Jenkins Agent

The Jenkins architecture is designed for distributed build environments. It allows us to use different environments for each build project balancing the workload among multiple agents running jobs in parallel.

The Jenkins controller is the original node in the Jenkins installation. The Jenkins controller administers the Jenkins agents and orchestrates their work, including scheduling jobs on agents and monitoring agents. Agents may be connected to the Jenkins controller using either local or cloud computers.

The agents require a Java installation and a network connection to the Jenkins controller.

Simply put. Any machine that can run java can be an agent for the Jenkins controller. You must make sure the java is installed. This can be a bare metal server or a vm.

The reason behind using Jenkins's agent is security and performance purposes.

Create a new amazon linux instance

Install java in it

Run below commands to install Java 11 on Amazon Linux:

$ sudo amazon-linux-extras install java-openjdk11

Run below commands to install Java 8 on Amazon Linux:

$ sudo yum install java-1.8.0-openjdk

Step 2 – Check Active Java Version

After successfully installing Java on Amazon Linux using the above steps, Let's verify the installed version using the following command.
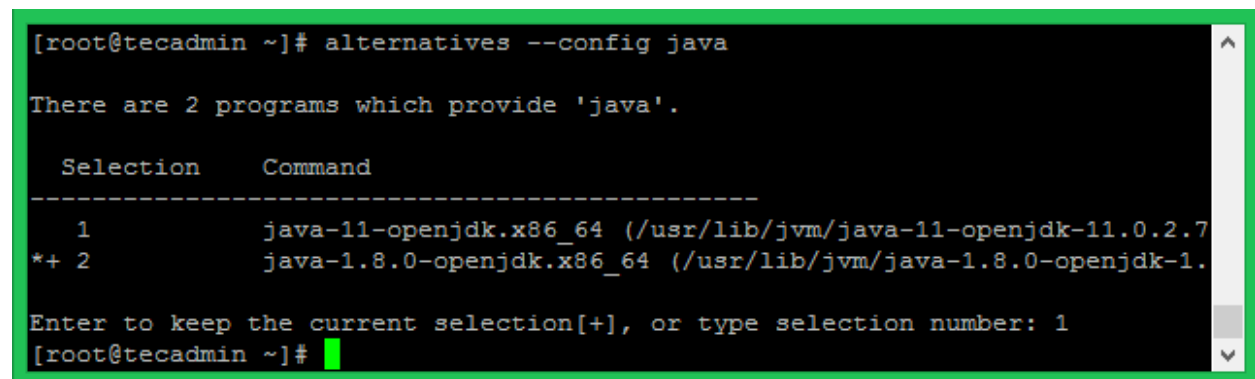
$ java -version

openjdk version "1.8.0_222"

OpenJDK Runtime Environment (build 1.8.0_222-8u222-b10-1ubuntu1~18.04.1-b10)

OpenJDK 64-Bit Server VM (build 25.222-b10, mixed mode)

Step 3 – Switch Java Version

Use alternatives command-line utility to switch active Java version on your Amazon Linux system. Run below command from the command line and select the appropriate Java version to make it default.

$ alternatives --config java

```
[root@tecadmin ~]# alternatives --config java

There are 2 programs which provide 'java'.

  Selection    Command
-----------------------------------------------
   1           java-11-openjdk.x86_64 (/usr/lib/jvm/java-11-openjdk-11.0.2.7
*+ 2           java-1.8.0-openjdk.x86_64 (/usr/lib/jvm/java-1.8.0-openjdk-1.

Enter to keep the current selection[+], or type selection number: 1
[root@tecadmin ~]#
```

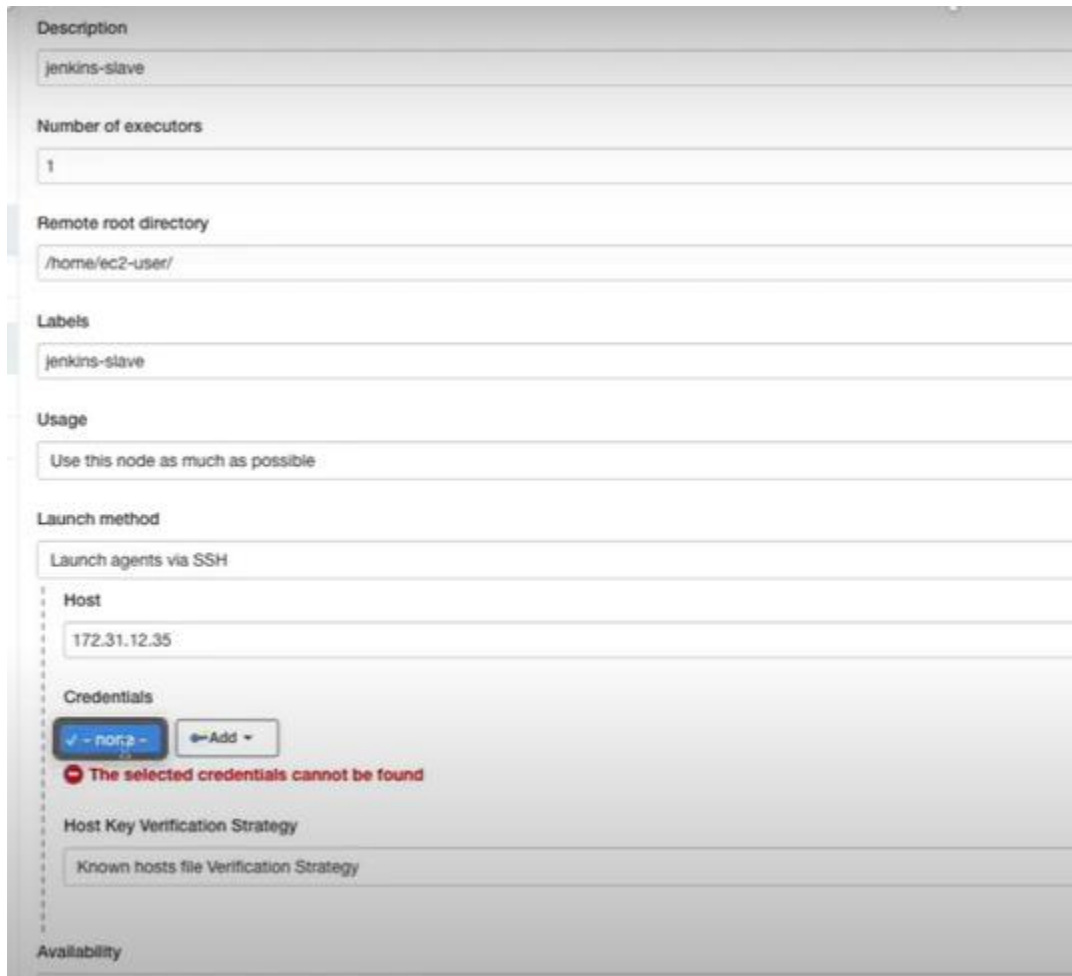After switching let's check again active Java version:

$ java -version

openjdk version "11.0.7" 2020-04-14 LTS

OpenJDK Runtime Environment 18.9 (build 11.0.7+10-LTS)

OpenJDK 64-Bit Server VM 18.9 (build 11.0.7+10-LTS, mixed mode, sharing)

Create Agent

Description

jenkins-slave

Number of executors

1

Remote root directory

/home/ec2-user/

Labels

jenkins-slave

Usage

Use this node as much as possible

Launch method

Launch agents via SSH

Host

172.31.12.35

Credentials

✓ - none -   ← Add ▾

🔴 The selected credentials cannot be found

Host Key Verification Strategy

Known hosts file Verification Strategy

Availability

Domain

Global credentials (unrestricted)

Kind

SSH Username with private key

Scope

Global (Jenkins, nodes, items, all child items, etc)

ID

jenkins-slave

Description

jenkins-slave

Username

ec2-user

☐ Treat username as secret

Private Key

● Enter directly

Key

Enter Ne

Host

172.31.12.35

Credentials

ec2-user (jenkins-slave) ▾   ⊕ Add ▾

Host Key Verification Strategy

Manually trusted key Verification Strategy

☐ Require manual verification of initial connection

Availability

Keep this agent online as much as possible

Node Properties

☐ Disable deferred wipeout on this node
☐ Environment variables
☐ Tool Locations

Save

S	Name ↓	Architecture	Clock Difference	Free Disk Space	Free Swap S
🖥	jenkins-slave ▾		N/A	N/A	
🖥	master	Linux (amd64)	In sync	5.66 GB	
	last checked	8 ms	5 ms	0 ms	

- Back to Dashboard
- Manage Jenkins
- New Node
- Configure Clouds
- Node Monitoring

Build Queue ⌃

No builds in the queue.

Build Executor Status ⌃

Click on slave and go to log and should connect

Create a simple pipeline