SSL-Verschlüsselung

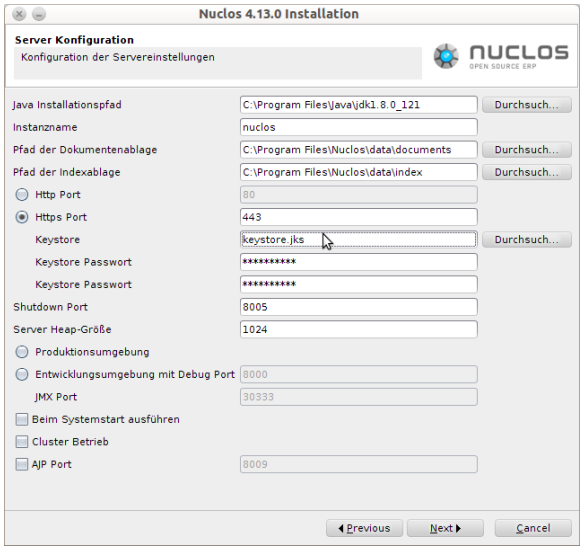
Es ist möglich, Nuclos mit SSL-Verschlüsselung laufen zu lassen. Dafür benötigen Sie ein SSL-Zertikat, dass in einen Java-Keystore eingebunden ist. Sollte Ihnen das SSL-Zertifikat noch nicht in einem Java-Keystore vorliegen, lesen Sie bitte unten auf dieser Seite nach, wie Sie einen Keystore erzeugen können.

> Es ist unbedingt zu empfehlen, ein von einer offiziellen Stelle signiertes Zertifikat zu verwenden. Selbst signierte Zertifikate werden von Java als unzuverlässig und somit als Sicherheitsrisiko eingestuft.

Ausführen des Nuclos-Installers

Liegt Ihnen das SSL-Zertifikat bereits in einem Java-Keystore vor, so müssen Sie auf Ihrer Nuclos-Instanz nur einmal erneut den Nuclos-Installer, der derzeit im Einsatz befindlichen Nuclos-Version ausführen, um die SSL-Verschlüsselung zu aktivieren.

Im Schritt "Server Konfiguration" aktivieren Sie dafür bitte den "Https Port" (die Portnummer können Sie frei vergeben; 443 ist der SSL-Standardport) und wählen den Ihnen vorliegende Java-Keystore aus. Sie müssten dabei das Passwort des Keystores eintragen und einmal bestätigen.



Fertig! Nach Abschluss des Installers und einem Start des Nuclos-Dienstes läuft Ihre Nuclos-Instanz mit SSL-Verschlüsselung.

> Bitte verifizieren Sie Ihr Zertifkat, wie es auf dieser Seite im unteren Abschnitt beschrieben ist, sollte das Zertifikat beim Aufruf der Nuclos-Webstartadresse oder beim Ausführen des Desktop-Clients nicht erkannt werden (bzw. als unzuverlässig oder ungültig eingestuft werden).

Erzeugen eines Java-Keystores

Liegt Ihnen noch kein Java-Keystore vor, so müssen Sie diesen zunächst aus den SSL-Zertifikaten erzeugen. Im folgenden Beispiel besteht das SSL-Zertifikat aus

- einem Root-Zerfikat *root.crt*,
- zwei Intermediate-Zerfikaten *chain1.crt* und *chain2.crt*
- sowie einem privaten Schlüssel *private.key*.

```
root.crt
chain1.crt
chain2.crt
private.key
```

Das Root-Zertifikat und die Intermediate-Zertifikate müssen zuerst zu einer Datei zusammengefügt werden. Das geht z.B. einfach über den Konkatenierbefehl **cat** auf der Kommandozeile. Es ist aber auch möglich, dies in einem Texteditor durchzuführen.

```
cat root.crt chain1.crt chain2.crt > certificate.
crt
```

Sind in Ihrem Fall keine Intermediate-Zertifkate vorhanden, dann kann dieser Schritt übersprungen werden.

Jetzt kann das zusammengefügte Zertifkat mit **openssl** verschlüsselt werden. Dabei können Sie über die Option "-name" einen Namen vergeben.

Nach dem Absetzen des Kommandos werden Sie dazu aufgefordert, ein Passwort für den Keystore zu vergeben und dieses einmal zu bestätigen.

```
openssl pkcs12 -export -inkey private.key -in
certificate.crt -name certificate_name -out
keystore.p12
```

- Es kann notwendig sein, dass die Erzeugung des Keystores auf dem Server selbst durchgeführt werden muss (d.h. nicht auf einem anderen Rechner).

Anschließend wird der Keystore im p12-Format noch mit dem Java-Keytool in einen Java-Keystore exportiert.

Nach dem Absetzen müssen Sie das Passwort des p12-Keystores eingeben.

Schließlich werden Sie dazu aufgefordert, ein Passwort für den Java-Keystore zu vergeben und dieses einmal zu bestätigen.

```
keytool -importkeystore -srckeystore keystore.p12 -
srcstoretype pkcs12 -destkeystore keystore.jks
```

Erzeugen eines selbst signierten Zertifikats zur Verwendung im Launcher

Um ein selbst signiertes Zertifikat in Verbindung mit dem Nuclos-Launcher verwenden zu können, müssen folgende Bedingungen erfüllt sein:

- Das Zertifikat muss mit einem passende "subject alternative name" erzeugt worden sein.
- Das Zertifikat muss dem Truststore des Launchers manuell hinzugefügt werden

```
keytool -genkey -keyalg RSA -alias localhost -
keystore keystore.jks -storepass password -keysize
2048 -ext SAN=dns:localhost,ip:127.0.0.1 -dname
"CN=test, OU=Unknown, O=Unknown, L=Unknown,
ST=Unknown, C=Unknown"
```

```
keytool -export -alias localhost -file localhost.
cer -keystore keystore.jks
```

```
keytool -import -v -trustcacerts -alias localhost -
file localhost.cer -keystore
"<path_to_launcher>\cacerts"
```

Verifizierung des Java-Keystores

Es ist möglich, den Java-Keystore über die Kommandozeile zu verifizieren.

Mit dem Java-Keytool können Sie sich die Inhalte des Keystores auf der Kommandozeile anzeigen lassen.

```
keytool -v -list -keystore keystore.jks
```