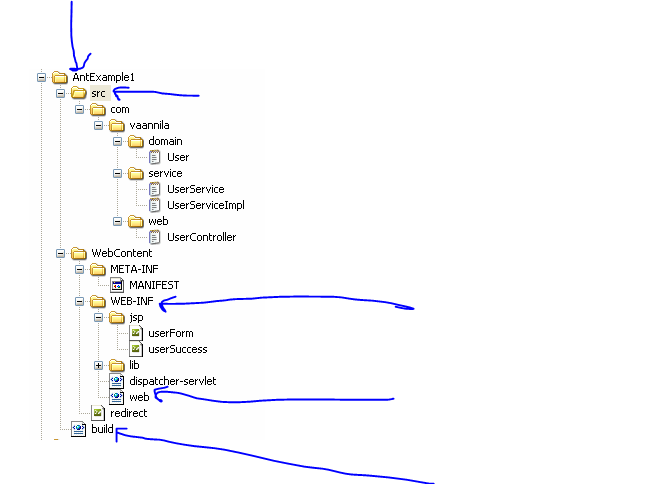
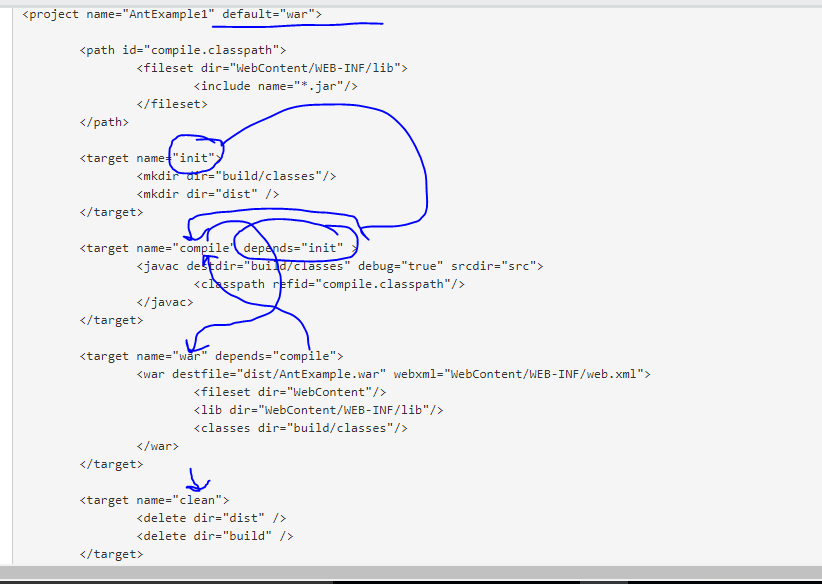
**LAB:-**

Login to the ec2 with your own user without directly.

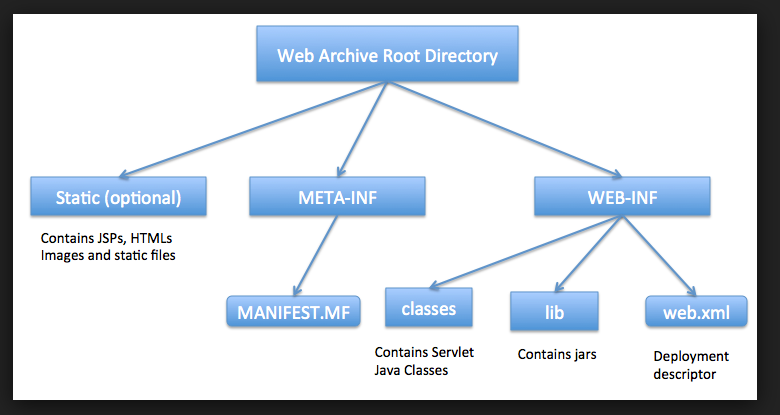
**Source Code structure(Soruce Code Mangememnt Tool(SVN/CVS/Github)**



**Build.xml**



**WAR file Structure**



**Difference between EAR and WAR ?**

**Flow of request**

**WAR**

**browser---->localhost:8080/filename------------->tomcat/webapps/<filename>---->web.xml---->welcomfile**

**EAR**

**browser---->localhost:8080/filename------------->jboss/deployments-->application.xml--->webmodule----->contextroot----**

**WAR---->web.xml---->welcomfile**

**Apache Tomcat**

Java based server(WAR)

Container: Web Container(Servlet Container)

Port number: http(8080),https(8443),ajp(advance jserv protocol-8009)

Important file: server.xml

Important log file: catalina.out

Pre-req: JDK/JRE(java development kit/Java Runtime Environment)

Version: 9.x

On the live: 6/7/8/9

License type: OpenSource

Tomcat Directory Structure

**ApacheTomcat-xx(CATALINA\_HOME/CATALINA\_BASE)(Tomcat home directory)**

→bin(startup.sh,shutdown.sh,setenv.sh,catalina.sh)

→conf(server.xml,context.xml,tomcat-users.xml,logging.properties)

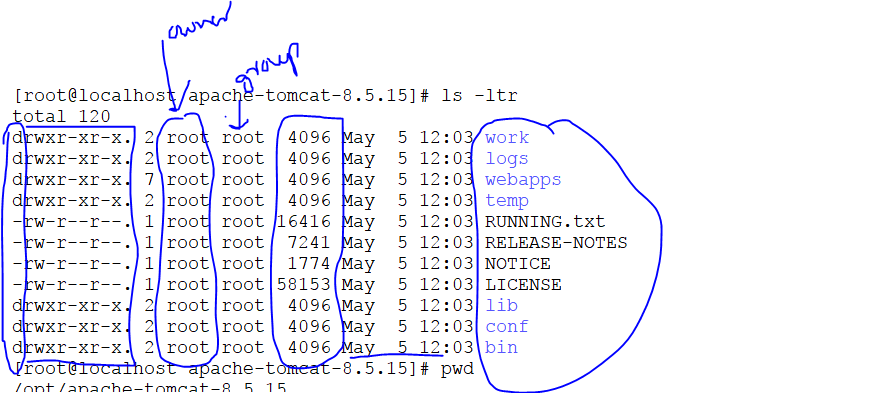
→lib(\*.jar(servlet-api.jar,jasper.jar))

→webapps(Deployment Directory(\*.war))

→work(temporary directory and it can be deleted)

→tmp(temporary directory)

→logs(catalina.out)



**JAVA Installation**

JDK/JRE Installation

JAVA ( SUN)-----→oracle java(jrocket-weblogic)

--→IBM (IBM java—IBM Products)

→Redhat →openjdk

→Apple →Apple Java

We use mostly sun java in the projects. SUN java(oracle).

To know which java or version we are using: **java –version**

JAVA Instalaltion Ways→RPM(you need to download the package from outside url(internet)) ,YUM(baseurl),zip/tar.gz(you need to download the package).

To install java with yum

yum install java

yum--→/etc/yum.repos.d/\*.repo -→checks for the package(baseurl).

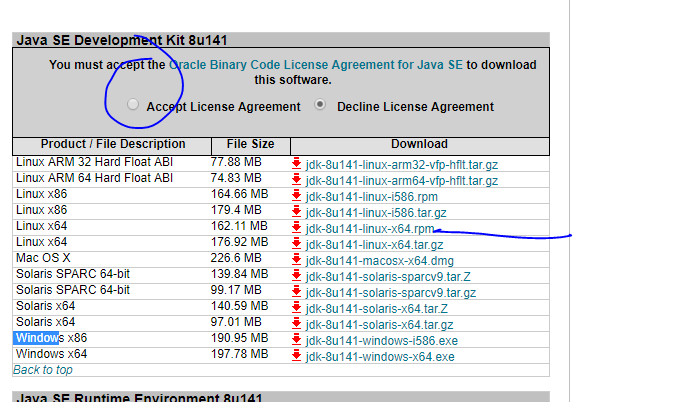
To check installed or not.

java –version

rpm installation

Download the package from repository url.

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>



Once you downloaded the jdk..xxx.rpm package,copy this file to remote unix server(filezilla/winscp).

wget --no-cookies --no-check-certificate --header "Cookie: gpw\_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u171-b11/512cd62ec5174c3487ac17c61aaa89e8/jdk-8u171-linux-x64.rpm"

Goto the rpm package residing location and execute the follow command to do installation.

rpm -ivh package.rpm

p

[root@localhost ot]# **rpm -ivh jdk-8u141-linux-x64.rpmI= install v=verbose h=hash value**

**[root@localhost opt]# java -version**

**java version "1.8.0\_141"**

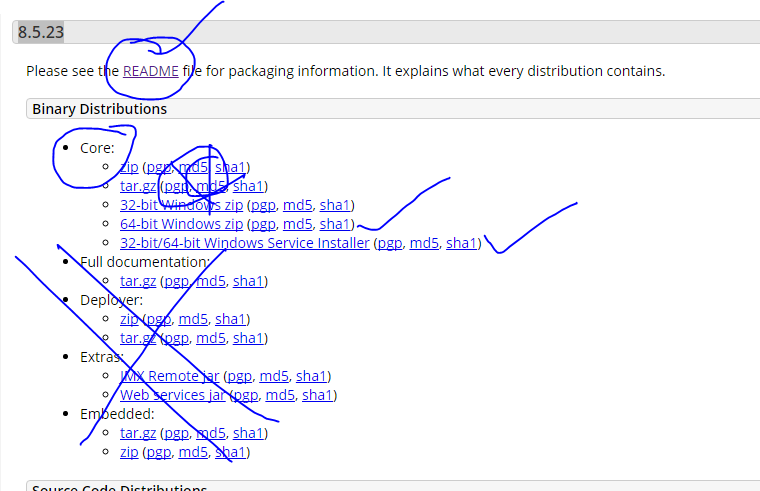
**Java(TM) SE Runtime Environment (build 1.8.0\_141-b15)**

**Java HotSpot(TM) 64-Bit Server VM (build 25.141-b15, mixed mode)**

**[root@localhost opt]#**

**Tomcat Installation.**

[**https://tomcat.apache.org/download-80.cgi**](https://tomcat.apache.org/download-80.cgi)

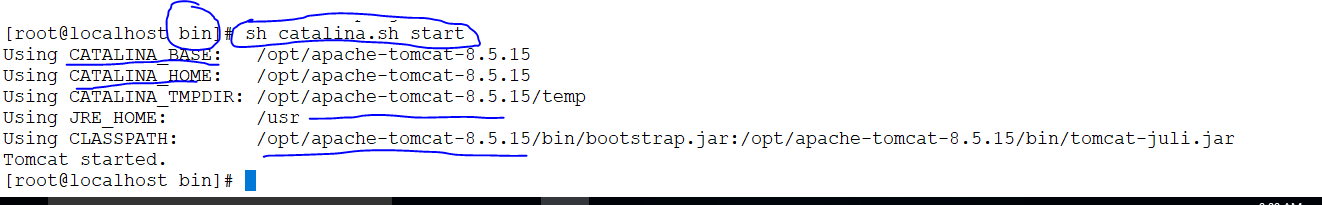


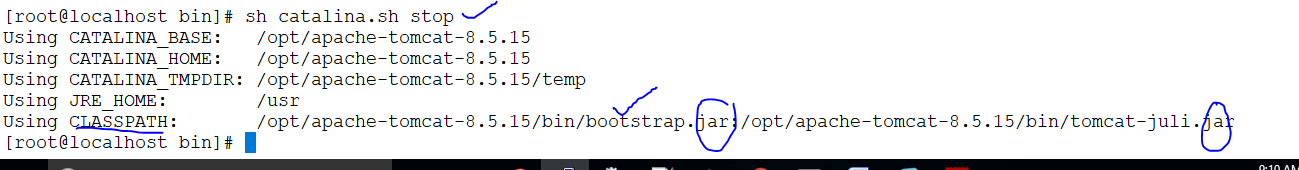
Extract the zip file

unzip apache-tomcat-xxx.zip

mv apache-tomcat tomcat ( rename)

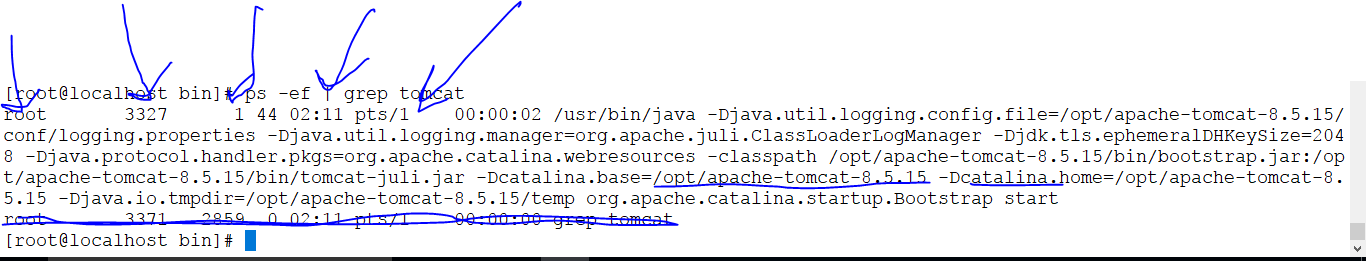
chmod -R 700 tomcat

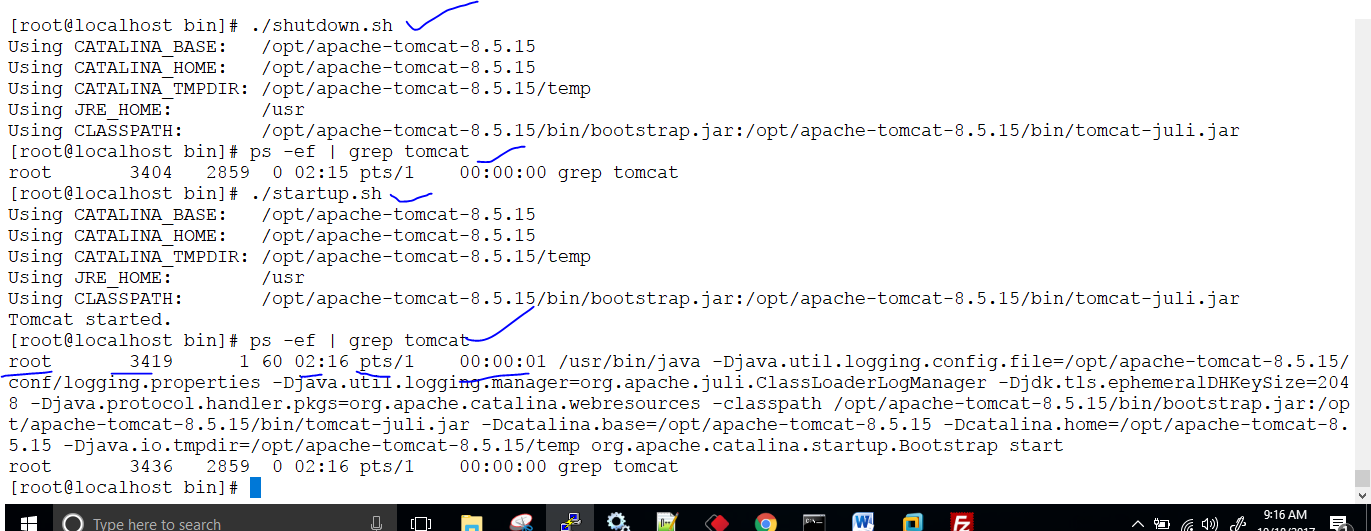




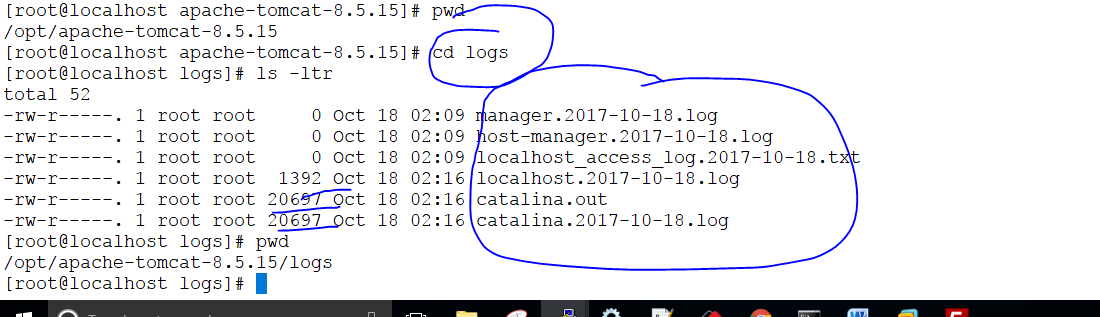
Check Tomcat process running or not

(Process owner and Process id , ppid,time and terminal)





Log Files



More filename

Less filename

Head filename

Tail filename

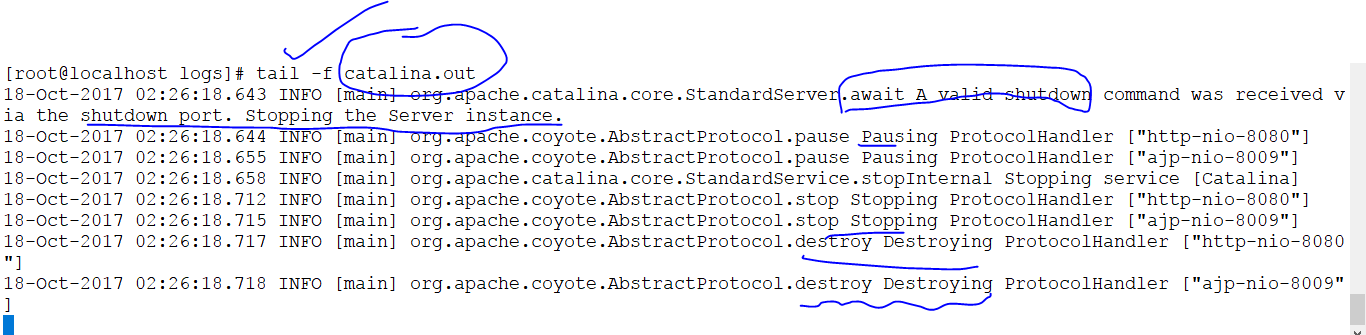
Vi filename

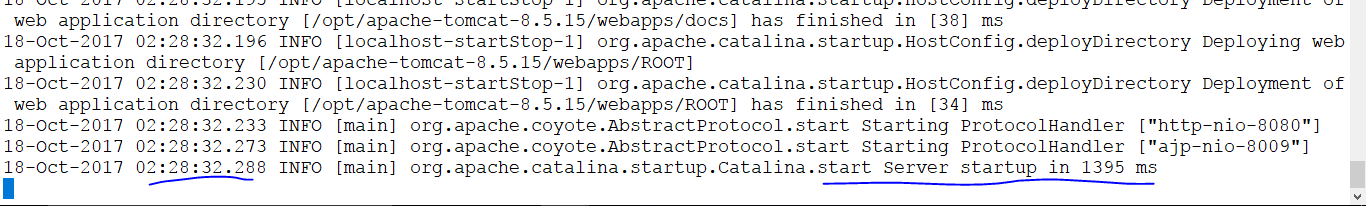
Head -100 filename

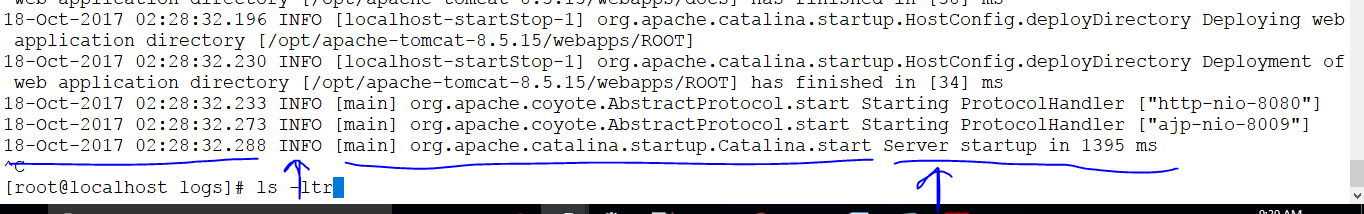
Tail -500 filename

Tail -50f filanem( last 50 lines + updated next lines )

Nullyfy a file (>filename)







Check Types of Log levels in Tomcat ?

INFO

SEVERE

WARN

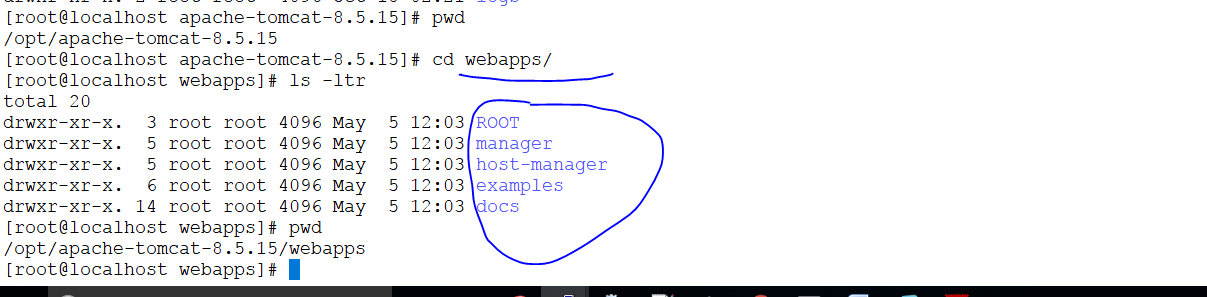
FINER

FINEST

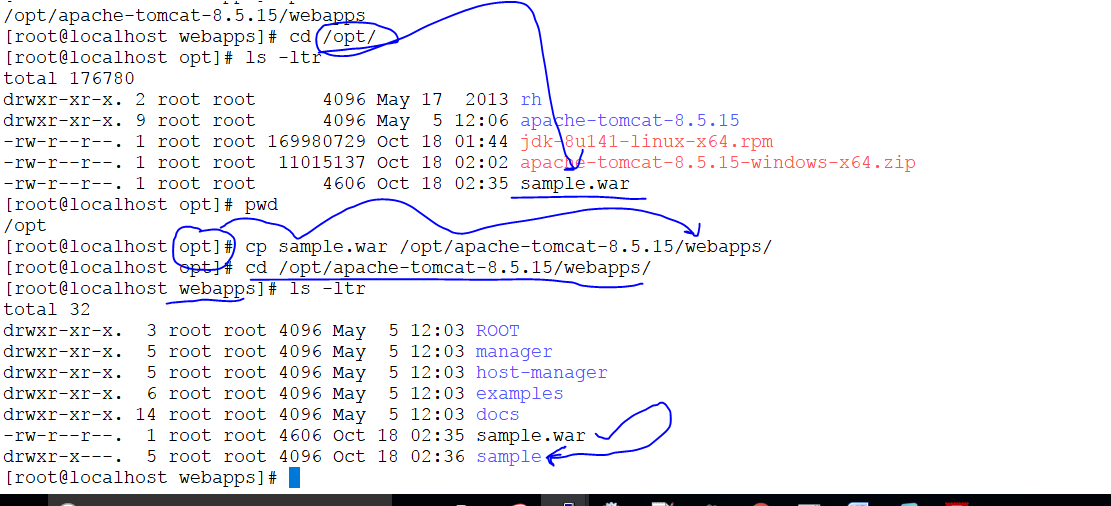
FINE

**About The WEBAPPS**

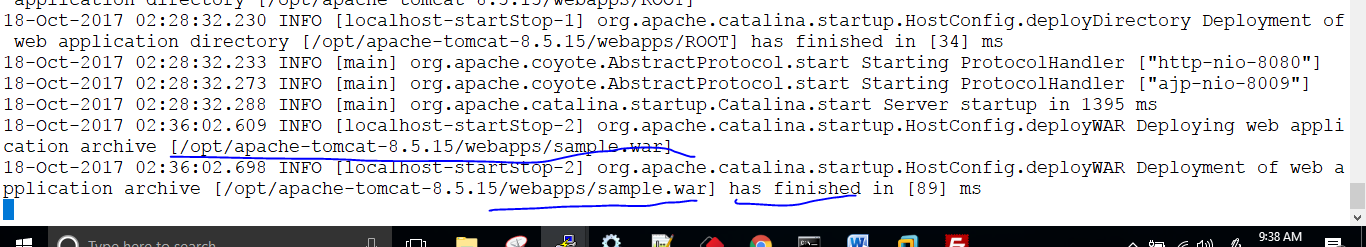
We do deployment of war files or exploded directories.



Deployment of Sample.war

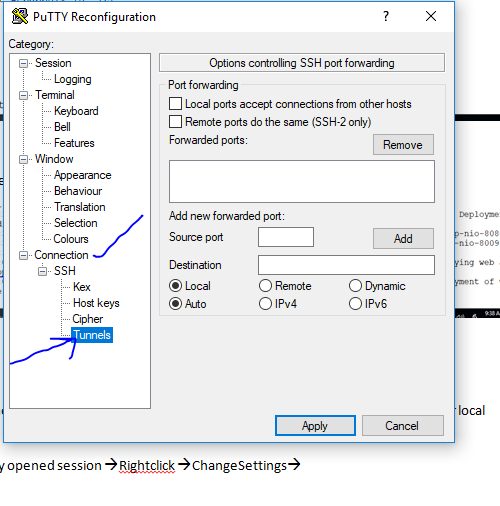


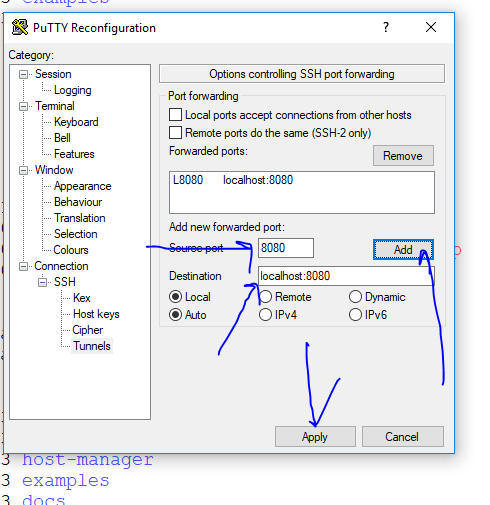
After deployment of the war file check log file and observe last lines of the file.



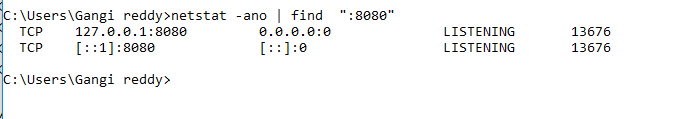
Access the remote application(Note: When remote server doesn’t have **public ip**) into your local system.

On Top of putty opened session →Rightclick →ChangeSettings→





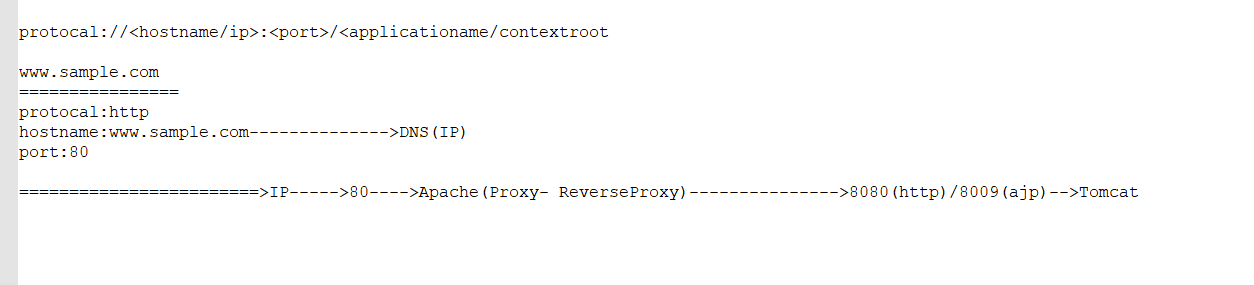
After enabling the tunneling check its active or not ( in your local system)



Access the page.



How request forwarded to Tomcat from apache(Without port number).



http://localhost:8080/sample---------->Directly Tomcat

www.sample.com/---->Via Apache To Tomcat(Benefit: No port number and security)

protocal://<hostname/ip>:<port>/<applicationame/contextroot

www.sample.com

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

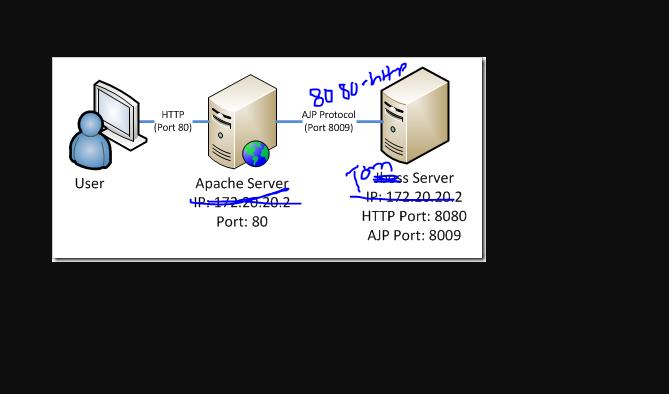
protocal:http

hostname:www.sample.com-------------->DNS(IP)

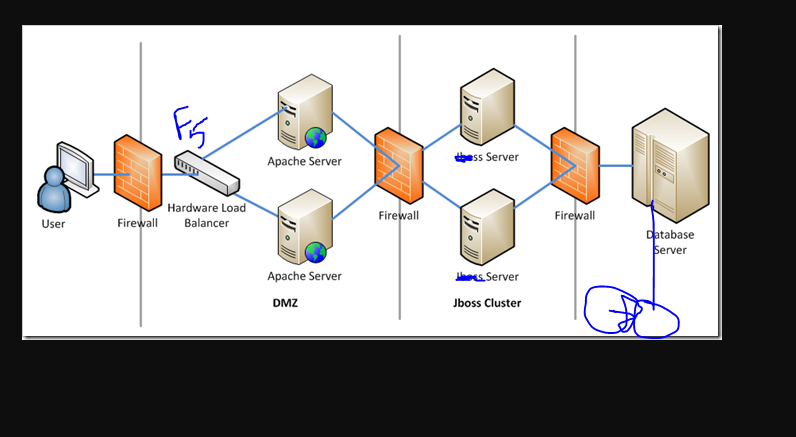
port:80

========================>IP----->80---->Apache(Proxy- ReverseProxy)--------------->8080(http)/8009(ajp)-->Tomcat

**Apache and Tomcat Integration**



**Enuser--→http(80)/https(443)-----→Apache---→http(8080)/ajp(8009)------→Tomcat-----→JDBC(mysql-3306/mssql-1433/oracle-1521/db2-50000)-----→DB**



**Apache and Tomcat integration done in multiple way**

**1.mod\_jk(external)**

**2.mod\_proxy(inbuilt in apache)**

**Request----→Apache-----------→Tomcat**

**Mod\_proxy**

**In httpd.conf-→Enable all \*proxy\*.so modules(.so)**

**Add below lines in httpd.conf (Single Tomcat)**

**<VirtualHost \*:80>**

**ServerName www.shoppingcart.test.com**

**RedirectMatch ^/$ /shoppingcart/**

**ProxyRequests off**

**ProxyPreserveHost on**

**ProxyPass /shoppingcart http://localhost:8080/shoppingcart/**

**ProxyPassReverse /shoppingcart http://localhost:8080/shoppingcart/**

**</virtualHost>**

**Restart apache.**

**With multiple Tomcat(change port numbers for second tomcat)**

**8005== some other port**

**8080= some other port**

**8443= some other port**

**8009= some other port**

**<VirtualHost \*:80>**

**ServerName www.shoppingcart.test.com**

**RedirectMatch ^/$ /shoppingcart/**

**ProxyRequests off**

**ProxyPreserveHost on**

**ProxyPass /shoppingcart balancer://shr**

**ProxyPassReverse /shoppingcart balancer://shr**

**<Proxy balancer://shr>**

**BalancerMember http://localhost:8080/shoppingcart**

**BalancerMember http://localhost:10000/shoppingcart**

**</Proxy>**

**</virtualHost>**

**About The Conext.xml(Datasouce info/DBinfo)**

**Application ------------------------------------------------------------→DB**

**DB username**

**DB password**

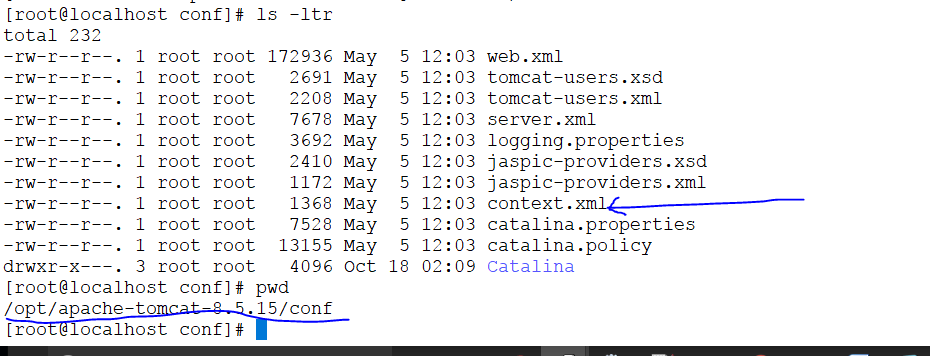
**DB Name**

**Where my DB is running**

**JNDI name**

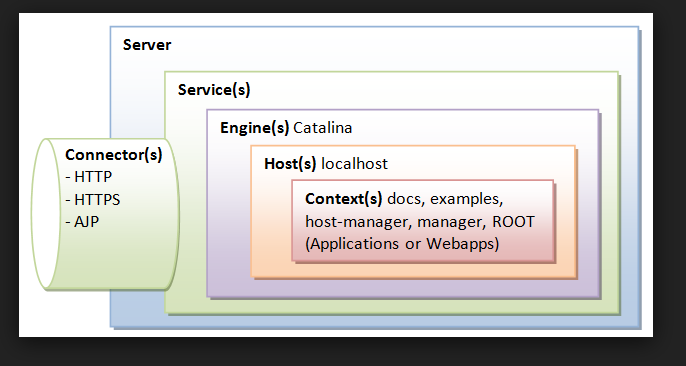
**Or**

**<CATINA\_HOME/conf/context.xml(JDBC/Datasource )(server.xml)**



**About server.xml**

**Tomcat architecture**



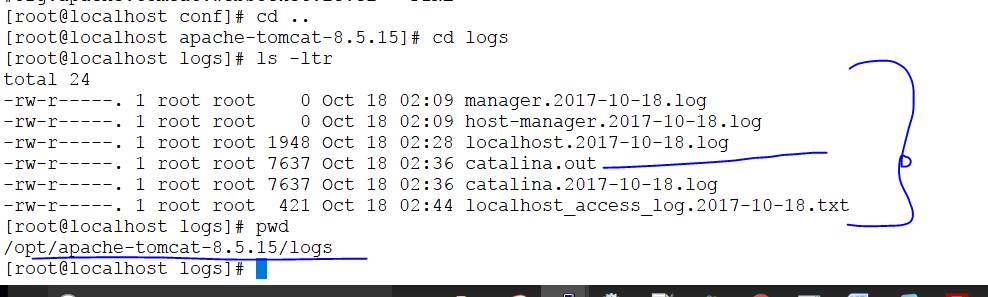
**About Logging .properties.(CATALINA\_BASE/conf/logging.properties)**

**Log-→Writing info**

**Log level**

**Log Appender-→console,file**

**Rotation-→size and time**



**Logging.properties**



**LAB:- Make a catalina.out to be with your name .**

**Adding the tomcat admin console users in tomcat-users.xml**

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

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

**Excerises:-**

**SSO**

**A=-----SSO--->username and pwd--->page---->ssopage------------->A page**

**B=---SSO-->username and pwd--->page----->ssopage**

**C=----SSO->username and pwd--->page------>ssopage**

**aws configure-->sso--->sso page-->wso2 page**

**SSO----------------------------->page(uname and pwd)--ssopage**

**gmail------------->gmail login(identity providers)(Idp)**

**facebook---->face login page)(identity providers)(Idp)**

**wso2---->sso---->(testapp---ssopage---username and pwd--->gmail)**

**testapp---->testapp page login--->ssopage(service provider)(sP)**

**testapp--->ssopage--->uname and pwd-->@gmail.com/pwd---->testapp pgage**

**gmail**

**youtube**

**google drive**

**--------->Google login page-->**

**facebook(identiy provider) --------------->ws02----------------->aws login(service provider)**

**LDAP/AD(Active Directory--Windows Server)-------->username and pwd**

**symante---->sso login page-->identity provider LDAP and Service Proider(register)---->**

**LAB:-**

Port number:10000--------→server.xml(connector)

Catalina.out--→name.out---------→logging.properties

/opt/logs/tomcat/--→logging.properties

/opt/apps/--→

<host--------- appBase=”/opt/apps”>

**Interview Questions**

Difference between kill -9 and kill -3 ?

Kill -3 <pid> is used to take thread dump.

Kill -9 <pid> is used to terminate process.

How to take thread and heap dumps ?(jstack and jmap are jdk tools )

Thread dump : jstack

Heap dump: jmap

Difference between thread dump and heap dump ?

0