Application Note 07/2025 AP050017DE

Verwalten von SSL Zertifikaten in Codesys 3

Level 4	1 – Fundamental – keine weiteren Kenntnisse nötig 2 – Basic – Grundwissen empfehlenswert 3 – Fortgeschritten – Grundwissen notwendig 4 – Expert – Praxiserfahrung in dem Thema empfehlenswert

Powering Business Worldwide

Services

Für Service und Support kontaktieren Sie bitte Ihre lokale Vertriebsorganisation.

Kontaktdaten: Eaton.com/contacts
Service Seite: Eaton.com/aftersales

Original Application Note

Die deutsche Ausführung dieser Application Note ist das Original.

Übersetzung des Originaldokuments

Alle nicht deutschen Sprachausgaben dieses Application Note sind Übersetzungen der
Original Application Note.

1. Auflage 2025, Redaktionsdatum 07/2025

GEFAHR!
GEFÄHRLICHE ELEKTRISCHE SPANNUNG!

Vor Beginn der Installationsarbeiten

- Installation erfordert Elektro-Fachkraft.
- Gerät spannungsfrei schalten
- Gegen Wiedereinschalten sichern
- Spannungsfreiheit feststellen
- Erden und kurzschließen
- Benachbarte, unter Spannung stehende Teile abdecken oder abschranken.
- Die für das Gerät angegebenen Montagehinweise (IL) sind zu beachten.
- Nur gemäß EN 50110-1/-2 (VDE 0105 Teil 100) qualifiziertes Personal darf Eingriffe an diesem Gerät/System vornehmen.
- Achten Sie bei Installationsarbeiten darauf, dass Sie sich statisch entladen, bevor Sie das Gerät berühren.
- Die Funktionserde (FE, PES) muss an die Schutzerde (PE) oder den Potentialausgleich angeschlossen werden. Die Ausführung dieser Verbindung liegt in der Verantwortung des Errichters.
- Anschluss- und Signalleitungen sind so zu installieren, dass induktive und kapazitive Einstreuungen keine Beeinträchtigung der Automatisierungsfunktionen verursachen.
- Einrichtungen der Automatisierungstechnik und deren Bedienelemente sind so einzubauen, dass sie gegen unbeabsichtigte Betätigung geschützt sind.
- Damit ein Leitungs- oder Aderbruch auf der Signalseite nicht zu undefinierten Zuständen in der Automatisierungseinrichtung führen kann, sind bei der E/A-Kopplung hard- und softwareseitig entsprechende Sicherheitsvorkehrungen zu treffen.
- Bei 24-Volt-Versorgung ist auf eine sichere elektrische Trennung der Kleinspannung zu achten. Es dürfen nur Netzgeräte verwendet werden, die die Forderungen der IEC 60364-4-41 bzw. HD 384.4.41 S2 (VDE 0100 Teil 410) erfüllen.
- Schwankungen bzw. Abweichungen der Netzspannung vom Nennwert dürfen die in den technischen Daten angegebenen Toleranzgrenzen nicht überschreiten. Andernfalls sind Funktionsausfälle und Gefahrenzustände nicht auszuschließen.
- NOT-AUS-Einrichtungen nach IEC/EN 60204-1 müssen in allen Betriebsarten der Automatisierungseinrichtung wirksam bleiben. Entriegeln der NOT-AUS-Einrichtungen darf keinen Wiederanlauf bewirken.
- Einbaugeräte für Gehäuse oder Schränke dürfen nur im eingebauten Zustand, Tischgeräte oder Portables nur bei geschlossenem Gehäuse betrieben und bedient werden.
- Es sind Vorkehrungen zu treffen, dass nach Spannungseinbrüchen und -ausfällen ein unterbrochenes Programm ordnungsgemäß wiederaufgenommen werden kann. Dabei dürfen auch kurzzeitig keine gefährlichen Betriebszustände auftreten. Ggf. ist NOT-AUS zu erzwingen.
- An Orten, an denen in der Automatisierungseinrichtung auftretende Fehler Personen- oder Sachschäden verursachen können, müssen externe Vorkehrungen getroffen werden, die auch im Fehler- oder Störfall einen sicheren Betriebszustand gewährleisten beziehungsweise erzwingen (z. B. durch unabhängige Grenzwertschalter, mechanische Verriegelungen usw.).

Gewährleistungsausschluss und Haftungsbeschränkung

Die Informationen, Empfehlungen, Beschreibungen und Sicherheitshinweise in diesem Dokument basieren auf den Erfahrungen und Einschätzungen der Eaton Corp. Und berücksichtigen möglicherweise nicht alle Eventualitäten.

Wenn Sie weitere Informationen benötigen, wenden Sie sich bitte an ein Verkaufsbüro von Eaton. Der Verkauf der in diesen Unterlagen dargestellten Produkte erfolgt zu den Bedingungen und Konditionen, die in den entsprechenden Verkaufsrichtlinien von Eaton oder sonstigen vertraglichen Vereinbarungen zwischen Eaton und dem Käufer enthalten sind. Es existieren keine Abreden, Vereinbarungen, Gewährleistungen ausdrücklicher oder stillschweigender Art, einschließlich einer Gewährleistung der Eignung für einen bestimmten Zweck oder der Marktgängigkeit, außer soweit in einem bestehenden Vertrag zwischen den Parteien ausdrücklich vereinbart. Jeder solche Vertrag stellt die Verpflichtung von Eaton abschließend dar.

Der Inhalt dieses Dokumentes wird weder Bestandteil eines Vertrages zwischen den Parteien noch führt er zu dessen Änderung. Eaton übernimmt gegenüber dem Käufer oder Nutzer in keinem Fall eine vertragliche, deliktische (einschließlich Fahrlässigkeit), verschuldensunabhängige oder sonstige Haftung für außergewöhnliche, indirekte oder mittelbare Schäden, Folgeschäden bzw. –verluste irgendeiner Art – unter anderem einschließlich, aber nicht beschränkt auf Schäden an bzw. Nutzungsausfälle von Geräten, Anlagen oder Stromanlagen, von Vermögensschäden, Stromausfällen, Zusatzkosten in Verbindung mit der Nutzung bestehender Stromanlagen, oder Schadensersatzforderungen gegenüber dem Käufer oder Nutzer durch deren Kunden – infolge der Verwendung der hierin enthaltenen Informationen, Empfehlungen und Beschreibungen. Wir behalten uns Änderungen der in diesem Handbuch enthaltenen Informationen vor. Fotos und Abbildungen dienen lediglich als Hinweis und begründen keine Verpflichtung oder Haftung seitens Eaton.

Inhalt

1 SSL – Secure Sockets Layer / TLS – Transport Layer Security 6
1.1 Asymmetrischer und Symmetrischer Schlüssel 6
1.2 SSL-Zertifikate 7
2 Erstellen von Zertifikaten in CoDeSys 7
3 Verschlüsselte Applikation 10
4 Verschlüsselte Online-Kommunikation 13
5 Verschlüsselter OPC-UA Server Zugriff 16
6 Verschlüsselter Webserver-Zugriff 18

1 SSL – Secure Sockets Layer / TLS – Transport Layer Security

SSL steht für “Secure Sockets Layer” und ist eine Standardtechnologie, um eine verschlüsselte Verbindung zwischen einem Server und einem Client aufzubauen. Oft wird die neue Variante TLS (Transport Layer Security) auch als SSL bezeichnet, jedoch bietet die aktualisierte Version höhere Sicherheit. Mit Hilfe der SSL-Technologie können zwei Systeme Daten verschlüsselt übertragen, so dass die Nachrichten nicht von Dritten mitgelesen werden können. Hierfür werden Zertifikate und Schlüssel benötigt, die wiederum Verschlüsselungsalgorithmen nutzen, um die Daten zu verschlüsseln.

1.1 Asymmetrischer und Symmetrischer Schlüssel

Um eine SSL verschlüsselte Kommunikation aufzubauen, werden zwei Verschlüsselungsmethoden angewendet.

- **Asymmetrische Verschlüsselung / Public-Key Verschlüsselung**

Nutzt zwei unterschiedliche Schlüssel für die Ver- und Entschlüsselung. Der Schlüssel für das Verschlüsseln einer Nachricht (Public-Key) kann von jedem genutzt werden. Diese verschlüsselte Nachricht kann dann aber nur mit dem Schlüssel für die Entschlüsselung (Private Key) entschlüsselt und gelesen werden. Häufigster Verschlüsselungsalgorithmus ist RSA, ECC oder Diffie-Hellman.

- **Symmetrische Verschlüsselung / Pre-Shared Key Verschlüsselung**

Nutzt einen Schlüssel für die Ver- und Entschlüsselung. Dieser Schlüssel muss dem Sender und dem Empfänger zur Verfügung stehen, damit Nachrichten sinnvoll ausgetauscht werden können.

Der verschlüsselte Kommunikationsaufbau lässt sich damit grob in vier Schritte zusammenfassen.

1. Nach der Anfrage vom Client, sendet der Server eine Kopie seines SSL-Zertifikats inkl. einem asymmetrischen öffentlichen Schlüssels zum Client.
2. Client prüft das Zertifikat und erstellt einen symmetrischen Schlüssel, verschlüsselt diesen mit dem asymmetrischen öffentlichen Schlüssel und schickt ihn an den Server.
3. Der Server entschlüsselt die Nachricht mit seinem asymmetrischen privaten Schlüssel, um den symmetrischen Schlüssel zu erhalten und bestätigt dies dem Client.
4. Server und Client verschlüsseln und entschlüsseln nun alle übertragenen Daten mit dem symmetrischen Schlüssel. Dadurch ist eine verschlüsselte Kommunikation für diese Verbindung möglich. Bei einem erneuten Verbindungsaufbau wird ein neuer Schlüssel erstellt.

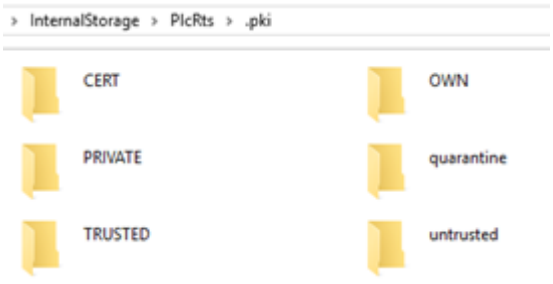
1.2 SSL-Zertifikate

SSL-Zertifikate sind kleine Dateien mit einer digitalen Bindung des Schlüssels zu einer Organisation. Auf einem Webserver führt das Zertifikat dazu, dass das https-Protokoll verwendet wird und somit eine sichere Verbindung von einem Webserver zu einem Browser möglich ist. Im Browser wird eine sichere Verbindung durch das Schloss visuell dargestellt.

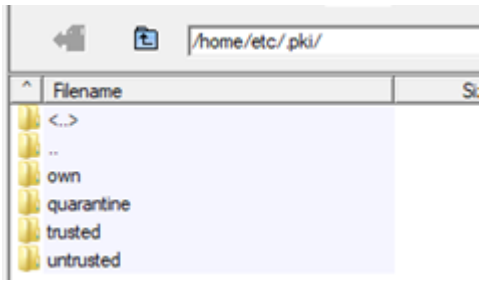
2 Erstellen von Zertifikaten in CoDeSys

CoDeSys bietet die Möglichkeit individuelle Security Features zu konfigurieren. Hierfür steht der „Security Screen“ zur Verfügung, in dem die Verwendung von Zertifikaten und die Verschlüsselungen eingestellt werden können, um den Zugriff auf Projektdaten nur mittels einer digitalen Signatur zu ermöglichen. CoDeSys nutzt für die Security Features Zertifikate im X.509 Format, die im Verzeichnis PKI abgelegt werden. Um eine Verschlüsselung zu ermöglichen müssen die Geräte OpenSSL unterstützen und über die Komponente CmpSecurityManager verfügen.

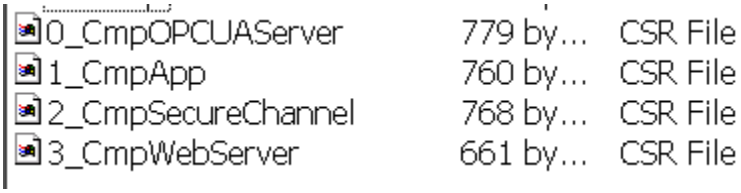
XV300:



XC300:



Es gibt zwei Arten von Zertifikaten, die in diesem Zusammenhang von der Runtime verwendet werden können. Zum einen selbstsignierte Zertifikate, die z.B. über die PLC-Shell generiert werden können, oder CA signierte Zertifikate, die von einer offiziellen Stelle (z.B. VeriSign, GeoTrust, …) generiert oder signiert werden. Hierfür muss für die entsprechende Anwendung in der PLC Shell der Befehl „cert-createcsr“ ausgeführt werden, wodurch ein CSR (Certificate Signing Request) erstellt und in dem Verzeichnis cert/export abgelegt wird.



Die CSR-Datei kann per File-Transfer übertragen und an eine Zertifizierungsstelle gesendet werden. Nachdem das Zertifikat signiert wurde, kann dieses per File-Transfer wieder auf das Gerät in das Verzeichnis cert/import übertragen und mit dem PLC-Shell Befehl „cert-import“ in die Runtime importiert werden.

Nachfolgend werden die vorhandenen PLC Shell Befehle zum Thema Security aufgeführt:

- cert-getapplist

Zeigt an, für welche Komponenten bereits ein Zertifikat registriert ist. Außerdem erhält man Informationen bzgl. der Laufzeit des Zertifikates und die ID-Nummer der Komponenten. Diese wird für die Verwendung anderer Befehle benötigt.

```
cert-getapplist

Nr.    ComponentName              CommonName                      CertAvailable DateNotBefore
-------------------------------------------------------------------------------------------------
0      CmpOPCUAServer             OPCUAServer@WM01                FALSE         --
1      CmpApp                     WM01                      FALSE         --
2      CmpSecureChannel           WM01                      TRUE          2021-1-13T22:58:57.0
3      CmpWebServer               WM01                      TRUE          2021-1-13T22:58:57.0
```

- cert-genselfsigned <number of the component in the applist>

Mit diesem Befehl können selbstsignierte Zertifikate für die jeweiligen Komponenten erstellt werden. Eine erfolgreiche Generierung der Zertifikate kann im PLC-Log eingesehen werden.

```
cert-genselfsigned 0

Generate selfsigned certificate with given index (0). Check logger to see when finished.
```

- cert-gendhparams [len in bits]

Erzeugt die Parameter für den Diffie-Hellman Schlüsselaustausch. Achtung: Dieser Vorgang kann mehrere Minuten dauern!

```
cert-gendhparams

Generation of the parameter started. This may take SEVERAL hours.
Please do not turn off the PLC.
Check the logger if the generation has finished.
```

- cert-getcertlist [<trustlevel>]

Listet alles Zertifikate der ausgewählten Vertrauensstufe auf. Wird diese nicht angegeben, so werden alle Zertifikate angezeigt. Mögliche Vertrauensstufen sind

  - untrusted (nicht vertrauenswürdig)
  - trusted (vertrauenswürdig)

- own (Zertifikate der Steuerung)
- quarantine (Zertifikate konnten nicht validiert werden)

```
cert-getcertlist


-------------------------------------------------------
List of own certificates
-------------------------------------------------------
Number: 0
Thumbprint: 401e6fb676da07ec5516f4765b62512c74be970c
Subjects:
 - commonName: OPCUAServer@WM01
Key usage(s): Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment, Certificate Sign
Extended key usage(s): TLS Web Server Authentication
Valid from: 27.1.2021 0:49:47
Valid until: 27.1.2022 0:49:47
-------------------------------------------------------
Number: 1
Thumbprint: 8b5793f43754450e1a2c69a5e70543e723ebf5bf
Subjects:
 - commonName: WM01
 - unstructuredName: Vendor: Eaton Automation
 - unstructuredName: Device: Plc3 V3.5.16.20-5427 192.168.119.133
 - serialNumber: 101200001813
Extended key usage(s): TLS Web Server Authentication
Valid from: 13.1.2021 22:58:57
Valid until: 12.2.2021 22:58:57
```

- cert-createcsr [<number retrieved by “cert-getapplist”>]

Generiert CSR-Dateien für die Komponenten.

```
cert-createcsr 2

Create CSR for application with given index. Check logger to see when finished.
```

- cert-import <trustlevel> <filename>

Importiert das angegebene Zertifikat aus dem Verzeichnis cert/import

```
cert-import own signature.cer

Import Succeeded.
```

- cert-export <trustlevel> [<number retrieved by "cert-getcertlist">]

Exportiert das angegebene Zertifikat in das Verzeichnis cert/export

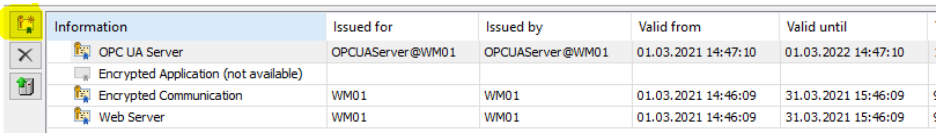
```
cert-export own 0

Export certificate Nr. 0 of specified trustlevel.
Certificate Nr. 0 exported to $cert$/export/4c5761fe84cfa941ad5da34b22d0dccd59a8d8be.cer.
```

- cert-remove <trustlevel> <number retrieved by "cert-getcertlist" or "all">

Entfernt die angegebenen Zertifikate

```
cert-remove own 0

Remove certificate Nr. 0 of specified trustlevel.
Successfully removed certificate Nr. 0.
```

Grundsätzlich lassen sich Zertifikate auch direkt in dem Security Screen erzeugen. Dafür muss die entsprechende Komponente ausgewählt und der Button "Neues Zertifikat erstellen" betätigt werden:



3 Verschlüsselte Applikation

Um den Download oder einen Online-Change einer Applikation auf eine Steuerung zu verschlüsseln, kann die Option „Verschlüsselung von Download, Online-Changes und Bootapplikationen erzwingen“ aktiviert werden. Dadurch können Applikationen auf der Steuerung nur mit einem gültigen Zertifikat ausgetauscht werden. Hierfür wird ein X.509 Zertifikat für die Komponente CmpApp benötigt. Dieses Zertifikat muss dann bei einem Download in der CoDeSys Software vorhanden sein.

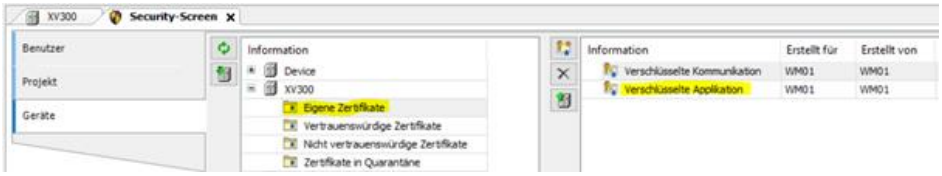
Zunächst wird geprüft, ob bereits ein Zertifikat generiert wurde:

```
cert-getapplist

Nr.     ComponentName           CommonName                  CertAvailable DateNotBefore
        DateNotAfter            Thumbprint
-------------------------------------------------------------------------------------------
-------------------------------------------------------
0       CmpOPCUAServer          OPCUAServer@WM01            FALSE         --
                    --
1       CmpApp                  WM01                        FALSE         --
            --
2       CmpSecureChannel        WM01                        TRUE          2021-1-27T2:49:36.0
27T2:49:36.0        87a2a53ade19f4be88c603877d2965013be014ee
3       CmpWebServer            WM01                        TRUE          2021-1-27T2:49:36.0
27T2:49:36.0        87a2a53ade19f4be88c603877d2965013be014ee
```

Ist dies nicht der Fall, so kann mit dem Befehl „cert-genselfsigned 1“ ein neues Zertifikat generiert werden. In der Registerkarte Log wird eine erfolgreiche Generierung gemeldet.

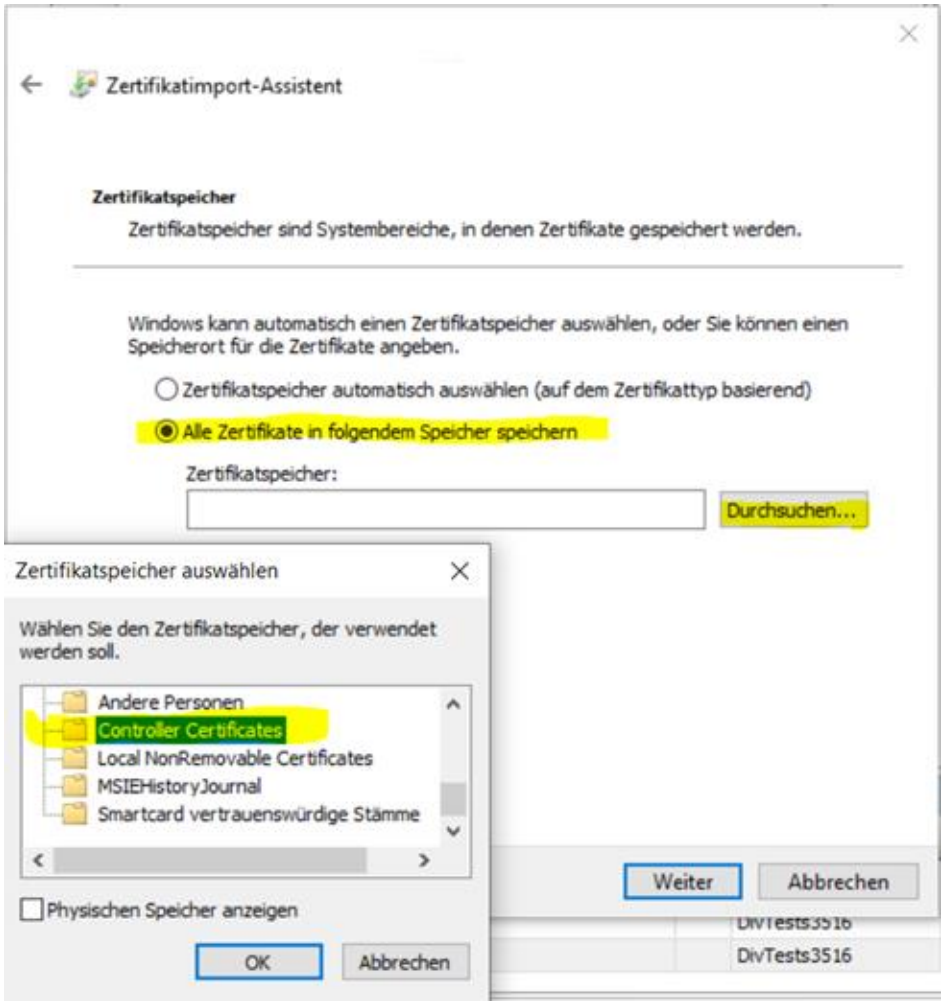
```
cert-genselfsigned 1

Generate selfsigned certificate with given index (1). Check logger to see when finished.
```

Das generierte Zertifikat muss nun auf den PC übertragen werden, damit dieses in der CoDeSys Software eingebunden werden kann. Im Security Screen in der Registerkarte Geräte kann man nun auch die vorhandenen Zertifikate auf dem Gerät einsehen.

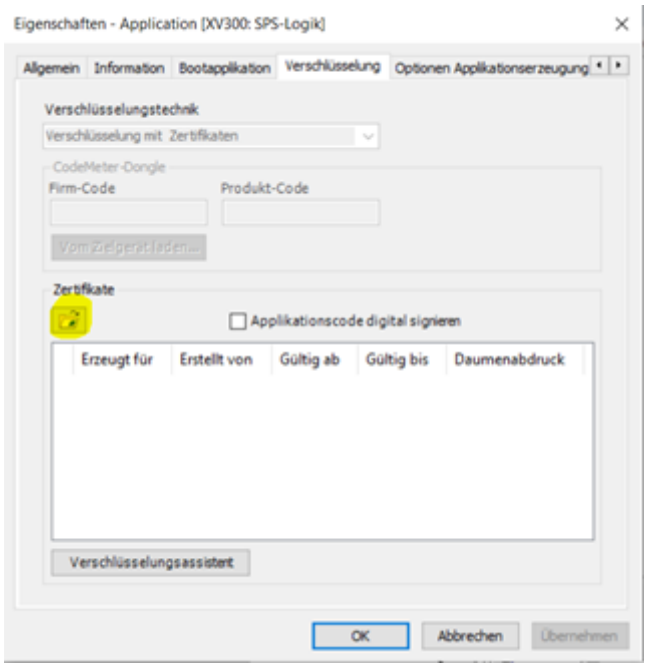


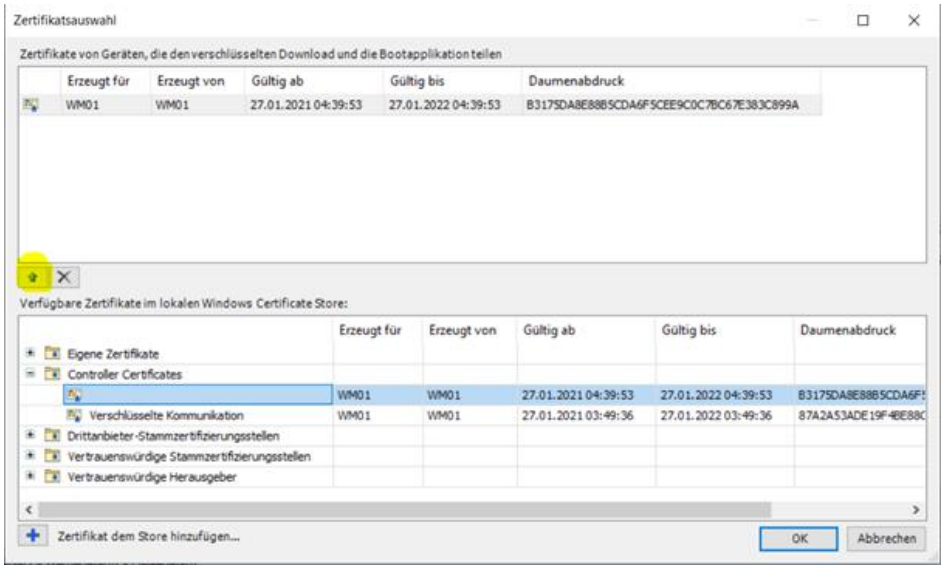
Durch einen Doppelklick auf das Zertifikat „Verschlüsselte Applikation“ öffnet sich der Windows-Standarddialog für Zertifikate.

Das Zertifikat wird in dem Zertifikatspeicher „Controller Certificates“ gespeichert.

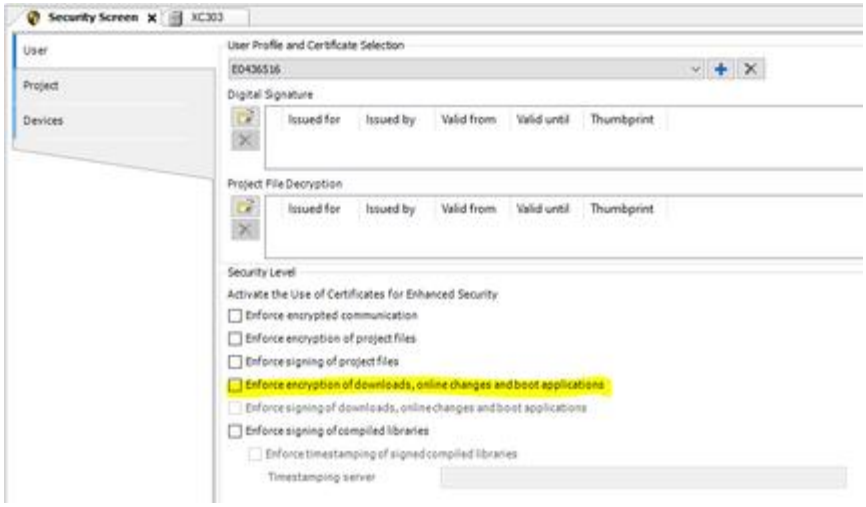


Nach erfolgreichem Import steht dieses Zertifikat zur Verschlüsselung von Download, Online-Change und Bootapplikation bereit. Nun muss dieses Zertifikat auch in der CoDeSys ausgewählt werden. Im Security Screen in der Registerkarte Projekt wird das Zertifikat dem Projekt zugewiesen. Ein Doppelklick auf die Applikation öffnet die Eigenschaften der Applikation, wo das Zertifikat ausgewählt werden muss.





Das Übertragen einer Bootapplikation, ein Download des Projektes oder ein Online Change sind nun verschlüsselt, wenn folgende Option aktiviert ist:



4 Verschlüsselte Online-Kommunikation

Die Sicherheitsfunktionen ermöglichen eine verschlüsselte Online-Kommunikation zu einer PLC Runtime. Die Verschlüsselung erfolgt über das TLS-Protokoll innerhalb des CoDeSys Kommunikationsprotokolls. Hierfür wird ein X.509 Zertifikat und ein Schlüssel für die Komponente CmpSecureChannel benötigt. Die Konfiguration erfolgt im Security Screen. Zunächst muss jedoch ein Zertifikat für CmpSecureChannel auf der Steuerung generiert werden. Dafür wird zunächst in der PLC-Shell der Befehl „cert-getapplist“ ausgeführt, um zu prüfen, ob bereits ein Zertifikat für diese Komponente erstellt wurde:

```
cert-getapplist

Nr.    ComponentName              CommonName                    CertAvailable DateNotBefore
-------------------------------------------------------------------------------------------
-----
0      CmpOPCUAServer             OPCUAServer@WM01              FALSE         --
1      CmpApp                     WM01                    FALSE       --
2      CmpSecureChannel           WM01                    FALSE       --
3      CmpWebServer               WM01                    FALSE       --
```

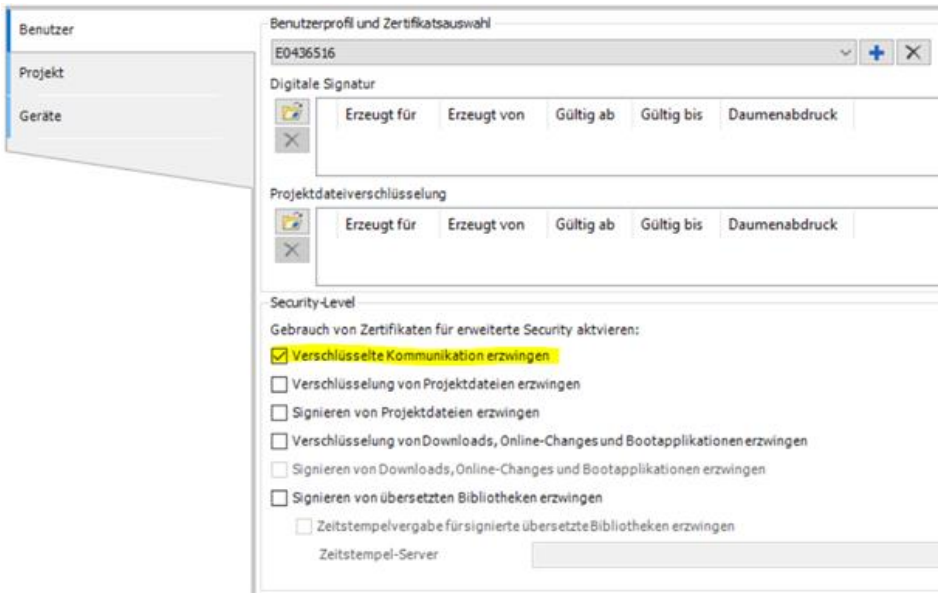
Ist noch kein Zertifikat für die Komponente CmpSecureChannel vorhanden, dann kann mit dem Befehl „cert-genselfsigned 2“ ein Zertifikat generiert werden.

```
cert-genselfsigned 2

Generate selfsigned certificate with given index (2). Check logger to see when finished.
```

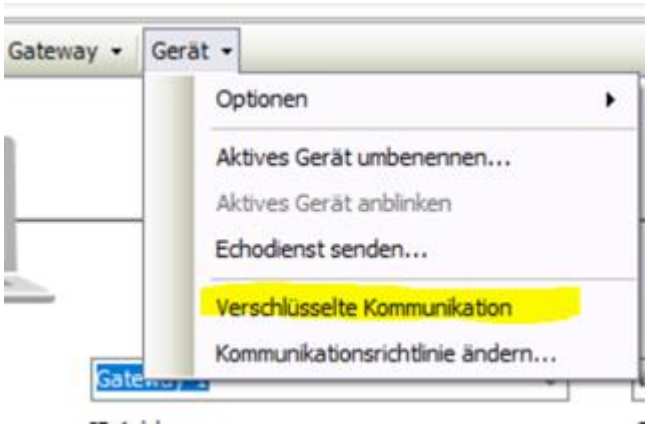
Sobald das Zertifikat erfolgreich generiert wurde, wird dies in der Registerkarte Log sichtbar.

Durch eine erneute Ausführung des Befehls „cert-getapplist“ ist eine Überprüfung auch in der PLC Shell möglich:

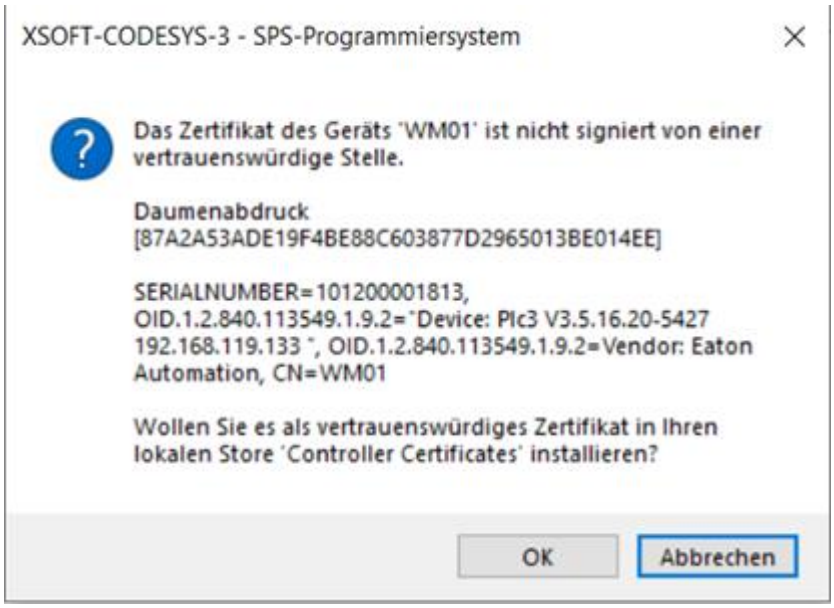
Nun kann im Security Screen die Option „Verschlüsselte Kommunikation erzwingen“ aktiviert werden.



Diese Option gilt für alle Steuerungen. Soll die Kommunikation nur zu einzelnen Steuerungen verschlüsselt erfolgen, besteht die Möglichkeit dies in der Registerkarte „Kommunikation“ zu aktivieren:



Beim ersten Einloggen auf die Steuerung, erscheint die Meldung, dass das Zertifikat der Steuerung nicht von einer vertrauenswürdigen Stelle signiert ist. Außerdem wird angeboten das Zertifikat in dem lokalen Store „Controller Certificates“ zu installieren. Wird dieser Dialog bestätigt, dann wird auf die Steuerung eingeloggt und zukünftige Verbindungen zu der Steuerung mit diesem Zertifikat verschlüsselt.

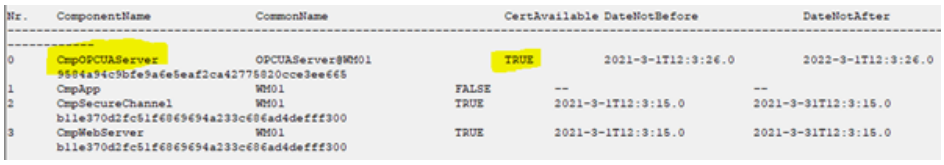


Eine verschlüsselte Verbindung wird in der Registerkarte Kommunikation entsprechend farblich dargestellt:

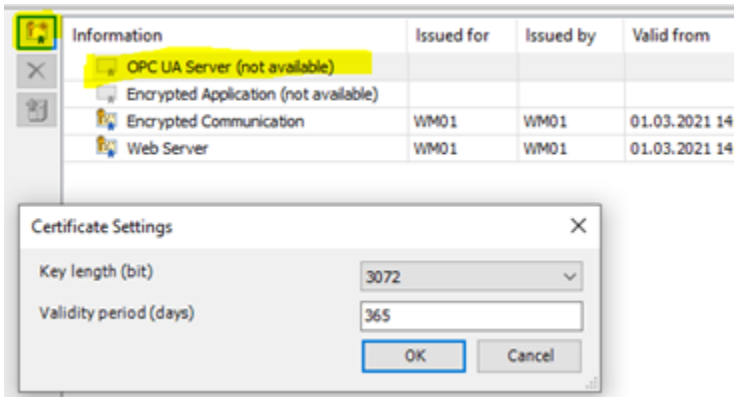
5 Verschlüsselter OPC-UA Server Zugriff

Für die verschlüsselte Kommunikation zu dem OPCUA-Server wird ein X.509 Zertifikat benötigt. Mit cert-getapplist wird zunächst geprüft, ob bereits ein Zertifikat für den OPUA-Server angelegt wurde.

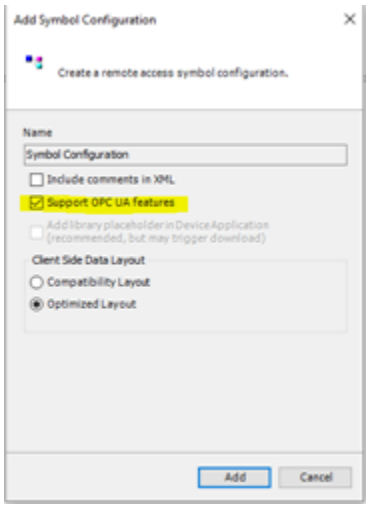
Mit dem Befehl cert-genselfsigned kann auch hier wieder ein Zertifikat für diese Komponente erstellt werden.



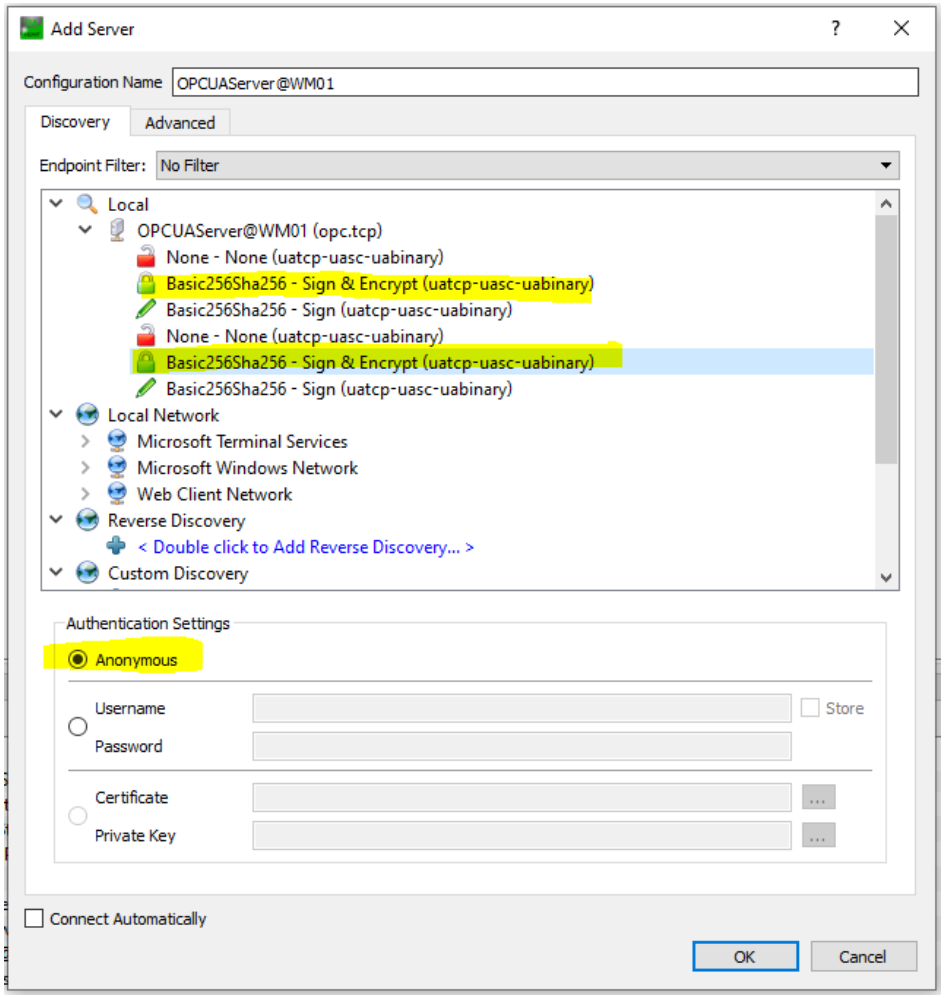
Alternativ kann das Zertifikate auch im Security Screen erstellt werden:



Nach der Erstellung muss die PLC-Laufzeit neu gestartet werden. Auf Seiten der CoDeSys muss nun eine Symbolkonfiguration hinzugefügt werden, bei der die Option „OPC UA Funktionalitäten unterstützen“ aktiviert ist.

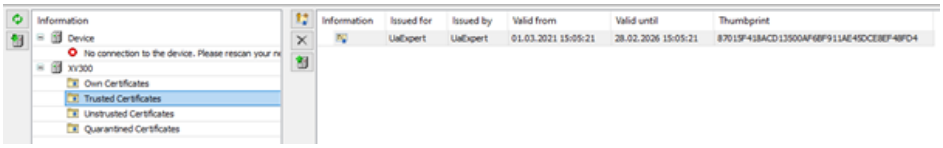


Nun kann der OPCUA-Client konfiguriert werden. In diesem Beispiel wird der Client UaExpert von Unified Automation verwendet. Zunächst wird ein Server hinzugefügt. Der UaExpert Client listet die möglichen Verbindungen zu den Servern auf. Hier wählt man die verschlüsselte Basic256 (Sign&Encrypt" Verbindung aus. Der Zugriff kann entweder über den Gerätenamen oder die Geräte IP-Adresse erfolgen.

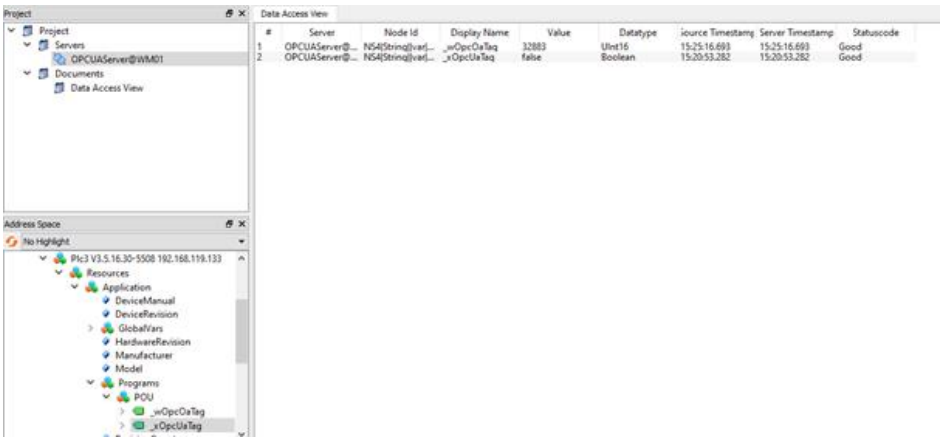


Nach dem ersten Verbinden wird nachfolgende Meldung angezeigt, die darauf hinweist, dass das vorhandene Zertifikat ein selbstsigniertes Zertifikat ist und nicht verifiziert werden kann. Durch Bestätigung der Meldung wird das Zertifikat als gültig anerkannt und das Client-Zertifikat an die Steuerung übertragen.

Anschließend muss dieses Client-Zertifikat aus dem Quarantined-Verzeichnis in das Trusted-Verzeichnis verschoben werden:



Verbindet sich man nun erneut mit dem OPCUA-Client, dann wird eine verschlüsselte Verbindung aufgebaut und die konfigurierten Variablen aus der Symbolkonfiguration können beobachtet und manipuliert werden.



Per Default akzeptiert die PLC alle eingehenden Verbindungen auf den OPCUA-Server. Um das zu verhindern und eine verschlüsselte Verbindung zu erzwingen, muss der CommunicationMode in der PLC3.cfg Datei angepasst werden:

```
[CmpOPCUAServer]
SECURITY.CommunicationMode=SIGNED_AND_ENCRYPTED
```

6 Verschlüsselter Webserver-Zugriff

Für die verschlüsselte Kommunikation zu dem Web-Server (https) wird ein X.509 Zertifikat für die TLS Web Server Authentication benötigt. Mit cert-getapplist wird zunächst geprüft, ob bereits ein Zertifikat für den Web-Server angelegt wurde.

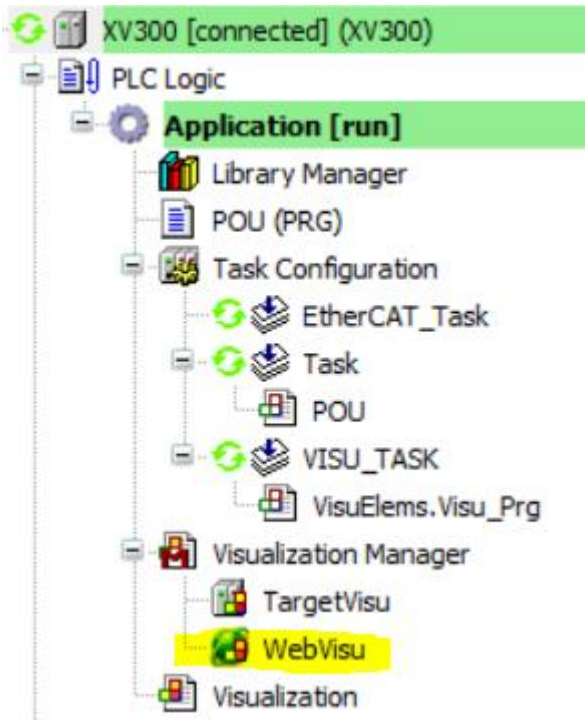
```
cert-getapplist
Nr.    ComponentName              CommonName                    CertAvailable DateNotBefore
       DateNotAfter               Thumbprint
-------------------------------------------------------------------------------------------
-------------------------------------------------------------
0      CmpOPCUAServer             OPCUAServer@WM01              FALSE         --
                        --
1      CmpApp                     WM01                    FALSE         --
                    --
2      CmpSecureChannel           WM01                    FALSE         --
                    --
3      CmpWebServer               WM01                    FALSE         --
                    --
```

Ist dies nicht der Fall, dann kann mit dem Befehl cert-genselfsigned ein Zertifikat für diese Komponente erstellt werden. Standardmäßig ist auf den Geräten bereits ein Zertifikat für den Web-Server vorhanden. Jedoch ist die Gültigkeit dieses Zertifikats auf einen kurzen Zeitraum begrenzt. Wenn ein neues Zertifikat erstellt wird, kann die Gültigkeitsdauer entsprechend konfiguriert werden.

```
----
cert-genselfsigned 3 365

Generate selfsigned certificate with given index (3). Check logger to see when finished.
```

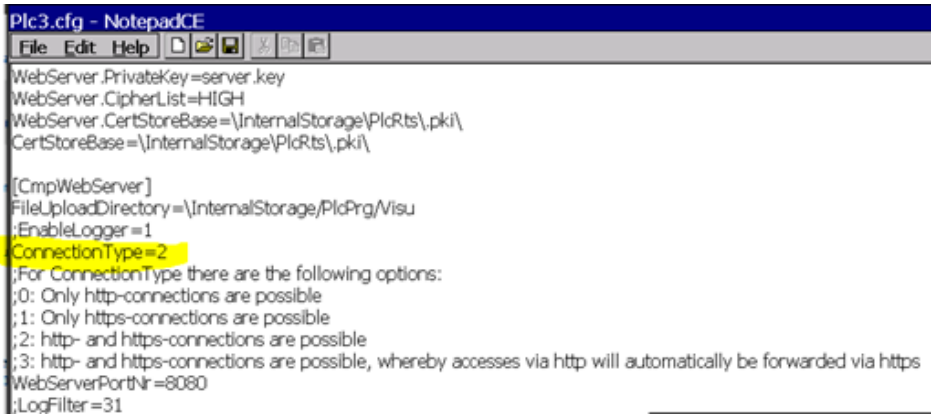
```
Nr.     ComponentName              CommonName                    CertAvailable DateNotBefore
            DateNotAfter               Thumbprint
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------
0       CmpOPCUAServer             OPCUAServer@WM01              FALSE         --
            --                         --
1       CmpApp                     WM01                    FALSE         --
        --                         --
2       CmpSecureChannel           WM01                    TRUE          2018-1-24T23:0:19.0
        2018-2-23T23:0:19.0        2a453b12f0836769b4ee40ce10a299ca499dbdc4
3       CmpWebServer               WM01                    TRUE          2018-1-24T23:3:54.0
        2019-1-24T23:3:54.0        0c31a21fe022b7cde5351cfdcef1f592ec6767f8
```

In dem CoDeSys Projekt ist das Anlegen und Übertragen einer Web-Visualisierung erforderlich:

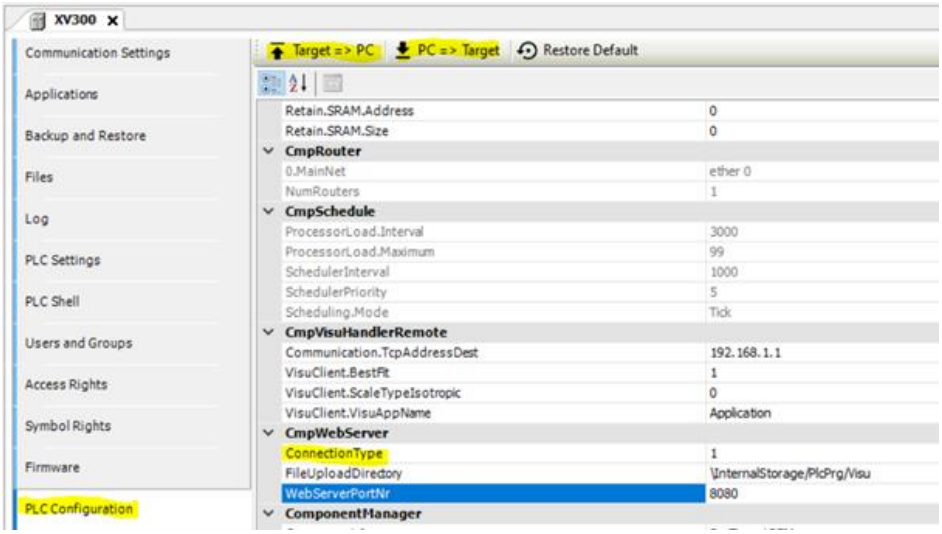


Anschließend kann über die URL https://<IP-Adresse>/Name_der_HTML_Datei.htm auf die Web-Visualisierung zugegriffen werden.

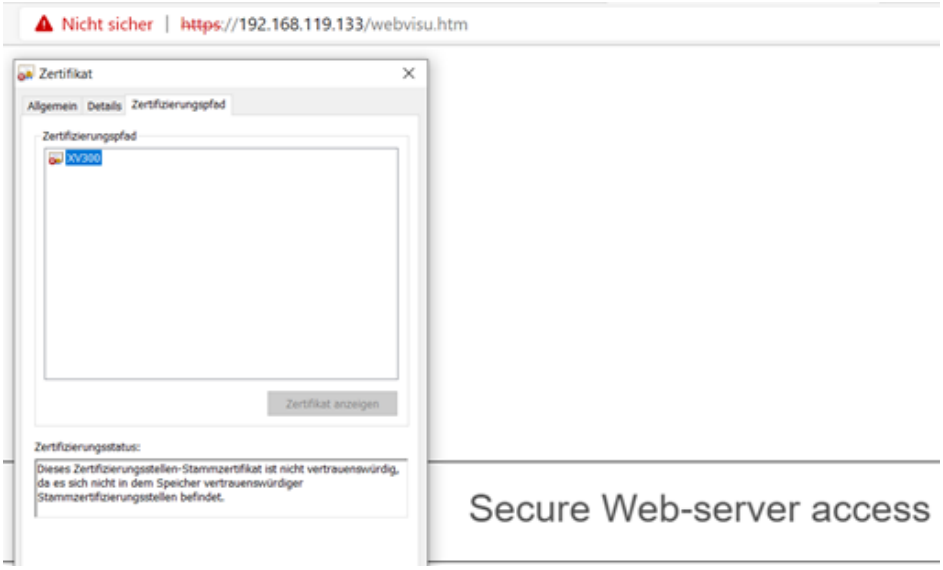
Damit auch zwingende eine https Verbindung verwendet werden kann, muss der entsprechende Eintrag in der PLC3.cfg erfolgen. Per Default sind http und https Verbindungen erlaubt. Sollen nur https Verbindungen erlaubt sein, muss der ConnectionType auf 1 oder 3 eingestellt werden.



Die Umstellung kann bei einem XV100/XV300 in der CoDeSys Software erfolgen. Nach dem Ändern und Übertragen muss das Gerät neugestartet werden.



Solange man selbstsignierte Zertifikate nutzt und diese nicht von einer offiziellen Stelle signieren lässt oder ein eigenes Stammzertifikat erstellt, wird in dem Browser die Meldung „Nicht sicher“ angezeigt.



Eaton ist ein intelligentes Energiemanagementunternehmen, das sich dem Ziel verschrieben hat,
für mehr Lebensqualität zu sorgen und die Umwelt zu schützen. Wir handeln verantwortlich und nachhaltig und -unterstützen unsere Kunden beim Energiemanagement – heute und in Zukunft.
Wir setzen auf die globalen Wachstumstrends Elektrifizierung und Digitalisierung und beschleunigen so die Umstellung der Welt auf erneuerbare Energien, tragen zur Lösung der weltweit dringendsten Herausforderungen im Energiemanagement bei und setzen uns für das Beste für unsere Stakeholder und die ganze Gesellschaft ein.

Das 1911 gegründete Unternehmen Eaton ist seit fast einem Jahrhundert an der NYSE notiert.
Im Jahr 2021 verzeichneten wir einen Umsatz von 19,6 Milliarden US-Dollar und wir sind in über 170 Ländern vertreten.

Weitere Informationen finden Sie unter Eaton.com. Folgen Sie uns auf Twitter und LinkedIn.

Eaton Industries GmbH
Hein-Moeller-Str. 7- 11
D-53115 Bonn

07/2025 AP050017DE

Powering Business Worldwide