

## JENKINS INSTALLATION AND CONFIGURATION

**Introduction:** Jenkins is an open source continuous integration tool that simplifies your build process. This application can be deployed on a app server and configured to build your code out of the source repository, run defined tests and report its stability on a schedule basis.

### **Dependencies:**

**Jenkins:** `http://jenkins-ci.org/`

```
wget http://mirrors.jenkins-ci.org/war/latest/jenkins.war
```

**Jetty 9:** `http://download.eclipse.org/jetty/`

```
wget http://download.eclipse.org/jetty/stable-9/dist/jetty-
```

```
distribution-9.2.10.v20150310.tar.g
```

**Pre-requisite:** A linux instance with an app server. I am using **Fedora 21** with jetty 9 here for my setup.

You could also launch the jenkins.war using the embedded jetty server using the following command.

```
$ nohup java -jar jenkins.war > $LOGFILE 2>&1
```

Alternately, I prefer using an app server for jenkins deployment as that would be the production deployment scenario. The app server could be Tomcat, JBoss or GlassFish. I think this is more of a preference.

### **Environment:**

Setup Variables:

Add an entry to the `~/.bashrc` file and source it.

```
export JETTY_HOME=/home/rahul/dev/jetty-distribution-9.2.10.v20150310/
```

```
export JENKINS_HOME=/home/rahul/dev/jenkins/
```

Create a context file to the deployment at `$JETTY_HOME/webapp/jenkins.xml`

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<!DOCTYPE Configure PUBLIC "-//Jetty//Configure//EN"
"http://www.eclipse.org/jetty/configure.dtd">
```

```
<!-- =====
```

```
Configure and deploy the jenkins web application in
$(jetty.home)/webapps/jenkins
```

Note. If this file did not exist or used a context path other than /jenkins then the default configuration of jetty.xml would discover the jenkins webapplication with a WebAppDeployer. By specifying a context in this directory, additional configuration may be specified and hot deployments detected.

```
===== -->
```

```
<Configure class="org.eclipse.jetty.webapp.WebAppContext">
```

```
<!-- - - - - - -->
```

```
<!-- Required minimal context configuration : -->
```

```
<!-- + contextPath -->
```

```
<!-- + war OR resourceBase -->
```

```
<!-- - - - - - -->
```

```
<Set name="contextPath">/jenkins</Set>

<Set name="war"><SystemProperty name="jetty.home"
default="."/>/webapps/jenkins.war</Set>

<!-- - - - - - -->

<!-- Optional context configuration -->

<!-- - - - - - -->

<Set name="extractWAR">true</Set>

<Set name="copyWebDir">false</Set>

<Set name="defaultsDescriptor"><SystemProperty name="jetty.home"
default="."/>/etc/webdefault.xml</Set>

<!--<Set name="overrideDescriptor"><SystemProperty name="jetty.home"
default="."/>/webapps/jenkins.d/override-web.xml</Set>-->

<Get name="securityHandler">

<Set name="loginService">

<New class="org.eclipse.jetty.security.HashLoginService">

<Set name="name">Jenkins Realm</Set>
```

```
<Set name="config"><SystemProperty name="jetty.home"
default="."/>/etc/realm.properties</Set>

<!-- To enable reload of realm when properties change, uncomment
the following lines -->

<!-- changing refreshInterval (in seconds) as
desired -->

<!--

<Set name="refreshInterval">5</Set>

<Call name="start"></Call>

-->

</New>

</Set>

<Set name="authenticator">

<New class="org.eclipse.jetty.security.authentication.FormAuthenticator">

<Set name="alwaysSaveUri">true</Set>

</New>

</Set>

<Set name="checkWelcomeFiles">true</Set>

</Get>
```

```
<!-- Add context specific logger

    <Set name="handler">

        <New id="RequestLog"
class="org.eclipse.jetty.server.handler.RequestLogHandler">

            <Set name="requestLog">

                <New id="RequestLogImpl" class="org.eclipse.jetty.server.NCSARequestLog">

                    <Set name="filename"><Property name="jetty.logs"
default="./logs"/>/jenkins-yyyy_mm_dd.request.log</Set>

                    <Set name="filenameDateFormat">yyyy_MM_dd</Set>

                    <Set name="append">true</Set>

                    <Set name="LogTimeZone">GMT</Set>

                </New>

            </Set>

        </New>

    </Set>

-->
```

</Configure>

## Configuration

**Setup Jetty** : Configure port in `$JETTY_HOME/start.ini` to 8085

**Deploy jenkins** : `cp jenkins.war $JETTY_HOME/webapp/`

**Stop Firewall** : `sudo service iptables stop`

**Run Jetty** : `java -jar start.jar`

**Access Jenkins** at [\*http://192.168.56.102:8085/jenkins/\*](http://192.168.56.102:8085/jenkins/)



Make sure that the jenkins folders have the relevant user privilege/access at the operating system. The home page would look like this screen shot. The url could be an IP or the domain name based on the setup. At this point we should start thinking on what we really expect out of this installation and the various repository and build environment expected here.