**Tomcat**

**Introduction**

Tomcat or Apache Tomcat is a light weight, open source web container used to deploy and running java-based web applications, developed by Apache Software Foundation (ASF).

**What is Webserver?**

A Web server is a program that uses HTTP (Hypertext Transfer Protocol) protocol to serves web content (HTML and static content) to users.

**What is Application Server?**

An application server is a software framework that provides both facilities to create web applications and a server environment to run them. Application Server Frameworks contains a comprehensive service layer model.

**Tomcat installation**

Tomcat software will available as a zip file. So, we can unzip it and use as follows.

Step 1: Create the folder called **Tomcat** in any directory and Unzip apache software into that folder.

Step 2: Open the command prompt and execute the below commands.

>cd C:\DevOps\Tomcat\apache-tomcat-7.0.79\bin

>startup.bat (OR) > catalina.bat start ---> Starting the Tomcat server.

**MAC/Linux:**

startup.sh (OR) > catalina.sh start ---> Starting the Tomcat server.

Once you execute the startup.bat, you will see below screens.

Tomcat server is started successfully.

Use the below URL and open in browser.

http://localhost:8080

**Note:** Tomcat by default will run in 8080 port number.

**Tomcat Architecture and Configuration**

Tomcat installation provides these directories:

**bin:** for Tomcat's binaries and startup scripts.

**conf:** global configuration applicable to all the webapps. The default installation provides:

One Policy File: catalina.policy for specifying security policy.

Two Properties Files: catalina.properties and logging.properties,

Four Configuration XML Files: server.xml (Tomcat main configuration file), web.xml (global

web application deployment descriptors), context.xml (global Tomcat-specific configuration options)

and tomcat-users.xml (a database of user, password and role for authentication and access control).

The conf also contain a sub-directory for each engine, e.g., Catalina, which in turn contains a subsub-

directory for each of its hosts, e.g., localhost. You can place the host-specific context

information (similar to context.xml, but named as webapp.xml for each webapp under the host).

**lib:** Keeps the JAR-file that are available to all webapps. The default installation include servletapi.

jar (Servlet), jasper.jar (JSP) and jasper-el.jar (EL). You may also keep the JAR files of external

package here, such as MySQL JDBC driver (mysql-connector-java-5.1.{xx}-bin.jar) and JSTL (jstl.jar

and standard.jar).

**logs:** contains the engine logfile Catalina.{yyyy-mm-dd}.log, host logfile localhost.{yyyy-mm-dd}.log,

and other application logfiles such as manger and host-manager. The access log (created by the

AccessLogValve) is also kept here.

**webapps:** the default appBase - web applications base directory of the host localhost.

**work:** contains the translated servlet source files and classes of JSP/JSF. Organized in hierarchy of engine name (Catalina), host name (localhost), webapp name, followed by the Java classes

package structure.

**temp:** temporary files.

**Install Tomcat in Linux**

#cp /Users/Lenovo/Desktop/DevOps/Softwares/Tomcat/apache-tomcat-7.0.78-windows-x64.zip/opt/

#unzip apache-tomcat-7.0.78-windows-x64.zip

#cd /opt/apache-tomcat-7.0.78/conf

# vim tomcat-users.xml

**8080** is the default tomcat port number. To access tomcat from browser, then use http://hostname:8080

**How to change the port number in Tomcat?**

Go to the **conf** directory and open the **server.xml** and you will find below lines.

**<Connector port="8080" protocol="HTTP/1.1"**

**connectionTimeout="20000"**

**redirectPort="8443" />**

Replace the 8080 with any number.

**Note:** After doing server configurations, we need to restart tomcat to affect the changes.

**How to stop the tomcat server?**

>cd C:\DevOps\Apache\apache-tomcat-7.0.79\bin

>shutdown.bat (OR) > catalina.bat stop

**MAC/Linux:**

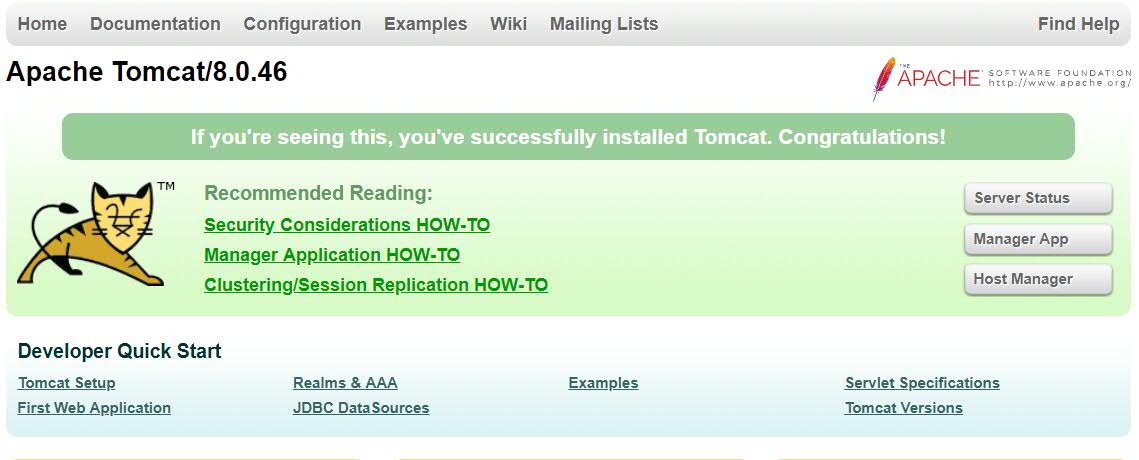
shutdown.sh (OR) > catalina.sh stop

**Working with tomcat admin Console**

Tomcat provides a web based administration console which can be started via the following link:

<http://localhost:8080>

Before this, tomcat server should be started, for this got to tomcat bin location and start tomcat. Then admin console looks like below.



**Adding users to tomcat:**

# The tomcat-users.xml file should look something like this:

<tomcat-users>

<!--

<role rolename="tomcat"/>

<role rolename="role1"/>

<user username="tomcat" password="tomcat" roles="tomcat" />

<user username="role1" password="tomcat" roles="role1" />

<user username="both" password="tomcat" roles="tomcat,role1" />

-->

<!-- **Note:** While copy pasting above line you have to remove double quotes and type manually. -->

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

**<user username="chaitanya" password="passw0rd" roles="manager-gui,admin-gui"/>**

</tomcat-users>

**Deployments in Tomcat:**

Deploying the application in Tomcat server is nothing but, putting war file in Tomcat’s webapps.

directly.

For every application deployment need to restart the tomcat server.

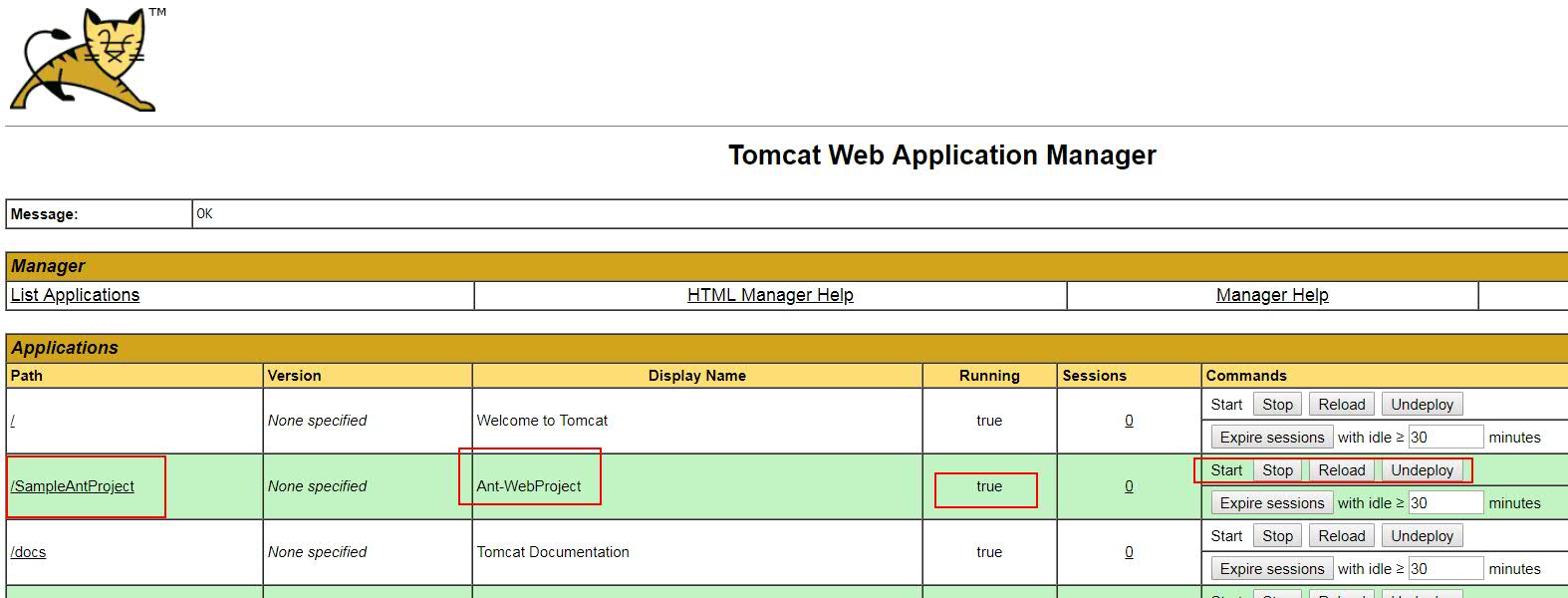
Once the server restart, tomcat starts it will unpack the war and make the application

available.

**Managing deployments from admin console:**

From admin console, choose **Manager App** option to do deployments. Then give the credentials.

Then you can see the screen like below.



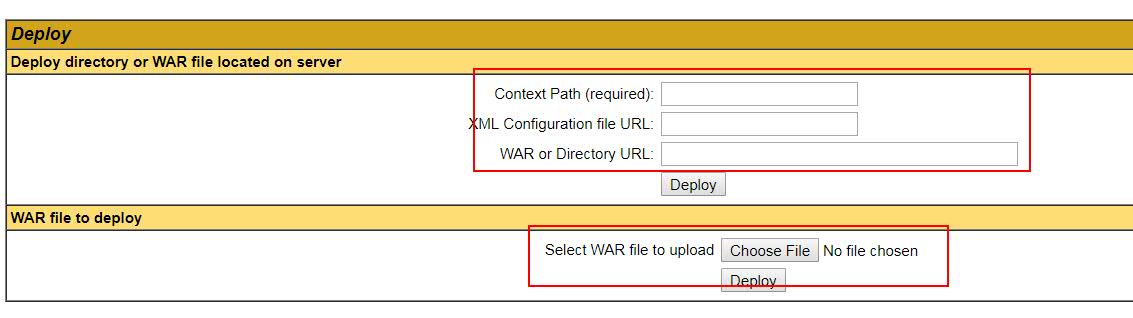
Start is to start app.

Stop is to stop the application.

Reload is to reload the application when we have done hot fixes.

Undeploy is to undeploy the application from server.

**To deploy an application from admin console:**



You can choose any option to deploy. For example, select the was file and click on Deploy.

**Server Status & Manager App**

To access the server status application and Manager app, tomcat 7 onwards below roles as

introduced.

**manager-gui** - allows access to the HTML GUI and the status pages

**manager-script** - allows access to the text interface and the status pages

**manager-jmx** - allows access to the JMX proxy and the status pages

**manager-status** - allows access to the status pages only

If you provide add below line in tomcat-users.xml, we will get the access for below apps(Buttons)

<user username="admin" password="passw0rd" roles="manager-gui"/>

**http://localhost:8083/manager/status**

**http://localhost:8083/manager/html**

If you provide add below line in tomcat-users.xml, we will get the access for below apps(Buttons)

<user username="admin" password="passw0rd" roles="admin-gui"/>

**http://localhost:8083/host-manager/html**

**Note:** Users with the manager-gui role should not be granted either the manager-script or managerjmx

roles.

**Host Manager**

To access the host manager application, tomcat 7 onwards below roles as introduced.

admin-gui - allows access to the HTML GUI

admin-script - allows access to the text interface

**Note:** Users with the admin-gui role should not be granted the admin-script role.

**Run the server as follows.**

#cd /opt/apache-tomcat-7.0.78/bin

# sh catalina.sh start

(OR)

#sh-3.2# ./startup.sh

# ./startup.sh: Permission denied

# chmod -R 777 \*.sh

# ./startup.sh

Using CATALINA\_BASE: /opt/apache-tomcat-7.0.78

Using CATALINA\_HOME: /opt/apache-tomcat-7.0.78

Using CATALINA\_TMPDIR: /opt/apache-tomcat-7.0.78/temp

Using JRE\_HOME: /Library/Java/JavaVirtualMachines/jdk1.8.0\_121.jdk/Contents/Home

Using CLASSPATH: /opt/apache-tomcat-7.0.78/bin/bootstrap.jar:/opt/apache-tomcat-

7.0.78/bin/tomcat-juli.jar

Tomcat started.

**Stop the server**

# ./shutdown.sh (OR) sh catalina.sh stop

**Important points about Tomcat**

**WebServer examples:**

Apache HTTP server

IBM HTTP server(IHS)

Nginx -->EngineX server

Internet Information Server(IIS)-- for .net

**Application Servers examples:**

Apache Tomcat--By Apache

JBoss/WildFly-->By Redhat

WAS--->By IBM

WLP(Websphere Liberty Profile)--->By IBM

Glassfish--->By Oracle, former BEA

WebLogic--->By Oracle, former BEA

* /bin folder contains all binary files like .sh and .bat files.bat files are for windows and .sh files are for Linux
* /conf folder contains all configurations.
* startup.bat/startup.sh (or) catalina.sh start/catalina.bat start: are the file to start tomcat server
* shutdown.bat/shutdown.sh (or) catalina.sh stop/ catalina.bat stop: are the files to stop tomcat server
* webapps is the folder where we deploy applications in tomcat.
* After putting the app in webapps, it will copy to work/ with extracted files
* tomcat-users.xml is the file where you add new users.

To change the http port number in tomcat, then go to settings.xml in conf/ directory and change the below line:

<Connector port="8083" protocol="HTTP/1.1"

* **In tomcat console,**
* **Reload** is required when we do a **hot fix** for java .class files or .jsp files. Reload is not required for static files. Html, css, js files are called static files.
* Hot fix means moving/updating a .class or html or .jsp files manually to the project under webpps folder.
* Tomcat will allow only .jar and .war files to deploy. It will not allow .ear files to deploy.
* **WAR file to deploy in Tomcat console is the another way to deploy .war in tomcat.**

In 2 ways we can deploy .war in tomcat.

1)Manually copying project .war to webaps folder

2)From console, by choosing WAR file to deploy option

* 8080 is Wildfly default http port number.
* 8443 is Wildfly default https port number.
* 8080 is the default tomcat console port number.
* From console, stop is the option to stop the application,

start is the option to start the application

Undeploy is the option to undeploy the application.

Deploy is the option is to deploy the project in tomcat server.

**Steps to install Tomcat on Linux RedHat:**

**yum install tomcat tomcat-webapps tomcat-admin-webapps**

**systemctl start tomcat**

**systemctl enable tomcat**

**vi /usr/share/tomcat/conf/tomcat-users.xml**

**Add below text:**

**<user username="admin" password="passw0rd" roles="admin-gui,manager-gui,manager-script"/>**

**<user username="admin1" password="passw0rd" roles="admin-gui"/>**

**<user username="admin2" password="passw0rd" roles="admin-gui,manager-gui"/>**

**Then change the port number in server.xml if required.**

**vi /usr/share/tomcat/conf/server.xml**

**systemctl restart tomcat**