One Way SSL

Introduction

In one way SSL the server is required to present the certificate to the client to verify the credentials of the server but client is not verified by the server.

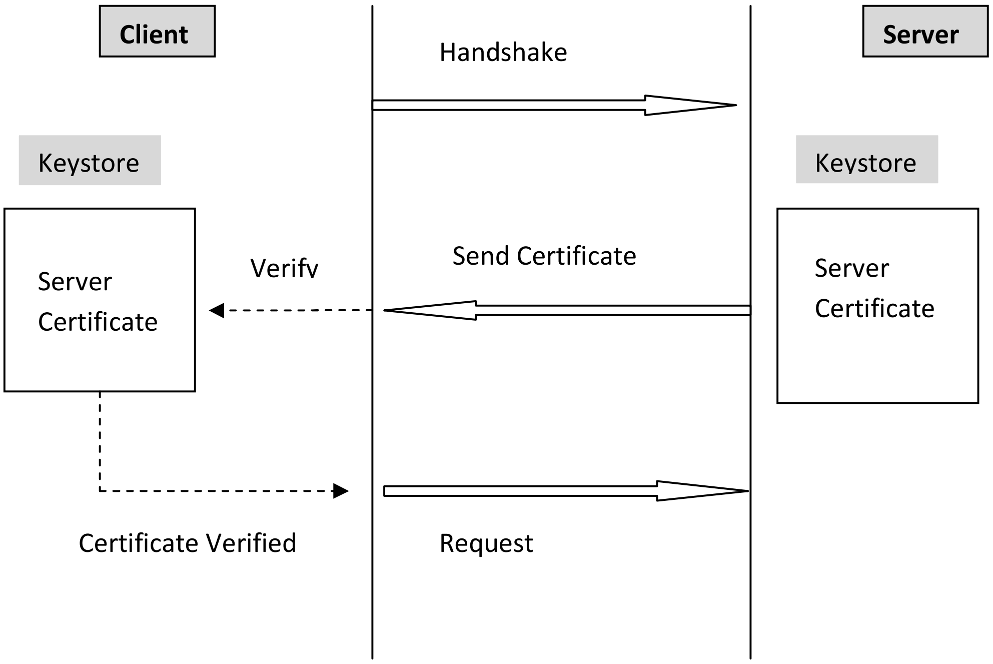


Figure: One Way SSL Process

Implementation

Example below shows how to configure one way SSL for client connecting to Weblogic/Glassfish Server. Both servers provide default keystore (database of private keys and certificate) which are complete in themselves for SSL implementation in testing environment. In production environment you should implement your own certificate signed by your own CA.

More information on configuring SSL on Weblogic at:

http://download-llnw.oracle.com/docs/cd/E11035_01/wls100/secmanage/ssl.html

Java provides **keytool**, a key and certificate management utility. It enables users to administer their own public/private key pairs and associated certificates for use in self-authentication.

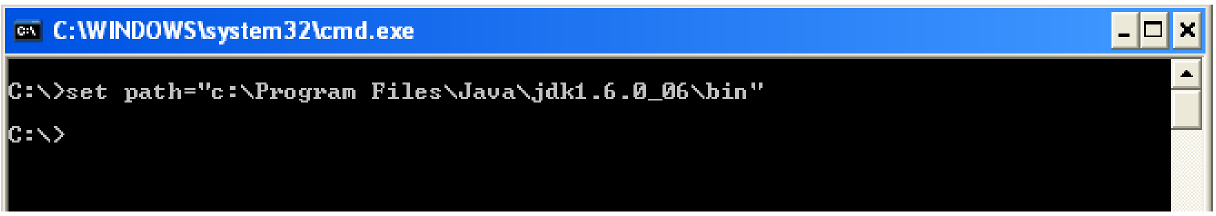
keytool stores the keys and certificates in a so-called *keystore.*

More information on keytool visit:

http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html

Following are the steps to implement One Way SSL:

1. Set the path to use keytool: set the path to your jdk

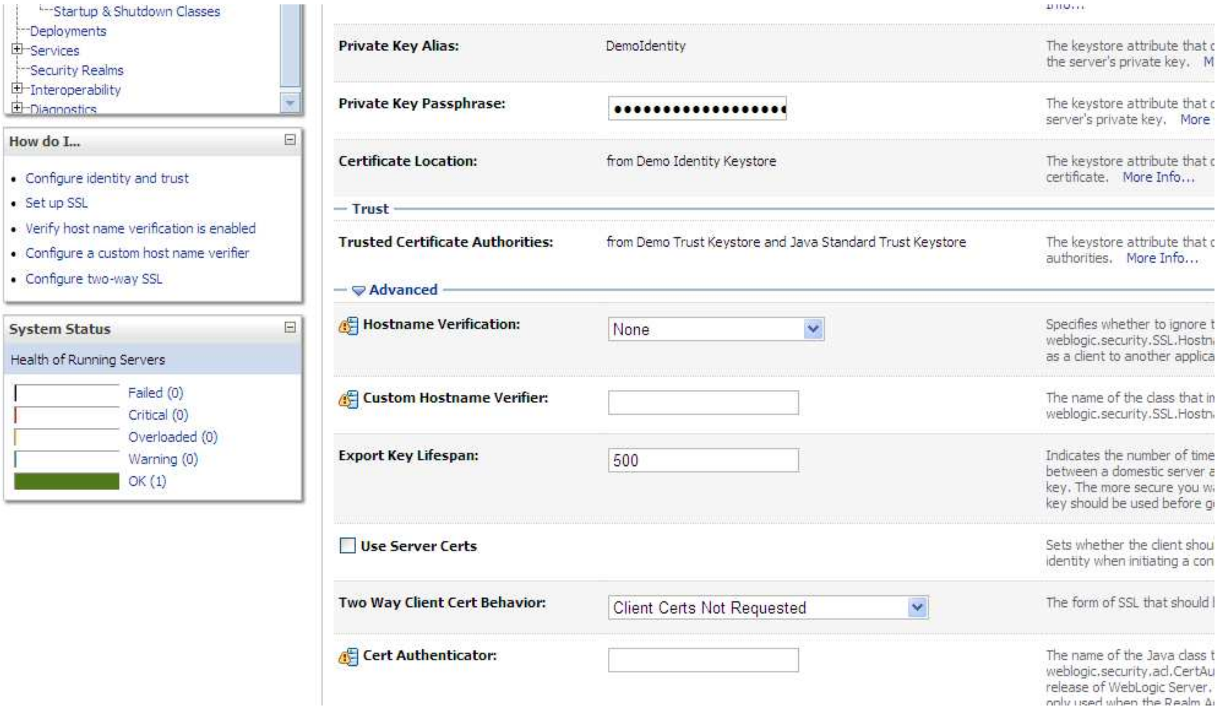


```
C:\>set path="c:\Program Files\Java\jdk1.6.0_06\bin"
C:\>
```

2. Configure the weblogic to implement One Way SSL

Start Weblogic -> Login to console -> Click on Environment -> Servers -> SSL ->Advanced

Make sure in Two Way Client Cert behavior option **Client certs not requested** is selected



Startup & Shutdown Classes
Deployments
⊞ Services
Security Realms
⊞ Interoperability
⊞ Diagnostics

How do I...

- Configure identity and trust
- Set up SSL
- Verify host name verification is enabled
- Configure a custom host name verifier
- Configure two-way SSL

System Status

Health of Running Servers

Failed (0)
Critical (0)
Overloaded (0)
Warning (0)
🟩 OK (1)

Private Key Alias: DemoIdentity — The keystore attribute that c the server's private key. M

Private Key Passphrase: •••••••••••••••••• — The keystore attribute that c server's private key. More

Certificate Location: from Demo Identity Keystore — The keystore attribute that c certificate. More Info...

Trust

Trusted Certificate Authorities: from Demo Trust Keystore and Java Standard Trust Keystore — The keystore attribute that c authorities. More Info...

Advanced

Hostname Verification: None — Specifies whether to ignore t weblogic.security.SSL.Hostn as a client to another applica

Custom Hostname Verifier: — The name of the class that in weblogic.security.SSL.Hostn

Export Key Lifespan: 500 — Indicates the number of time between a domestic server a key. The more secure you w key should be used before g

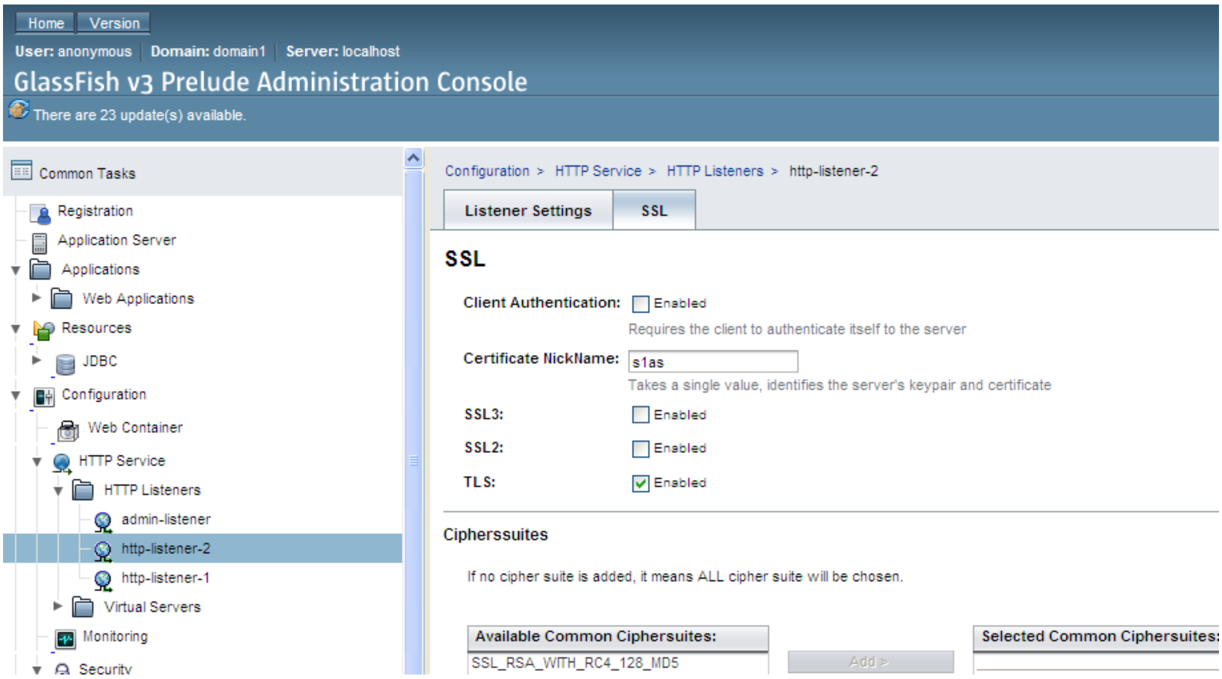
□ Use Server Certs — Sets whether the client shou identity when initiating a con

Two Way Client Cert Behavior: Client Certs Not Requested — The form of SSL that should l

Cert Authenticator: — The name of the Java class t weblogic.security.acl.CertAu release of WebLogic Server. only used when the Realm A

For Glassfish

Make sure that client authentication is not selected



SSL

Client Authentication: □ Enabled

Requires the client to authenticate itself to the server

Certificate NickName: s1as

Takes a single value, identifies the server's keypair and certificate

SSL3: □ Enabled

SSL2: □ Enabled

TLS: ☑ Enabled

Cipherssuites

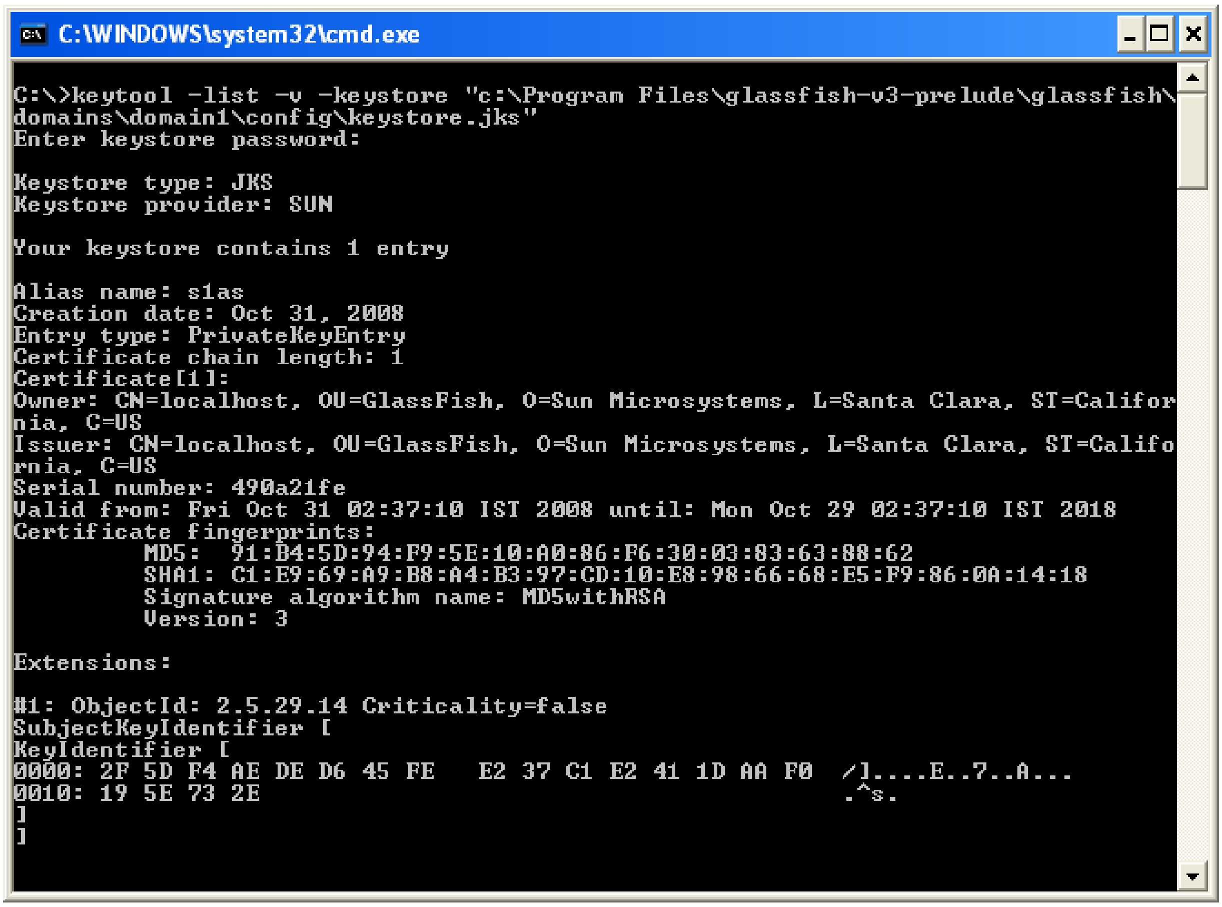
If no cipher suite is added, it means ALL cipher suite will be chosen.

3. To view the information about certificate(s) in default keystore

a) Glassfish Keystore

```
C:\>keytool -list -v -keystore "c:\Program Files\glassfish-v3-prelude\glassfish\domains\domain1\config\keystore.jks
```

Keystore password is masterpassword of domain that is defined by user during domain creation.
(For netbeans glassfish the password is "changeit")

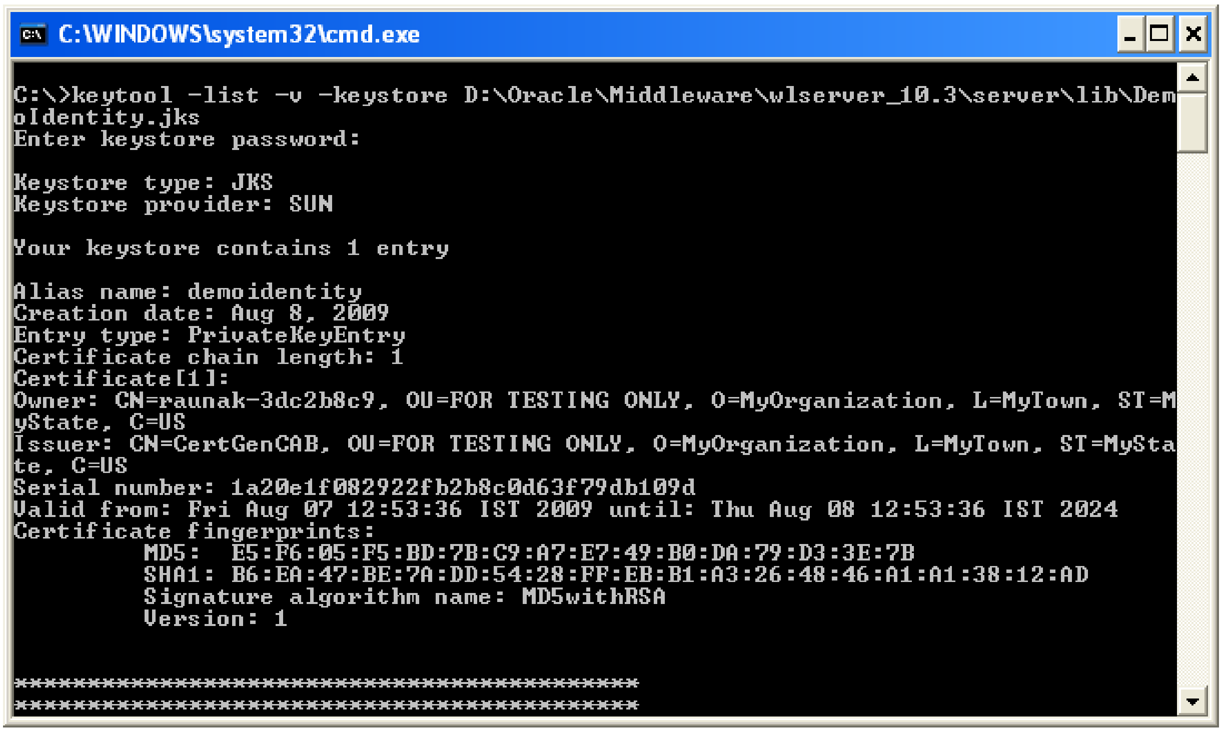


```
C:\>keytool -list -v -keystore "c:\Program Files\glassfish-v3-prelude\glassfish\
domains\domain1\config\keystore.jks"
Enter keystore password:

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 1 entry

Alias name: s1as
Creation date: Oct 31, 2008
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=localhost, OU=GlassFish, O=Sun Microsystems, L=Santa Clara, ST=Califor
nia, C=US
Issuer: CN=localhost, OU=GlassFish, O=Sun Microsystems, L=Santa Clara, ST=Califo
rnia, C=US
Serial number: 490a21fe
Valid from: Fri Oct 31 02:37:10 IST 2008 until: Mon Oct 29 02:37:10 IST 2018
Certificate fingerprints:
         MD5:  91:B4:5D:94:F9:5E:10:A0:86:F6:30:03:83:63:88:62
         SHA1: C1:E9:69:A9:B8:A4:B3:97:CD:10:E8:98:66:68:E5:F9:86:0A:14:18
         Signature algorithm name: MD5withRSA
         Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 2F 5D F4 AE DE D6 45 FE   E2 37 C1 E2 41 1D AA F0  /]....E..7..A...
0010: 19 5E 73 2E                                        .^s.
]
]
```

b) Weblogic Keystore

```
C:\>keytool -list -v -keystore
D:\Oracle\Middleware\wlserver_10.3\server\lib\DemoIdentity.jks
```

Default Password for DemoIdentity.jks is DemoIdentityKeyStorePassPhrase

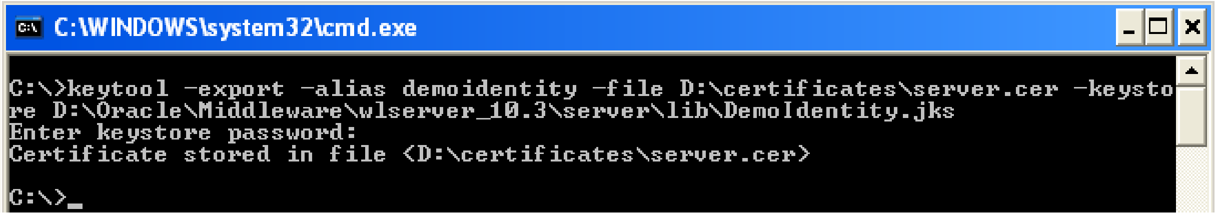


```
C:\>keytool -list -v -keystore D:\Oracle\Middleware\wlserver_10.3\server\lib\Dem
oIdentity.jks
Enter keystore password:

Keystore type: JKS
Keystore provider: SUN

Your keystore contains 1 entry

Alias name: demoidentity
Creation date: Aug 8, 2009
Entry type: PrivateKeyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=raunak-3dc2b8c9, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=M
yState, C=US
Issuer: CN=CertGenCAB, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=MySta
te, C=US
Serial number: 1a20e1f082922fb2b8c0d63f79db109d
Valid from: Fri Aug 07 12:53:36 IST 2009 until: Thu Aug 08 12:53:36 IST 2024
Certificate fingerprints:
         MD5:  E5:F6:05:F5:BD:7B:C9:A7:E7:49:B0:DA:79:D3:3E:7B
         SHA1: B6:EA:47:BE:7A:DD:54:28:FF:EB:B1:A3:26:48:46:A1:A1:38:12:AD
         Signature algorithm name: MD5withRSA
         Version: 1


*******************************************
*******************************************
```

4. Export the certificate in keystore to a file. This certificate file will be imported to client keystore.
   (Implementation steps including this step are explained by taking example of weblogic.
   Note: Following steps are same for both the server)

   a) Weblogic Certificate

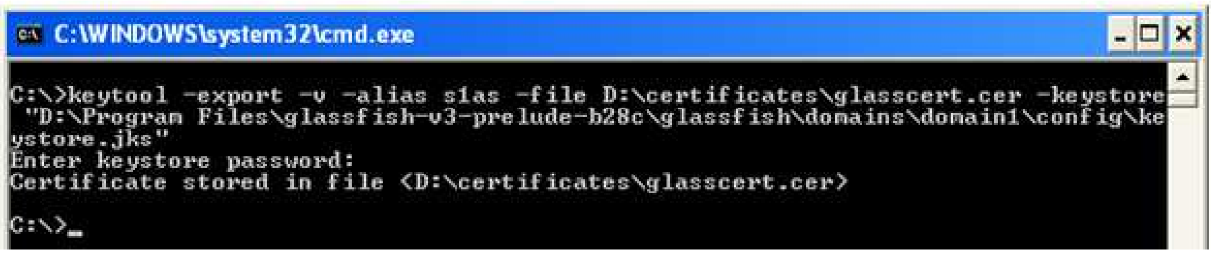
```
C:\>keytool -export -alias demoidentity -file D:\certificates\server.cer –keystore
D:\Oracle\Middleware\wlserver_10.3\server\lib\DemoIdentity.jks
```



```
C:\>keytool -export -alias demoidentity -file D:\certificates\server.cer -keysto
re D:\Oracle\Middleware\wlserver_10.3\server\lib\DemoIdentity.jks
Enter keystore password:
Certificate stored in file <D:\certificates\server.cer>

C:\>_
```

b) Glassfish Certificate

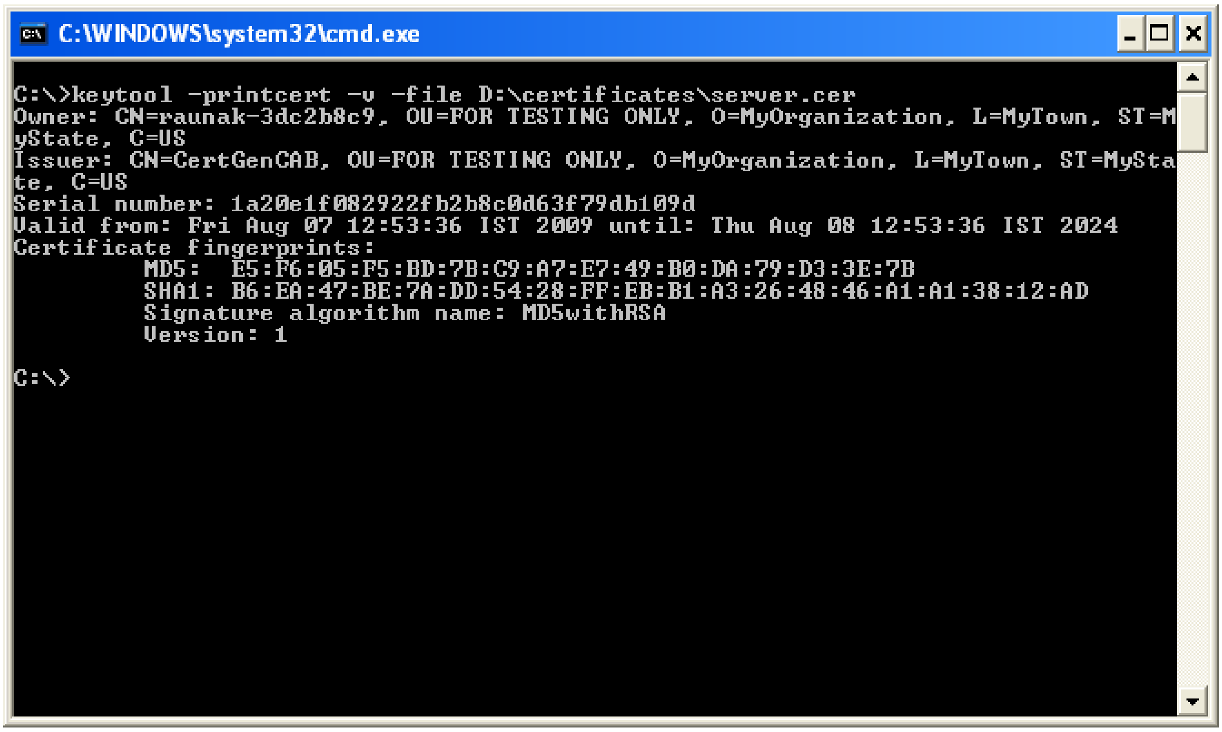
```
C:\> keytool -export -v -alias s1as -file D:\certificates\glasscert.cer –keystore "D:\Program Files\glassfish-v3-prelude-b28c\glassfish\domains\domain1\config\keystore.jks"
```



```
C:\>keytool -export -v -alias s1as -file D:\certificates\glasscert.cer -keystore
 "D:\Program Files\glassfish-v3-prelude-b28c\glassfish\domains\domain1\config\ke
ystore.jks"
Enter keystore password:
Certificate stored in file <D:\certificates\glasscert.cer>

C:\>_
```

5. To print the information about the certificate created

```
C:\>keytool -printcert -v -file D:\certificates\server.cer
```

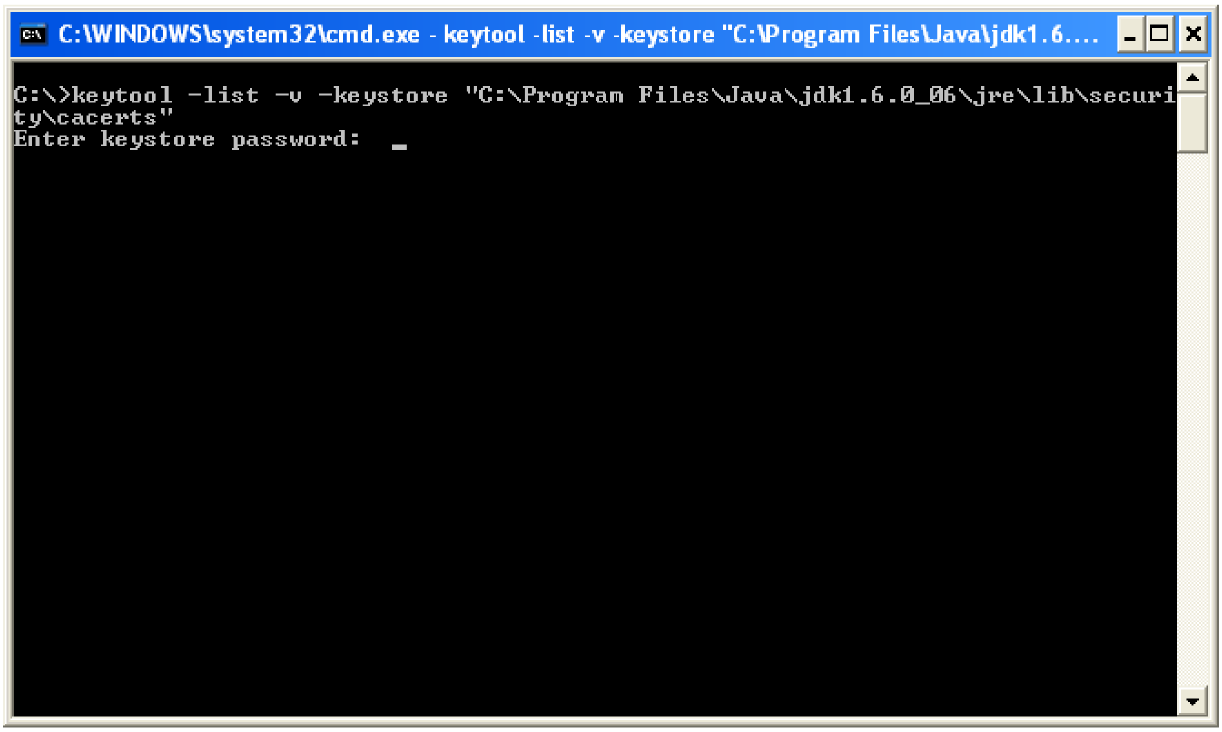


```
C:\>keytool -printcert -v -file D:\certificates\server.cer
Owner: CN=raunak-3dc2b8c9, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=M
yState, C=US
Issuer: CN=CertGenCAB, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=MySta
te, C=US
Serial number: 1a20e1f082922fb2b8c0d63f79db109d
Valid from: Fri Aug 07 12:53:36 IST 2009 until: Thu Aug 08 12:53:36 IST 2024
Certificate fingerprints:
         MD5:  E5:F6:05:F5:BD:7B:C9:A7:E7:49:B0:DA:79:D3:3E:7B
         SHA1: B6:EA:47:BE:7A:DD:54:28:FF:EB:B1:A3:26:48:46:A1:A1:38:12:AD
         Signature algorithm name: MD5withRSA
         Version: 1

C:\>
```

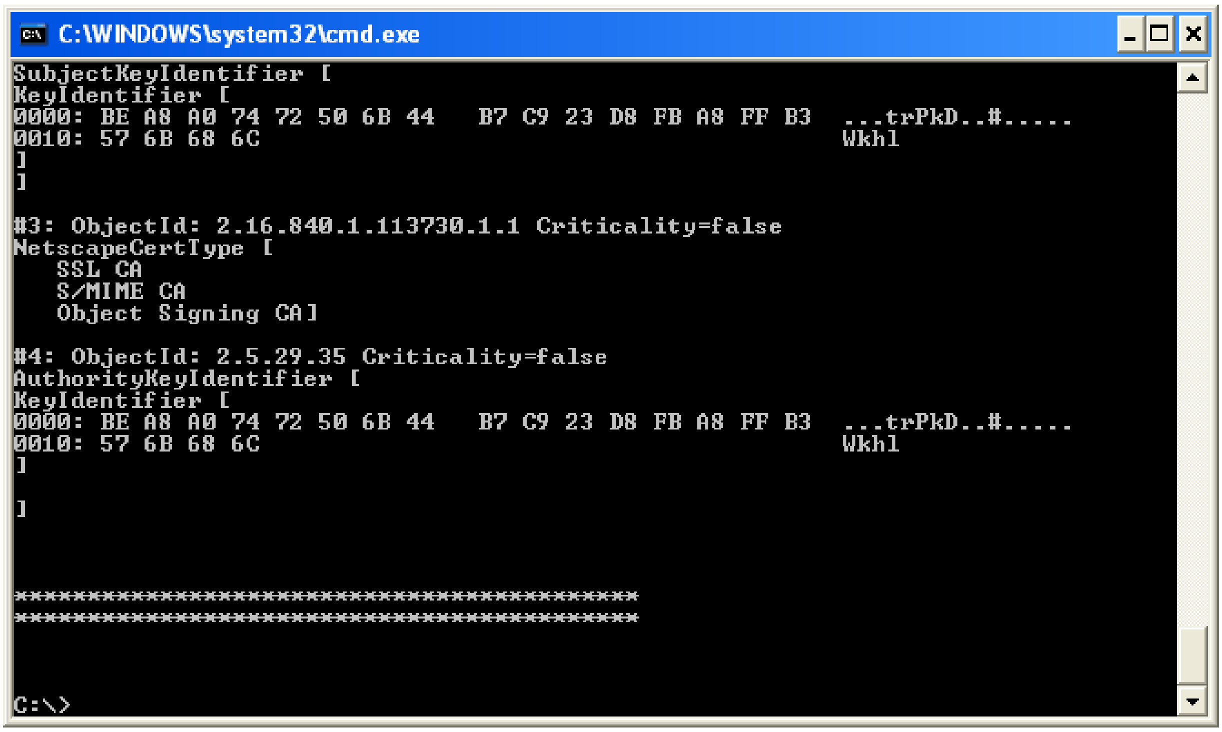
6. To view the information about certificates in the client keystore

   (Java provides its own truststore which is placed in "C:\Program Files\Java\jdk1.6.0_06\jre\lib\security" directory with name cacerts)

```
C:\>keytool -list -v -keystore "C:\Program Files\Java\jdk1.6.0_06\jre\lib\security\cacerts"
```


C:\WINDOWS\system32\cmd.exe - keytool -list -v -keystore "C:\Program Files\Java\jdk1.6....

```
C:\>keytool -list -v -keystore "C:\Program Files\Java\jdk1.6.0_06\jre\lib\securi
ty\cacerts"
Enter keystore password:  _
```



```
SubjectKeyIdentifier [
KeyIdentifier [
0000: BE A8 A0 74 72 50 6B 44   B7 C9 23 D8 FB A8 FF B3  ...trPkD..#.....
0010: 57 6B 68 6C                                        Wkhl
]
]

#3: ObjectId: 2.16.840.1.113730.1.1 Criticality=false
NetscapeCertType [
   SSL CA
   S/MIME CA
   Object Signing CA]

#4: ObjectId: 2.5.29.35 Criticality=false
AuthorityKeyIdentifier [
KeyIdentifier [
0000: BE A8 A0 74 72 50 6B 44   B7 C9 23 D8 FB A8 FF B3  ...trPkD..#.....
0010: 57 6B 68 6C                                        Wkhl
]

]



*******************************************
*******************************************



C:\>
```

Start the execution of client (in this example client is a java program) before importing the certificate to client keystore (default java keystore)

(Note: Service deployed on server has addition and subtraction operation exposed)

Client Run

```
Enter your choice:
 1. Addition
 2. Subtraction

1
Enter the first number for the operation.
10
Enter the second number for the operation.
20
```

The following error will occur indicating certificate is missing

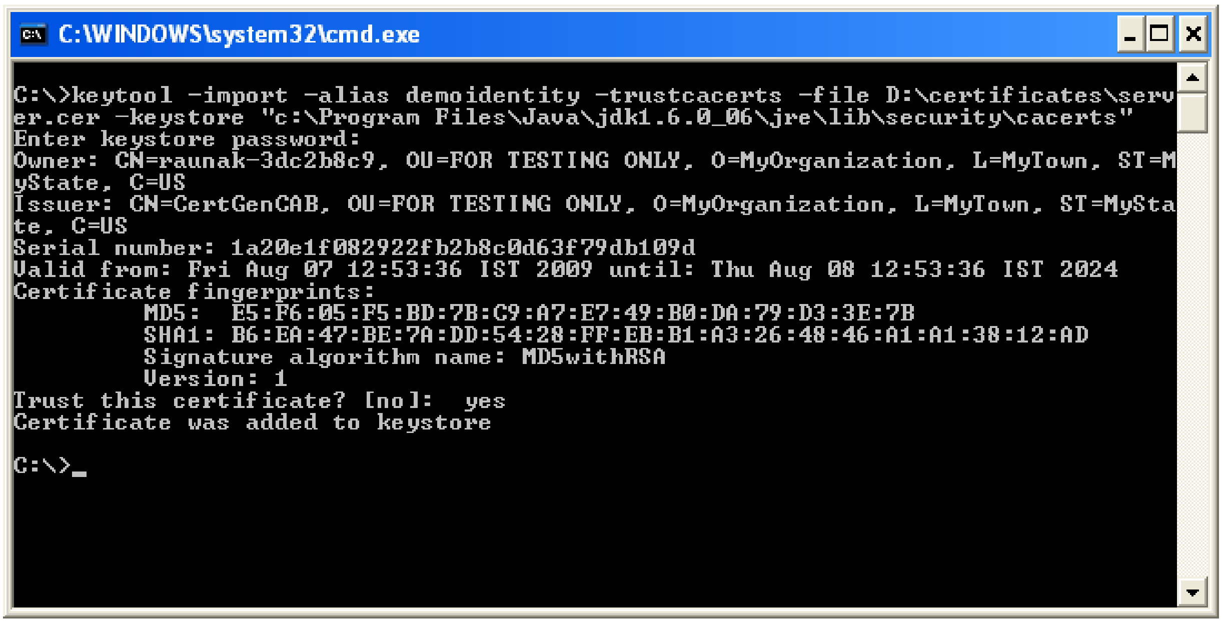
```
javax.xml.ws.WebServiceException: Failed to access the WSDL at:
https://116.73.230.57:7002/WebServices/AdditionsubtractionService?WSDL. It failed with:
	sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid
certification path to requested target.
	at
com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.tryWithMex(RuntimeWSDLParser.
java:136)
	at
com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.parse(RuntimeWSDLParser.java:1
22)
	at
com.sun.xml.internal.ws.client.WSServiceDelegate.parseWSDL(WSServiceDelegate.java:22
6)
	at
com.sun.xml.internal.ws.client.WSServiceDelegate.<init>(WSServiceDelegate.java:189)
	at
com.sun.xml.internal.ws.client.WSServiceDelegate.<init>(WSServiceDelegate.java:159)
	at
com.sun.xml.internal.ws.spi.ProviderImpl.createServiceDelegate(ProviderImpl.java:81)
	at javax.xml.ws.Service.<init>(Service.java:56)
	at com.service.AdditionsubtractionService.<init>(AdditionsubtractionService.java:46)
	at client.ClientAccess.add(ClientAccess.java:23)
	at
	at client.ClientAccess.main(ClientAccess.java:75)
Caused by: javax.net.ssl.SSLHandshakeException:
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid
certification path to requested target
	at com.sun.net.ssl.internal.ssl.Alerts.getSSLException(Alerts.java:174)
	at com.sun.net.ssl.internal.ssl.SSLSocketImpl.fatal(SSLSocketImpl.java:1591)
	at com.sun.net.ssl.internal.ssl.Handshaker.fatalSE(Handshaker.java:187)
	at com.sun.net.ssl.internal.ssl.Handshaker.fatalSE(Handshaker.java:181)
	at
com.sun.net.ssl.internal.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:975)
```

```
	at
com.sun.net.ssl.internal.ssl.ClientHandshaker.processMessage(ClientHandshaker.java:123)
	at com.sun.net.ssl.internal.ssl.Handshaker.processLoop(Handshaker.java:516)
	at com.sun.net.ssl.internal.ssl.Handshaker.process_record(Handshaker.java:454)
	at com.sun.net.ssl.internal.ssl.SSLSocketImpl.readRecord(SSLSocketImpl.java:884)
	at
com.sun.net.ssl.internal.ssl.SSLSocketImpl.performInitialHandshake(SSLSocketImpl.java:1
096)
	at
com.sun.net.ssl.internal.ssl.SSLSocketImpl.startHandshake(SSLSocketImpl.java:1123)
	at
com.sun.net.ssl.internal.ssl.SSLSocketImpl.startHandshake(SSLSocketImpl.java:1107)
	at sun.net.www.protocol.https.HttpsClient.afterConnect(HttpsClient.java:405)
	at
sun.net.www.protocol.https.AbstractDelegateHttpsURLConnection.connect(AbstractDelegate
HttpsURLConnection.java:166)
	at
sun.net.www.protocol.http.HttpURLConnection.getInputStream(HttpURLConnection.java:97
7)
	at
sun.net.www.protocol.https.HttpsURLConnectionImpl.getInputStream(HttpsURLConnectionI
mpl.java:234)
	at java.net.URL.openStream(URL.java:1009)
	at
com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.createReader(RuntimeWSDLParse
r.java:785)
	at
com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.resolveWSDL(RuntimeWSDLParse
r.java:236)
	at
com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.parse(RuntimeWSDLParser.java:1
07)
	... 8 more
Caused by: sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid
certification path to requested target
	at sun.security.validator.PKIXValidator.doBuild(PKIXValidator.java:285)
	at sun.security.validator.PKIXValidator.engineValidate(PKIXValidator.java:191)
	at sun.security.validator.Validator.validate(Validator.java:218)
	at
com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.validate(X509TrustManagerImpl.java:12
6)
	at
com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.checkServerTrusted(X509TrustManagerI
mpl.java:209)
Result of addition is 0
	at
com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.checkServerTrusted(X509TrustManagerI
mpl.java:249)
	at
com.sun.net.ssl.internal.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:954)
	... 23 more
Caused by: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid
certification path to requested target
```

```
	at 
sun.security.provider.certpath.SunCertPathBuilder.engineBuild(SunCertPathBuilder.java:174
)
	at java.security.cert.CertPathBuilder.build(CertPathBuilder.java:238)
	at sun.security.validator.PKIXValidator.doBuild(PKIXValidator.java:280)
	... 29 more
```

7. Import the server certificate into the client cacert

```
C:\>keytool -import -alias demoidentity -trustcacerts -file D:\certificates\server.cer -
keystore "c:\Program Files\Java\jdk1.6.0_06\jre\lib\security\cacerts"
```



```
C:\>keytool -import -alias demoidentity -trustcacerts -file D:\certificates\serv
er.cer -keystore "c:\Program Files\Java\jdk1.6.0_06\jre\lib\security\cacerts"
Enter keystore password:
Owner: CN=raunak-3dc2b8c9, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=M
yState, C=US
Issuer: CN=CertGenCAB, OU=FOR TESTING ONLY, O=MyOrganization, L=MyTown, ST=MySta
te, C=US
Serial number: 1a20e1f082922fb2b8c0d63f79db109d
Valid from: Fri Aug 07 12:53:36 IST 2009 until: Thu Aug 08 12:53:36 IST 2024
Certificate fingerprints:
         MD5:  E5:F6:05:F5:BD:7B:C9:A7:E7:49:B0:DA:79:D3:3E:7B
         SHA1: B6:EA:47:BE:7A:DD:54:28:FF:EB:B1:A3:26:48:46:A1:A1:38:12:AD
         Signature algorithm name: MD5withRSA
         Version: 1
Trust this certificate? [no]:  yes
Certificate was added to keystore

C:\>_
```

Note: For glassfish import the glasscert.cer into the cacert

Start the execution of client

Client Run

```
Enter your choice:
 1. Addition
 2. Subtraction
1
Enter the first number for the operation.
10
Enter the second number for the operation.
20

Port is JAX-WS RI 2.1.1 in JDK 6:
Stub for https://116.73.230.57:7002/WebServices/AdditionsubtractionService
Result of addition is 30
BUILD SUCCESSFUL (total time: 33 seconds)
```

Additional Information

If you get the following error while running your client:

Error:

```
javax.xml.ws.WebServiceException: Failed to access the WSDL at: https://116.73.230.57:7002/WebServices/AdditionsubtractionService?WSDL. It failed with:
	java.security.cert.CertificateException: No subject alternative names present.
	at com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.tryWithMex(RuntimeWSDLParser.java:136)
	at com.sun.xml.internal.ws.wsdl.parser.RuntimeWSDLParser.parse(RuntimeWSDLParser.java:122)
	at com.sun.xml.internal.ws.client.WSServiceDelegate.parseWSDL(WSServiceDelegate.java:226)
	at com.sun.xml.internal.ws.client.WSServiceDelegate.<init>(WSServiceDelegate.java:189)
	at com.sun.xml.internal.ws.client.WSServiceDelegate.<init>(WSServiceDelegate.java:159)
	at com.sun.xml.internal.ws.spi.ProviderImpl.createServiceDelegate(ProviderImpl.java:81)
	at javax.xml.ws.Service.<init>(Service.java:56)
	at com.service.AdditionsubtractionService.<init>(AdditionsubtractionService.java:46)
	at client.ClientAccess.subtraction(ClientAccess.java:38)
	at client.ClientAccess.main(ClientAccess.java:78)
```

Solution:

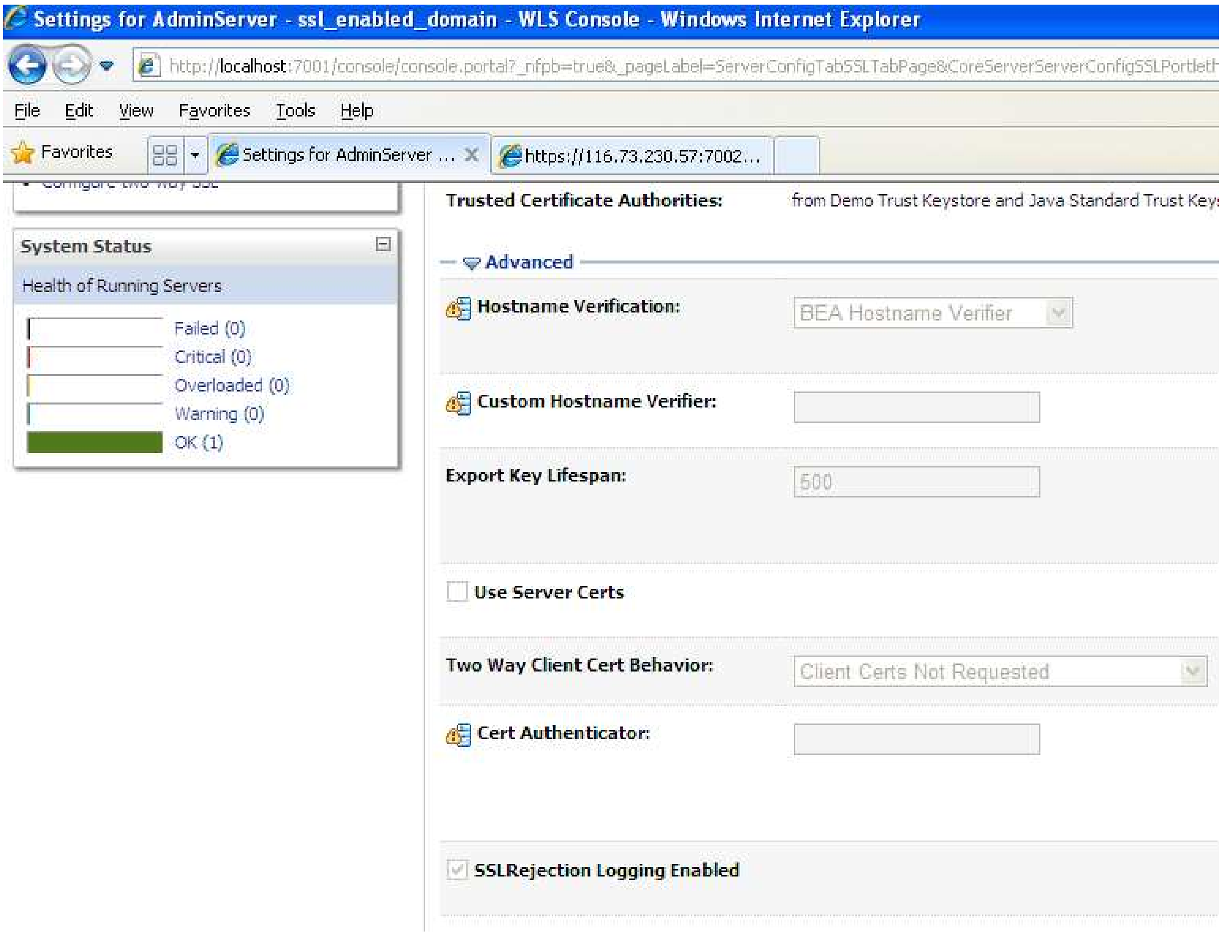
Include the following code in your code

```
static {
	//WORKAROUND. TO BE REMOVED.

	javax.net.ssl.HttpsURLConnection.setDefaultHostnameVerifier(new
javax.net.ssl.HostnameVerifier() {

	    public boolean verify(String hostname,
	            javax.net.ssl.SSLSession sslSession) {
	        return true;
	    }
	});

   }
```

Or

In weblogic

Start Weblogic -> Login to console -> Click on Environment -> Servers -> SSL ->Advanced


Settings for AdminServer - ssl_enabled_domain - WLS Console - Windows Internet Explorer
http://localhost:7001/console/console.portal?_nfpb=true&_pageLabel=ServerConfigTabSSLTabPage&CoreServerServerConfigSSLPortlet
File Edit View Favorites Tools Help
Favorites
Settings for AdminServer ...
https://116.73.230.57:7002...
System Status
Health of Running Servers
Failed (0)
Critical (0)
Overloaded (0)
Warning (0)
OK (1)
Advanced

Trusted Certificate Authorities: from Demo Trust Keystore and Java Standard Trust Key

Hostname Verification: BEA Hostname Verifier

Custom Hostname Verifier:

Export Key Lifespan: 500

□ Use Server Certs

Two Way Client Cert Behavior: Client Certs Not Requested

Cert Authenticator:

☑ SSLRejection Logging Enabled

Set the Hostname Verification to None