**Purpose:**

To install security certificates and modify related jetty files to permit SSL (secure sockets layer), http/2 and https access. Several applications are involved:

* OpenSSL is a library that provides cryptographic functionality to applications such as secure web servers. OpenSSL includes a command line utility that can be used to perform a variety of cryptographic functions. It is described in the openssl manpage. OpenSSL manpages are installed in /usr/local/ssl/man.
* The keytool application manages a keystore (database) of cryptographic keys, X.509 certificate chains, and trusted certificates. Keytool is bundled with the JDK. You can use either Keytool or OpenSSL tools to generate keys and certificates.

Creating secure login involves three steps:

* configure the SSL keystore password and save
* add SSL (secure sockets layer transport) and https modules to start.ini
* modify start.ini for ssl listen port, and update Keystore password

The following process follows that documented by Rishi Khandelwal (2013Sept10) and the Jetty manual.

| **Step** | **Major Activity** | **References, Forms and Details** |
| --- | --- | --- |
| **1** | cd /*usr/*share/jetty9/etc  sudo rm keystore | Move to etc directory  Remove any existing keystore entry |
| **2** | keytool -keystore keystore -alias jetty -genkey -keyalg RSA -sigalg SHA256withRSA –validity 5000 -ext 'SAN=dns:jetty.eclipse.org,dns:\*.jetty.org' | The following command generates a key pair and certificate directly into file keystore  Add -validity <days> to the keytool call above, otherwise your certificate is only valid for one month  Use the SAN extension to set one or more names that the certificate applies to  This command prompts for information about the certificate and for passwords to protect both the keystore and the keys within it.  The only mandatory response is to provide the fully qualified host name of the server at the "first and last name" prompt (e.g. jetty.eclipse.org)  The browser will not trust the certificate you have generated, and prompts the user to this effect. |
| **3** | keytool -certreq -alias jetty -keystore keystore -file jetty.csr | This command generates the file jetty.csr using keytool for a key/cert already in the keystore |
| **4** |  |  |
| **5** | cd /usr/share/jetty9/lib  java -cp jetty-util-9.4.26.v20200117.jar org.eclipse.jetty.util.security.Password <password> | Obfuscate <password>  NOTE: adjust references to the specific jetty-util jar.  Save the OBF password |
| **6** | cd $JETTY\_BASE  keytool -list \  -keystore etc/keystore \  -storetype jks \  -storepass ''<password>” \  -v | Confirm the keystore contents  Change directory to JETTY\_BASE  Provide in plain format presumed <password>  (quotes optional)  -v for verbose output |
| **7** | sudo java -jar start.jar --add-to-start=ssl | Should respond:  INFO: ssl initialized in  ${jetty.base}/start.ini  INFO: Base directory was modified |
| **8** | Replace first default keystore password in start.ini with saved password | Can be obfuscated or not. This is not encrypted in either case. |
| **9** | sudo java -jar start.jar --add-to-start=https |  |