Tomcat download

<http://www-us.apache.org/dist/tomcat/tomcat-7/v7.0.90/bin/apache-tomcat-7.0.90.zip>

Add this to .bashrc in home

export CATALINA\_HOME=/home/student/lab4/apache-tomcat-7.0.90

export JAVA\_HOME=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.79-2.5.5.0.fc20.x86\_64

export JRE\_HOME=/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.79-2.5.5.0.fc20.x86\_64/jre

export PATH=$PATH:$JAVA\_HOME/bin

Generate server keystore

**keytool -genkey -alias server -keyalg RSA -keystore keystore.jks -keysize 2048 -validity 365 -sigalg SHA256withRSA -storepass password -keypass password -dname "CN=server, OU=ID, O=IBM, L=Hursley, S=Hants, C=GB"**

Enable HTTPS in tomcat by adding this to tomcat conf/server.xml

<Connector  
 protocol="org.apache.coyote.http11.Http11NioProtocol"  
 port="8443" maxThreads="200"  
 scheme="https" secure="true" SSLEnabled="true"  
  **keystoreFile="/home/student/lab4/keystore.jks" keystorePass="password"** **clientAuth="false"** sslProtocol="TLS" ciphers="TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256,TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA,TLS\_ECDHsE\_RSA\_WITH\_RC4\_128\_SHA,TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA256,TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA,TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA256,TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA,SSL\_RSA\_WITH\_RC4\_128\_SHA"

/>

Go to bin of tomcat and start server.

Linux

sh catalina.sh run

Windows

catalina.bat run

Access tomcat via <https://localhost:8443>

Generate client keystore

**keytool -genkey -alias client -keyalg RSA -keystore clientkeystore.jks -keysize 2048 -validity 365 -sigalg SHA256withRSA -storepass password -keypass password -dname "CN=client, OU=ID, O=IBM, L=Hursley, S=Hants, C=GB"**

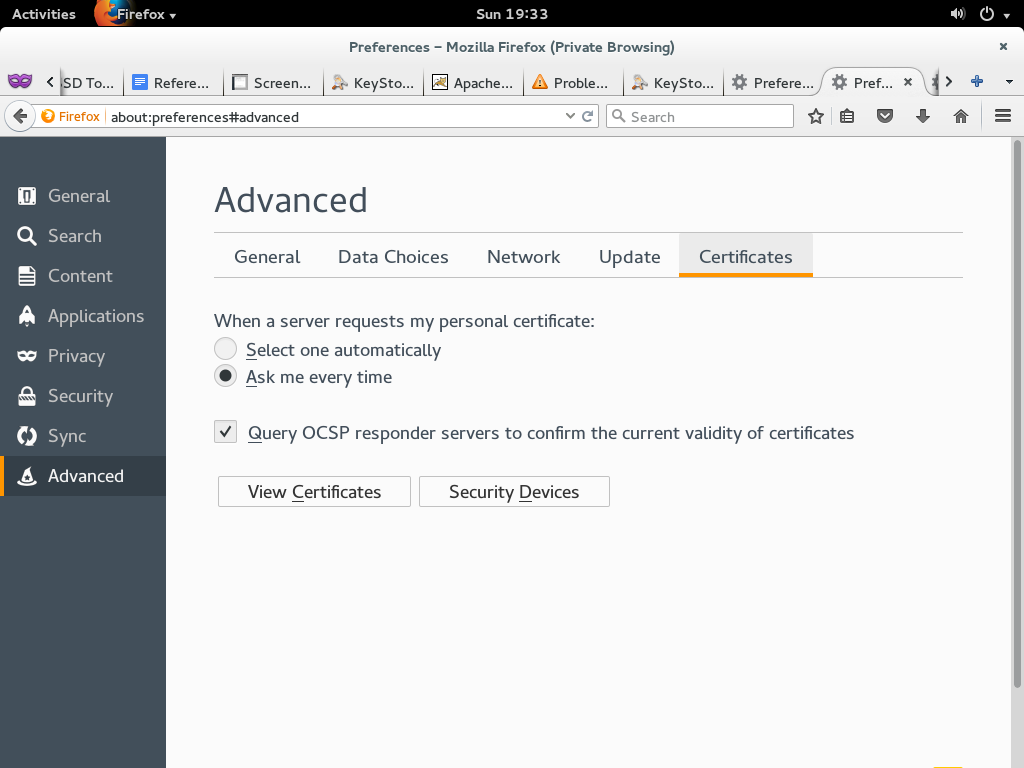
Keystore Explorer

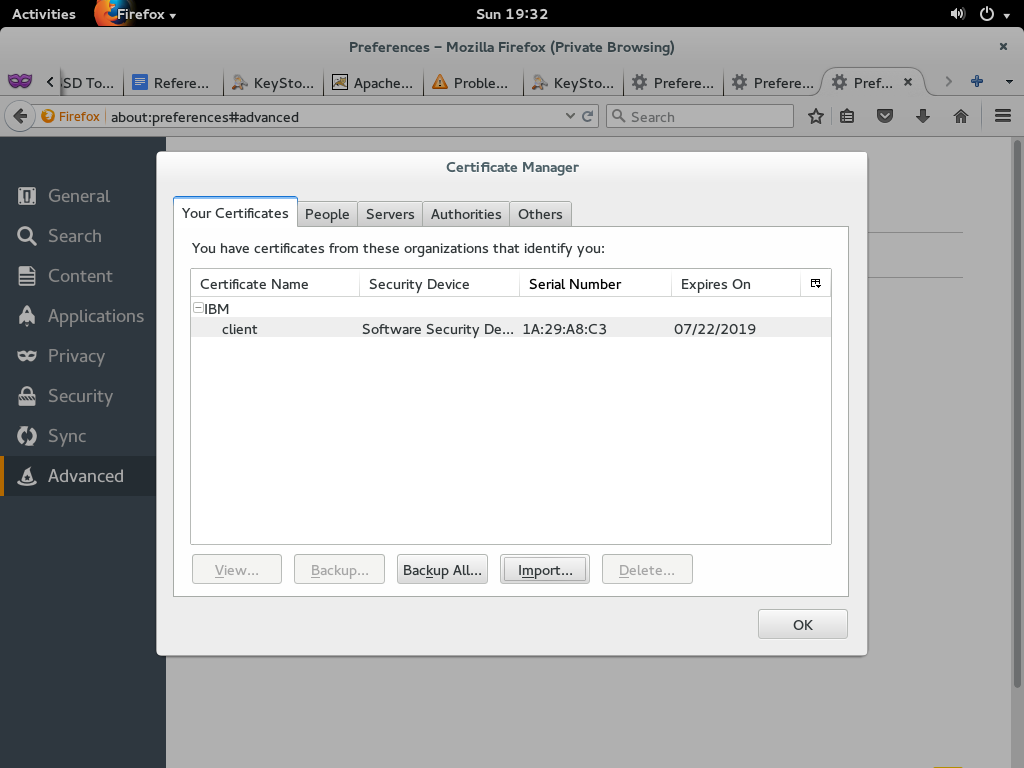
<http://keystore-explorer.org/downloads.html>

Convert JKS to P12

keytool -importkeystore -srckeystore clientkeystore.jks -destkeystore clientkeystore.p12 -srcstoretype JKS -deststoretype PKCS12 -srcstorepass password -deststorepass password -srcalias client -destalias client -srckeypass password -destkeypass password -noprompt

Go to Browser preferences and import the .p12 file created above as a trusted certificate and keystore.





<Connector

protocol="org.apache.coyote.http11.Http11NioProtocol"

port="8443" maxThreads="200"

scheme="https" secure="true" SSLEnabled="true"

keystoreFile="/home/student/lab4/keystore.jks" keystorePass="password"

**clientAuth="true"** sslProtocol="TLS"

**truststoreFile="/home/student/lab4/truststore.jks"**

**truststorePass="password"** ciphers="TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA256,TLS\_ECDHE\_RSA\_WITH\_AES\_128\_CBC\_SHA,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA384,TLS\_ECDHE\_RSA\_WITH\_AES\_256\_CBC\_SHA,TLS\_ECDHsE\_RSA\_WITH\_RC4\_128\_SHA,TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA256,TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA,TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA256,TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA,SSL\_RSA\_WITH\_RC4\_128\_SHA"

/>

Export client’s public certificate

keytool -export -alias *client* -file *clientcert*.crt -keystore client*keystore*.jks -storepass password

r

Import the above client certificate file to server’s trustrue

keytool -import -trustcacerts -file *clientcert.crt* -alias *client* -keystore truststore.jks -storepass password

How to read ssl debug logs

<http://www.smartjava.org/content/how-analyze-java-ssl-errors>

Add this to catalina.sh for enabling SSL debug logs within

-Djavax.net.debug=ssl:handshake

Delete trusted certificate from truststore

keytool -delete -alias client -keystore truststore.jks -storepass password

Keystore explorer

<http://keystore-explorer.org/>