**07-DevOps - 17-OCT-24**

**Webservers**

=====================================================================

Note – We can deploy multiple application in same server but with different war files if we use same war file then it will override another application.

- The process of keeping war file into the server is called as deployment.

- Server is a software which is used to run web applications.

- To run any web applications server is mandatory.

- Users can access our web application by sending request to server.

- Users will use client s/w to send request to server

Ex: Browser (google chrome, Firefox, Edge)

- Server is responsible to handle user requests & response.

- We have several servers in the market to run our web applications.

**1) Tomcat----------🡪**

JAVA Applications

**2) JBoss----------🡪**

**3) Glassfish----------🡪**

**4) WebLogic----------🡪**

**5) WebSphere----------🡪**

**6) IIS etc.. ----------🡪 For .Net Applications**

Note: To run web application, server is mandatory.

- **The process of executing web application by using server is called as Deployment.**

=====================================================================

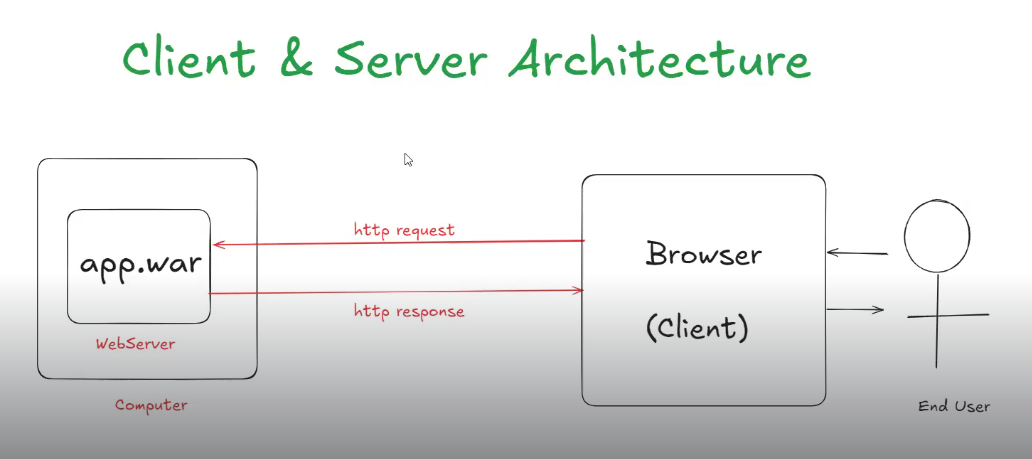
**QUE - What is Build & Deployment ?**

* As a devops engineer We are responsible for build and deployment.

=====================================================================

(Maven will build the code ) **Build = compile + test + package**

**Deployment = Execution in server**



* Every application will use client server architecture.
* Client is nothing but the browser.
* We are end users want to access some other location web-applications
* Ex.- Naukri website we don’t know in which computer naukari server is available but by using browser we are able to access the application
* We will send the request and the naukaris server will send a response to us.

**Tomcat Server**

=====================================================================

- Tomcat is free & open source s/w

- Tomcat is a web server developed by Apache Organization.

- Tomcat server developed using Java language

**Note: To run tomcat server, java should be installed.**

- Tomcat server is used to run Java Web Applications.

- Tomcat supports multiple operating systems.

- Tomcat server runs on 8080 port number (we can change it).

=====================================================================

**Tomcat Setup In Linux**

=====================================================================

- Create Linux VM using Amazon Linux AMI in AWS Cloud

- Connect to Linux VM using ssh client

Steps-

* Open keypairs file location from your system.
* Open git client from path
* And paste ssh id

- Install maven software

ex: sudo yum install maven

Note: When we install maven, java s/w also gets installed automatically.

mvn -version

java -version

- We can download tomcat software from its offical website

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

- Download tomcat server zip file

Ex: wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.96/bin/apache-tomcat-9.0.96.zip

- Extract zip file

Ex: unzip <zip-file-name>

- Go inside tomcat directory and see directory structure

cd <tomcat-dir>

ll

=====================================================================

**Tomcat Server directory structure**

=====================================================================

* Will work on bin , conf and webapps

1) **bin :** It contains files to start & stop server (windows : bat , Linux : sh)

windows: startup.bat & shutdown.bat

Linux : statup.sh & shutdown.sh

2) **conf :** It contains **tomcat server configuration files.**

ex: server.xml, tomcat-users.xml

I -3) **webapps :** It is called as deployment folder. We will keep war files here for execution.

4) lib : It contains libraries required for server (jars).

5) temp : Temporary files will be created here (We can delete them).

6) logs : Server log messages will be stored here.

* Use maven notes

=====================================================================

**Web app deployment process Using - CLI**

=====================================================================

## Step-1 :: Create Maven web application in ec2-user home directory ##

mvn archetype:generate -DgroupId=in.ashokit -DartifactId=my-web-app -DarchetypeArtifactId=maven-archetype-webapp -DarchetypeVersion=1.4 -DinteractiveMode=false

cd <project-dir>

ls -l

## Step-2 :: Build project using maven goal ##

mvn clean package

* It will compile our project and create war file in target folder

ls -l

ls -l target

cd ..

## Step-3 :: Copy application war file into tomact-server webapps folder for execution ##

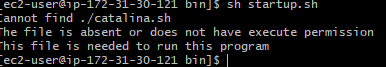
cp <app-war-file-path> <tomcat-webapps-dir-path>

Ex: cp my-web-app/target/my-web-app.war apache-tomcat-9.0.96/webapps/my-web-app.war

## Step-4 :: Start tomcat server from bin directory ##

cd apache-dir/bin

ls –l



chmod 777 catalina.sh

chmod 777 startup.sh

sh startup.sh

## Step-5 :: Enable Tomcat server port number 8080 in Ec2 VM Security Group Inbound Rules.

## Step-6 :: Access our web application using browser

URL : http://public-ip:8080/my-web-app/

## 18-OCT-24

=====================================================================

**How to change tomcat server port number ?**

=====================================================================

- Tomcat server default port is 8080

- We can change this port number using "**server.xml**" file

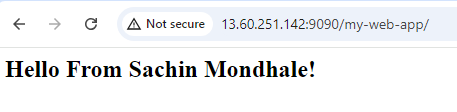
File location : tomcat-dir/conf/server.xml

- After changing the port number stop and start tomcat server.

- Enable new port number in EC2 VM Security Group inbound rules.

Server URL : http://public-ip:port-num/

Que - How to change display message in Web-browser?

* 

Ans – cd webapps 🡪 cd my-web-app 🡪 vi index.jsp

=====================================================================

Que –if latest code is not deployed to the server then ?

Ans – edit index.jsp according to your need

* Go to the my-web-app folder and execute mvn clean package
* Copy war file to the apache-tomcat/webapps
* Go to apache-tomcat-9.0.96/bin and run sh shutdown.sh
* And restart the tomcat server sh startup.sh
* Go to the AWS copy public ip and run
* latest code deployed to the server.

If you have created a new VM in aws for tomcat then do below setting for running server.

=====================================================================

**Web app deployment process Using - GUI**

=====================================================================

* 
* Remove everything and add allow=”.\*”
* For deploying applications will use tomcat gui

=====================================================================

**Enable Tomcat Admin Console Access**

=====================================================================

- By default tomcat server admin console is accessible in the same machine in which tomcat server is running.

Note: If you wish to modify this restriction, you'll need to edit the Host Manager's context.xml file.

- File Location : <tomcat-folder>/webapps/manager/META-INF/context.xml

- In Manager context.xml file, change <Valve> section like below (allow attribute value changed)

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

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

</Context>

=====================================================================

**Add tomact users in "<tomact>/conf/tomact-users.xml" file like below**

=====================================================================

<role rolename="manager-gui" />

<role rolename="manager-script" />

<role rolename="admin-gui" />

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

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

- add above script second last line of the users.xml file

- file location cd apache-tomcat-9.0.96/conf tomcat-users.xml

- Once above changes completed, then start tomcat server and access tomcat admin console in browser.

Server URL : http://public-ip:port-num/

=====================================================================

**Tomcat - Summary**

=====================================================================

1) What is client-server architecture

2) What is Tomcat

3) Tomcat Setup in Linux

4) Tomcat Server directory structure

5) War file deployment

6) Enable Tomcat Admin Console

7) Changing Tomcat Server Port Number