

# Optionale Erweiterungen

- [Single-Sign-On: Auslagerung der web.xml](#)
- [SSL](#)

# Single-Sign-On: Auslagerung der web.xml

Ein häufiger Kundenwunsch ist die automatische Anmeldung am System via Single-Sign-On. Hierfür müssen Anpassungen in der **web.xml** vorgenommen werden, welche sich im **webapps**-Ordner des Tomcats befindet. Nach einem Documents-Update soll jedoch immer der **webapps**-Ordner gelöscht werden und dadurch gehen auch diese SSO-Anpassungen in der **web.xml** verloren.

Hierfür kann die **web.xml** in den **..\DEXPRO\WEB-INF**-Ordner kopiert werden. Bei einem Documents-Update gehen die Anpassungen an der Datei nicht mehr verloren. Der Nachteil ist allerdings, dass dadurch ggf. benötigte Erweiterungen an der **web.xml** manuell hinzugefügt werden müssen.

Die Auslagerung der **web.xml** ist ausdrücklich nur als Option zu sehen.

# SSL

Bei einer Umstellung auf https müssen alle Programme über dasselbe Protokoll miteinander kommunizieren! In der Regel sind folgende Programme betroffen und es werden folgende Zertifikate benötigt:

- **Tomcat8**

PFX / P12 Datei mit Kennwort. Eine PFX-Datei kann einfach in .p12 umbenannt werden!

```
<Connector port="443" protocol="org.apache.coyote.http11.Http11NioProtocol"
  maxThreads="200"
    SSLEnabled="true"
    scheme="https"
    secure="true"
  keystoreFile="D:\Applikationen\Documents5\tomcat8\conf\zertifikatsdatei.p12"
    keystorePass="ZertifikatPasswort"
  clientAuth="false"
    sslProtocol="TLS"
    maxPostSize="8388608"
    maxHttpHeaderSize="32768"
/>
```

- **TableService**

Zertifikat BASE64 codiert (PEM Datei, wobei die Endung keine große Rolle spielt)

Private Key (unverschlüsselt – kein RSA Key) ebenfalls BASE64 codiert

- **Squeeze**

Zertifikat BASE64 codiert (PEM Datei, wobei die Endung keine große Rolle spielt)

Private Key (unverschlüsselt – kein RSA Key) ebenfalls BASE64 codiert

Zertifikatskette (wenn vorhanden) ebenfalls BASE64 codiert.

Der Zertifikatsspeicher (PFX) bzw. das Zertifikat muss mindestens mit dem Signaturalgorithmus SHA-256 signiert sein.

Es kann bei veraltet konfigurierten CAs (Certificate Authority) vorkommen, dass ein Exportvorgang mit SHA-256 erfolgreich abgeschlossen, das Zertifikat jedoch immer noch mit SHA-1 signiert wurde. SHA-1 gilt seit 2017 als **NICHT SICHER!**

Der Kunde soll einen PFX Zertifikatsspeicher bereitstellen und das Passwort mitteilen. Gegebenenfalls wird im DEXPRO-Ordner bereits der Unterordner **OpenSSL** ausgeliefert. Der Ordner enthält die openssl.exe und 2 DLL's sowie eine **ExtractPfx.bat**. In der BAT-Datei müssen

lediglich der Dateiname, das Passwort und der Pfad angepasst werden. Die PFX-Datei muss in den Unterordner "IN" abgelegt werden.

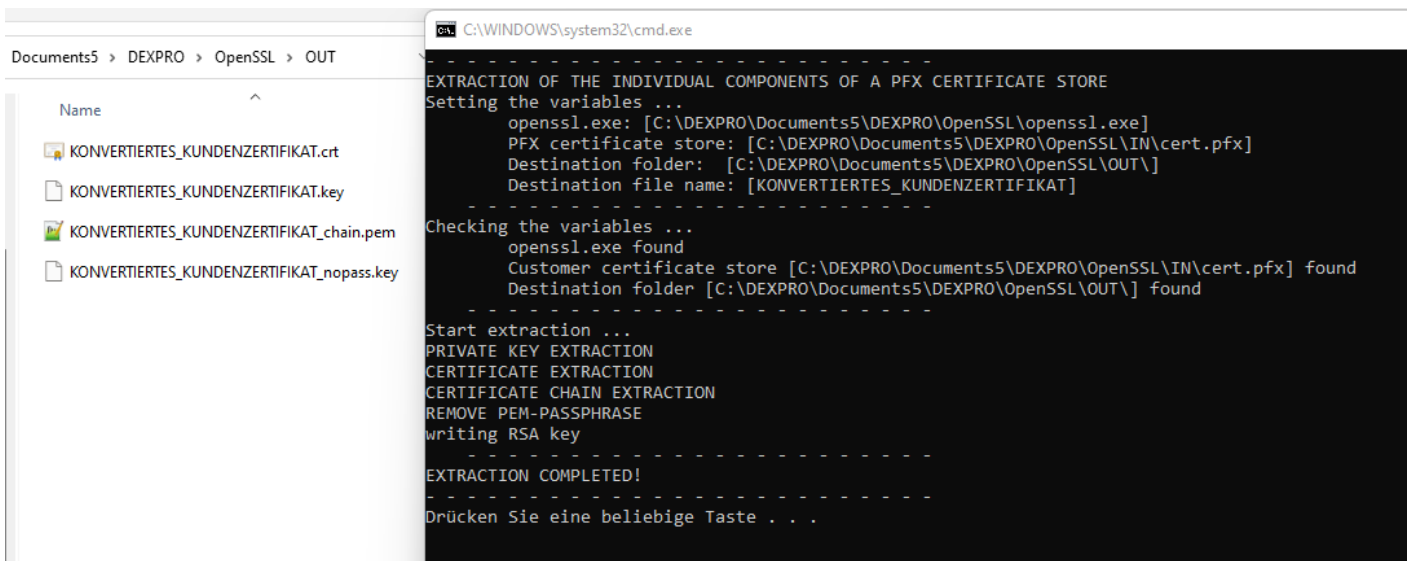
```
REM VARIABLES TO SET!  
SET FILENAME=cert.pfx  
SET PW=mySecretPW  
SET PATH=D:\Documents5\DEXPRO\OpenSSL\
```

## OPENSSL BINARIES

Durch die Extraktion werden die folgenden Dateien erstellt:

- Das eigentliche Zertifikat ist die `.cert` Datei.
- Der Private Key (mit PEM-Passphrase) ist `.key` Datei.
- Die Zertifikatskette ist die `_chain.pem` Datei, wenn diese leer ist, dann gibt es keine weitere Kette.
- Der Private Key (ohne PEM-Passphrase) ist die `_nopass.key` Datei.

Die **ExtrctPfx.bat** sollte mit Administrator-Berechtigungen ausgeführt werden. Das Ergebnis sollte wie im folgenden Screenshot zu sehen aussehen.



Die konvertierte crt- und die ky-Datei enthalten zusätzliche Informationen, welche manuell aus der Datei gelöscht werden müssen. Die Dateien können beliebig umbenannt und verwendet werden.

