

Deploying a Web Application using Jenkins and Tomcat

1. Installing Jenkins on Server 1

- Install Jenkins, Git, and Java on Server 1.
- Store all project files in a Git repository.
- In Jenkins, select the Freestyle Project.
- Install the required Jenkins plugin: Deploy to Container.

2. Installing Tomcat on Server 2

- Install Tomcat on Server 2.
- Enable Maven in Jenkins by adding the latest version.

Tomcat Installation Steps:

```
$ wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.39/bin/apache-tomcat-10.1.39.tar.gz
$ tar -zxvf apache-tomcat-10.1.39.tar.gz
$ cd apache-tomcat-10.1.39/bin
$ ./startup.sh
```

(Optional) Changing Tomcat Port:

If needed, change the default Tomcat port (8080) to 8090:

```
$ cd apache-tomcat-10.1.39/conf
$ vi server.xml
# Change the port number to 8090
```

Restart Tomcat:

```
$ ./shutdown.sh
$ ./startup.sh
```

3. Configuring Tomcat Manager for Deployment

Host Manager & Manager Authorization Fix:

By default, the Tomcat Manager is only accessible from localhost.

To modify this restriction:

1. Edit the context.xml file:

```
$ vi apache-tomcat-10.1.39/webapps/manager/META-INF/context.xml
```

```
$ vi apache-tomcat-10.1.39/webapps/host-manager/META-INF/context.xml
```

2. Comment out the following line:

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

Updating Tomcat User Credentials:

To allow Jenkins to deploy the application, update tomcat-users.xml:

```
$ vi apache-tomcat-10.1.39/conf/tomcat-users.xml
```

- Set up a deployer user with the necessary roles.
- Use the same credentials in Jenkins.

4. Deploying the Web Application via Jenkins

- Configure Jenkins to deploy the WAR file to Tomcat.
- After a successful build, access the deployed application at:

Example:

<http://54.197.83.63:8080/my-project-1.0-SNAPSHOT/>

(Replace with your actual server IP and project name.)