

Install Tomcat 7 on CentOS / RHEL

PLEASE NOTE: This post covers installation of Tomcat 7 along with JDK 6. For installation of Tomcat 7 with JDK 6 or JDK 7, please see my updated and expanded post here:

http://www.davidghedini.com/pg/entry/install_tomcat_7_on_centos

This post will cover installing and basic configuration of Tomcat 7 on CentOS 5.x.

The procedure can be used for Fedora and RHEL as well.

Tomcat 7 implements the JavaServer Pages 2.2 and Servlet 3.0 specifications and a number of new features. The Manager application also has a new look with finer-grain roles and access than 6.x

In this post, we'll install the required JDK, Tomcat, configure Tomcat as a service, create a start/stop/restart script, and (optionally) configure Tomcat to run under a non-root user.

For this installation, we'll use Tomcat 7.0.19, the current stable release of Tomcat 7. This post began with the first Tomcat 7 release and I have tried to keep it updated to keep things as "copy and paste" as possible.

I've also updated the post for JDK 6, Update 26.

To begin, we'll need to install the Java Development Kit (JDK) 1.6

JDK 1.6 is the minimum JDK version for Tomcat 7.

If you do have the JDK installed, you can skip to: Step 2: Download and Unpack Tomcat 7.0.19:

Step 1: Install JDK 1.6

You can download the JDK here:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

We'll install the latest JDK, which is JDK 6 Update 26. The JDK is specific to 32 and 64 bit versions.

My CentOS box is 64 bit, so I'll need: `jdk-6u26-linux-x64.bin`

If you are on 32 bit, you'll need: `jdk-6u26-linux-i586.bin`

Download the appropriate JDK and save it to a directory. I'm saving it to /root.

Move (mv) or copy (cp) the file to the /opt directory:

[view plainprint?](#)

1. [root@srv6 ~]# mv jdk-6u26-linux-x64.bin /opt/jdk-6u26-linux-x64.bin

Create a new directory /usr/java.

[view plainprint?](#)

1. [root@srv6 ~]# mkdir /usr/java

Change to the /usr/java directory we created and install the JDK using 'sh /opt/jdk-6u26-linux-x64.bin'

[view plainprint?](#)

1. [root@srv6 ~]# cd /usr/java
2. [root@srv6 java]# sh /opt/jdk-6u26-linux-x64.bin

Set the JAVA_HOME path. This is where we installed our JDK above.

To set it for your current session, you can issue the following from the CLI:

[view plainprint?](#)

1. [root@srv6 java]# JAVA_HOME=/usr/java/jdk1.6.0_26
2. [root@srv6 java]# export JAVA_HOME
3. [root@srv6 java]# PATH=\$JAVA_HOME/bin:\$PATH
4. [root@srv6 java]# export PATH

To set the JAVA_HOME permanently, we add below to either the ~/.bashrc or ~/.bash_profile of the user (in this case, root).

We can also add it /etc/profile and then source it to give to all users.

[view plainprint?](#)

1. JAVA_HOME=/usr/java/jdk1.6.0_26
2. export JAVA_HOME
3. PATH=\$JAVA_HOME/bin:\$PATH
4. export PATH

Once you have added the above to ~/.bash_profile or ~/.bashrc, you should log out, then log back in and check that the JAVA_HOME is set correctly.

[view plainprint?](#)

1. [root@srv6 ~]# echo \$JAVA_HOME
2. /usr/java/jdk1.6.0_26

Step 2: Download and Unpack Tomcat 7.0.19

Download apache-tomcat-7.0.19.tar.gz [here](#)

Alternatively, you can download using wget.

[view plainprint?](#)

1. [root@srv6 ~]# wget http://apache.mivzakim.net/tomcat/tomcat-7/v7.0.19/bin/apache-tomcat-7.0.19.tar.gz

Save the file to a directory. I'm saving it to /root/apache-tomcat-7.0.19.tar.gz

Before proceeding, you should verify the MD5 Checksum for your Tomcat download (or any other download).

Since we saved the Tomcat download to /root/apache-tomcat-7.0.19.tar.gz, we'll go to the /root directory and use the md5sum command.

[view plainprint?](#)

1. [root@srv6 ~]# md5sum apache-tomcat-7.0.19.tar.gz
2. 5a5e9bc742714d1b7210d9f68764fd8e *apache-tomcat-7.0.19.zip

Compare the output above to the MD5 Checksum provided by [here the Apache Tomcat MD5 page](#) and insure that they match exactly.

Now, move (mv) or copy (cp) the file to the /usr/share directory:

[view plainprint?](#)

1. [root@srv6 ~]# mv apache-tomcat-7.0.19.tar.gz /usr/share/apache-tomcat-7.0.19.tar.gz

Change to the /usr/share directory and unpack the file using tar -xzf:

[view plainprint?](#)

1. [root@srv6 ~]# cd /usr/share
2. [root@sv2 srv6]# tar -xzf apache-tomcat-7.0.19.tar.gz

This will create the directory /usr/share/apache-tomcat-7.0.19

Step 3: Configure Tomcat to Run as a Service.

We will now see how to run Tomcat as a service and create a simple Start/Stop/Restart script, as well as to start Tomcat at boot.

Change to the /etc/init.d directory and create a script called 'tomcat' as shown below.

[view plainprint?](#)

1. [root@srv6 share]# cd /etc/init.d

2. [root@srv6 init.d]# vi tomcat

[view plainprint?](#)

```
1. #!/bin/bash
2. # description: Tomcat Start Stop Restart
3. # processname: tomcat
4. # chkconfig: 234 20 80
5. JAVA_HOME=/usr/java/jdk1.6.0_26
6. export JAVA_HOME
7. PATH=$JAVA_HOME/bin:$PATH
8. export PATH
9. CATALINA_HOME=/usr/share/apache-tomcat-7.0.19
10.
11.
12.case $1 in
13.start)
14.sh $CATALINA_HOME/bin/startup.sh
15.;;
16.stop)
17.sh $CATALINA_HOME/bin/shutdown.sh
18.;;
19.restart)
20.sh $CATALINA_HOME/bin/shutdown.sh
21.sh $CATALINA_HOME/bin/startup.sh
22.;;
23.esac
24.exit 0
```

The above script is simple and contains all of the basic elements you will need to get going.

As you can see, we are simply calling the startup.sh and shutdown.sh scripts located in the Tomcat bin directory (/usr/share/apache-tomcat-7.0.19/bin).

You can adjust your script according to your needs and, in subsequent posts, we'll look at additional examples.

CATALINA_HOME is the Tomcat home directory (/usr/share/apache-tomcat-7.0.19)

Now, set the permissions for your script to make it executable:

[view plainprint?](#)

```
1. [root@srv6 init.d]# chmod 755 tomcat
```

We now use the chkconfig utility to have Tomcat start at boot time. In my script above, I am using chkconfig: 234 20 80. 2445 are the run levels and 20 and 80 are the stop and start priorities respectively. You can adjust as needed.

[view plainprint?](#)

1. [root@srv6 init.d]# chkconfig --add tomcat
2. [root@srv6 init.d]# chkconfig --level 234 tomcat on

Verify it:

[view plainprint?](#)

1. [root@srv6 init.d]# chkconfig --list tomcat
2. tomcat 0:off 1:off 2:on 3:on 4:on 5:off 6:off

Now, let's test our script.

Start Tomcat:

[view plainprint?](#)

1. [root@srv6 ~]# service tomcat start
2. Using CATALINA_BASE: /usr/share/apache-tomcat-7.0.19
3. Using CATALINA_HOME: /usr/share/apache-tomcat-7.0.19
4. Using CATALINA_TMPDIR: /usr/share/apache-tomcat-7.0.19/temp
5. Using JRE_HOME: /usr/java/jdk1.6.0_26
6. Using CLASSPATH: /usr/share/apache-tomcat-7.0.19/bin/bootstrap.jar:/usr/share/apache-tomcat-7.0.19/bin/tomcat-juli.jar

Stop Tomcat:

[view plainprint?](#)

1. [root@srv6 ~]# service tomcat stop
2. Using CATALINA_BASE: /usr/share/apache-tomcat-7.0.19
3. Using CATALINA_HOME: /usr/share/apache-tomcat-7.0.19
4. Using CATALINA_TMPDIR: /usr/share/apache-tomcat-7.0.19/temp
5. Using JRE_HOME: /usr/java/jdk1.6.0_26
6. Using CLASSPATH: /usr/share/apache-tomcat-7.0.19/bin/bootstrap.jar:/usr/share/apache-tomcat-7.0.19/bin/tomcat-juli.jar

Restarting Tomcat (Must be started first):

[view plainprint?](#)

1. [root@srv6 ~]# service tomcat restart
2. Using CATALINA_BASE: /usr/share/apache-tomcat-7.0.19
3. Using CATALINA_HOME: /usr/share/apache-tomcat-7.0.19
4. Using CATALINA_TMPDIR: /usr/share/apache-tomcat-7.0.19/temp
5. Using JRE_HOME: /usr/java/jdk1.6.0_26
6. Using CLASSPATH: /usr/share/apache-tomcat-7.0.19/bin/bootstrap.jar:/usr/share/apache-tomcat-7.0.19/bin/tomcat-juli.jar
7. Using CATALINA_BASE: /usr/share/apache-tomcat-7.0.19
8. Using CATALINA_HOME: /usr/share/apache-tomcat-7.0.19
9. Using CATALINA_TMPDIR: /usr/share/apache-tomcat-7.0.19/temp

10.Using JRE_HOME: /usr/java/jdk1.6.0_26

11.Using CLASSPATH: /usr/share/apache-tomcat-7.0.19/bin/bootstrap.jar:/usr/share/apache-tomcat-7.0.19/bin/tomcat-juli.jar

We should review the Catalina.out log located at /usr/share/apache-tomcat-7.0.19/logs/catalina.out and check for any errors.

[view plainprint?](#)

1. [root@srv6 init.d]# more /usr/share/apache-tomcat-7.0.19/logs/catalina.out

We can now access the swanky new Tomcat Manager page at:

http://yourdomain.com:8080 or http://yourIPAddress:8080 and we should see the Tomcat home page.



Step 4: Configuring Tomcat Manager Access.

Tomcat 7 contains a number of changes that offer finer-grain roles.

For security reasons, no users or passwords are created for the Tomcat manager roles by default. In a production deployment, it is always best to remove the Manager application.

To set roles, user name(s) and password(s), we need to configure the tomcat-users.xml file located at \$CATALINA_HOME/conf/tomcat-users.xml.

In the case of our installation, \$CATALINA_HOME is located at /usr/share/apache-tomcat-7.0.19.

By default the Tomcat 7 tomcat-users.xml file will look as below.

[view plainprint?](#)

1. <!--

2. Licensed to the Apache Software Foundation (ASF) under one or more
3. contributor license agreements. See the NOTICE file distributed with
4. this work for additional information regarding copyright ownership.
5. The ASF licenses this file to You under the Apache License, Version 2.0
6. (the "License"); you may not use this file except in compliance with
7. the License. You may obtain a copy of the License at
8.
9. <http://www.apache.org/licenses/LICENSE-2.0>
10.
11. Unless required by applicable law or agreed to in writing, software
12. distributed under the License is distributed on an "AS IS" BASIS,
13. WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
14. See the License for the specific language governing permissions and
15. limitations under the License.
16.-->
17.<tomcat-users>
18.<!--
19. NOTE: By default, no user is included in the "manager-gui" role required
20. to operate the "/manager/html" web application. If you wish to use this app,
21. you must define such a user - the username and password are arbitrary.
22.-->
23.<!--
24. NOTE: The sample user and role entries below are wrapped in a comment
25. and thus are ignored when reading this file. Do not forget to remove
26. <!-- ..> that surrounds them.
27.-->
28.<!--
29. <role rolename="tomcat"/>
30. <role rolename="role1"/>
31. <user username="tomcat" password="tomcat" roles="tomcat"/>
32. <user username="both" password="tomcat" roles="tomcat,role1"/>
33. <user username="role1" password="tomcat" roles="role1"/>
34.-->
35.</tomcat-users>

Note that while examples are provided, the elements between the <tomcat-users> and </tomcat-users> tags have been commented-out.

New roles for Tomcat 7 offer finer-grained access.

The following roles are available:

manager-gui
manager-status
manager-jmx
manager-script
admin-gu
admin-script.

We can enable access for the manager-gui role, for example as below:

[view plainprint?](#)

1. <tomcat-users>
2. <role rolename="manager-gui">
3. <user username="tomcat" password="secret" roles="manager-gui">
4. </user>
5. </role></tomcat-users>

Caution should be exercised in granting multiple roles so as not to under-mind security.

Step 5 (Optional): How to Run Tomcat using Minimally Privileged (non-root) User.

In our Tomcat configuration above, we are running Tomcat as Root.

For security reasons, it is always best to run services with the only those privileges that are necessary.

There are some who make a strong case that this is not required, but it's always best to err on the side of caution.

To run Tomcat as non-root user, we need to do the following:

1. Create the group 'tomcat':

[view plainprint?](#)

1. [root@srv6 ~]# groupadd tomcat

2. Create the user 'tomcat' and add this user to the tomcat group we created above.

[view plainprint?](#)

1. [root@srv6 ~]# useradd -s /bin/bash -g tomcat tomcat

The above will create a home directory for the user tomcat in the default user home as /home/tomcat

If we want the home directory to be elsewhere, we simply specify so using the -d switch.

[view plainprint?](#)

1. [root@srv6 ~]# useradd -g tomcat -d /usr/share/apache-tomcat-7.0.19/tomcat tomcat

The above will create the user tomcat's home directory as /usr/share/apache-tomcat-7.0.19/tomcat

3. Change ownership of the tomcat files to the user tomcat we created above:

[view plainprint?](#)

1. [root@srv6 ~]# chown -Rf tomcat.tomcat /usr/share/apache-tomcat-7.0.19/

Note: it is possible to enhance our security still further by making certain files and directories read-only. This will not be covered in this post and care should be used when setting such permissions.

4. Adjust the start/stop service script we created above. In our new script, we need to su to the user tomcat:

[view plainprint?](#)

1. #!/bin/bash
2. # description: Tomcat Start Stop Restart
3. # processname: tomcat
4. # chkconfig: 234 20 80
5. JAVA_HOME=/usr/java/jdk1.6.0_26
6. export JAVA_HOME
7. PATH=\$JAVA_HOME/bin:\$PATH
8. export PATH
9. TOMCAT_HOME=/usr/share/apache-tomcat-7.0.19/bin
- 10.
- 11.
- 12.case \$1 in
- 13.start)
- 14./bin/su tomcat \$TOMCAT_HOME/startup.sh
- 15.;;
- 16.stop)
- 17./bin/su tomcat \$TOMCAT_HOME/shutdown.sh
- 18.;;
- 19.restart)
- 20./bin/su tomcat \$TOMCAT_HOME/shutdown.sh
- 21./bin/su tomcat \$TOMCAT_HOME/startup.sh
- 22.;;
- 23.esac
- 24.exit 0

Step 6 (Optional): How to Run Tomcat on Port 80 as Non-Root User.

Note: the following applies when you are running Tomcat in "stand alone" mode with Tomcat running under the minimally privileged user Tomcat we created in the previous step.

To run services below port 1024 as a user other than root, you can add the following to your IP tables:

[view plainprint?](#)

1. [root@srv6 ~]# iptables -t nat -A PREROUTING -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 8080
2. [root@srv6 ~]# iptables -t nat -A PREROUTING -p udp -m udp --dport 80 -j REDIRECT --to-ports 8080

Be sure to save and restart your iptables for the above change to take affect.