How to Install Apache Tomcat in RHEL-8

Apache Tomcat is an open-source, lightweight, powerful and widely-used web server developed and maintained by **Apache Foundation**. It is an implementation of the **Java Servlet**, **JavaServer Pages (JSP)**, **Java Expression Language (EL)** and **Java WebSocket** technologies, and provides a pure Java HTTP server to run Java web-based applications.

□ Installing JAVA on RHEL 8

Initially, you need to install the JAVA OPENJDK package.

```
# dnf install java-11-openjdk-devel -y
```

Once the installation setup done, you can verify the installed java version on the system using the following command below:

```
# java -version
```

□ Installing Apache Tomcat on RHEL 8

Once JAVA has been installed on the system, now it’s time to download the latest version of Apache Tomcat.
Here, the officisal Apache download page and check if there is a newer version available to download.

https://tomcat.apache.org/download-90.cgi

Alternatively, you can download the latest version of Apache Tomcat using the following wget command and setup it as shown.

```
# cd /usr/local
# wget https://mirrors.estointernet.in/apache/tomcat/tomcat-9/v9.0.46/bin/apache-tomcat-9.0.46.tar.gz
# tar xf apache-tomcat-9.0.46.tar.gz
# mv apache-tomcat-* tomcat9
```

The Apache Tomcat server is now deployed in the /usr/local/tomcat directory, you can verify it.

```
# pwd tomcat9/
# ls -l tomcat9/
```

The Following is a description of each of the sub-directories in the installation directory of Apache tomcat.

- **bin - contains the executables.**
- **conf - contains configuration files.**
- **lib - store library files.**
- **log - store log files.**
- **temp - contains temporary files.**
- **webapps - stores web application files.**

□ Running Apache Tomcat Under Systemd in RHEL 8

To easily manage the Apache Tomcat daemon, you need to run it as a service under systemd. You will need to create a system user account named **tomcat.**

```
# useradd -r tomcat
```

Once the **tomcat** user is created, apply the ownership rights to the tomcat directory.

```
# chown -R tomcat:tomcat /usr/local/tomcat9
# ls -l /usr/local/tomcat9
```

Next, create a **tomcat.service** unit file under **/etc/systemd/system/** directory using any of the editors you would like.

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

Copy and paste the following configuration in the tomcat.service file.

```
[Unit]
Description=Apache Tomcat Server
After=syslog.target network.target

[Service]
Type=forking
User=tomcat
Group=tomcat

Environment=CATALINA_PID=/usr/local/tomcat9/temp/tomcat.pid
Environment=CATALINA_HOME=/usr/local/tomcat9
Environment=CATALINA_BASE=/usr/local/tomcat9
ExecStart=/usr/local/tomcat9/bin/catalina.sh start
ExecStop=/usr/local/tomcat9/bin/catalina.sh stop
```

```
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
```

Save the file and reload the systemd daemon service.

```
# systemctl daemon-reload
```

Then start the tomcat service, enable it to auto-start at system boot and check the status

```
# systemctl start tomcat.service
# systemctl enable tomcat.service
# systemctl status tomcat.service
```

Tomcat uses port 8080 and 8443 for HTTP and HTTPs requests respectively. Listening by checking the HTTP port among all listening ports on the system using netstat command.

```
# netstat -tulnp
```

If you have the firewall service running, you must add the ports 8080 and 8443 before accessing the web interface for tomcat, using firewall-cmd command as shown below:

```
# firewall-cmd --permanent --zone=public --add-port={8080,8443}/tcp
# firewall-cmd --reload
```

Now Finally, Tomcat you have Installed and started Tomcat as a service, and allowed requests to the daemon via the firewall.

□ Access Apache Tomcat Web Interface

You can test the installation by trying to access the web interface by using the URL.

```
# http://localhost:8080
OR
# http://SERVER_IP:8080
```

□ Enable HTTP Authentication for Tomcat Manager and Host Manager

To configure basic HTTP authentication in the **/usr/local/tomcat9/conf/tomcat-users.xml**

```
# vi /usr/local/tomcat9/conf/tomcat-users.xml
```

Copy and paste the following configuration within the <tomcat-users> and </tomcat-users>. This configuration adds the admin-gui and manager-gui roles to a user named “**admin**” with a password of “**redhat**”.

```
<role rolename="admin-gui,manager-gui"/>
<user username="admin" password="tomhost@80"roles="admin-gui,manager-gui"/>
```

Save the changes in the file and exit.

□ Enable Remote Access to Tomcat Manage and Host Manager

By default, access to the Manager and Host Manager apps is restricted to the localhost, the server on which Tomcat is installed and running. But you can enable remote access to a specific IP address or network e.g your **LAN**.

```
# vi /usr/local/tomcat9/webapps/manager/META-INF/context.xml
```

Then look for the following line.

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

change it to this to allow tomcat access from IP address **192.168.0.X**.

```
allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1 |192.168.56.10" />
```

You can also allow tomcat access from the local network **192.168.0.X**

```
allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1 |192.168.0.*" />
```

or allow tomcat access from any host or network.

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

Similarly, enable remote access to the Host Manager app in the file **/usr/local/tomcat9/webapps/manager/META-INF/context.xml** As shown above.

Next, restart the tomcat service to apply to recent changes.

```
# systemctl restart tomcat.service
```

- □ **To access Tomcat Manager Web Apps**

```
# http://localhost:8080/manager
Or
# http://SERVER_IP/manager
```