**Step1:Create three EC2 instances and configure them accordingly**

In one of the EC2 machines install Jenkins and use it as the master node

The rest of the two machines are acting as slave nodes.

A screenshot of a computer

Description automatically generated

Set the inbound rule in the security group for the instances as below to allow all kind of traffic .

A screenshot of a computer

Description automatically generated

As you can see Jenkins is installed and up and running in the master node.

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Use the below commands to install Java in the slave nodes

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Step 2: Adding two slave nodes in Jenkins**

-Navigate to ManageJenkins--> Security--> Agent

-Enable TCP port for inbound agents

- Select Fixed, and give port value as **50000**

A screenshot of a computer

Description automatically generated

The name of the node is **Javaslave** (EC2 machine slave1)

The **launch** **method** is **launch agent by connecting it to the controller**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

For connecting the node to the Jenkins master we need to run the below script in the slave1 EC2 machine

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

You can see the node is connected now.

A screenshot of a computer

Description automatically generated

The name of the node is **sshslave** (EC2 machine slave2)

The **launch** **method** is **launch agent via ssh**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Step 3: Create a Maven app and save it Git repository**

Create a maven app using start.spring.io

A screenshot of a computer

Description automatically generated

Create a Git repository named SpringbootMaven-Jenkins-Demo-varthini

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Step 4: Building the app in respective nodes using the agent label name specified in the Jenkinsfile pipeline script.**

Creating a pipelinedemo project in Jenkins for building the maven app and testing the maven app. As a **pre**-**requisite** we need to install the **Maven** plugin in **Jenkins**.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The label used in the pipeline script is **java** so the Javaslave node is selected for building the application.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

In the same project/new project, we can modify the label as **ssh** in the pipeline script.

The label used in the pipeline script is ssh so the sshslave node is selected for building the application.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**The label** used in the pipeline script is ssh so the **sshslave** node is selected for building the application.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated