# SPECCHIO VM Authentication Setup

## On Server (CentOS)

**Preparation**: ensure the keytool is available. See <http://www.webupd8.org/2012/09/install-oracle-java-8-in-ubuntu-via-ppa.html>

Attention: setting the trust key password to something different than changeit will lead to errors in Glassfish: Exception during lifecycle processing

MultiException stack 1 of 4

java.lang.Error: java.security.UnrecoverableKeyException: Cannot recover key

* It is supposed that the password for the truststore entry would need changing in glassfish as well -> to be tested …

SPECCHIO super trust key password: changeit

cd /opt/glassfish4/glassfish/domains/domain1/config

# create backup

[root@specchiovm2 config]# mkdir backup

[root@specchiovm2 config]# cp cacerts.jks backup/

[root@specchiovm2 config]# cp keystore.jks backup/

[root@specchiovm2 config]# cp domain.xml backup/

### Creation of SPECCHIO super trust (done only once for SPECCHIO\_VM, then re-used in other distributions)

# remove old entry

keytool -delete -alias specchio\_trust -keystore keystore.jks -keypass changeit -storepass changeit

# create SPECCHIO super trust CA (Certificate Authority): create this private key in keystore.jks

*Note: My reasoning is that cacerts contains only trustedCertEntries and no private keys ... in any case, doing it the other way around does not let me import the SPECCHIO\_Trust as trustedCertEntry into cacerts (keytool -import -alias specchio\_trust -keystore cacerts.jks -storepass changeit -keypass changeit -file SPECCHIO\_Trust.cer*

*----> keytool error: java.lang.Exception: Certificate reply and certificate in keystore are identical)  
Having the private key of SPECCHIO\_Trust in the cacerts.jks gives this error when trying to sign the s1as key entry:  
keytool -import -alias s1as -keystore keystore.jks -storepass changeit -keypass changeit -file SPECCHIOVM2.cer*

*keytool error: java.lang.Exception: Failed to establish chain from reply*

keytool -genkey -keyalg RSA -keysize 4096 -sigalg SHA256WithRSA -keystore keystore.jks -keypass changeit -storepass changeit -validity 3650 -alias specchio\_trust -dname "cn=SPECCHIO Trust"

# show new key entry

keytool -list -keystore keystore.jks -alias specchio\_trust -storepass changeit -v

Alias name: specchio\_trust

Creation date: Apr 22, 2018

Entry type: PrivateKeyEntry

Certificate chain length: 1

Certificate[1]:

Owner: CN=SPECCHIO Trust

Issuer: CN=SPECCHIO Trust

Serial number: 1423ae1e

Valid from: Sun Apr 22 09:32:25 EDT 2018 until: Wed Apr 19 09:32:25 EDT 2028

Certificate fingerprints:

MD5: 97:92:30:DF:1A:DD:85:8C:FC:09:E9:FA:70:12:4B:9B

SHA1: A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

SHA256: 97:3F:9E:0D:C2:76:5E:59:78:59:E4:19:85:56:C9:88:93:B0:71:36:07:48:71:A1:B4:00:F6:FF:9E:8D:A1:55

Signature algorithm name: SHA256withRSA

Subject Public Key Algorithm: 4096-bit RSA key

Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 96 5D E6 3E 6A 5E C4 F1 B2 9F 49 FC 7B 2A 85 21 .].>j^....I..\*.!

0010: C8 82 CA 8C ....

]

]

The Fingerprints above are important. Copy them to an editor (eg. Notepad). We will use them later to verify if all is OK.

# export super trust certificate for use in client trust store

keytool -export -rfc -alias specchio\_trust -file SPECCHIO\_Trust.cer -keystore keystore.jks -keypass changeit -storepass changeit

# shows content of trust file:

openssl x509 -in SPECCHIO\_Trust.cer -text

Export the private key and certificate to use SPECCHIO super trust on a different server. This should allow to use the SPECCHIO\_Trust certifcate on the client side to validate the connection. Apparently, PKCS12 format must be used and then the private key entry exported via openssl (java keytool will not export the private key).

# conversion[[1]](#footnote-1): use keystore password = changeit

sudo keytool -importkeystore -srckeystore cacerts.jks -destkeystore cacerts.p12 -deststoretype PKCS12 -storepass changeit

Enter source keystore password:

Entry for alias digicertassuredidrootca successfully imported.

Entry for alias trustcenterclass2caii successfully imported.

Entry for alias thawtepremiumserverca successfully imported.

Entry for alias swisssignplatinumg2ca successfully imported.

Entry for alias swisssignsilverg2ca successfully imported.

Entry for alias thawteserverca successfully imported.

Entry for alias equifaxsecureebusinessca1 successfully imported.

Entry for alias utnuserfirstclientauthemailca successfully imported.

Entry for alias thawtepersonalfreemailca successfully imported.

Entry for alias s1as successfully imported.

Entry for alias entrustevca successfully imported.

Entry for alias utnuserfirsthardwareca successfully imported.

Entry for alias specchiovm2 successfully imported.

Entry for alias certumca successfully imported.

Entry for alias addtrustclass1ca successfully imported.

Entry for alias equifaxsecureca successfully imported.

Entry for alias quovadisrootca3 successfully imported.

Entry for alias quovadisrootca2 successfully imported.

Entry for alias digicerthighassuranceevrootca successfully imported.

Entry for alias secomvalicertclass1ca successfully imported.

Entry for alias equifaxsecureglobalebusinessca1 successfully imported.

Entry for alias geotrustuniversalca successfully imported.

Entry for alias verisignclass3ca successfully imported.

Entry for alias thawteprimaryrootcag3 successfully imported.

Entry for alias deutschetelekomrootca2 successfully imported.

Entry for alias utnuserfirstobjectca successfully imported.

Entry for alias geotrustprimaryca successfully imported.

Entry for alias baltimorecodesigningca successfully imported.

Entry for alias verisignclass1ca successfully imported.

Entry for alias baltimorecybertrustca successfully imported.

Entry for alias starfieldclass2ca successfully imported.

Entry for alias camerfirmachamberscommerceca successfully imported.

Entry for alias ttelesecglobalrootclass3ca successfully imported.

Entry for alias verisignclass3g5ca successfully imported.

**Entry for alias specchio\_trust successfully imported.**

Entry for alias ttelesecglobalrootclass2ca successfully imported.

Entry for alias trustcenteruniversalcai successfully imported.

Entry for alias verisignclass3g3ca successfully imported.

Entry for alias certplusclass3pprimaryca successfully imported.

Entry for alias certumtrustednetworkca successfully imported.

Entry for alias verisignclass3g2ca successfully imported.

Entry for alias globalsignr3ca successfully imported.

Entry for alias utndatacorpsgcca successfully imported.

Entry for alias secomscrootca2 successfully imported.

Entry for alias gtecybertrustglobalca successfully imported.

Entry for alias secomscrootca1 successfully imported.

Entry for alias trustcenterclass4caii successfully imported.

Entry for alias verisignuniversalrootca successfully imported.

Entry for alias globalsignr2ca successfully imported.

Entry for alias certplusclass2primaryca successfully imported.

Entry for alias digicertglobalrootca successfully imported.

Entry for alias globalsignca successfully imported.

Entry for alias thawteprimaryrootca successfully imported.

Entry for alias geotrustglobalca successfully imported.

Entry for alias soneraclass2ca successfully imported.

Entry for alias verisigntsaca successfully imported.

Entry for alias soneraclass1ca successfully imported.

Entry for alias quovadisrootca successfully imported.

Entry for alias valicertclass2ca successfully imported.

Entry for alias comodoaaaca successfully imported.

Entry for alias addtrustqualifiedca successfully imported.

Entry for alias keynectisrootca successfully imported.

Entry for alias aolrootca2 successfully imported.

Entry for alias addtrustexternalca successfully imported.

Entry for alias verisignclass2g3ca successfully imported.

Entry for alias aolrootca1 successfully imported.

Entry for alias verisignclass2g2ca successfully imported.

Entry for alias geotrustprimarycag3 successfully imported.

Entry for alias swisssigngoldg2ca successfully imported.

Entry for alias entrust2048ca successfully imported.

Entry for alias camerfirmachambersignca successfully imported.

Entry for alias camerfirmachambersca successfully imported.

Entry for alias glassfish-instance successfully imported.

Entry for alias godaddyclass2ca successfully imported.

Entry for alias entrustsslca successfully imported.

Entry for alias verisignclass1g3ca successfully imported.

Entry for alias secomevrootca1 successfully imported.

Entry for alias verisignclass1g2ca successfully imported.

Import command completed: 78 entries successfully imported, 0 entries failed or cancelled

# View the content of the pkcs12 keystore by:

openssl pkcs12 -alias specchio\_trust -in cacerts.p12

# conversion[[2]](#footnote-2): use keystore password = changeit

sudo keytool -importkeystore -srckeystore keystore.jks -destkeystore keystore.p12 -deststoretype PKCS12 -storepass changeit

Enter source keystore password:

Entry for alias specchio\_trust successfully imported.

Entry for alias glassfish-instance successfully imported.

Entry for alias s1as successfully imported.

Import command completed: 3 entries successfully imported, 0 entries failed or cancelled

Export the certificate:

sudo openssl pkcs12 -in cacerts.p12 -nokeys -out cert.pem

Export the private key (unencrypted): use changeit for all pass phrases

sudo openssl pkcs12 -in keystore.p12 -nocerts -out private\_keys.pem

sudo openssl pkcs12 -in keystore.p12 -out private\_keys.pem

Enter Import Password:

MAC verified OK

Enter PEM pass phrase:

Verifying - Enter PEM pass phrase:

Enter PEM pass phrase:

Verifying - Enter PEM pass phrase:

Enter PEM pass phrase:

Verifying - Enter PEM pass phrase:

# show contents

cat private\_keys.pem

* Shows both private keys and certificates

### Import of SPECCHIO Super Trust and Creation of key signed with Super Trust (to be done on new servers using SPECCHIO Super Trust)

*Note: IPs and host names must be matched with server in question.*

# remove existing trusted certificate if required:

keytool -delete -alias specchio\_trust -keystore cacerts.jks -storepass changeit

# import signed certificate into trust store:

keytool -import -alias specchio\_trust -keystore cacerts.jks -storepass changeit -file SPECCHIO\_Trust.cer

Issuer: CN=SPECCHIO Trust

Serial number: 1423ae1e

Valid from: Sun Apr 22 09:32:25 EDT 2018 until: Wed Apr 19 09:32:25 EDT 2028

Certificate fingerprints:

MD5: 97:92:30:DF:1A:DD:85:8C:FC:09:E9:FA:70:12:4B:9B

SHA1: A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

SHA256: 97:3F:9E:0D:C2:76:5E:59:78:59:E4:19:85:56:C9:88:93:B0:71:36:07:48:71:A1:B4:00:F6:FF:9E:8D:A1:55

Signature algorithm name: SHA256withRSA

Subject Public Key Algorithm: 4096-bit RSA key

Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 96 5D E6 3E 6A 5E C4 F1 B2 9F 49 FC 7B 2A 85 21 .].>j^....I..\*.!

0010: C8 82 CA 8C ....

]

]

Trust this certificate? [no]: yes

Certificate was added to keystore

# remove old s1as entry

keytool -delete -alias s1as -keystore keystore.jks -keypass changeit -storepass changeit

# generate new key (including IP of SPECCHIO VM via host only adapter of Oracle VM

keytool -genkey -keyalg RSA -keysize 2048 -sigalg SHA256WithRSA -keystore keystore.jks -keypass changeit -storepass changeit -validity 3650 -ext SAN=dns:localhost,ip:127.0.0.1,ip:192.168.56.101 -alias s1as -dname "cn=SPECCHIOVM2"

# signed with SPECCHIO super trust CA (written into SPECCHIOVM2.cer)

keytool -keystore keystore.jks -storepass changeit -keypass changeit -alias s1as -certreq | keytool -keystore keystore.jks -keypass changeit -storepass changeit -alias specchio\_trust -gencert -rfc -validity 3650 -ext SAN=dns:localhost,ip:127.0.0.1,ip:192.168.56.101 >SPECCHIOVM2.cer

# show certificate content

openssl x509 -in SPECCHIOVM2.cer -text

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 1332524819 (0x4f6cb713)

Signature Algorithm: sha256WithRSAEncryption

Issuer: CN=SPECCHIO Trust

Validity

Not Before: Apr 22 13:35:45 2018 GMT

Not After : Apr 19 13:35:45 2028 GMT

Subject: CN=SPECCHIOVM2

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

Public-Key: (2048 bit)

Modulus:

00:d3:07:94:2c:d5:76:8c:6b:b6:15:0a:0d:f1:9d:

3f:7b:29:ff:64:b4:32:82:78:cf:e8:c8:aa:74:0f:

42:a0:4a:b1:19:db:00:d1:c5:26:e5:86:3a:20:46:

25:cd:d4:33:b4:b4:4b:0d:25:3f:28:2a:3f:1a:c3:

dc:4d:9c:a7:d5:af:e7:17:3d:13:f4:3d:32:fb:b8:

d7:f7:80:4a:3f:8c:c3:08:f9:27:f4:c9:c5:ce:93:

e9:0c:7b:bc:e0:67:5f:57:09:08:de:a8:d3:ce:84:

02:f4:00:07:a6:4b:a2:71:dd:60:b8:5f:98:ce:8f:

7d:14:8d:f0:32:14:cb:59:fa:19:3f:3d:27:dd:3c:

73:89:30:e9:ed:e6:1b:ba:2e:e0:cb:9d:24:6f:e8:

d6:86:e9:0a:97:38:e2:0f:9e:a7:4e:92:d4:fd:66:

ae:40:7c:03:71:df:31:5a:df:91:94:ff:fb:21:a1:

c5:6b:4c:a9:dc:f5:9b:bf:49:3e:b7:f1:d4:dc:bc:

47:78:90:53:84:20:94:46:d6:a4:0c:e1:b8:39:1d:

ff:e3:d5:c3:cb:17:d0:eb:46:e1:60:14:23:d3:0a:

09:90:91:03:57:03:4f:e1:8d:ec:fa:0e:9a:fc:12:

85:2b:d7:5d:eb:ff:90:c1:83:99:81:90:71:05:ee:

c4:e7

Exponent: 65537 (0x10001)

X509v3 extensions:

X509v3 Authority Key Identifier:

keyid:96:5D:E6:3E:6A:5E:C4:F1:B2:9F:49:FC:7B:2A:85:21:C8:82:CA:8C

X509v3 Subject Alternative Name:

DNS:localhost, IP Address:127.0.0.1, IP Address:192.168.56.101

X509v3 Subject Key Identifier:

D4:46:AF:8A:AE:9B:4B:15:B3:77:36:BB:AE:05:A3:C6:11:94:02:2B

Signature Algorithm: sha256WithRSAEncryption

11:77:43:2a:3c:51:d5:ef:a1:4d:3d:58:f5:3c:7a:14:8d:dd:

d3:7b:35:a3:29:73:3b:3b:5f:47:77:22:54:ac:93:93:5d:0e:

a8:fb:e5:6c:36:0c:ab:66:77:17:ba:31:44:50:4d:71:6d:27:

3e:04:60:e3:78:cb:5b:6f:b8:44:a5:19:6f:ad:b1:8d:15:e6:

8b:80:17:87:4a:5f:b7:8e:39:b6:10:bb:25:ed:4e:3b:e2:8e:

7e:eb:4d:a6:06:31:ee:69:01:d8:3b:d7:c8:85:6b:2f:be:77:

7e:66:d5:c7:ab:ba:eb:1d:bd:78:6f:1a:43:61:2c:93:93:15:

81:a2:53:c8:0f:78:84:c0:94:b7:00:5d:43:1e:58:1b:88:ec:

b4:6b:a7:4f:63:51:c3:c1:6b:22:fe:3e:38:7b:6d:d5:49:98:

fe:23:db:00:57:52:cd:0d:ef:16:6c:66:1a:e4:c2:f1:7d:b5:

01:b7:8b:e6:e9:7f:1a:17:39:df:4b:e5:8e:32:ab:b3:5a:ec:

b1:a3:64:5c:77:cb:bf:a9:f1:11:fa:a3:98:af:d3:d7:dd:94:

a2:5d:2a:78:0c:99:10:f4:65:f7:82:ae:b4:86:ff:a5:79:18:

aa:56:c4:5b:ff:88:ff:f6:19:fe:9a:c7:68:f9:7b:0a:fa:47:

56:f0:35:2c:88:eb:46:b4:ec:e5:6b:a4:13:d4:95:de:f8:ad:

d9:fc:ef:f5:a0:ae:8b:30:17:55:ae:21:57:91:ba:7e:af:e5:

bf:69:7a:08:27:05:d7:15:b4:cc:c5:76:01:8c:9c:b4:ef:bc:

31:ed:98:60:ce:47:8a:a8:e0:7b:95:44:a8:7b:de:47:2d:a5:

c3:99:34:1c:27:17:ba:83:f6:2a:dc:d7:41:1f:4e:12:73:08:

a0:12:36:64:26:48:37:90:1c:89:74:07:7a:b4:70:50:f9:09:

79:56:28:01:13:12:76:a9:06:94:69:df:0c:90:b1:a8:5c:6b:

30:94:e9:8f:85:5d:9d:35:8f:e0:da:ad:27:0c:e3:1d:f9:69:

79:7c:19:66:a9:df:5c:b7:3d:8f:2e:e3:91:cb:8c:20:ca:90:

c0:7b:39:18:91:51:a2:0b:95:ce:98:84:95:86:e3:aa:a0:04:

f8:fb:20:48:2c:e0:18:83:03:90:40:e3:69:d3:7c:74:a8:ac:

09:0a:00:69:89:7c:d5:4c:c2:fd:b6:84:ea:6e:f5:39:f8:2f:

82:f4:7e:8b:5f:e7:ed:fc:31:d2:59:00:dc:63:77:39:28:71:

2f:5b:b0:b1:25:54:9a:0a:0d:2f:8e:7c:ed:a4:a7:39:8e:e5:

17:da:71:6b:5a:ca:90:58

-----BEGIN CERTIFICATE-----

MIIEETCCAfmgAwIBAgIET2y3EzANBgkqhkiG9w0BAQsFADAZMRcwFQYDVQQDEw5T

UEVDQ0hJTyBUcnVzdDAeFw0xODA0MjIxMzM1NDVaFw0yODA0MTkxMzM1NDVaMBYx

FDASBgNVBAMTC1NQRUNDSElPVk0yMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB

CgKCAQEA0weULNV2jGu2FQoN8Z0/eyn/ZLQygnjP6MiqdA9CoEqxGdsA0cUm5YY6

IEYlzdQztLRLDSU/KCo/GsPcTZyn1a/nFz0T9D0y+7jX94BKP4zDCPkn9MnFzpPp

DHu84GdfVwkI3qjTzoQC9AAHpkuicd1guF+Yzo99FI3wMhTLWfoZPz0n3TxziTDp

7eYbui7gy50kb+jWhukKlzjiD56nTpLU/WauQHwDcd8xWt+RlP/7IaHFa0yp3PWb

v0k+t/HU3LxHeJBThCCURtakDOG4OR3/49XDyxfQ60bhYBQj0woJkJEDVwNP4Y3s

+g6a/BKFK9dd6/+QwYOZgZBxBe7E5wIDAQABo2QwYjAfBgNVHSMEGDAWgBSWXeY+

al7E8bKfSfx7KoUhyILKjDAgBgNVHREEGTAXgglsb2NhbGhvc3SHBH8AAAGHBMCo

OGUwHQYDVR0OBBYEFNRGr4qum0sVs3c2u64Fo8YRlAIrMA0GCSqGSIb3DQEBCwUA

A4ICAQARd0MqPFHV76FNPVj1PHoUjd3TezWjKXM7O19HdyJUrJOTXQ6o++VsNgyr

ZncXujFEUE1xbSc+BGDjeMtbb7hEpRlvrbGNFeaLgBeHSl+3jjm2ELsl7U474o5+

602mBjHuaQHYO9fIhWsvvnd+ZtXHq7rrHb14bxpDYSyTkxWBolPID3iEwJS3AF1D

HlgbiOy0a6dPY1HDwWsi/j44e23VSZj+I9sAV1LNDe8WbGYa5MLxfbUBt4vm6X8a

FznfS+WOMquzWuyxo2Rcd8u/qfER+qOYr9PX3ZSiXSp4DJkQ9GX3gq60hv+leRiq

VsRb/4j/9hn+msdo+XsK+kdW8DUsiOtGtOzla6QT1JXe+K3Z/O/1oK6LMBdVriFX

kbp+r+W/aXoIJwXXFbTMxXYBjJy077wx7ZhgzkeKqOB7lUSoe95HLaXDmTQcJxe6

g/Yq3NdBH04ScwigEjZkJkg3kByJdAd6tHBQ+Ql5VigBExJ2qQaUad8MkLGoXGsw

lOmPhV2dNY/g2q0nDOMd+Wl5fBlmqd9ctz2PLuORy4wgypDAezkYkVGiC5XOmISV

huOqoAT4+yBILOAYgwOQQONp03x0qKwJCgBpiXzVTML9toTqbvU5+C+C9H6LX+ft

/DHSWQDcY3c5KHEvW7CxJVSaCg0vjnztpKc5juUX2nFrWsqQWA==

-----END CERTIFICATE-----

# import signed certificate into keystore to sign existing s1as key entry in keystore

keytool -import -alias s1as -keystore keystore.jks -storepass changeit -keypass changeit -file SPECCHIOVM2.cer

Certificate reply was installed in keystore

# show keystore content

keytool -list -keystore keystore.jks -keypass changeit -storepass changeit

Keystore type: JKS

Keystore provider: SUN

Your keystore contains 3 entries

specchio\_trust, Apr 22, 2018, PrivateKeyEntry,

Certificate fingerprint (SHA1): A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

glassfish-instance, Mar 23, 2017, PrivateKeyEntry,

Certificate fingerprint (SHA1): 60:64:1C:FE:E8:61:74:30:03:39:B9:90:9D:43:98:68:07:B7:86:63

s1as, Apr 22, 2018, PrivateKeyEntry,

Certificate fingerprint (SHA1): 63:07:D0:BA:A5:A5:11:86:3C:8B:AB:E4:FC:0E:8A:B5:9B:F8:1B:85

specchio\_trust, Apr 21, 2018, PrivateKeyEntry,

Certificate fingerprint (SHA1): 49:56:C7:23:42:AD:3E:D3:60:BA:50:55:12:56:AE:5B:13:F8:81:64

glassfish-instance, Mar 23, 2017, PrivateKeyEntry,

Certificate fingerprint (SHA1): 60:64:1C:FE:E8:61:74:30:03:39:B9:90:9D:43:98:68:07:B7:86:63

s1as, Apr 21, 2018, PrivateKeyEntry,

should show two certificates (not more)

# show signed s1as certificate

keytool -list -keystore keystore.jks -alias s1as -keypass changeit -storepass changeit -v

Alias name: s1as

Creation date: Apr 22, 2018

Entry type: PrivateKeyEntry

Certificate chain length: 2

Certificate[1]:

Owner: CN=SPECCHIOVM2

Issuer: CN=SPECCHIO Trust

Serial number: 4f6cb713

Valid from: Sun Apr 22 09:35:45 EDT 2018 until: Wed Apr 19 09:35:45 EDT 2028

Certificate fingerprints:

MD5: B4:50:3C:D5:6E:62:5E:46:1A:1A:BE:DE:66:B5:57:72

SHA1: 63:07:D0:BA:A5:A5:11:86:3C:8B:AB:E4:FC:0E:8A:B5:9B:F8:1B:85

SHA256: 58:99:3A:3C:09:B6:BE:2A:DF:DB:13:E3:49:97:99:AB:F1:21:BD:53:CC:1F:9F:F0:F0:34:0E:62:71:91:39:3B

Signature algorithm name: SHA256withRSA

Subject Public Key Algorithm: 2048-bit RSA key

Version: 3

Extensions:

#1: ObjectId: 2.5.29.35 Criticality=false

AuthorityKeyIdentifier [

KeyIdentifier [

0000: 96 5D E6 3E 6A 5E C4 F1 B2 9F 49 FC 7B 2A 85 21 .].>j^....I..\*.!

0010: C8 82 CA 8C ....

]

]

#2: ObjectId: 2.5.29.17 Criticality=false

SubjectAlternativeName [

DNSName: localhost

IPAddress: 127.0.0.1

IPAddress: 192.168.56.101

]

#3: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: D4 46 AF 8A AE 9B 4B 15 B3 77 36 BB AE 05 A3 C6 .F....K..w6.....

0010: 11 94 02 2B ...+

]

]

Certificate[2]:

Owner: CN=SPECCHIO Trust

Issuer: CN=SPECCHIO Trust

Serial number: 1423ae1e

Valid from: Sun Apr 22 09:32:25 EDT 2018 until: Wed Apr 19 09:32:25 EDT 2028

Certificate fingerprints:

MD5: 97:92:30:DF:1A:DD:85:8C:FC:09:E9:FA:70:12:4B:9B

SHA1: A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

SHA256: 97:3F:9E:0D:C2:76:5E:59:78:59:E4:19:85:56:C9:88:93:B0:71:36:07:48:71:A1:B4:00:F6:FF:9E:8D:A1:55

Signature algorithm name: SHA256withRSA

Subject Public Key Algorithm: 4096-bit RSA key

Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 96 5D E6 3E 6A 5E C4 F1 B2 9F 49 FC 7B 2A 85 21 .].>j^....I..\*.!

0010: C8 82 CA 8C ....

]

]

This should show two certificates: First the SPECCHIOVM2 certificate (must have Owner cn=SPECCHIOVM2 and Issuer cn=SPECCHIO TRUST). The second should show Owner cn=SPECCHIO TRUST and Issuer cn=SPECCHIO TRUST

# Mount host folder

mount -t vboxsf SPECCHIO\_Project /mnt

# copy “SPECCHIO super trust certificate” outside of VM to import into SPECCHIO keystore on Mac

cp SPECCHIO\_Trust.cer /mnt

# Mac

keytool -delete -alias specchio\_trust -keystore /Users/andyhueni/git/dc10/src/client/specchio.keystore

cd /Users/andyhueni/Data/Studies/RSL/SPECCHIO\ Project/Certificates/SPECCHIO\_VM2\_Glassfish4

keytool -import -alias SPECCHIO\_Trust -file SPECCHIO\_Trust.cer -trustcacerts -keystore /Users/andyhueni/git/dc10/src/client/specchio.keystore

Owner: CN=SPECCHIO Trust

Issuer: CN=SPECCHIO Trust

Serial number: 1423ae1e

Valid from: Sun Apr 22 15:32:25 CEST 2018 until: Wed Apr 19 15:32:25 CEST 2028

Certificate fingerprints:

MD5:  97:92:30:DF:1A:DD:85:8C:FC:09:E9:FA:70:12:4B:9B

SHA1: A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

SHA256: 97:3F:9E:0D:C2:76:5E:59:78:59:E4:19:85:56:C9:88:93:B0:71:36:07:48:71:A1:B4:00:F6:FF:9E:8D:A1:55

Signature algorithm name: SHA256withRSA

Version: 3

Extensions:

#1: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 96 5D E6 3E 6A 5E C4 F1   B2 9F 49 FC 7B 2A 85 21  .].>j^....I..\*.!

0010: C8 82 CA 8C                                        ....

]

]

Trust this certificate? [no]:  yes

Certificate was added to keystore

keytool -list -keystore /Users/andyhueni/git/dc10/src/client/specchio.keystore

Keystore type: JKS

Keystore provider: SUN

Your keystore contains 7 entries

dc\_10\_wollongong, May 24, 2016, trustedCertEntry,

Certificate fingerprint (SHA1): 02:FA:F3:E2:91:43:54:68:60:78:57:69:4D:F5:E4:5B:68:85:18:68

specchio\_rsl\_sc08, Apr 3, 2017, trustedCertEntry,

Certificate fingerprint (SHA1): 05:C4:21:EE:BB:18:60:4B:37:AB:0F:C5:79:B6:9F:8C:C1:DD:24:D9

specchio\_ah\_development\_machine, Sep 23, 2017, trustedCertEntry,

Certificate fingerprint (SHA1): 36:47:8E:AB:37:81:49:36:EE:2A:BC:02:E9:4A:93:A9:34:9D:14:54

specchio\_vm\_, Dec 10, 2015, trustedCertEntry,

Certificate fingerprint (SHA1): E4:30:72:FB:F8:BB:05:2C:B9:07:BE:40:81:A9:E3:E1:27:20:5F:D1

specchio\_rsl, Apr 28, 2014, trustedCertEntry,

Certificate fingerprint (SHA1): EC:76:30:0F:7E:8B:D5:58:89:2E:17:FD:60:1E:1D:D5:17:AD:6A:E2

specchio\_trust, Apr 22, 2018, trustedCertEntry,

Certificate fingerprint (SHA1): A7:37:6B:63:2A:A8:E0:1C:63:78:85:41:5F:A1:6A:BA:25:D5:0A:09

uow, May 25, 2016, trustedCertEntry,

Certificate fingerprint (SHA1): 5A:6F:1C:F2:4B:B8:96:90:57:66:C5:6E:AD:9D:90:12:82:EA:DB:4C

openssl s\_client -connect 192.168.56.101:8181 -CAfile SPECCHIO\_Trust.cer

CONNECTED(00000003)

depth=1 CN = SPECCHIO Trust

verify return:1

depth=0 CN = SPECCHIOVM2

verify return:1

---

Certificate chain

 0 s:/CN=SPECCHIOVM2

   i:/CN=SPECCHIO Trust

 1 s:/CN=SPECCHIO Trust

   i:/CN=SPECCHIO Trust

---

Server certificate

-----BEGIN CERTIFICATE-----

MIIEETCCAfmgAwIBAgIET2y3EzANBgkqhkiG9w0BAQsFADAZMRcwFQYDVQQDEw5T

UEVDQ0hJTyBUcnVzdDAeFw0xODA0MjIxMzM1NDVaFw0yODA0MTkxMzM1NDVaMBYx

FDASBgNVBAMTC1NQRUNDSElPVk0yMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB

CgKCAQEA0weULNV2jGu2FQoN8Z0/eyn/ZLQygnjP6MiqdA9CoEqxGdsA0cUm5YY6

IEYlzdQztLRLDSU/KCo/GsPcTZyn1a/nFz0T9D0y+7jX94BKP4zDCPkn9MnFzpPp

DHu84GdfVwkI3qjTzoQC9AAHpkuicd1guF+Yzo99FI3wMhTLWfoZPz0n3TxziTDp

7eYbui7gy50kb+jWhukKlzjiD56nTpLU/WauQHwDcd8xWt+RlP/7IaHFa0yp3PWb

v0k+t/HU3LxHeJBThCCURtakDOG4OR3/49XDyxfQ60bhYBQj0woJkJEDVwNP4Y3s

+g6a/BKFK9dd6/+QwYOZgZBxBe7E5wIDAQABo2QwYjAfBgNVHSMEGDAWgBSWXeY+

al7E8bKfSfx7KoUhyILKjDAgBgNVHREEGTAXgglsb2NhbGhvc3SHBH8AAAGHBMCo

OGUwHQYDVR0OBBYEFNRGr4qum0sVs3c2u64Fo8YRlAIrMA0GCSqGSIb3DQEBCwUA

A4ICAQARd0MqPFHV76FNPVj1PHoUjd3TezWjKXM7O19HdyJUrJOTXQ6o++VsNgyr

ZncXujFEUE1xbSc+BGDjeMtbb7hEpRlvrbGNFeaLgBeHSl+3jjm2ELsl7U474o5+

602mBjHuaQHYO9fIhWsvvnd+ZtXHq7rrHb14bxpDYSyTkxWBolPID3iEwJS3AF1D

HlgbiOy0a6dPY1HDwWsi/j44e23VSZj+I9sAV1LNDe8WbGYa5MLxfbUBt4vm6X8a

FznfS+WOMquzWuyxo2Rcd8u/qfER+qOYr9PX3ZSiXSp4DJkQ9GX3gq60hv+leRiq

VsRb/4j/9hn+msdo+XsK+kdW8DUsiOtGtOzla6QT1JXe+K3Z/O/1oK6LMBdVriFX

kbp+r+W/aXoIJwXXFbTMxXYBjJy077wx7ZhgzkeKqOB7lUSoe95HLaXDmTQcJxe6

g/Yq3NdBH04ScwigEjZkJkg3kByJdAd6tHBQ+Ql5VigBExJ2qQaUad8MkLGoXGsw

lOmPhV2dNY/g2q0nDOMd+Wl5fBlmqd9ctz2PLuORy4wgypDAezkYkVGiC5XOmISV

huOqoAT4+yBILOAYgwOQQONp03x0qKwJCgBpiXzVTML9toTqbvU5+C+C9H6LX+ft

/DHSWQDcY3c5KHEvW7CxJVSaCg0vjnztpKc5juUX2nFrWsqQWA==

-----END CERTIFICATE-----

subject=/CN=SPECCHIOVM2

issuer=/CN=SPECCHIO Trust

---

No client certificate CA names sent

Peer signing digest: SHA512

Server Temp Key: ECDH, P-256, 256 bits

---

SSL handshake has read 2769 bytes and written 433 bytes

---

New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES128-GCM-SHA256

Server public key is 2048 bit

Secure Renegotiation IS supported

Compression: NONE

Expansion: NONE

No ALPN negotiated

SSL-Session:

    Protocol  : TLSv1.2

    Cipher    : ECDHE-RSA-AES128-GCM-SHA256

    Session-ID: 5ADC92454D27486B73D70D5E1D7DEB39B6539392219773EE737D3B6F98CB9705

    Session-ID-ctx:

    Master-Key: DAF5BF470DFB47B9E3FE07E4C3073D16168B4698C88F3BC9CC8B8B088C3C942DDA02E9ABBD435F508303FB977CB73E3C

    Key-Arg   : None

    PSK identity: None

    PSK identity hint: None

    SRP username: None

    Start Time: 1524404805

    Timeout   : 300 (sec)

    Verify return code: 0 (ok)

---

CONNECTED(00000003)

depth=1 CN = SPECCHIO Trust

verify error:num=19:self signed certificate in certificate chain

---

Certificate chain

 0 s:/CN=SPECCHIOVM2

   i:/CN=SPECCHIO Trust

 1 s:/CN=SPECCHIO Trust

   i:/CN=SPECCHIO Trust

---

Server certificate

-----BEGIN CERTIFICATE-----

MIID7zCCAdegAwIBAgIENVM7hjANBgkqhkiG9w0BAQsFADAZMRcwFQYDVQQDEw5T

UEVDQ0hJTyBUcnVzdDAeFw0xODA0MTIwOTA1MzZaFw0yODA0MDkwOTA1MzZaMBYx

FDASBgNVBAMTC1NQRUNDSElPVk0yMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB

CgKCAQEAozFU6xMHQ+Vm5yVC4KTw5xOCqkqMgDSwgkWUyhbsz8KzAsV7KTw3RPxm

qNyAk7pB+ot0sXtSqzYX6L9pJZeoAOaCPXBzW5iAIANYzcC6aVOAPHFZhZjoeuMx

QBp7Zsx9ki+z8BxeUSHRcosVBGBr6Jmsq6ZKvexUC33E0ucN7l7PwrlozOZmjQ1I

JGXj0+kk2Z3k4i78jbLW+Gn0F7JrUQIZMT8jZnU/UTkZMsQ84Cp2M+zlTiQfvGbF

NfWN6BkyN9AwOfNRosnZ/XZ5XFsFP/WtXhKNL/gtZYHZd4e8HDMKMMrHRv0BsBPs

Z1qiDxKwTNAwcd6DjqwxdXRA0cD1hwIDAQABo0IwQDAfBgNVHSMEGDAWgBSABhh4

RKrW/32o2qGb+Ioy0fmqfjAdBgNVHQ4EFgQUrhRP9Yvu0QBYt4SSqhb52g8xIqcw

DQYJKoZIhvcNAQELBQADggIBAMiztfJ+bQh1xh+nYs4SE39Rdfm1zESRxCozvEdp

INHl+1C07zZomoiIk7wxPPA5TIjdsZzLo8ckaHCzf3QRxnIYrkaLiipQC52k60fa

yJOmLW4e3nfkHJ63Z28UFJvNO/0/pfAJ/qOmh/q2/br+ld+JkycxQBhoaMXwm50d

hZMxlQ9YusQlz5aCXrKp19wQuZkE2o54TGUQ7MAs6R7WxhKE2M8xZp6rF6vtIUNT

mLdkTgF6j3CJuAyS3++MzpX5RO4MJSR1C4JHPIAWP6toPs/U/r2oCGUWZL2cvWDR

mR7S9ZB9q/+pvtLHYVbJxmg7xdlQViPN4TX3uo8x7LaPYXRD1hoPKVTpHFDN/0xv

c4FDYCYXbmnGBP7boGAcmetbp0IQ/GVxmhdScFnCgKW7AUxlm0T1vlUpa+u2oKwz

kPZNMDD7jWY0d2byFbhWTKO9m4xX3UyHB5xDxbi7uJ73jyidGkK4hmxzk19lM+z1

NF6bCNOt+tMPVhWPMC9ln0TUwxk/KNDswQRx6RgS9hUhzXXLDR9N2TD5+/ijYxVu

+RIObswZhDrDOFzv2+eiO5VEFdu/7OIqFYd+RV/DbPgRzXhTrOvqVE9qA1ND2WAF

H8Gds9hlhKt+jvLPRfsXhTb3RNJlLCEFxQ82KsB0AwUHtHltjXmOFVfmV7cYKX9R

ZsOi

-----END CERTIFICATE-----

subject=/CN=SPECCHIOVM2

issuer=/CN=SPECCHIO Trust

---

No client certificate CA names sent

Peer signing digest: SHA512

Server Temp Key: ECDH, P-256, 256 bits

---

SSL handshake has read 2735 bytes and written 433 bytes

---

New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES128-GCM-SHA256

Server public key is 2048 bit

Secure Renegotiation IS supported

Compression: NONE

Expansion: NONE

No ALPN negotiated

SSL-Session:

    Protocol  : TLSv1.2

    Cipher    : ECDHE-RSA-AES128-GCM-SHA256

    Session-ID: 5ADB0CB7131B1D5B5EC2AC15B4A513427F8122B1B18753965ED502E2C6E4E03B

    Session-ID-ctx:

    Master-Key: 89ED757F6056C5C35DC7D31F99E924B1EDEE493B687EEC1AEAD3F1C3E001818F4151464BFF295A7DF5EEF7E586628DD7

    Key-Arg   : None

    PSK identity: None

    PSK identity hint: None

    SRP username: None

    Start Time: 1524305079

    Timeout   : 300 (sec)

    Verify return code: 19 (self signed certificate in certificate chain)

---

read:errno=0

# Test with SPECCHIO Client

https, 192.168.56.101, 8181, /specchio\_service, sdb\_admin, 5p3cch10\_SDB\_VM, jdbc/specchio,

javax.net.ssl.SSLHandshakeException: java.security.cert.CertificateException: No subject alternative names present

com.sun.jersey.client.urlconnection.URLConnectionClientHandler.handle(URLConnectionClientHandler.java:155)

# Troubleshooting

1. http://asaf.github.io/tech\_posts/export\_certificate\_and\_private\_key\_from\_jks/ [↑](#footnote-ref-1)
2. http://asaf.github.io/tech\_posts/export\_certificate\_and\_private\_key\_from\_jks/ [↑](#footnote-ref-2)