Jenkins Build Node Requirements

January 24, 2018

Geoff Rosenthal ([Geoffrey.Rosenthal@perficient.com)](mailto:Geoffrey.Rosenthal@perficient.com))



© 2018 Perficient, Inc. All Rights Reserve

This document lists the requirements for Jenkins build nodes that are used by **ACME Organization**.

# General Build Server Requirements

Build servers should have a reasonable amount of resources in order to perform builds. More information on Jenkins can be found:

<https://jenkins.io/doc/book/hardware-recommendations/>

<https://jenkins.io/doc/book/architecting-for-scale/#distributed-builds-architecture>

In general, the amount of system resources that Jenkins uses (either a master or slave) highly depends upon the types of jobs that are running, which tools they use (like NodeJs), and the number of jobs that are concurrently running.

As general guidance…

-Allocate at least 1 GB of RAM for each executor that you allocate on the build node.

-Install the Jenkins workspace directory on a disk partition that can be expanded and is not the main disk for the machine

-Implement disk storage monitoring to be warned before the machine runs out of disk space so that corrective action can be taken before issues arise

-Continually monitor the node from the Jenkins Master for free space, swap space, and free temp space issues

-Sync the system clocks on all build nodes with the Jenkins Master

# All Build Nodes

1. A JRE or JDK should be installed first. Java 8 is required (use the latest release).
2. Install GIT: <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>
3. For Linux build nodes, ensure that SSH is enabled between the Jenkins Master and slave servers. Here is some documentation on one approach to ensure SSH is working: <https://support.cloudbees.com/hc/en-us/articles/222978868-How-to-Connect-to-Remote-SSH-Slaves->
   1. If the jenkins user is disabled by default on the build node or master, enable the user by running the command:

su –s /bin/bash jenkins

* 1. While running as jenkins user, from the master, ssh into build server and from the build server, ssh into the master in order to update known\_hosts file in .ssh directory to allow access to target server

ssh root@<IP address>

* 1. When prompted type “yes” to add the target’s RSA key to the list of known hosts

1. Use the Jenkins UI to create/setup an SSH slave on the deployment target environments i.e. devint, devint2, QA, QA2, etc.
2. Add the **ACME Organization** security certificates to the Java Keystore that is configured as the default Java instance for the server. This should be the Java instance that the Jenkins process is using to run the node.

update-alternatives --configure java

1. Configure the build node to use the **ACME Organization** Nexus Repo instance as a proxy repository for NPM. See the documentation section “Configuring npm”: <https://help.sonatype.com/display/NXRM3/Node+Packaged+Modules+and+npm+Registries>
   1. You need to set the registry property in the user home directory to the URL of the Nexus proxy group for NPM packages.

# To Build Node.js Applications

You must install both NPM and Node.js. You must install Node.js before you install NPM. See the documentation:

<https://docs.npmjs.com/getting-started/installing-node>