



Guida installazione

# Apache Tomcat 9



Capitolo	Pagina
Indice	2
Introduzione	3
Install Java	3
Install Tomcat	5
Download Tomcat	5
Configure firewall	6
Test tomcat	6
enable service	7
conf tomcat user	8
configure Apache Reverse Proxy	9

# Install Tomcat 9.0.13

## Introduction

This guide explains all the step-by-step procedures to install a web server in Tomcat on a CentOS 7 machine.

OS: CentOS 7

## Install Java

To use Tomcat is required java, before you install java update the system. Install the Java JRE (Java Runtime Environment) and the JDK (Java Development Kit).

```
yum update
yum install java-1.8.0-openjdk.x86_64 java-1.8.0-openjdk-devel.x86_64
```

Check if java is installed, checking the version.

```
java -version
javac -version
```

Set the last version as the usual Java version. Use the following command and select the last java version.

```
update-alternatives --config java
```

Set the java environment, editing the following file.

```
/etc/environment
```

Insert in the file the variable `JAVA_HOME` as follows:

```
JAVA_HOME="/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.111-1.b15.el7_2.x86_64/jre"
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-1.el7_6.x86_64/jre
```

then edit the profile bash script ( `~/.bash_profile` ) and insert the following lines.

```
export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.111-1.b15.el7_2.x86_64/jre
export PATH=$JAVA_HOME/bin:$PATH
```



reload the profile bash script

```
source ~/.bash_profile
```

check if the variable works correctly displaying it

```
echo $JAVA_HOME
```

It should return something like this

```
/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.111-1.b15.el7_2.x86_64/jre
```

## Install Tomcat

Create the user and group for tomcat:

```
groupadd tomcat
useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat
```

The `-s /bin/false` means that this user has no shell.

## Download Tomcat

Move to the directory where you want to install tomcat.

```
cd /opt
```

Install wget, it's a small software to download files by http.

```
yum install wget
```

Download tomcat, with the software you just installed.

```
http://www-eu.apache.org/dist/tomcat/tomcat-9/v9.0.13/bin/apache-tomcat-9.0.13.tar.gz
```

Decompress tomcats files

```
tar -xzvf apache-tomcat-9.0.13.tar.gz
```

Move tomcats files to the tomcat user home directory

```
mv apache-tomcat-9.0.13/* tomcat/
```

change owner of all the files under the tomcat home directory

```
chown -hR tomcat:tomcat tomcat
```

## Configure firewall

Install, start and enable the firewall

```
yum install firewalld
systemctl start firewalld
systemctl enable firewalld
```

Add to the firewall tables the port 8080 used by Tomcat.

```
firewall-cmd --zone=public --permanent --add-port=8080/tcp
firewall-cmd --reload
```

Check the ports and the services in the firewall tables.

```
firewall-cmd --list-ports
firewall-cmd --list-services
```

## Test tomcat

To test tomcat, start it.

```
cd /opt/tomcat/bin
./startup.sh
```

Open the server ip, with the port on the **8080** on a Browser, like **80.19.194.99:8080**  
If the reserver shows the page, shut it down

```
./shutdown.sh
```

## enable service

edit file:

```
/etc/systemd/system/tomcat.service
```

write in the file:

```
[Unit]
Description=Apache Tomcat 8 Servlet Container
After=syslog.target network.target

[Service]
User=tomcat
Group=tomcat
Type=forking
Environment=CATALINA_PID=/opt/tomcat/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

reload services, start service and enable it

```
systemctl start tomcat
systemctl enable tomcat
```

check service status

```
systemctl status tomcat
```

## conf tomcat user

edit file:

```
/opt/tomcat/conf/tomcat-users.xml
```

add rows:

```
<role rolename="manager-gui"/>
<user username="admin" password="password" roles="manager-gui,admin-gui"/>
```

then edit the file:

```
/opt/tomcat/webapps/manager/META-INF/context.xml
```

comment the rows 19, 20:

```
<Context antiResourceLocking="false" privileged="true" >
  <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
</Context>
```

edit file:

```
/opt/tomcat/webapps/host-manager/META-INF/context.xml
```

comment the rows 19,20:

```
<Context antiResourceLocking="false" privileged="true" >
  <!-- <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
</Context>
```

reboot the server and try to connect again on **ip:8080**



## configure Apache Reverse Proxy

Apache will be used as Reverse Proxy, to redirect the requests from the apache web server to Apache Tomcat.

Install Apache on CentOS, enable it and start it.

```
yum install httpd          # install
systemctl enable httpd     # enable
systemctl start httpd      # start
```

Then create the virtual host. Create the file `/etc/httpd/conf.d/tomcat_rev_proxy.conf`.

```
<VirtualHost example.domain.com:80>
    ServerName domotics.domain.com

    ProxyRequests Off
    ProxyPass /examples ajp://localhost:8009/examples
    ProxyPassReverse /examples ajp://localhost:8009/examples
</VirtualHost>
```

With this it will be displayed the the Apache Tomcat web-app `/example` on the request to the Proxy `example.domain.com/examples`.

After you configured the reverse proxy, check the configuration:

```
apachectl configtest
```

It should return something like `Syntax OK`, if it returns it, restart Apache httpd, with:

```
systemctl restart httpd
```

Now configure tomcat to accept connections from the Apache httpd reverse proxy.

Edit configuration tomcats file, `/opt/tomcat/conf/server.xml`, and go to the line that contains `Connector` and edit it as follows:

```
<Connector address="127.0.0.1" port="8009"...
```

Then restart tomcat:

```
/opt/tomcat/bin/shutdown.sh
/opt/tomcat/bin/startup.sh
```



Before you test if everything works open the port 80 on the firewall, with the command:

```
firewall-cmd --permanent --add-port=80/tcp  
firewall-cmd --reload
```

Now open your browser and open the page [example.domain.com/example](http://example.domain.com/example) .