

Tomcat neues Zertifikat beantragen und mit Zertifizierungskette installieren

Version 1.0
19. Februar 2008

QuoVadis Trustlink Schweiz AG
Teufenerstrasse 11
9000 St. Gallen

Phone +41 71 272 60 60
Fax +41 71 272 60 61
www.quovadis.ch

Einleitung

Dieses Dokument beschreibt wie im Apache Tomcat ein CSR (Certificate Signin Request) für ein SSL Zertifikat angefordert wird und das Zertifikat anschliessend mit der gesamten Zertifizierungskette importiert wird. Diese Vorgehen ist mit dem Apache Tomcat 6.0 und Java Runtime Environment Version 1.5.0_14 getestet worden.

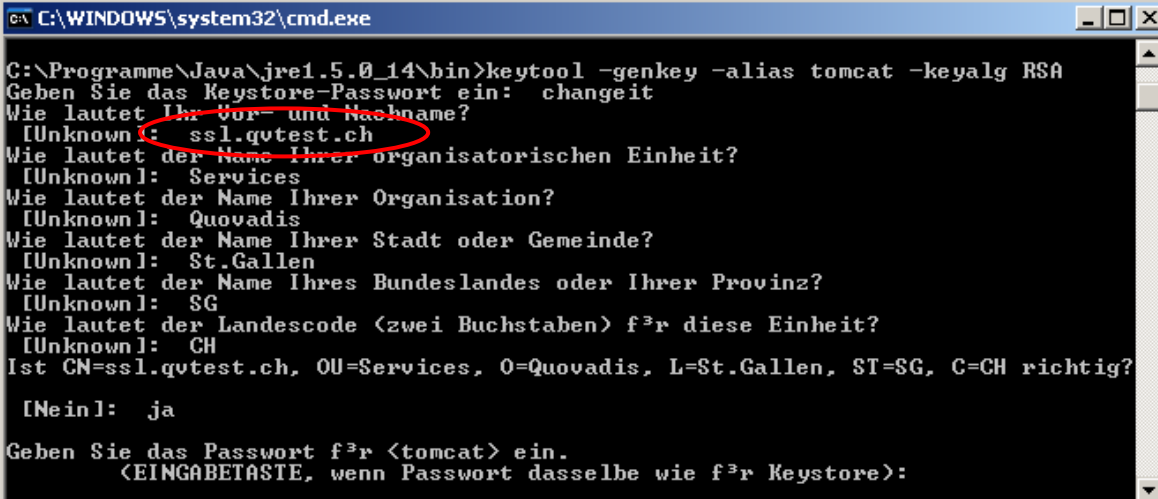
Übersicht Vorgehen

1. Key generieren mit keytool
2. Den CSR generieren (Certificate Signing Request)
3. Das Stamm- und Zwischenstellenzertifikat importieren
4. Das Serverzertifikat importieren
5. Apache Tomcat für SSL konfigurieren
6. Service unter dem Benutzer Administrator starten (nur Windows)

1. Key generieren mit keytool

Mit dem keytool aus dem Java JRE muss zuerst ein Key generiert werden. Dazu ist der folgende Befehl auszuführen:

```
keytool -genkey -alias tomcat -keyalg RSA
```



```
C:\WINDOWS\system32\cmd.exe
C:\Programme\Java\jre1.5.0_14\bin>keytool -genkey -alias tomcat -keyalg RSA
Geben Sie das Keystore-Passwort ein: changeit
Wie lautet Ihr Vor- und Nachname?
 [Unknown]: ssl.qvtest.ch
Wie lautet der Name Ihrer organisatorischen Einheit?
 [Unknown]: Services
Wie lautet der Name Ihrer Organisation?
 [Unknown]: Quovadis
Wie lautet der Name Ihrer Stadt oder Gemeinde?
 [Unknown]: St.Gallen
Wie lautet der Name Ihres Bundeslandes oder Ihrer Provinz?
 [Unknown]: SG
Wie lautet der Landescode (zwei Buchstaben) f³r diese Einheit?
 [Unknown]: CH
Ist CN=ssl.qvtest.ch, OU=Services, O=Quovadis, L=St.Gallen, ST=SG, C=CH richtig?
 [Nein]: ja
Geben Sie das Passwort f³r <tomcat> ein.
 <EINGABETASTE, wenn Passwort dasselbe wie f³r Keystore>:
```



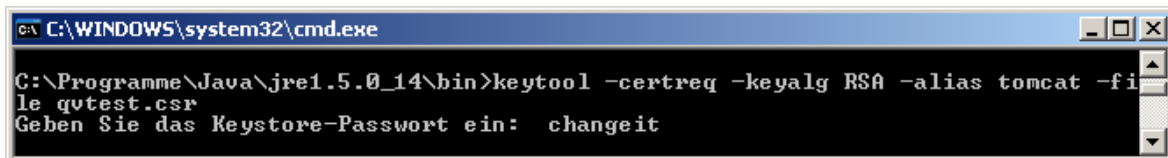
Achtung

Bei der Frage „Wie lautet Ihr Vor- und Nachname?“ muss die URL des Servers eingegeben werden. Ist dieser nicht korrekt wird das Zertifikat beim Browserzugriff als ung³ltig gemeldet.

2. Den CSR generieren (Certificate Signing Request)

Mit keytool kann ein CSR generiert werden. Dieser CSR wird benötigt um das Zertifikat zu erzeugen. Dazu ist der folgende Befehl auszuführen:

```
keytool -certreq -keyalg RSA -alias tomcat -file qvtest.csr
```



```
C:\WINDOWS\system32\cmd.exe
C:\Programme\Java\jre1.5.0_14\bin>keytool -certreq -keyalg RSA -alias tomcat -file qvtest.csr
Geben Sie das Keystore-Passwort ein: changeit
```

3. Stamm- und Zwischenstellenzertifikat importieren

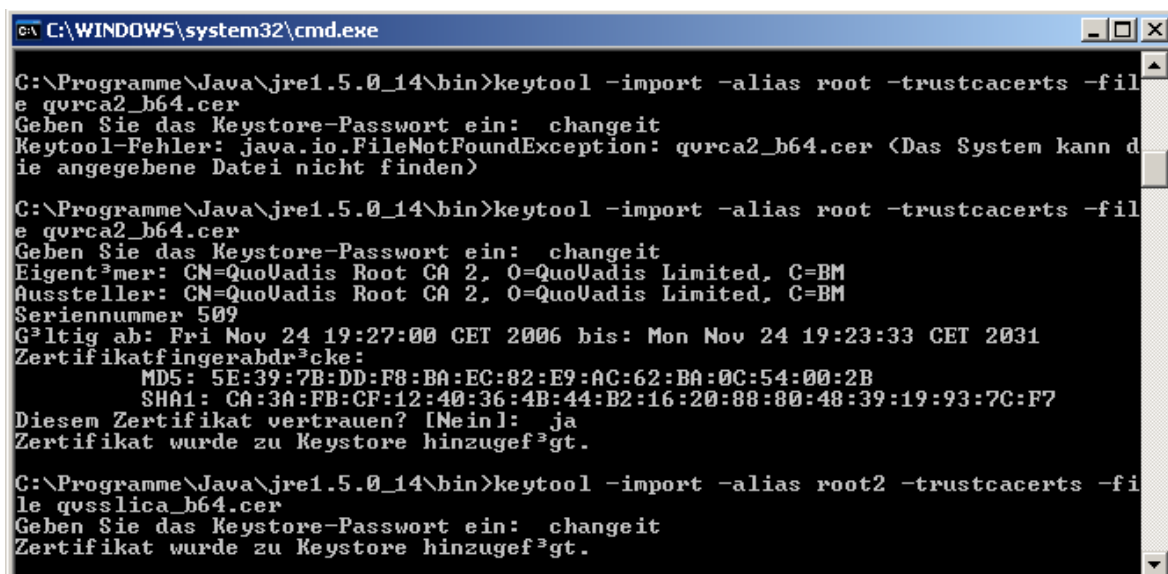
Um die Zertifizierungskette sicher zustellen muss das Stamm- und Zwischenstellen Zertifikat importiert werden. Diese Zertifikate können Sie unter:

http://downloads.quovadisglobal.com/certificates/qvrca2_b64.crt
http://downloads.quovadisglobal.com/certificates/qvsslica_b64.crt

herunterladen.

Mit den folgenden Befehlen werden die Zertifikate importiert:

```
keytool -import -alias root -trustcacerts -file qvrca2_b64.cer
keytool -import -alias root2 -trustcacerts -file qvsslica_b64.cer
```

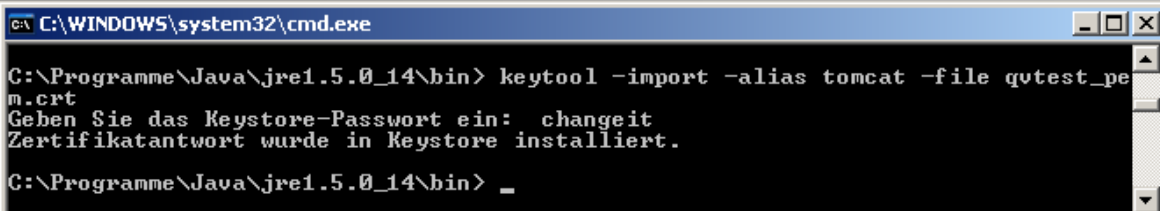


```
C:\WINDOWS\system32\cmd.exe
C:\Programme\Java\jre1.5.0_14\bin>keytool -import -alias root -trustcacerts -file qvrca2_b64.cer
Geben Sie das Keystore-Passwort ein: changeit
Keytool-Fehler: java.io.FileNotFoundException: qvrca2_b64.cer (Das System kann die angegebene Datei nicht finden)
C:\Programme\Java\jre1.5.0_14\bin>keytool -import -alias root -trustcacerts -file qvrca2_b64.cer
Geben Sie das Keystore-Passwort ein: changeit
Eigentümer: CN=QuoVadis Root CA 2, O=QuoVadis Limited, C=BM
Aussteller: CN=QuoVadis Root CA 2, O=QuoVadis Limited, C=BM
Seriennummer 509
Gültig ab: Fri Nov 24 19:27:00 CET 2006 bis: Mon Nov 24 19:23:33 CET 2031
Zertifikatfingerabdruck:
MD5: 5E:39:7B:DD:F8:BA:EC:82:E9:AC:62:BA:0C:54:00:2B
SHA1: CA:3A:FB:CF:12:40:36:4B:44:B2:16:20:88:80:48:39:19:93:7C:F7
Diesem Zertifikat vertrauen? [Nein]: ja
Zertifikat wurde zu Keystore hinzugefügt.
C:\Programme\Java\jre1.5.0_14\bin>keytool -import -alias root2 -trustcacerts -file qvsslica_b64.cer
Geben Sie das Keystore-Passwort ein: changeit
Zertifikat wurde zu Keystore hinzugefügt.
```

4. Das Serverzertifikat importieren

Nun muss noch das eigentliche SSL-Serverzertifikat importiert werden.
Dazu wird dieser Befehl verwendet:

```
keytool -import -alias tomcat -file qvtest_pem.crt
```

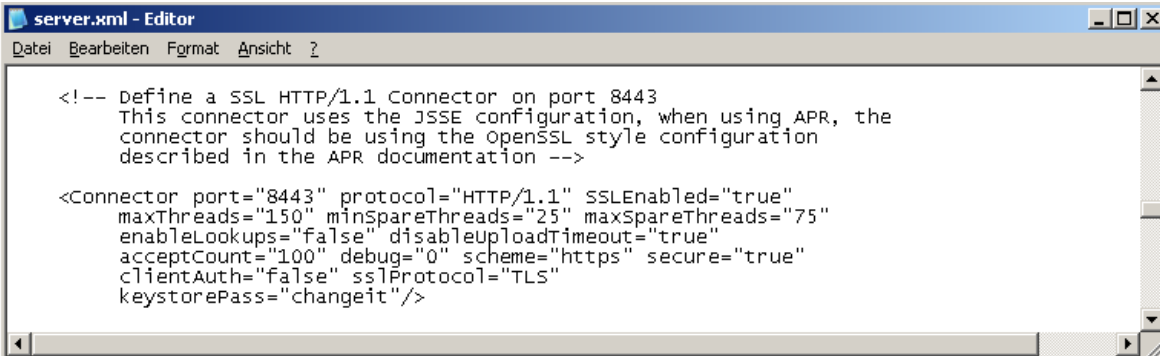


```
C:\WINDOWS\system32\cmd.exe
C:\Programme\Java\jre1.5.0_14\bin> keytool -import -alias tomcat -file qvtest_pem.crt
Geben Sie das Keystore-Passwort ein: changeit
Zertifikatantwort wurde in Keystore installiert.
C:\Programme\Java\jre1.5.0_14\bin> _
```

5. Apache Tomcat für SSL konfigurieren

Im Tomcat .../conf Verzeichnis befindet sich das File server.xml. Dieses File muss
gemäss dem folgendem Beispiel konfiguriert werden.

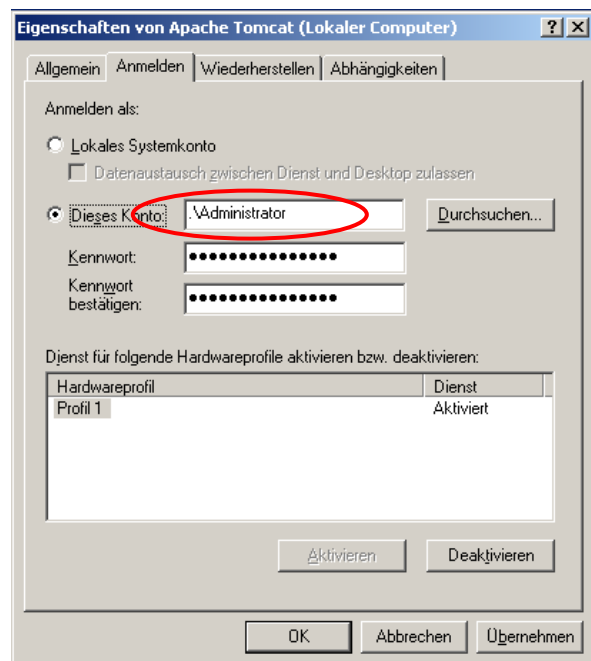
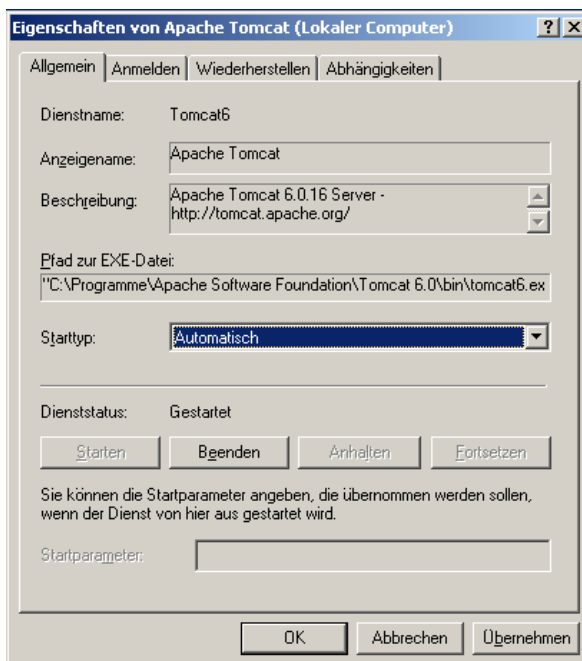
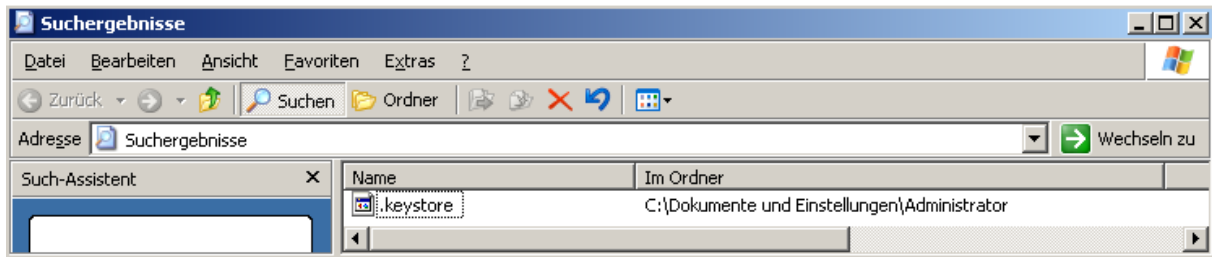
```
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" debug="0" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystorePass="changeit"/>
```



```
server.xml - Editor
Datei Bearbeiten Format Ansicht ?
<!-- Define a SSL HTTP/1.1 Connector on port 8443
This connector uses the JSSE configuration, when using APR, the
connector should be using the openssl style configuration
described in the APR documentation -->
<Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" debug="0" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystorePass="changeit"/>
```

6. Service unter dem Benutzer Administrator starten

Der Speicher für die Zertifikate (.keystore) wird im Home-Verzeichnis des Benutzers der bei der Generierung angemeldet ist abgelegt. Damit der Apache Tomcat Service Zugriff auf dieses File hat, muss der Service unter dem richtigen Benutzer (in diesem Fall der Administrator) gestartet werden.



Ab jetzt funktioniert der Apache Tomcat mit SSL.