Maven and Tomcat Installation on CentOS:

Install Maven

Let us download the latest version from the project download page. At the time of writing this article, the stable Ant version was **3.2.3**. For the purpose of this tutorial, I will be using Apache Maven binary file in tar format

```
$ wget https://www-us.apache.org/dist/maven/maven-3/3.6.1/binaries/apache-maven-3.6.1-bin.tar.gz
```

Extract it using command:

```
$ tar -xvf apache-maven-3.6.1-bin.tar.gz
```

Move the extracted contents to /opt/ directory.

```
$ sudo mv apache-maven-3.6.1-bin/ /opt/maven
```

Make a symlink to maven/bin folder as shown below.

```
$ sudo ln -s /opt/maven/bin/mvn /usr/bin/mvn
```

Setup Maven environment variable

Create a file called maven.sh under /etc/profile.d/ directory.

```
vi /etc/profile.d/maven.sh
```

Add the following contents:

```
#!/bin/bash

MAVEN_HOME=/opt/maven

PATH=$MAVEN_HOME/bin:$PATH
```

```
export PATH MAVEN_HOME
```

```
export CLASSPATH=.
```

Save and close the file. Make it executable using the following command.

```
chmod +x /etc/profile.d/maven.sh
```

Then, set the environment variables permanently by running the following command:

```
source /etc/profile.d/maven.sh
```

Log out or reboot your system.

Now, check the ant version using command:

```
mvn -version
```

Sample output:

```
Maven home: /opt/maven
Java version: 1.8.0_171, vendor: Oracle Corporation
Java home: /usr/java/jdk1.8.0_171-amd64/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "3.10.0-693.2.2.el7.x86_64", arch: "amd64", family: "unix"
```

Check the environment variables:

```
echo $MAVEN_HOME
```

Sample output:

```
/opt/maven
```

Or

```
echo $PATH
```

Sample output:

/opt/tomcat/bin:/opt/maven/bin:/usr/java/jdk1.8.0_171-amd64/bin:/sbin:/bin:/usr/sbin:/usr/bin

Install Tomcat

From the **Apache Tomcat project home page**,

First, download the latest version from the project [download page](). At the time of writing this article, the stable Tomcat version was **8.0.15**. For the purpose of this tutorial, I will be using Apache Tomcat binary zip file.

```
wget http://www-us.apache.org/dist/tomcat/tomcat-
8/v8.5.31/bin/apache-tomcat-8.5.31.tar.gz
```

Extract it using command:

```
tar -xvf apache-tomcat-8.5.31.tar.gz
```

Move the extracted contents to **/opt/** directory.

```
mv apache-tomcat-8.0.15/ /opt/tomcat
```

Setup Tomcat environment variable

Create a file called tomcat.sh under /etc/profile.d/ directory.

```
vi /etc/profile.d/tomcat.sh
```

Add the following contents:

```
#!/bin/bash

CATALINA_HOME=/opt/tomcat

PATH=$CATALINA_HOME/bin:$PATH

export PATH CATALINA_HOME

export CLASSPATH=.
```

Save and close the file. Make it executable using the following command.

```
chmod +x /etc/profile.d/tomcat.sh
```

Then, set the environment variables permanently by running the following command:

```
source /etc/profile.d/tomcat.sh
```

Now, we have to start the Tomcat server. Before starting Tomcat, give executable permission to the following script files.

```
chmod +x $CATALINA_HOME/bin/startup.sh

chmod +x $CATALINA_HOME/bin/shutdown.sh

chmod +x $CATALINA_HOME/bin/catalina.sh
```

Start Tomcat server by running the following command from your Terminal.

```
cd $CATALINA_HOME/bin

./startup.sh
```

Sample output:

```
Using CATALINA_BASE:   /opt/tomcat

Using CATALINA_HOME:   /opt/tomcat

Using CATALINA_TMPDIR: /opt/tomcat/temp

Using JRE_HOME:        /usr/java/jdk1.8.0_25/

Using
CLASSPATH:      /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/to
mcat-juli.jar

Tomcat started.
```

As you see in the above output, the Tomcat server has been started. Now, open up the browser and navigate to http://ip-address:8080/.

Make sure you've allowed the Tomcat default port **8080** through iptables.

On CentOS 6.x systems:

Edit file **/etc/sysconfig/iptables**,

```
vi /etc/sysconfig/iptables
```

Add the following line:

```
[...]

-A INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j
ACCEPT

[...]
```

Save and close the file and restart the iptables service.

```
service iptables restart
```

On CentOS 7 systems:

```
firewall-cmd --permanent --add-port=8080/tcp

firewall-cmd --reload
```

Now, navigate to **http://ip-address:8080** from your web browser. You should see the following screen.

Voila! Congratulations! The Tomcat server is up and working.

To stop the server, just enter the following commands:

```
cd $CATALINA_HOME/bin

./shutdown.sh
```

Auto start Tomcat

We haven't finished yet. The Tomcat server won't start after the system reboots. You have to make it to start it manually on every reboot.

If you want it to start automatically on every reboot, Just follow the steps given below.

Create a file called **tomcat** under **/etc/init.d/** directory.

```
vi /etc/init.d/tomcat
```

Copy/paste the following lines in it. Replace the JAVA and Tomcat paths and environment variables with your own.

```
#!/bin/sh

# chkconfig: 2345 80 20

# Description: Tomcat Start/Shutdown script

export JAVA_HOME=/usr/java/jdk1.8.0_25

case $1 in

start)

cd /opt/tomcat/bin/

./startup.sh

;;

stop)

cd /opt/tomcat/bin/

./shutdown.sh

;;

restart)

cd /opt/tomcat/bin/

./shutdown.sh
```

```
cd /opt/tomcat/bin/

./startup.sh

;;

esac

exit 0
```

Make sure you have added the correct path of the jdk. Save and close the file. Set executable permission to the above file.

```
chmod a+x /etc/init.d/tomcat
```

Run the following command to add the tomcat service.

```
chkconfig --add tomcat
```

NB: Add double dashes(–) in front of **'add'** word.

Now, start the tomcat service and make it to start automatically as shown below.

On CentOS 6.x systems:

```
service tomcat start

chkconfig tomcat on
```

or, for CentOS 7.x

```
systemctl tomcat start

chkconfig tomcat on
```