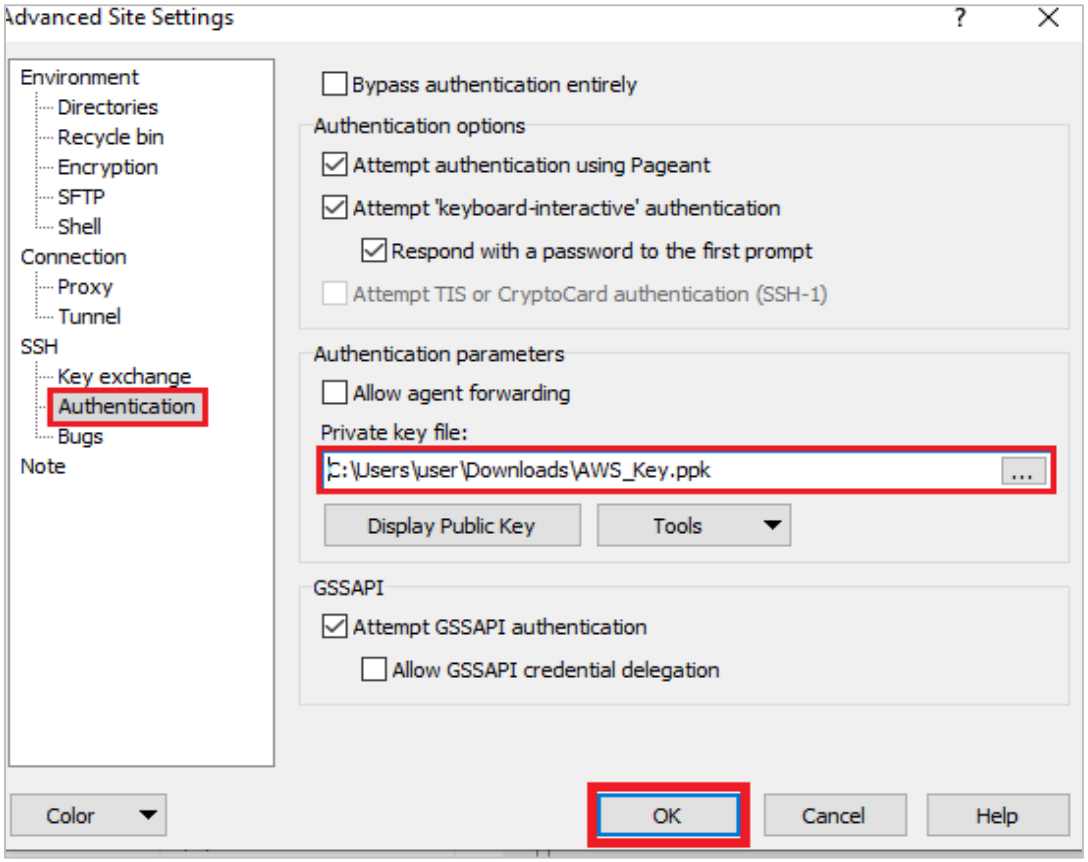
Note: Download and install WinSCP if it is not installed in your local system. It is an open source FTP tool.



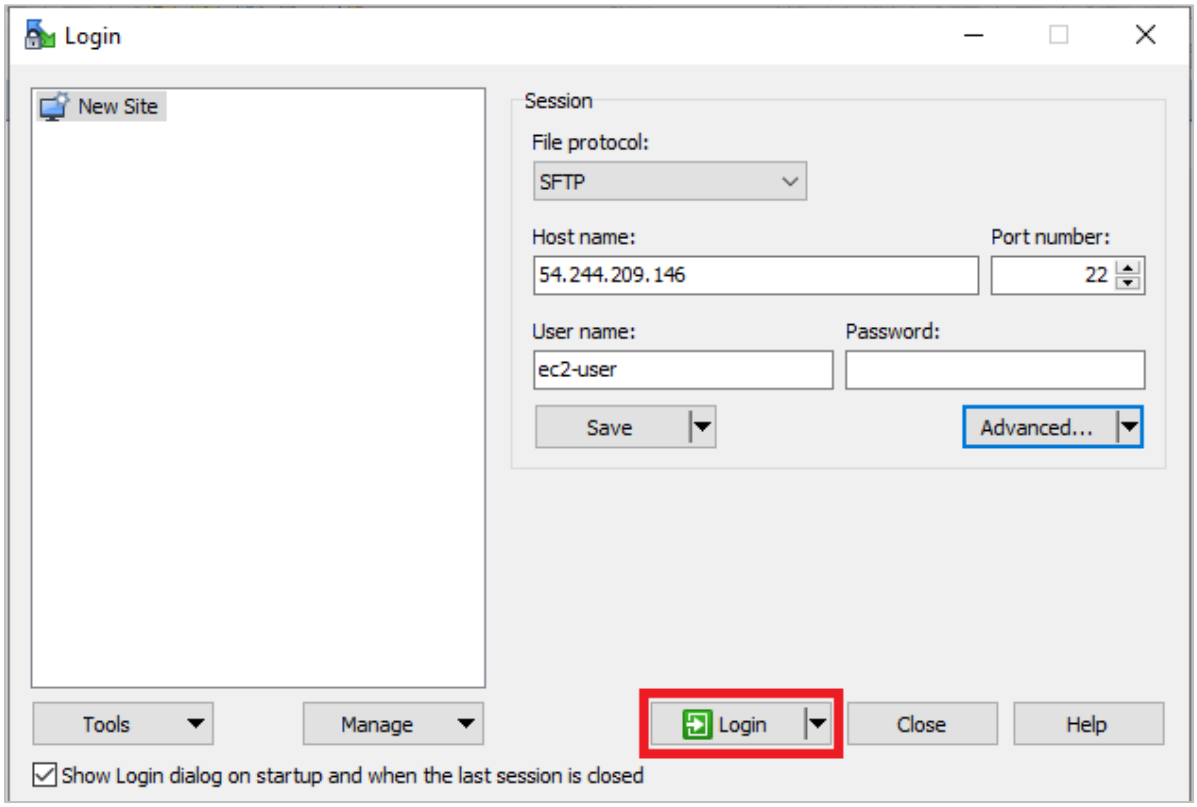
e. Provide the **Public IP address** of the **agent-node** instance for the **Host name**, **ec2-user** for **User name** and click **Advance** to provide the **Password** as shown below.



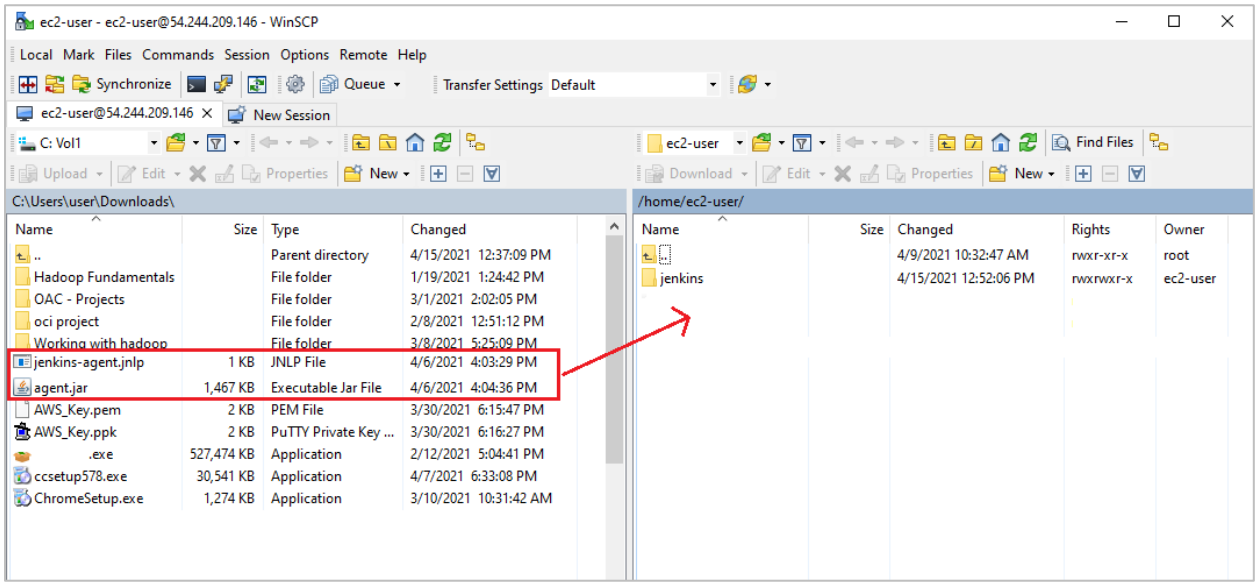
f. In **Advanced Site Settings** navigate to **SSH** and select **Authentication**. Browse the path for the **Private key file** for the instance connection and click **OK** as shown below.



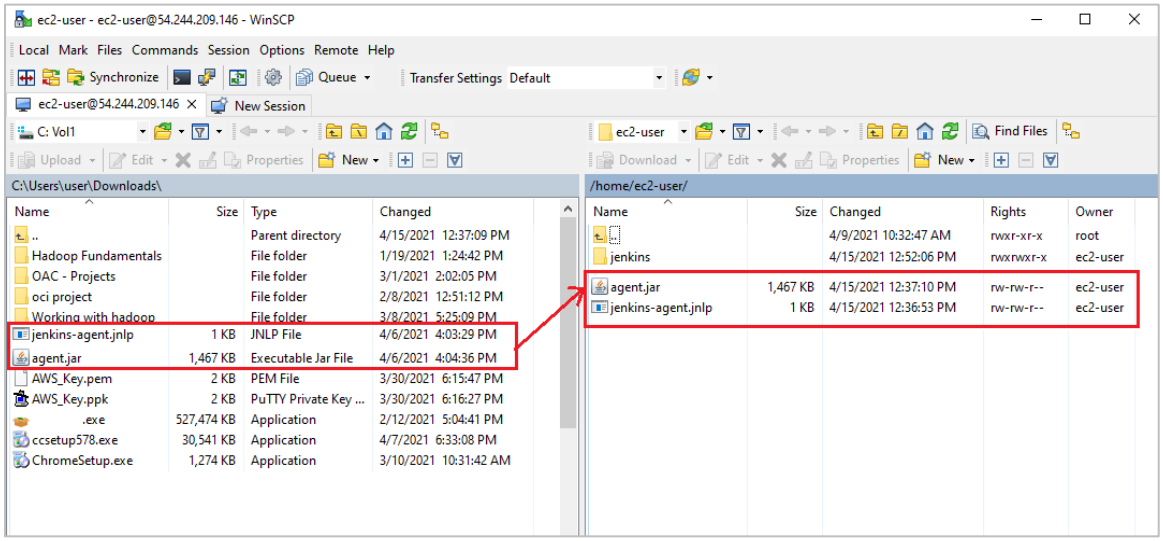
g. Click **Login** to connect to the **agent-node** instance to transfer the files.



h. As shown below, select the files from left box, drag and drop it to the right box.



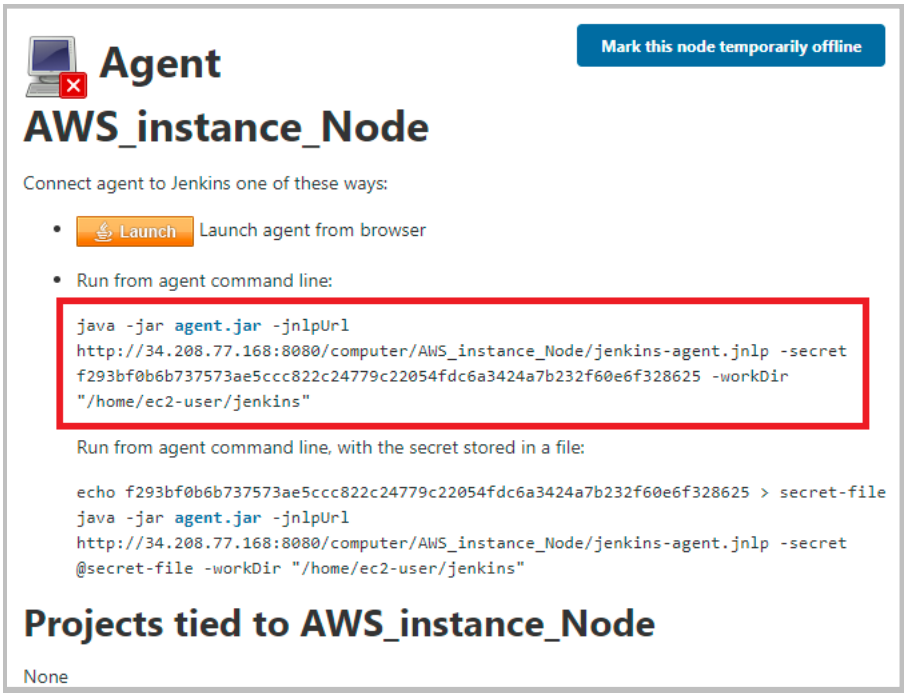
i. The files have been transferred to the connected **agent-node** AWS instance as shown below.



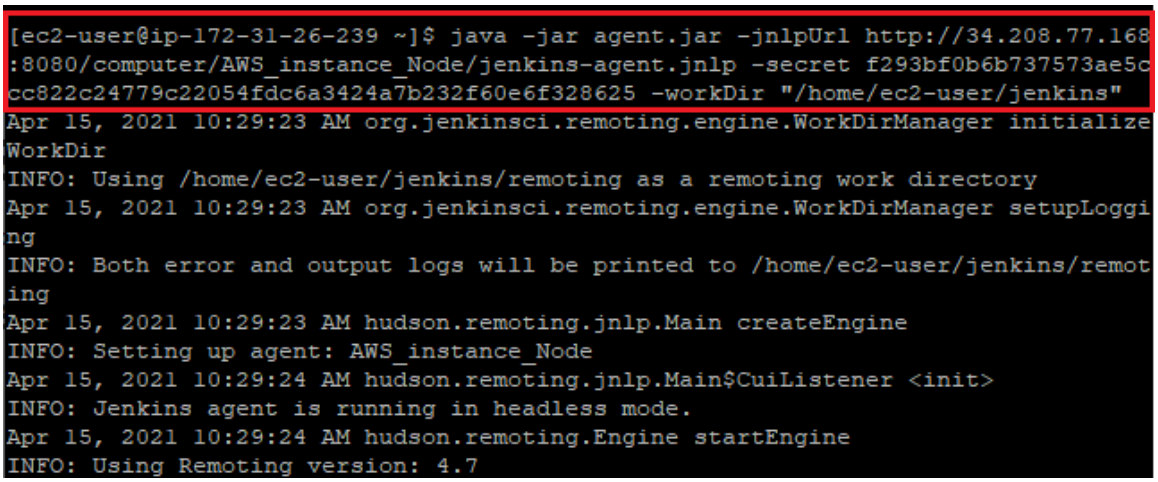
j. Connect to the **agent-node** instance from Putty and list the files in it as shown below.

5. Launch agent.jar in the agent-node instance to establish the connection to the master node.

   a. In Jenkins, click on the agent node and copy the code as shown below.

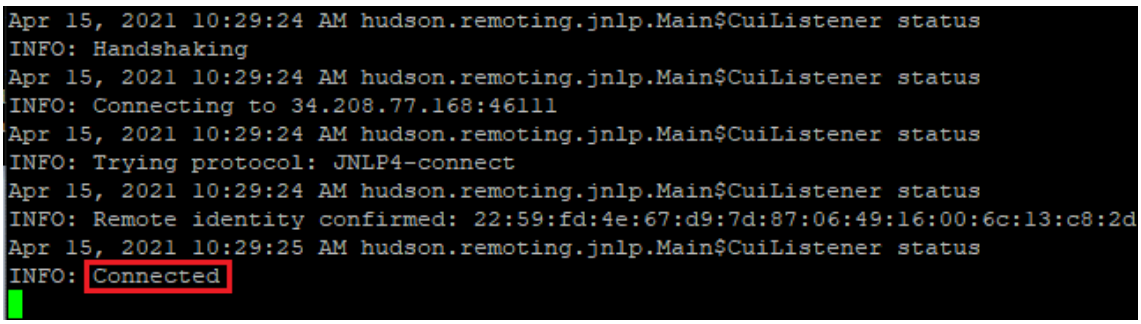


b. Open Putty terminal connected to the **agent-node** and paste the command to execute as shown below.

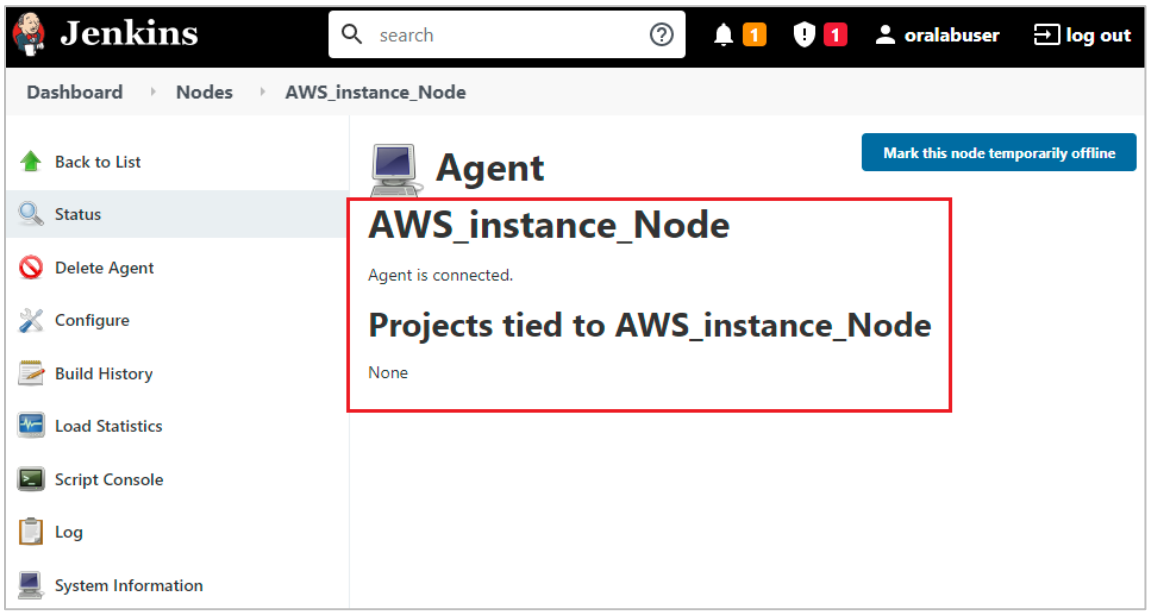


c. On successful execution, the connection will be established by displaying **Connected** as shown below.

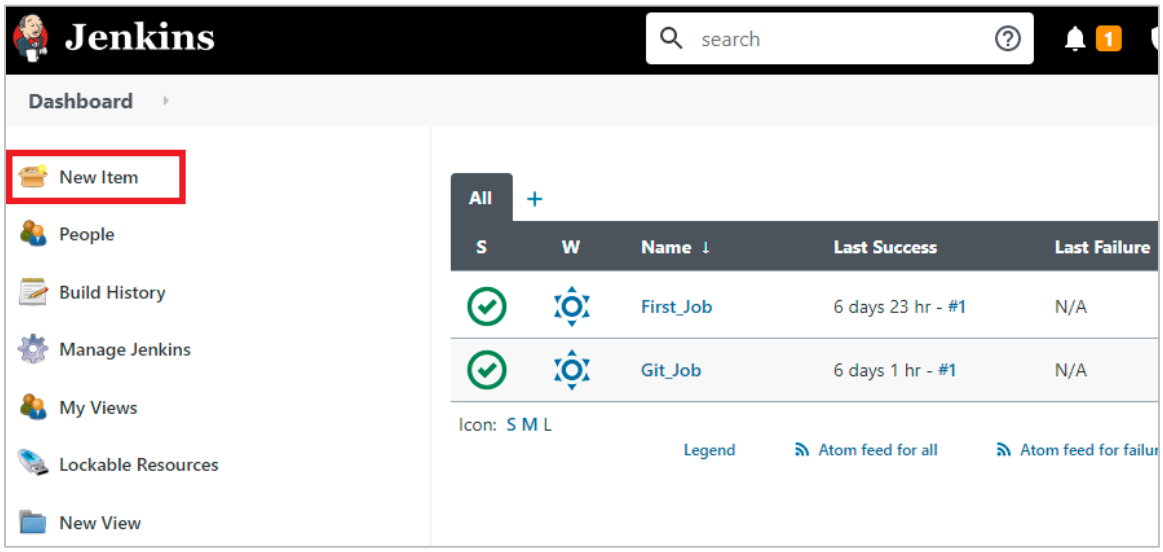
Note: As long as this command prompt is up and running, the agent would be connected.



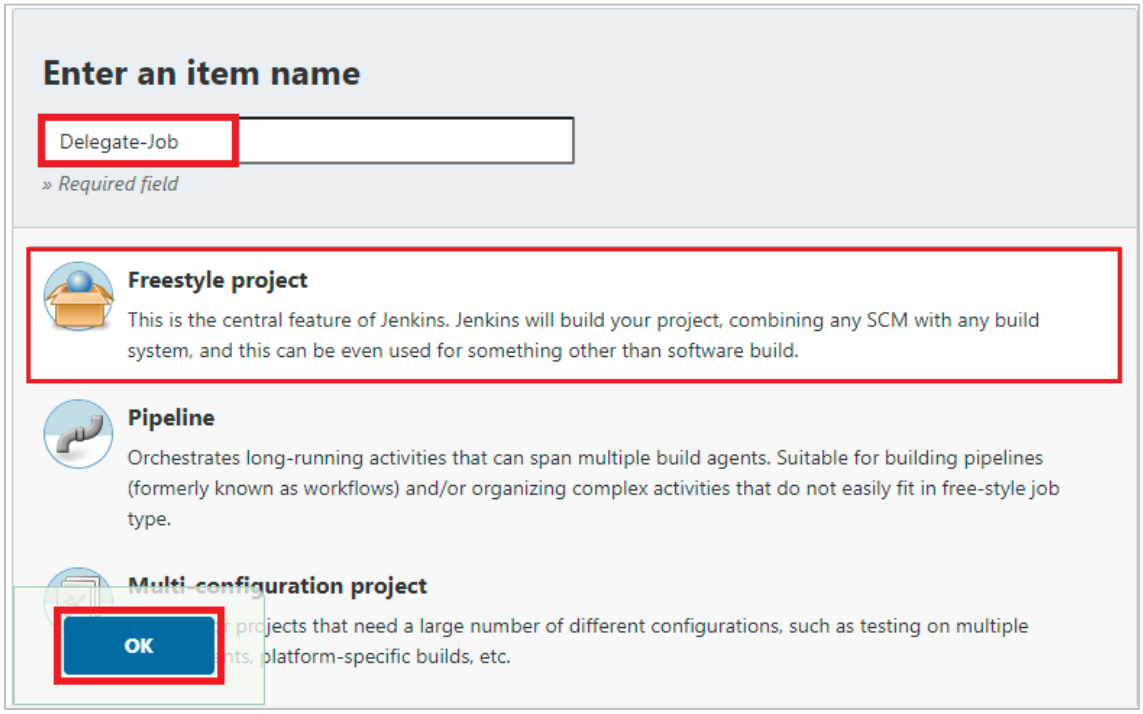
d. Open Jenkins dashboard and refresh the agent node page to verify the connection as shown below.



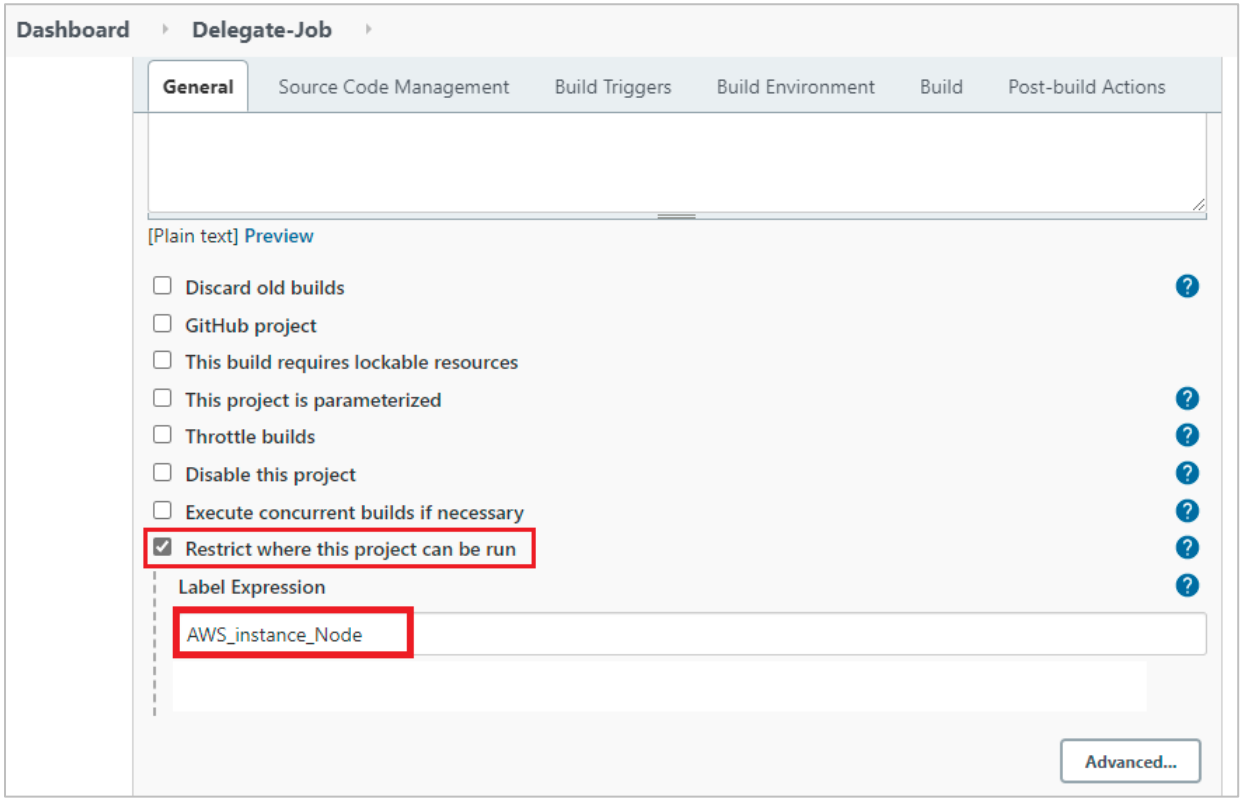
6. Create a new job to be executed on the agent node in Jenkins.
   a. In Jenkins Dashboard, navigate to main menu and select **New item** as shown below.



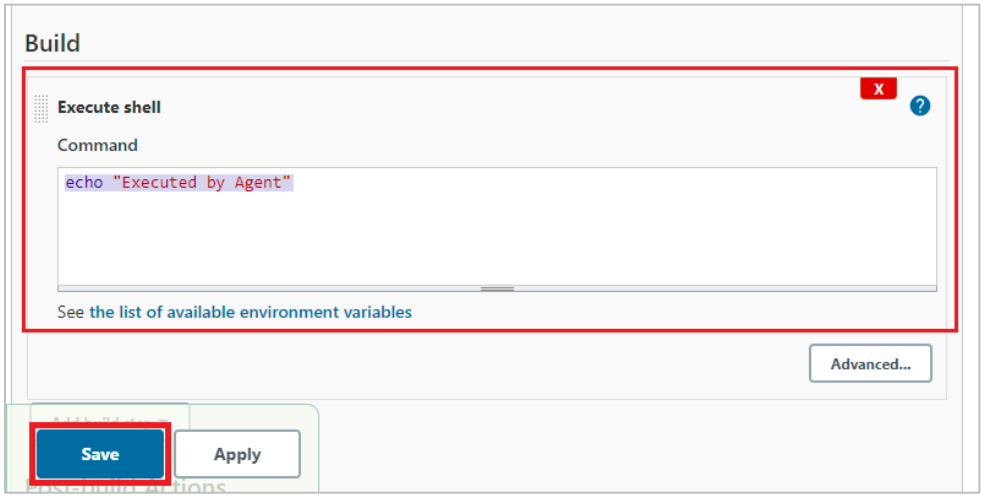
   b. Enter the name for the job, select **Freestyle project** and click **OK**.



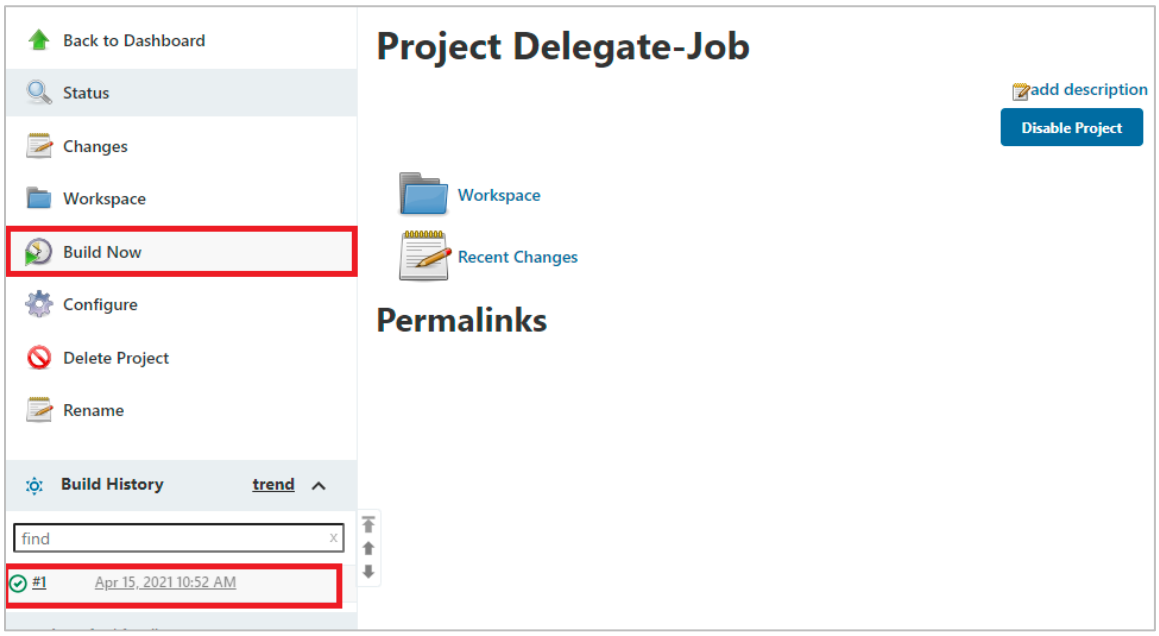
c. Select the check box for **Restrict where this project can be run** and provide the label of the agent node as shown below.



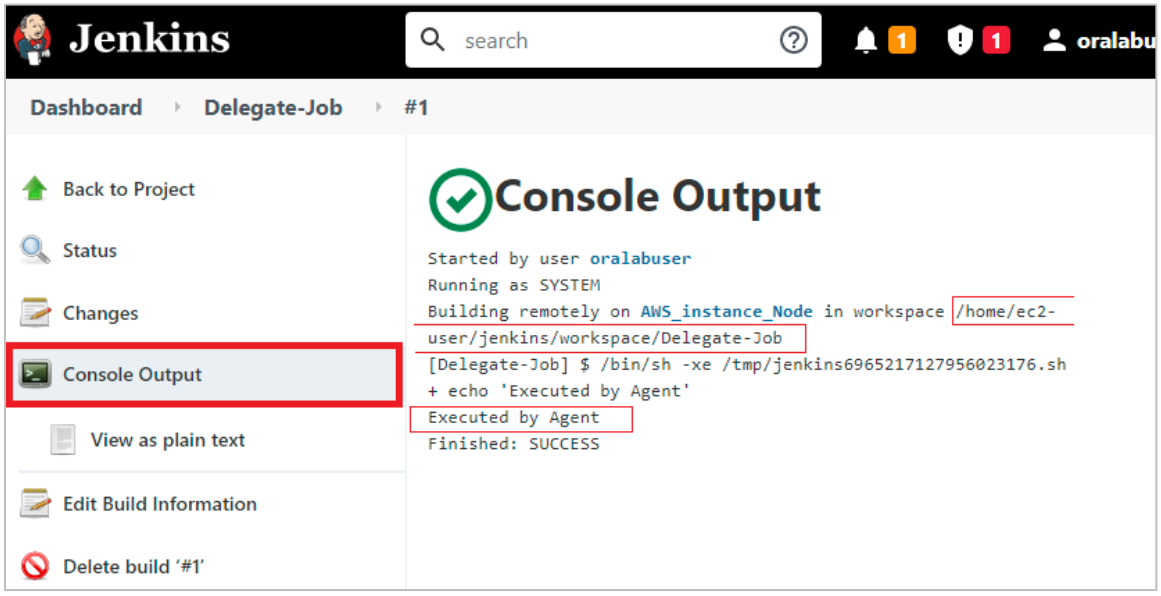
d. Scroll down to **Build** and select **Execute shell**. Type the **echo** command as shown below in the command prompt and click **Save**.



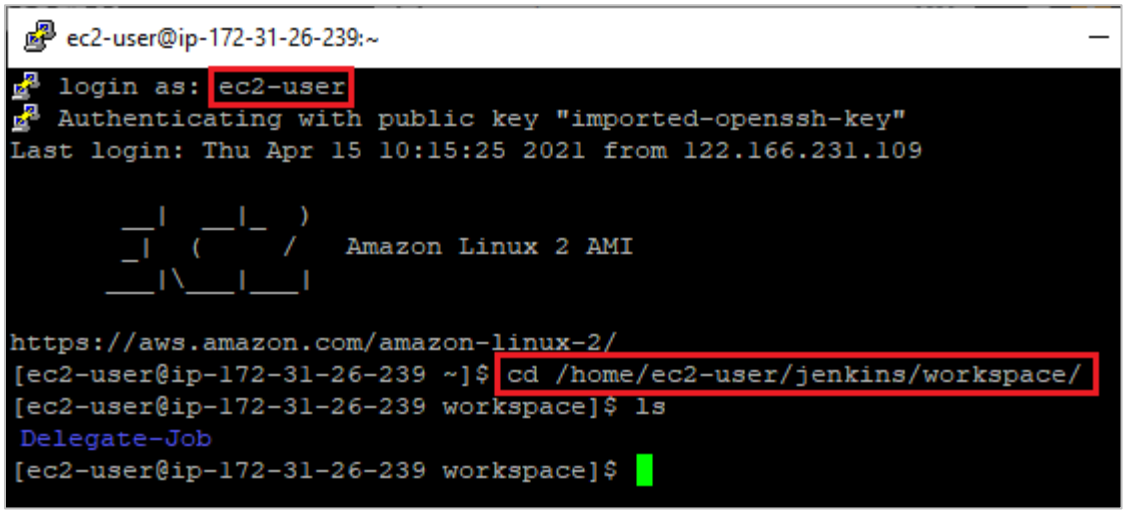
e. Click **Build Now** to build the job in Jenkins and click on the build job link provided in the bottom to verify the execution as shown below.



f. Select Console Output to view the execution. Note the path of the workspace highlighted.



g. Open Putty new terminal and navigate to the workspace path to view the Job name in the **agent-node** instance.



7. Keep the Jenkins Dashboard and the AWS Management Console open for the next practice.