APACHE TOMCAT

* It is an application server which executes on java servlets. is can be used as application server along with its web server (or) you can user apache web server. both are developed by same company.
* You deploy an war file in tomcat dir, it automatically pulls it to deploy an application.
* Before installing tomcat, you have to install java(jdk) and set java homepath correctly pointed to jdk. Only after this tomcat will start.
* Install tomcat from outside repo. wget tomcat link.
* Extract the gz file and go to tomcat directory.
* There are several directories and files in that directory.
* **webapps** = we deploy war file in this directory.
* **bin** = start and stop scripts are there in this directory.
* **conf** = we can change port of tomcat in server.xml file in this dir.
* To change port, go to server.xml file, change in **connector port=8080**(default).
* To start tomcat server = Go to bin, execute the start script = **sh startup.sh**
* To stop - execute the stop script = **sh shutdown.sh**.
* open your browser and type your server ip with tomcat port. it will shows the default tomcat homepage…..ip:8080 (default).
* For the first time, the page won't display. You have to add an rule in iptables (or) flush the iptables to display the tomcat homepage.
* To configure tomcat as a service. Create a file called **tomcat** in **/etc/init.d** dir and paste this.

**#!/bin/sh**

**# chkconfig: 2345 80 20**

**# Description: Tomcat Start/Shutdown script**

**export JAVA\_HOME=write your java home path**

**case $1 in**

**start)**

**cd /opt/tomcat/bin/**

**./startup.sh**

**;;**

**stop)**

**cd /opt/tomcat/bin/**

**./shutdown.sh**

**;;**

**restart)**

**cd /opt/tomcat/bin/**

**./shutdown.sh**

**cd /opt/tomcat/bin/**

**./startup.sh**

**;;**

**esac**

**exit 0**

* Change this file according to your requiremewnts like tomcat homepath, java homepath.
* Configure **tomcat-users.xml** file to create users to login to manager app from tomcat console to deploy war and jar files.
* Go to **/conf/tomcat-users.xml** file, remove all the lines up to license line and add the following lines.

-->

**<tomcat-users>**

**<user username="admin" password="red123" roles="admin-gui,manager-gui"/>**

**</tomcat-users>**

* Now, you can login to tomcat manager console with these user and password.
* By default, you can login to tomcat manager console only in local pc not in remote host.
* To do this we have to create a **manager.xml** file in **/conf/Catalina/localhost** dir and paste in some code to access manager console.

**<Context privileged="true" antiResourceLocking="false"**

**docBase="${catalina.home}/webapps/manager">**

**<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="^.\*$" />**

**</Context>**

* Now you can login to tomcat manager app console from remote host.

**VIRTUAL HOSTING**

* You can create virtual hosts in tomcat by creating hosts line in **server.xml** file and create new webapps directory for the new virtual hosts.
* You have to add a host line for each virtual host in server.xml file.
* Add dns entries for each virtual host in **/etc/hosts** file. Now, you can see your tomcat page with your domain name.
* By default, virtual hosts don’t have manager access, you have to copy the manager.xml file to their respective dir to access manager console.
* To create virtual hosts. Go to **tomcat/conf/server.xml** file, copy the default host line, paste and edit the lines according to your requirement like below and save the file.

**<Host name="www.kanth666.com" appBase="kanth\_webapps"**

**unpackWARs="true" autoDeploy="true"/>**

* Here, I have already created kanth\_webapps directory and pasted it in tomcat directory.
* Go to /etc/hosts file to create dns entries.

**Server-ip domain name**

**Server-ip 2nd domain name**

* Stop and start the tomcat server and type this domain in your browser to access tomcat.
* Whenever your create a new virtual host, a directory named as your domain will be created in **Conf/Catalina** path by default.
* To access manager-gui. Go to **conf/Catalina/localhost**, copy the **manager.xml** file to **conf/catalina/domain-name** dir.

**REVERSE PROXY FOR TOMCAT**

* When you configured an application with tomcat, by default it will give us an url along with tomcat port like ip:8080. But end users, can’t access with the server port and all those.
* For this we create virtual hosts with apache and redirect the application server with apache as normal domain. So, end users can access application without port and other urls.
* For this, we have to configure apache virtual hosts for tomcat applications and you have to uncomment the reverse proxy lines in httpd.conf file.

**LoadModule proxy\_module modules/mod\_proxy.so**

**LoadModule proxy\_http\_module modules/mod\_proxy\_http.so**

* Create virtual hosts with your tomcat server and dir files in httpd.conf.

**<VirtualHost \*:80>**

**ServerName example.com**

**ProxyRequests On**

**ProxyPass / http://localhost:8080/demo1/**

**ProxyPassReverse / http://localhost:8080/demo1/**

**<Location "/application-dir">**

**Order allow,deny**

**Allow from all**

**</Location>**

**</VirtualHost>**

* Edit the hosts file and resolve your ip with your domain name. restart the apache server and browser the domain, you will see your tomcat application.
* Copy the these virtual hosts for all your tomcat applications and make an entry in hosts file.