Using a self signed certificate in the Industrial HiVision web server

Hendrik Lepple - 2022-04-07 - Industrial HiVision

If you want to use your own self signed certificate in the web server of Industrial HiVision, you will need to go through the steps in this article.

1. Prerequisites:
   - Activate HTTPS for the Industrial HiVision web server. You can do that in the preferences of Industrial HiVision under 'Services Access'. Enable the checkbox and set the protocol to HTTPS.
   - Backup the currently used keystore. You can find this keystore in the folder 'lib' under the installation directory. The file is called 'keystore' without any file extension. Copy this file to a safe location on your PC.
   - Make sure your Industrial HiVision services are stopped from now on!
2. Create a new 'keystore' file
   - Open a console window and navigate into the following path under the installation directory of Industrial HiVision: */lib/java_amd64/bin*
   - Execute the following command:
   keytool -genkey -alias hirschmann -keyalg RSA -keysize 2048 -keystore keystore -keypass password -storepass password
   (Please note: You cannot change the alias or password here because those values are hard coded inside the web server of Industrial HiVision, which will run with the final certificate and keystore later)
   - Enter the following data as an answer to the first question when prompted:
   -- IP address of your computer
   OR
   -- the hostname of your computer
   OR
   -- the URL/domain to access your computer
   depending on how you are going to access your web server later.
   - You will then find a 'keystore' file in this folder to work with from now on
3. Create a CSR from the new keystore
   - Execute the following command to get a certificate signing request file called 'hirschmann.csr' that you can sign with your CA;
   keytool -certreq -alias hirschmann -file hirschmann.csr -keystore keystore -storepass password

4. Sign the CSR with your CA

   - Send the file 'hirschmann.csr' to your trusted CA for signing.

   - You should get a file back with the '.crt' file extenstion. In this article I will refer to it as 'certificate.crt' from now on for demonstration purposes. You will of course need to replace 'certificate.crt' in the following commands if your file is named differently.

   - Copy the 'certificate.csr' into the folder where youhave your console window open to access it there.

5. Trust the root CA

   - This step is neccessary to establish trust through the whole signing chain of your CA and the certificate you signed as Industrial HiVision is not aware of any CAs other than it's own. You will need the root certificate of your CA here, I will refer to it as 'caRootCertificate'. Please replace it with the name of your corresponding root certificate.

   - Execute the following command to trust the root CA:
   keytool -import -trustcacerts -alias root -file caRootCertificate.crt -keystore keystore -storepass password

   - Execute the following command to import the signed certificate into the web server of Industrial HiVision:
   keytool -importcert -alias hirschmann -file certificate.crt -keystore keystore -storepass password

6. Copy the new keystore into the correct location and start the services

   - Take the 'keystore' file from the path you created it (was */lib/java_amd64/bin* under the installation directory) and copy it to the folder */lib* under the installation directory. You should replace the old 'keystore' file if it is still present. (We backed that up earlier)

   - Start the Industrial HiVision services again

   - Import the root CA certificate into your browser

   - Try to access your Industrial HiVision web server and check whether the certificate is used correctly

7. Troubleshooting

   - 'BAD_CERT_DOMAIN' is displayed - this means that the name of the domain you are accessing is not included in the certificate. (see steps under point #2)

   - 'SEC_ERROR_UNKNOWN_ISSUER' is displayed - you haven't imported and trusted the root CA certificate correctly. Please note that your browser and your operating system could use different stores for their trusted certificates, importing it into the operating systems store does not always necessarily mean it is also trusted and used by the browser.