



SetUp Tomcat & Deploy To it

[Click Here To Enrol To Batch-5 | DevOps & Cloud DevOps](#)

Apache Tomcat, often referred to simply as Tomcat, is an open-source implementation of the Java Servlet, JavaServer Pages (JSP), Java Expression Language (EL), and Java WebSocket technologies. It is developed and maintained by the Apache Software Foundation and is widely used for deploying Java-based web applications.

Key Components of Tomcat

1. **Servlet Container:** Tomcat provides a "pure Java" HTTP web server environment in which Java code can run. It is responsible for managing the lifecycle of servlets, mapping URLs to servlets, and ensuring that web applications are secure and performant.
2. **JSP Engine:** Tomcat includes a JSP engine that enables developers to create dynamic web pages based on HTML, XML, or other document types. JSPs are essentially servlets that are dynamically compiled.
3. **Catalina:** This is Tomcat's servlet container. Catalina implements the specifications of the servlet and JSP technologies.
4. **Coyote:** This is the connector component that allows Catalina to communicate with the web. It supports HTTP 1.1 and can be configured to use SSL/TLS.
5. **Jasper:** This is Tomcat's JSP engine. Jasper parses JSP files to compile them into Java servlets for execution.
6. **Cluster:** Tomcat supports clustering, which allows for load balancing and failover across multiple instances of Tomcat.

9.0.65

1. Installing Apache Tomcat:

```
sudo su
```

This command switches to the superuser account for performing administrative tasks.

```
cd /opt
```

Navigate to the `/opt` directory where Apache Tomcat will be installed.

```
sudo wget https://archive.apache.org/dist/tomcat/tomcat-9/v9.0.65/bin/apache-tomcat-9.0.65.tar.gz
```

Download the Apache Tomcat 9 distribution archive.

```
sudo tar -xvf apache-tomcat-9.0.65.tar.gz
```

Extract the downloaded archive file.

2. Configuring Tomcat Users:

```
cd /opt/apache-tomcat-9.0.65/conf
```

Navigate to the `conf` directory within the Tomcat installation directory.

```
sudo vi tomcat-users.xml
```

Open the `tomcat-users.xml` file for editing.

Add the following line as the second-last line in the file:

```
<user username="admin" password="admin1234" roles="admin-gui, manager-gui, manager-script"/>
```

This line adds a user with the username "admin" and password "admin1234" with roles allowing access to the administrative GUI and manager scripts.

3. Creating Symbolic Links for Startup and Shutdown Scripts:

```
sudo ln -s /opt/apache-tomcat-9.0.65/bin/startup.sh /usr/bin/startTomcat
```

```
sudo ln -s /opt/apache-tomcat-9.0.65/bin/shutdown.sh /usr/bin/stopTomcat
```

Create symbolic links to the `startup.sh` and `shutdown.sh` scripts in the Tomcat bin directory, allowing easy access to start and stop Tomcat from anywhere in the terminal.

4. Configuring Access Control for Manager and Host-Manager Applications:

```
sudo vi /opt/apache-tomcat-9.0.65/webapps/manager/META-INF/context.xml
```

Open the `context.xml` file for the manager web application.

Comment out the following lines:

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

This disables the `RemoteAddrValve`, which restricts access based on IP addresses.

```
sudo vi /opt/apache-tomcat-9.0.65/webapps/host-manager/META-INF/context.xml
```

Open the `context.xml` file for the host-manager web application.

Comment out the following lines:

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

Similar to the manager application, this disables the RemoteAddrValve for host-manager as well.

5. Starting and Stopping Tomcat:

```
sudo stopTomcat
```

Stop the Tomcat server.

```
sudo startTomcat
```

Start the Tomcat server.

Following these steps, you should have successfully installed Apache Tomcat 9 on your Linux system, configured user authentication, created symbolic links for easy access to startup and shutdown scripts, and configured access control for the manager and host-manager web applications.

10.1.20

Step 1: Switch to Superuser

```
sudo su
```

This command switches the current user to the superuser, granting elevated privileges necessary for system-wide operations.

Step 2: Navigate to the Opt Directory

```
cd /opt
```

This command changes the current directory to `/opt`, where optional software packages are often installed.

Step 3: Download Apache Tomcat 10.1.20

```
sudo wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.20/bin/apache-tomcat-10.1.20.tar.gz
```

This command uses `wget` to download the Apache Tomcat 10.1.20 tarball from the official Apache Tomcat website.

Step 4: Extract the Apache Tomcat Tarball

```
sudo tar -xvf apache-tomcat-10.1.20.tar.gz
```

This command extracts the contents of the downloaded tarball into a directory named `apache-tomcat-10.1.20`.

Step 5: Navigate to the Tomcat Configuration Directory

```
cd /opt/apache-tomcat-10.1.20/conf
```

This command changes the current directory to the `conf` directory within the extracted Tomcat directory.

Step 6: Edit the Tomcat Users Configuration File

```
sudo vi tomcat-users.xml
```

This command opens the `tomcat-users.xml` file using the `vi` text editor with superuser privileges for editing.

Add the following line before the closing `</tomcat-users>` tag:

```
<user username="admin" password="admin1234" roles="admin-gui, manager-gui, manager-script"/>
```

This line adds a user with the username "admin" and password "admin1234" to the Tomcat user database with roles that grant access to the administrative interfaces.

Step 7: Create Symbolic Links for Start and Stop Commands

```
sudo ln -s /opt/apache-tomcat-10.1.20/bin/startup.sh /usr/bin/startTomcat
sudo ln -s /opt/apache-tomcat-10.1.20/bin/shutdown.sh /usr/bin/stopTomcat
```

These commands create symbolic links named `startTomcat` and `stopTomcat` in the `/usr/bin` directory, pointing to the respective startup and shutdown scripts in the Tomcat installation directory. This enables easy execution of start and stop commands from any directory in the terminal.

Step 8: Configure Tomcat Manager Context

```
sudo vi /opt/apache-tomcat-10.1.20/webapps/manager/META-INF/context.xml
```

This command opens the `context.xml` file for editing, which configures the context for the Tomcat Manager web application.

Comment out the following lines by adding `<!--` at the beginning and `-->` at the end:

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

This disables the remote address valve, allowing access to the Tomcat Manager interface from any IP address.

Step 9: Configure Host Manager Context

```
sudo vi /opt/apache-tomcat-10.1.20/webapps/host-manager/META-INF/context.xml
```

This command opens the `context.xml` file for editing, which configures the context for the Tomcat Host Manager web application.

Comment out the following lines by adding `<!--` at the beginning and `-->` at the end:

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

This disables the remote address valve, allowing access to the Host Manager interface from any IP address.

Step 10: Start and Stop Tomcat

```
sudo stopTomcat
sudo startTomcat
```

These commands utilize the symbolic links created earlier to easily start and stop the Tomcat server. `stopTomcat` shuts down the Tomcat server, while `startTomcat` starts it up again.

Deploy to Tomcat

Repo--> <https://github.com/jaiswaladi2468/maven-tomcat-sample.git>