**Jenkins Installation!!**

Jenkins is typically run as a standalone application in its own process with the built-in Java servlet container/application server (Jetty).

Jenkins can also be run as a servlet in different Java servlet containers such as Apache Tomcat or GlassFish.

**Prerequisites**

1. Minimum hardware requirements:
   1. 256 MB of RAM
   2. 1 GB of drive space (although 10 GB is a recommended minimum if running Jenkins as a Docker container)
2. Recommended hardware configuration for a small team:
   1. 1 GB+ of RAM
   2. 50 GB+ of drive space
3. Sofware requirements:

Java 8 - either a Java Runtime Environment (JRE) or a Java Development Kit (JDK) is fine >> Java 8 or Java 11 is required

>> Java 9 and 10 are not supported

**Note:** This is not a requirement if running Jenkins as a Docker container.

**Runtime Requirements: Jvm**

* Java Virtual Machine (JVM) must be tuned:
* Memory Heap size: -Xms1G -Xmx2G
* G1 garbage collector for heap > 4GB : -XX:+UseG1GC
* Check your Java Memory documentation for details

**Runtime Requirements: Operating System**

* + - Beware of operating system Limits
    - Maximum number of open files
    - Maximum number of forked processes
    - Network tuning (packet size, TCP timeouts)

**Jenkins Is Disk I/O Intensive**

* Jenkins writes a lot of files as it processes builds
  + Some of these can be offloaded to external repositories such as Nexus and Artifactory
* Disk usage grows over time
  + Especially when you start to host jobs from people who are not close to you
  + Be sure you add more disks when necessary

**Disk Requirements**

* No need to waste money on 15000rpm SCSI disks
  + But bigger disk is nice
  + Low latency is the big winner (SSD)
* Can be network mounted
  + NFS
  + SAN

**Disk Space**

* Artifacts, logs and other files written by builds can be very large
  + Consider external storage such as Nexus or Artifactory for storing  
    these large files off the Jenkins master
  + Retention policy affects the amount of space these consume
    - If you need to retain all (or most) artifacts for a long time, the  
      disk consumption issues are more serious

**Disk Space For Backups**

* Be sure to configure enough disk space for backups
* Can configure a separate disk on the master just to hold backups
* Can back up Jenkins files to a local directory then copy those files to an external location
* Must have enough space on the disk to accommodate the backed up files
* Recommend creating a separate filesystem (mountpoint) for the backup directory

**Expandable Volume**

* + “Spanned volume” on Windows
  + LVM on Linux
    - This needs the most upfront planning
  + ZFS
    - There’s no viable free Solaris flavor
  + The Jenkins home directory ($JENKINS\_HOME) should be its own filesystem
    - If it’s too late, symlink is your friend

**Jenkins Installation Platform**

* Run Jenkins as a standalone application on Windows/Ubuntu/Red Hat Linux/Fedora
* Run Jenkins on a servlet container (Apache Tomcat)
* Run Jenkins behind Nginx server
* Run Jenkins behind a reverse proxy server (Nginx)
* Run Jenkins with Docker

**Installing a standalone Jenkins server on Red Hat Linux**

**Step 1 : Install EPEL repo & update repo**

* $ sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
* $ sudo dnf update -y

**Step 2 : Install Open JDK [JAVA]**

* $ sudo dnf install java-11-openjdk-devel -y
* $ java –version
* $ javac –version
* $whereis java

**SET ENV Variable [JAVA\_HOME]**

* $sudo vim /etc/profile

#JAVA PATH

export JAVA\_HOME=" /usr/lib/jvm/java-11-openjdk-11.0.10.0.9-4.el8\_3.x86\_64"

export PATH=$JAVA\_HOME/bin:$PATH

* Verification >> source /etc/profile

echo $JAVA\_HOME

java –version

javac -verison

**Step 3 : Install Jenkins**

* sudo wget -O /etc/yum.repos.d/jenkins.repo \
* https://pkg.jenkins.io/redhat-stable/jenkins.repo
* sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
* sudo yum upgrade
* sudo yum install jenkins java-1.8.0-openjdk-devel
* sudo systemctl daemon-reload
* sudo systemctl start jenkins
* sudo systemctl enable jenkins

**Step 4 : Launch** [**http://localhost:8080/**](http://localhost:8080/)

**Installing a standalone Jenkins server on Windows**

* STEP 1 : Install Java & JAVA\_HOME path variable
* STEP 2: Download & install Jenkins
* STEP 3: Launch <http://localhost:8080/>

java -jar jenkins.war --httpPort=8088

**Run Jenkins on a servlet container on Apache Tomcat**

**Step 1 : Install EPEL repo & update repo**

* $ sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
* $ sudo dnf update -y

**Step 2 : Install Open JDK [JAVA]**

* $ sudo dnf install java-11-openjdk-devel -y
* $ java –version
* $ javac --version

**SET ENV Variable [JAVA\_HOME]**

* $sudo vim /etc/profile

#JAVA PATH

export JAVA\_HOME="/usr/lib/jvm/java-11-openjdk-11.0.7.10-1.el8\_1.x86\_64"

export PATH=$JAVA\_HOME/bin:$PATH

* Verification >> source /etc/profile

echo $JAVA\_HOME

java –version

javac -verison

**Step 3 : Install Tomcat on Redhat 8**

* useradd -m –d /opt/tomcat –U –s /bin/false tomcat
* cd /opt/tomcat
* wget <https://downloads.apache.org/tomcat/tomcat-9/v9.0.37/bin/apache-tomcat-9.0.37-fulldocs.tar.gz>
* tar -zxvf apache-tomcat-9.0.37.tar.gz
* chown -R tomcat:tomcat apache-tomcat-9.0.37
* cd apache-tomcat-9.0.37

**SET TOMCAT\_HOME**

* sudo vim /etc/profile

#TOMCAT PATH

export TOMCAT\_HOME="/opt/tomcat/apache-tomcat-9.0.37"

export PATH=$TOMCAT\_HOME/bin:$PATH

* source /etc/profile
* echo $TOMCAT\_HOME

**Set Startup & Shutdown Script**

sudo ln –s /opt/tomcat/apache-tomcat-9.0.37/bin/startup.sh /usr/local/bin/tomcatup

sudo ln –s /opt/tomcat/apache-tomcat-9.0.37/bin/shutdown.sh /usr/local/bin/tomcatdown

$ tomcatup

Make sure you give necessary permsion to startup & shutdown script & logs.

**Tomcat as a system service**

vim /etc/systemd/system/tomcat.service

****

**TOMCAT Configuration**

**Port management** [Default 8080] Below steps needed if port cange is req

* vim /usr/lib/tomcat/apache-tomcat-9.0.31/conf/server.xml  
  <Connector port="8090" protocol="HTTP/1.1"
* Restart tomcat

**Access Manager App**## edit context.xml##

* vim /opt/tomcat/apache-tomcat-9.0.37/webapps/manager/META-INF/context.xml

##coment this line

<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->

* vim /opt/tomcat/apache-tomcat-9.0.37/webapps/host-manager/META-INF/context.xml

##coment this line

<!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> --> ##

**TO CREATE USERS WITH ROLES**###EDIT tomcat-users.xml###

* vim /opt/tomcat/apache-tomcat-9.0.37/conf/tomcat-users.xml
* Place between

<tomat-user

----------------------------------

----------------

</tomcat-users>

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>

<user username="deployer" password="deployer" roles="manager-script"/>

<user username="tomcat" password="tomcat" roles="manager-gui"/>

**Deploy Jenkins**

* cd /tmp
* wget <http://mirrors.jenkins.io/war-stable/latest/jenkins.war>
* mv Jenkins.war ROOT.war
* cd /opt/tomcat/apache-tomcat-9.0.37/webapps
* rm –rvf \*
* cp /tmp/ROOT.war .
* It creates ROOT folder & deploy success
* Launcg url:8080

**Setting up the Jenkins home path [TOMCAT]**

Before we start using Jenkins, there is one important thing to configure, the jenkins\_home

path. When you install Jenkins as a service on Tomcat, the jenkins\_home path is automatically set to /root/.jenkins/. This is the location where all of the Jenkins configurations, logs, and builds are stored. Everything that you create and configure on the Jenkins dashboard is stored here.

We need to make it something more accessible, something like /var/jenkins\_home. This

can be done in the following way:

1. Stop the Apache Tomcat server using the following command:

**sudo systemctl stop tomcat**

2. Open the context.xml file for editing, which is present

inside /opt/tomcat/conf:

**sudo nano /opt/tomcat/conf/context.xml**

3. The file will look like this (comments removed):

<?xml version="1.0" encoding="UTF-8"?>

<Context>

<WatchedResource>WEB-INF/web.xml</WatchedResource>

<WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>

</Context>

4. Add the following line between <Context> </Context>:

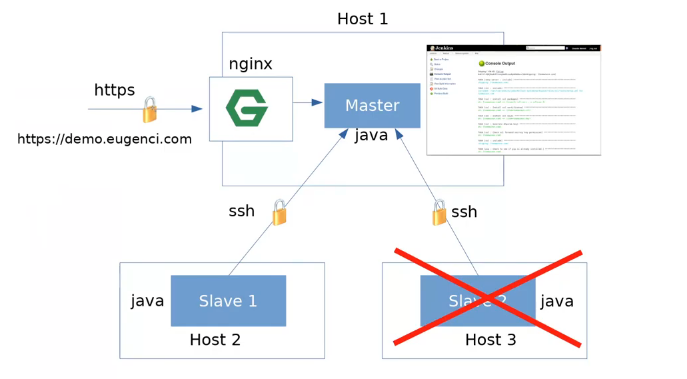
<Environment name="JENKINS\_HOME" value="/var/jenkins\_home"

type="java.lang.String"/>

5. Start the Tomcat service using the following command:

**sudo systemctl start tomcat**

**Jenkins as a Revers Proxy [NGINX]**



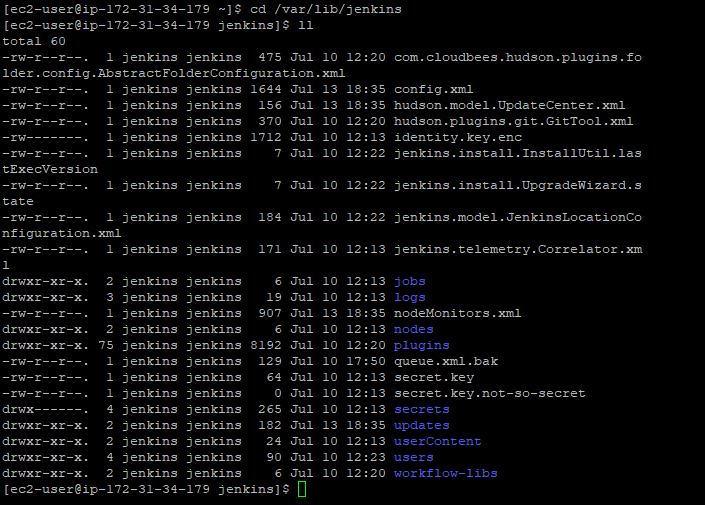
**Running Jenkins on Docker**

* sudo dnf install docker –y
* docker pull jenkins
* docker run –d –name jenkinsmaster –p 8081:8080 jenkins

**Jenkins HOME DIRECTORY**

**Default: /var/lib/Jenkins/**

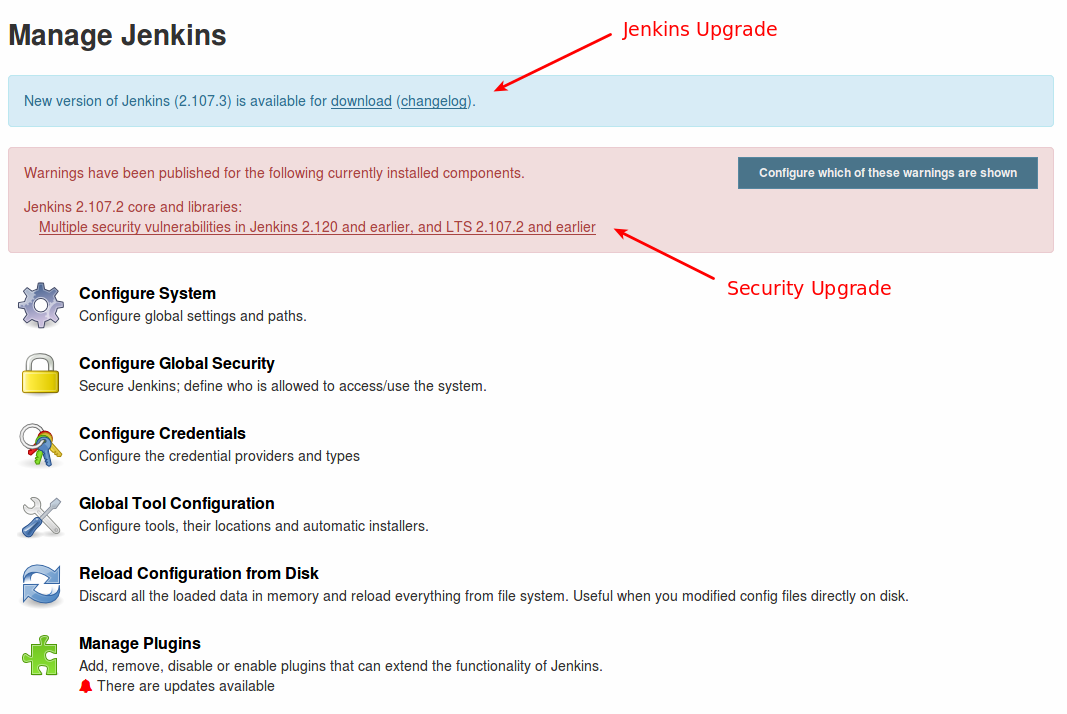
**Directory structure**

****

**JENKINS UPGRADE**

Manage console will display the newer version availability.

**Download war file & install the Jenkins & Restart**

****