# SSL Certificates
# SignOn Soltuions 2016

19 July 2023

# Table of contents


**1. Introduction ..... 3**
**2. Object identifiers ..... 3**
**3. Create the certificates ..... 4**
3.1 Using OpenSSL ..... 4
3.1.1 Preparing a Certificate Authority ..... 4
3.1.2 Certificate for the SignOn Proxy ..... 5
3.1.3 Certificate for the SignOn Agents ..... 6
3.1.4 Certificate for the Logon Client ..... 8
3.1.5 Certificate for the Comtarsia Web Gateway ..... 9
3.1.6 Certificate for the Comtarsia LDAP Directory Replicator ..... 9
3.2 Using Microsoft Certificate Authority ..... 10
3.2.1 Creating a certificate request ..... 10
3.2.2 Certificate for the SignOn Proxy ..... 13
3.2.3 Certificate for the Comtarsia SignOn Agent ..... 14
3.2.4 Certificate for the Comtarsia Logon Client ..... 16
3.2.5 Certificate for the Comtarsia Web Gateway ..... 17
3.2.6 Certificate for the Comtarsia LDAP Directory Replicator ..... 17
3.2.7 Issue the certificates ..... 17
3.2.8 Retrieve and convert the certificates ..... 17
3.2.9 CA certificate ..... 24
**4. Configuration of the Comtarsia products ..... 27**
4.1 Comtarsia SignOn Proxy ..... 27
4.2 Comtarsia SignOn Agent ..... 29
4.3 Comtarsia Logon Client ..... 30
4.4 Comtarsia LDAP Directory Replicator ..... 31
4.5 Comtarsia Web Gatewateway ..... 32
**5. Enhancing TLS Security ..... 33**
5.1 Comtarsia internal TLS communication ..... 33
5.1.1 Logon Client, SignOn Proxy and SignOn Agents ..... 33
5.1.2 LDAP Gateway, RADIUSG Gateway and AuthSRVM under Windows ..... 33
5.2 LDAP Gateway LDAP communication under Windows ..... 34
5.3 Logon Client and SignOn Proxy LDAP communication under Windows ..... 34

# 1. Introduction

The trust relationship between the Comtarsia components (Logon Client, Webgateway, Proxy, Agents and LDAP Directory Replicator) is done by using certificates.
For every product, different trust options can be defined. A common option is to compare the FQDN of the peer with the subject name in the certificate. Additionally, the certificates can be checked for specific Comtarsia extended key usage OIDs. The certificates and keys used are industry standard X509 certificates, which can be created with nearly every CA product available.

This document gives a guide on how to create the needed certificates using the free OpenSSL tool as well as using the Microsoft Server 2016 Certificate Authority. Furthermore, it also explains how to install and configure these certificates in the Comtarsia products.

# 2. Object identifiers

| Object identifier | Product |
|---|---|
| 1.3.6.1.4.1.13823.1.3.1 | SignOn Agent |
| 1.3.6.1.4.1.13823.1.3.2 | SignOn Proxy |
| 1.3.6.1.4.1.13823.1.3.3 | Logon Client |

The usage of the above mentioned OIDs is optional and provides extended security. Additionally, the following standard OIDs need to be set:

| Object identifier | Product |
|---|---|
| 1.3.6.1.5.5.7.3.1 (id_kp_serverAuth) | SignOn Proxy and SignOn Agent |
| 1.3.6.1.5.5.7.3.2 (id_kp_clientAuth) | SignOn Proxy and Logon Client |

# 3. Create the certificates

## 3.1 Using OpenSSL

The following paragraphs show how to create a certificate authority certificate and then certificates for the Comtarsia products using OpenSSL. OpenSSL is available for Windows and Linux platforms. Further information about the OpenSSL configuration options can be found in the OpenSSL manual at https://www.openssl.org/docs/manmaster/man5/x509v3_config.html

### 3.1.1 Preparing a Certificate Authority

If you do not have a certificate authority certificate and key yet, follow these steps to create one.

Create a text file named "ca.txt" with the following content:

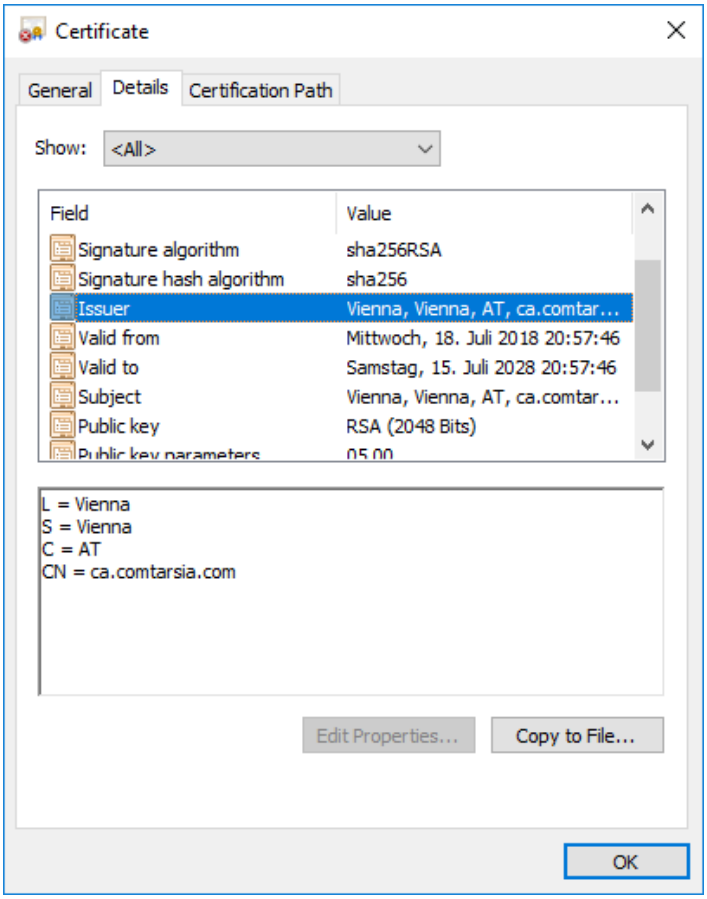
```
[ req ]
default_md = sha256
prompt = no
distinguished_name = req_distinguished_name
x509_extensions = v3_ca
[ req_distinguished_name ]
commonName = ca.comtarsia.com
countryName = AT
stateOrProvinceName = Vienna
localityName = Vienna
[ v3_ca ]
basicConstraints = critical,CA:TRUE
keyUsage = critical,digitalSignature,keyCertSign,cRLSign
```

The values written in bold in the section "req_distinguished_name" must be replaced with your own values.

Afterwards, execute the following commands:

```
openssl genrsa -out ca.key 2048
openssl req -key ca.key -new -x509 -days 3650 -sha256 -out ca.cer -
config ca.txt
```

The first command generates a key pair used for the certificate authority. Keep the resulting "ca.key" file private. The second command creates the CA certificate file, which is public and can be distributed to all computers running a Comtarsia installation. The parameter "days" specifies the validity in days, in this example 10 years.
Under Windows, when opening the created "ca.cer" file a certificate info dialog will open showing the attributes of the certificate:




### 3.1.2 Certificate for the SignOn Proxy

Create a text file named „proxy.txt" with the following content:

```
[ req ]
default_md = sha256
prompt = no
distinguished_name = req_distinguished_name
[ req_distinguished_name ]
commonName = proxy.comtarsia.com
countryName = AT
stateOrProvinceName = Vienna
localityName = Vienna
[ v3_ca ]
basicConstraints = critical,CA:FALSE
keyUsage = critical,digitalSignature,keyEncipherment
extendedKeyUsage =
critical,serverAuth,clientAuth,1.3.6.1.4.1.13823.1.3.2
```

The values written in bold in the section "req_distinguished_name" must be replaced with your own values. commonName is the FQDN of the SignOn Proxy server. It is important that the IP address used by the SignOn Proxy resolves reverse to this hostname.

Afterward, execute the following commands:

```
openssl genrsa -out proxy.key 2048
openssl req -new -nodes -key proxy.key -config proxy.txt -out
proxy.csr
```

The first command generates a key pair used for the SignOn Proxy. This key is private and should only reside on the SignOn Proxy machine. The second

command creates a CSR (certificate sign request) that needs to be signed with the CA key created in step 3.1.1.

The so created certificate sign request (CSR) can now be signed using the CA key. Every signed certificate contains a serial number. OpenSSL takes care of this serial number by using a special file, called "ca.srl". If you sign a certificate the first time, use the following command:

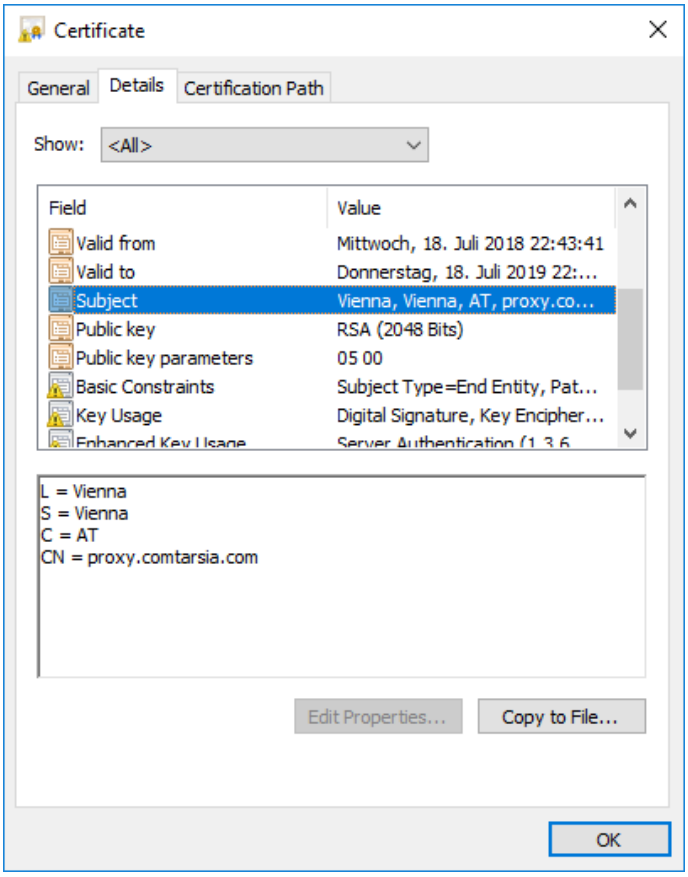
```
openssl x509 -req -days 365 -in proxy.csr -CA ca.cer -CAkey ca.key -CAcreateserial -out proxy.cer -sha256 -extensions v3_ca -extfile proxy.txt
```

If you already signed at least one certificate, use this command:

```
openssl x509 -req -days 365 -in proxy.csr -CA ca.cer -CAkey ca.key -CAserial ca.srl -out proxy.cer -sha256 -extensions v3_ca -extfile proxy.txt
```

This command performed the actual sign process, the parameter "days" specifies the validity of the resulting certificate in days, in this example 1 year.

Side note: Normally, with certificate authorities, the key usage and extended key usage is already specified for the certificate sign request. But because OpenSSL does not transfer extensions from the CSR to the certificate, they must be specified when signing. For details, see the OpenSSL manual: https://www.openssl.org/docs/man1.1.0/apps/x509.html#BUGS




### 3.1.3 Certificate for the SignOn Agents

This process is valid for all of the Comtarsia SignOn Agents.

Create a text file named „agent.txt“ with the following content:

```
[ req ]
default_md = sha256
prompt = no
distinguished_name = req_distinguished_name
[ req_distinguished_name ]
commonName = agent.comtarsia.com
countryName = AT
stateOrProvinceName = Vienna
localityName = Vienna
[ v3_ca ]
basicConstraints = critical,CA:FALSE
keyUsage = critical,digitalSignature,keyEncipherment
extendedKeyUsage = critical,serverAuth,1.3.6.1.4.1.13823.1.3.1
```

The values written in bold in the section “req_distinguished_name” must be replaced with your own values. commonName is the FQDN of the SignOn Agent server. It is important that the IP address used by the SignOn Agent resolves reverse to this hostname.

Afterwards, execute the following commands:

```
openssl genrsa -out agent.key 2048
openssl req -new -nodes -key agent.key -config agent.txt -out
agent.csr
```

The first command generates a key pair used for the SignOn Agent. This key is private and should only reside on the SignOn Agent machine. The second command creates a CSR (certificate sign request) that needs to be signed with the CA key created in step 3.1.1.

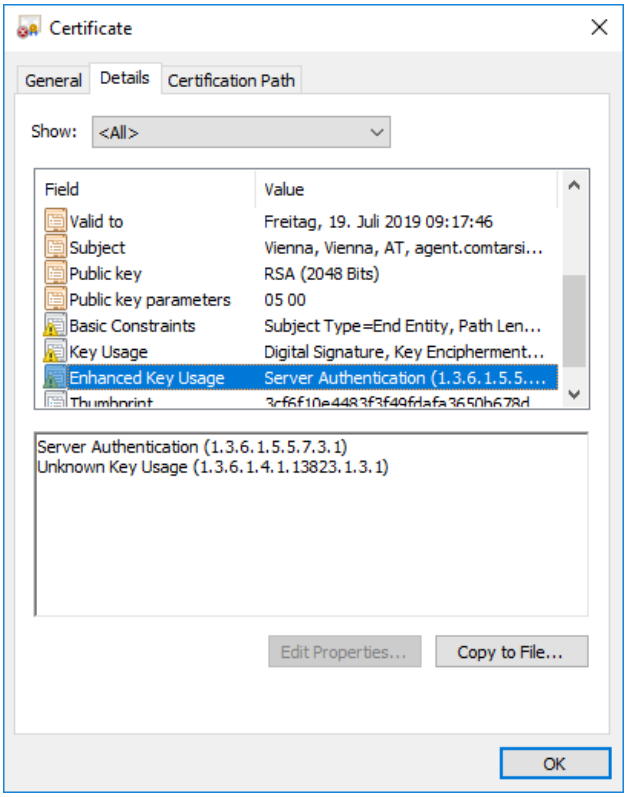
The so created certificate sign request (CSR) can now be signed using the CA key. Every signed certificate contains a serial number. OpenSSL takes care of this serial number by using a special file, called “ca.srl”. If you sign a certificate the first time, use the following command:

```
openssl x509 -req -days 365 -in agent.csr -CA ca.cer -CAkey ca.key -
CAcreateserial -out agent.cer -sha256 -extensions v3_ca -extfile
agent.txt
```

If you already signed at least one certificate, use this command:

```
openssl x509 -req -days 365 -in agent.csr -CA ca.cer -CAkey ca.key -
CAserial ca.srl -out agent.cer -sha256 -extensions v3_ca -extfile
agent.txt
```

This command performed the actual sign process, the parameter “days” specifies the validity of the resulting certificate in days, in this example 1 year.




### 3.1.4 Certificate for the Logon Client

Create a text file named „client.txt" with the following content:

```
[ req ]
default_md = sha256
prompt = no
distinguished_name = req_distinguished_name
[ req_distinguished_name ]
commonName = client.comtarsia.com
countryName = AT
stateOrProvinceName = Vienna
localityName = Vienna
[ v3_ca ]
basicConstraints = critical,CA:FALSE
keyUsage = critical,digitalSignature,keyEncipherment
extendedKeyUsage =
critical,serverAuth,clientAuth,1.3.6.1.4.1.13823.1.3.3
```

The values written in bold in the section "req_distinguished_name" must be replaced with your own values. commonName is the FQDN of the Logon Client. If FQDN checks are used it is important that the IP address used by the Logon Client resolves reverse to this hostname.

Afterwards, execute the following commands:

```
openssl genrsa -out client.key 2048
openssl req -new -nodes -key client.key -config client.txt -out
client.csr
```

The first command generates a key pair used for the Logon Client. This key is private and should only reside on the SignOn Proxy machine. The second command creates a CSR (certificate sign request) that needs to be signed with the CA key created in step 3.1.1.

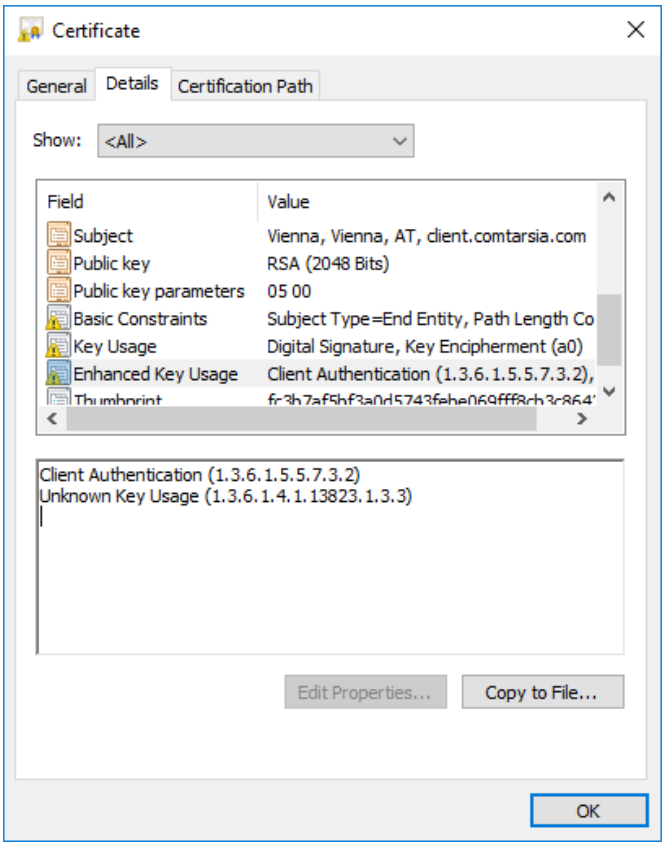
The so created certificate sign request (CSR) can now be signed using the CA key. Every signed certificate contains a serial number. OpenSSL takes care of this serial number by using a special file, called "ca.srl". If you sign a certificate the first time, use the following command:

```
openssl x509 -req -days 365 -in client.csr -CA ca.cer -CAkey ca.key
-CAcreateserial -out client.cer -sha256 -extensions v3_ca -extfile
client.txt
```

If you already signed at least one certificate, use this command:

```
openssl x509 -req -days 365 -in client.csr -CA ca.cer -CAkey ca.key
-CAserial ca.srl -out client.cer -sha256 -extensions v3_ca -extfile
client.txt
```

This command performed the actual sign process, the parameter "days" specifies the validity of the resulting certificate in days, in this example 1 year.




### 3.1.5 Certificate for the Comtarsia Web Gateway

The Comtarsia Web Gateway acts as a client in the SignOn Gate family and therefore a certificate like for the Logon Client can be used.

### 3.1.6 Certificate for the Comtarsia LDAP Directory Replicator

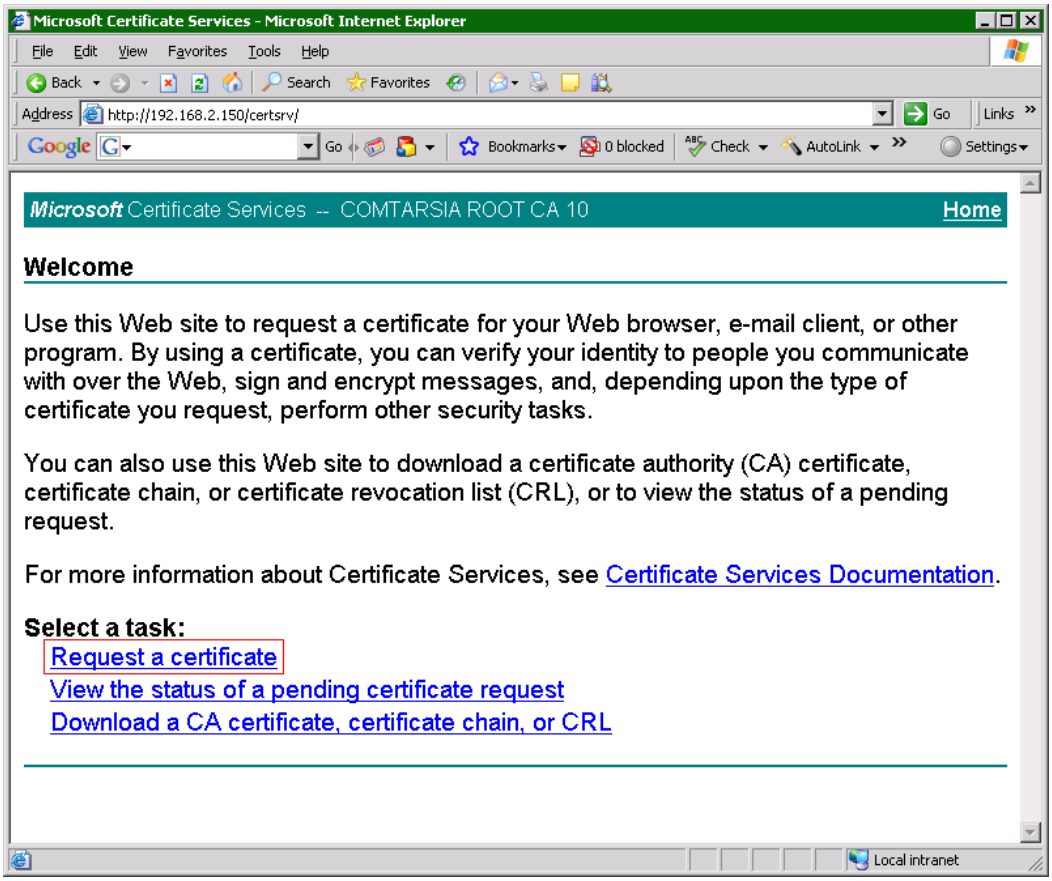
The Comtarsia LDAP Directory Replicator acts as a client in the SignOn Gate family and therefore a certificate like for the Logon Client can be used.

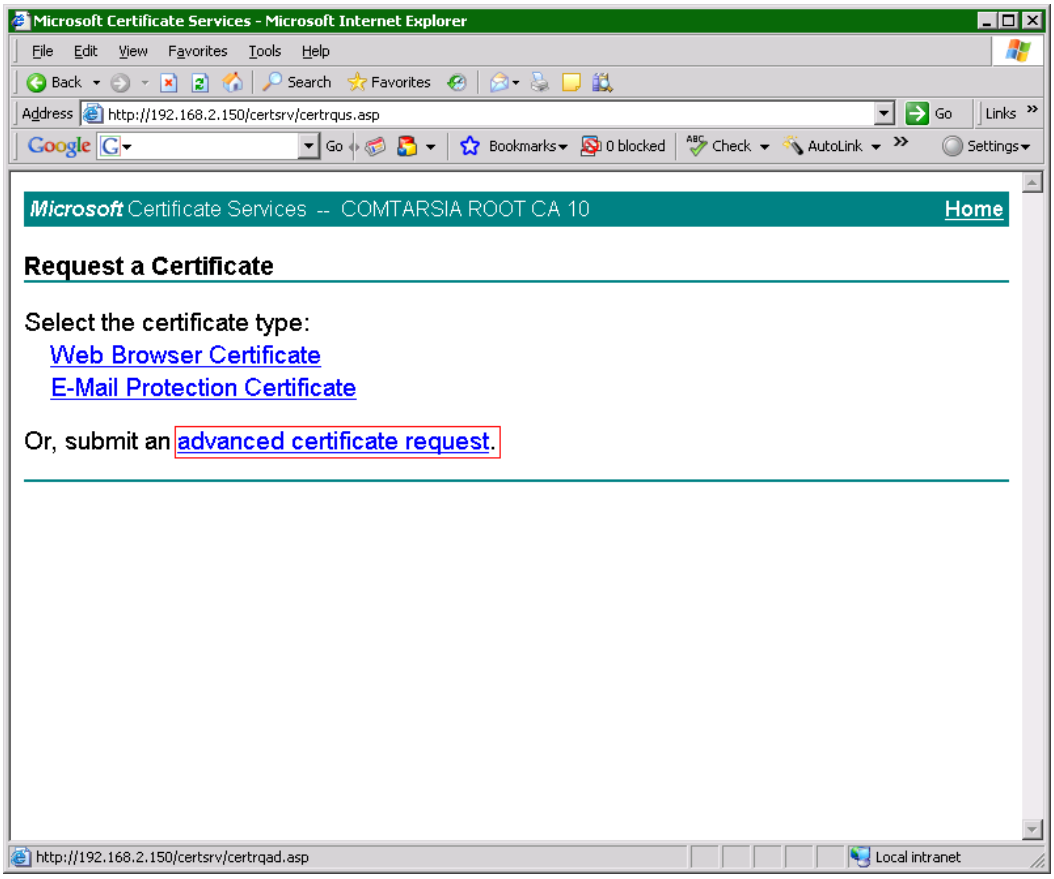
## 3.2 Using Microsoft Certificate Authority

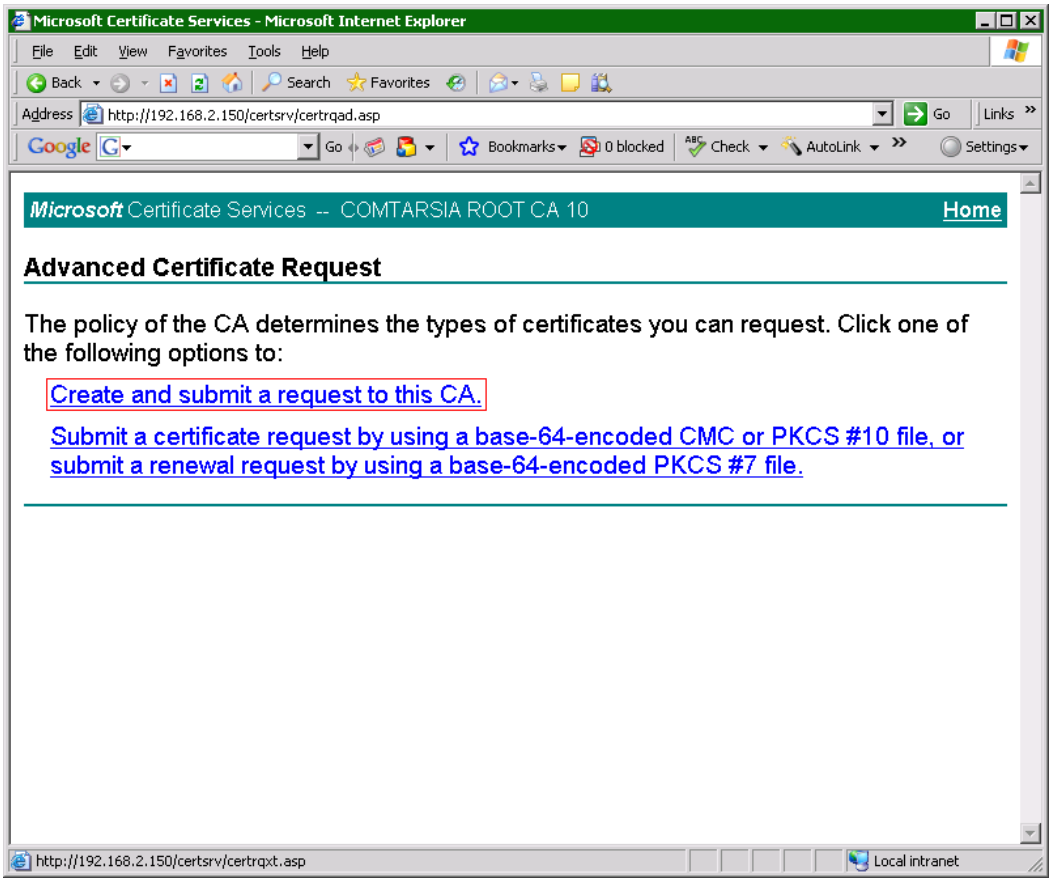
### 3.2.1 Creating a certificate request

Throughout the following chapter, a Microsoft Server 2016 Standalone Certificate Authority together with the Certificate Authority Web Enrolment is used. Older Microsoft Server Certificate Authorities can be used as well. The HTTPS protocol must be enabled on the IIS that serves the CA web enrolment pages. A good documentation on how to create a CSR for the IIS and then afterward install the issued certificate, see this document: https://www.digicert.com/csr-creation-ssl-installation-iis-10.htm

Open Internet Explorer and navigate to the following page: https://<CA-Server>/certsrv/
This page can be used to request and collect certificates.




Microsoft Certificate Services - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.2.150/certsrv/certrqus.asp
Go Links
Google Go Bookmarks 0 blocked Check AutoLink Settings
Microsoft Certificate Services -- COMTARSIA ROOT CA 10 Home
Request a Certificate
Select the certificate type:
Web Browser Certificate
E-Mail Protection Certificate
Or, submit an advanced certificate request.
http://192.168.2.150/certsrv/certrqad.asp
Local intranet


Microsoft Certificate Services - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.2.150/certsrv/certrqad.asp
Go Links
Google Go Bookmarks 0 blocked Check AutoLink Settings
Microsoft Certificate Services -- COMTARSIA ROOT CA 10 Home
Advanced Certificate Request
The policy of the CA determines the types of certificates you can request. Click one of the following options to:
Create and submit a request to this CA.
Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file.
http://192.168.2.150/certsrv/certrqxt.asp
Local intranet

### 3.2.2 Certificate for the SignOn Proxy

The SignOn Proxy needs a certificate for client authentication and server authentication, as connections from clients are accepted as well as connections to agents are made.
Additionally, the Comtarsia object identifier for the SignOn Proxy (1.3.6.1.4.1.13823.1.3.2) can be added to the certificate for enhanced security.

If the hostname check will be enabled in the trust options it is important that the common name in the certificate subject matches the FQDN of the SignOn Proxy server. This check is done using DNS reverse lookup on the IP address of the SignOn Proxy.

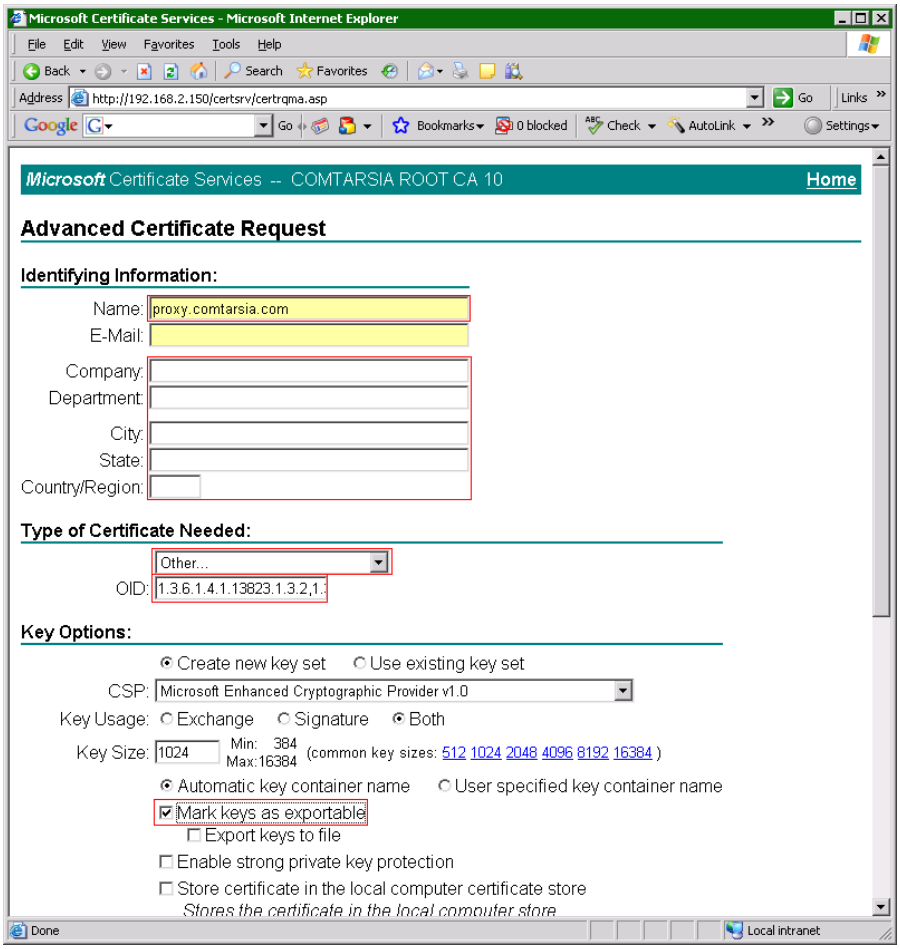
For „Type of Certificate" choose „other" which allows specifying the required OIDs.
The OIDs are seperated with a comma:

| | |
|---|---|
| 1.3.6.1.5.5.7.3.1 | server authentication (id_kp_serverAuth) |
| 1.3.6.1.5.5.7.3.2 | client authentication (id_kp_clientAuth) |
| 1.3.6.1.4.1.13823.1.3.2 | Comtarsia SignOn Proxy OID |

OID: 1.3.6.1.5.5.7.3.1,1.3.6.1.5.5.7.3.2,1.3.6.1.4.1.13823.1.3.2

Fields like „Company", „Department", City, State, Country/Region are optional and are not checked.

**Important: To be able to export the private key later, the checkbox "Mark key as exportable" must be checked.**




### 3.2.3 Certificate for the Comtarsia SignOn Agent

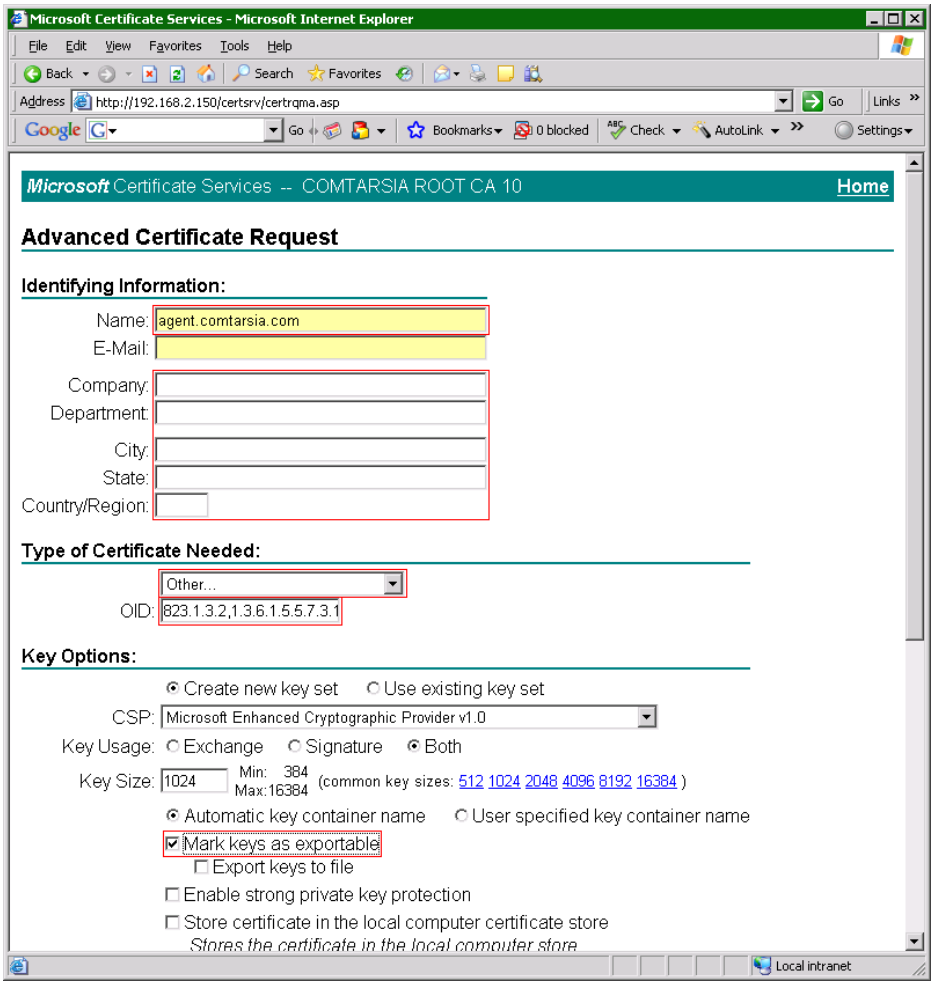
The SignOn Agent needs a certificate for server authentication, as connections from SignOn Proxies are accepted.
Additionally, the Comtarsia object identifier for the SignOn Agent (1.3.6.1.4.1.13823.1.3.1) can be added to the certificate for enhanced security.

If the hostname check will be enabled in the trust options it is important that the common name in the certificate subject matches the FQDN of the SignOn Agent server. This check is done using DNS reverse lookup on the IP address of the SignOn Agent.

For „Type of Certificate" choose „other" which allows specifying the required OIDs.
The OIDs are seperated with a comma:
1.3.6.1.5.5.7.3.1 server authentication (id_kp_serverAuth)
1.3.6.1.4.1.13823.1.3.1 Comtarsia SignOn Agent OID
OID: 1.3.6.1.5.5.7.3.1,1.3.6.1.4.1.13823.1.3.1

Fields like „Company", „Department", City, State, Country/Region are optional and are not checked.

**Important: To be able to export the private key later, the checkbox "Mark key as exportable" must be checked.**


Microsoft Certificate Services - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.2.150/certsrv/certrqma.asp
Go Links
Google Go Bookmarks 0 blocked Check AutoLink Settings
Microsoft Certificate Services -- COMTARSIA ROOT CA 10
Home
Advanced Certificate Request
Identifying Information:
Name: agent.comtarsia.com
E-Mail:
Company:
Department:
City:
State:
Country/Region:
Type of Certificate Needed:
Other...
OID: 823.1.3.2,1.3.6.1.5.5.7.3.1
Key Options:
Create new key set
Use existing key set
CSP: Microsoft Enhanced Cryptographic Provider v1.0
Key Usage: Exchange Signature Both
Key Size: 1024 Min: 384 Max:16384 (common key sizes: 512 1024 2048 4096 8192 16384 )
Automatic key container name
User specified key container name
Mark keys as exportable
Export keys to file
Enable strong private key protection
Store certificate in the local computer certificate store
Stores the certificate in the local computer store
Local intranet

### 3.2.4 Certificate for the Comtarsia Logon Client

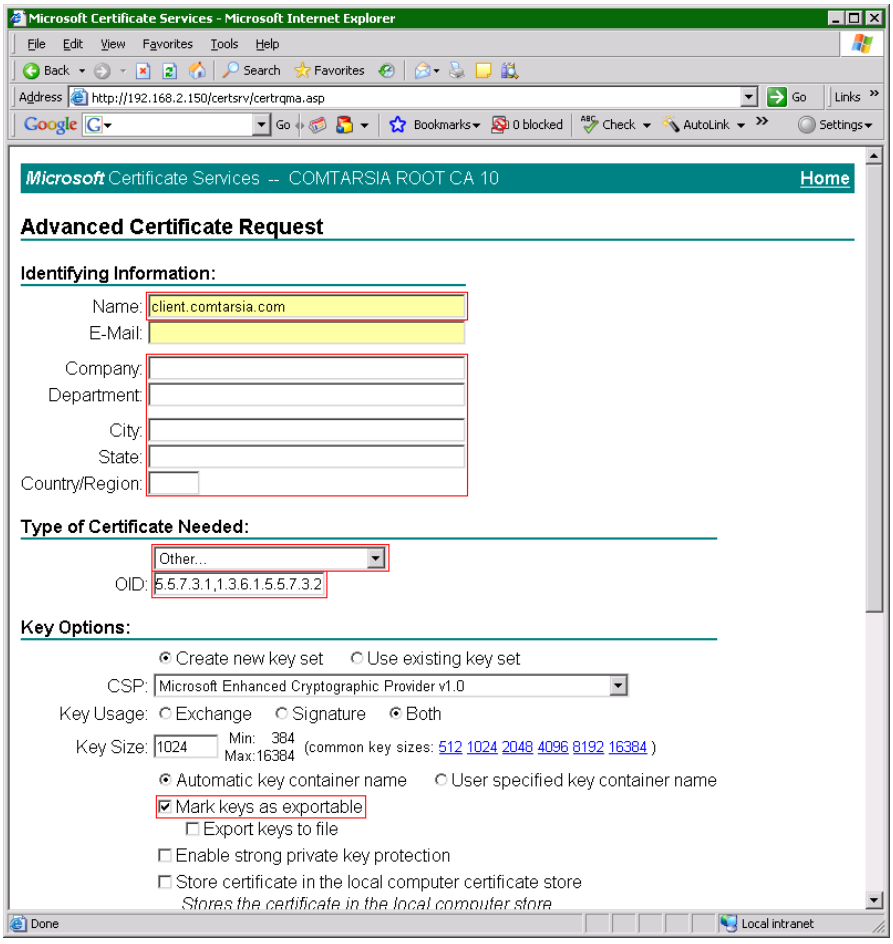
The Logon Client needs a certificate for client authentication, as it connects to the SignOn Proxy.
Additionally, the Comtarsia object identifier for the Logon Client (1.3.6.1.4.1.13823.1.3.3) can be added to the certificate for enhanced security.

If the hostname check will be enabled in the trust options it is important that the common name in the certificate subject matches the FQDN of the Logon Client server. This check is done using DNS reverse lookup on the IP address of the Logon Client.

For „Type of Certificate" choose „other" which allows specifying the required OIDs.
The OIDs are seperated with a comma:
1.3.6.1.5.5.7.3.2 client authentication (id_kp_serverAuth)
1.3.6.1.4.1.13823.1.3.3 Comtarsia Logon Client OID
OID: 1.3.6.1.5.5.7.3.2,1.3.6.1.4.1.13823.1.3.3

Fields like „Company", „Department", City, State, Country/Region are optional and are not checked.

**Important: To be able to export the private key later, the checkbox "Mark key as exportable" must be checked.**



### 3.2.5 Certificate for the Comtarsia Web Gateway

The Comtarsia Web Gateway acts as a client in the SignOn Gate family and therefore a certificate like for the Logon Client can be used.

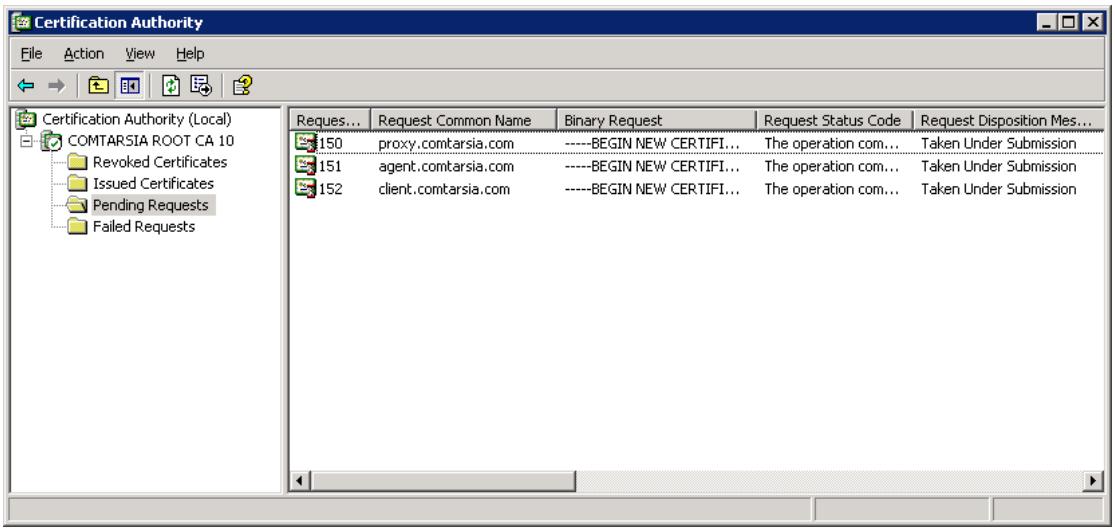
### 3.2.6 Certificate for the Comtarsia LDAP Directory Replicator

The Comtarsia LDAP Directory Replicator acts as a client in the SignOn Gate family and therefore a certificate like for the Logon Client can be used.

### 3.2.7 Issue the certificates

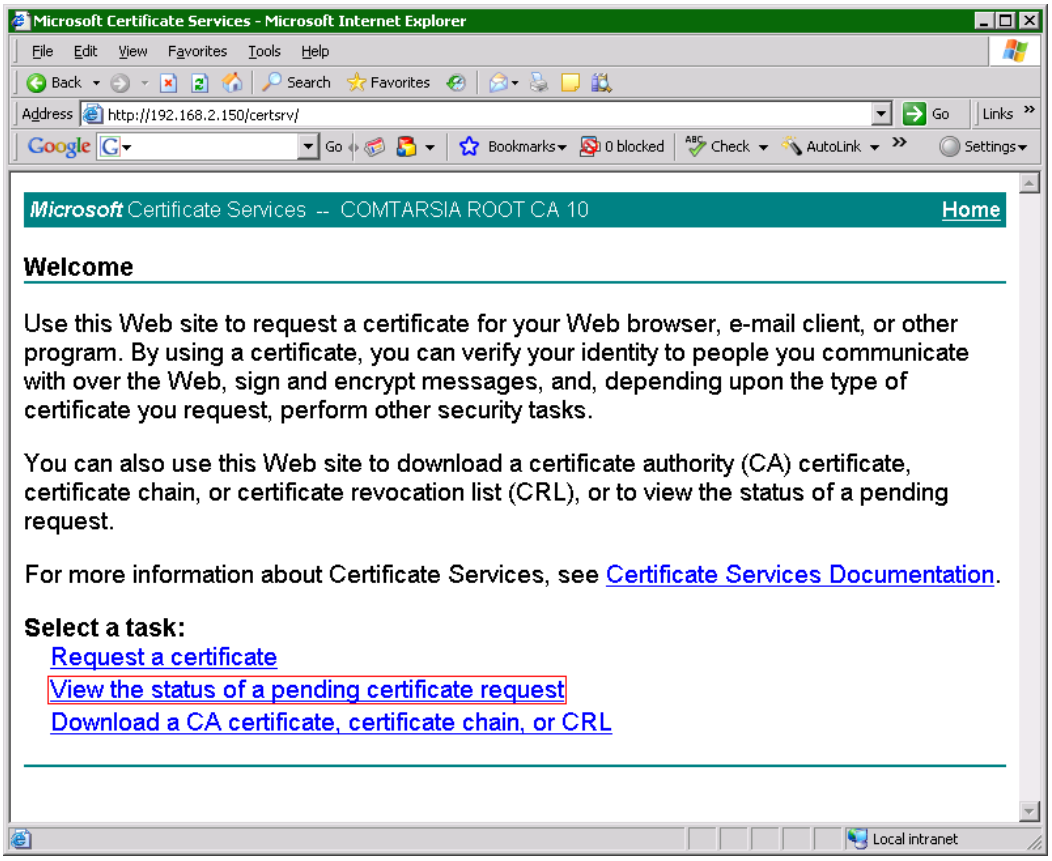
Depending on the CA configuration the requested certificates are either issues automatically and immediately available for download, or the need to be manually issued, which is shown in the following paragraph.

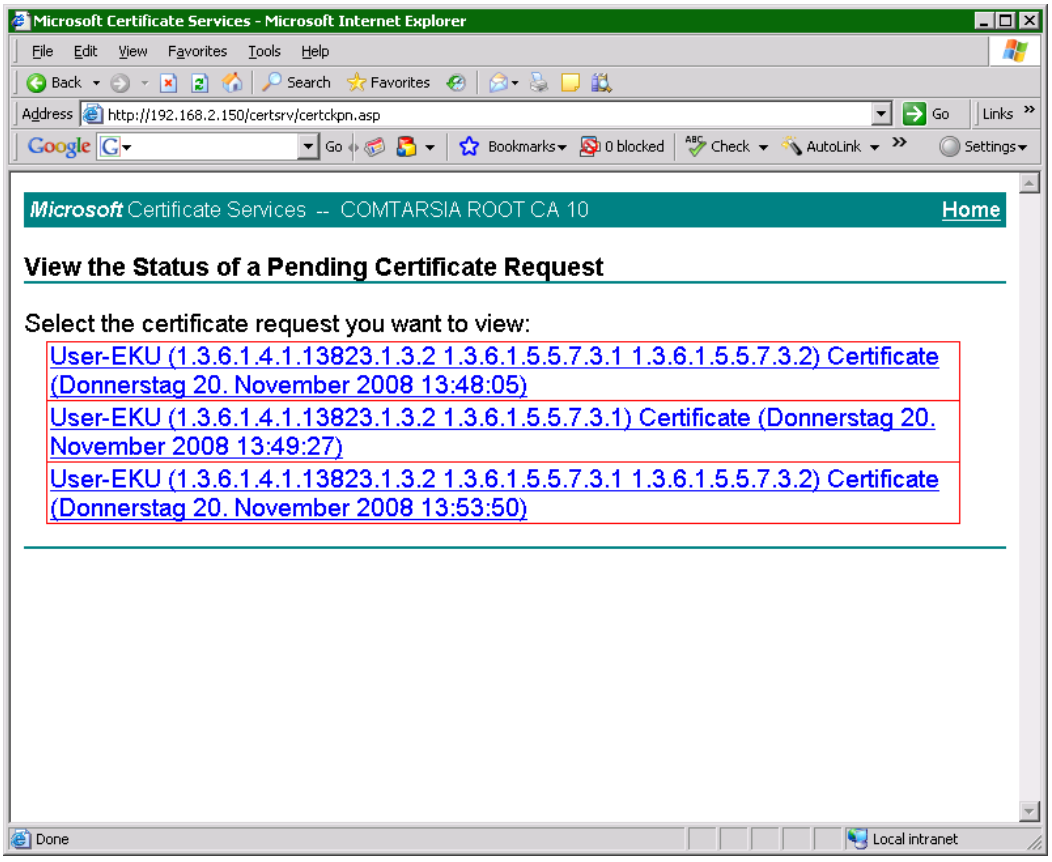
To manually issue a certificate, open up the „Certificate Authority" MMC snap-in. Under „Pending Requests" the previous requested certificates can be seen. Click with the right mouse button on an entry opens up a menu which allows approving or denying the request.

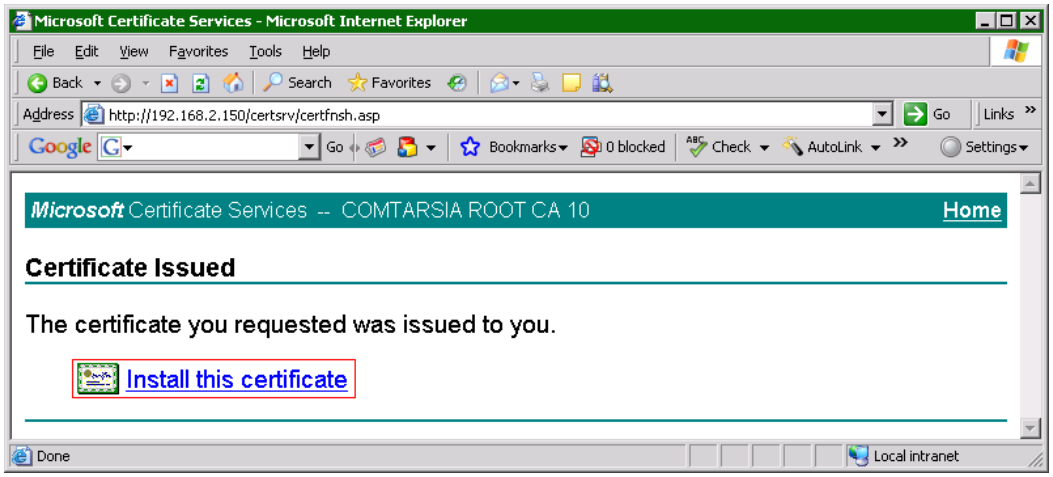



### 3.2.8 Retrieve and convert the certificates

After the CA has issued the certificate, the web interface can be used again to pick up the certificate.
**Important: The certificate download over the web interface is only possible once.**

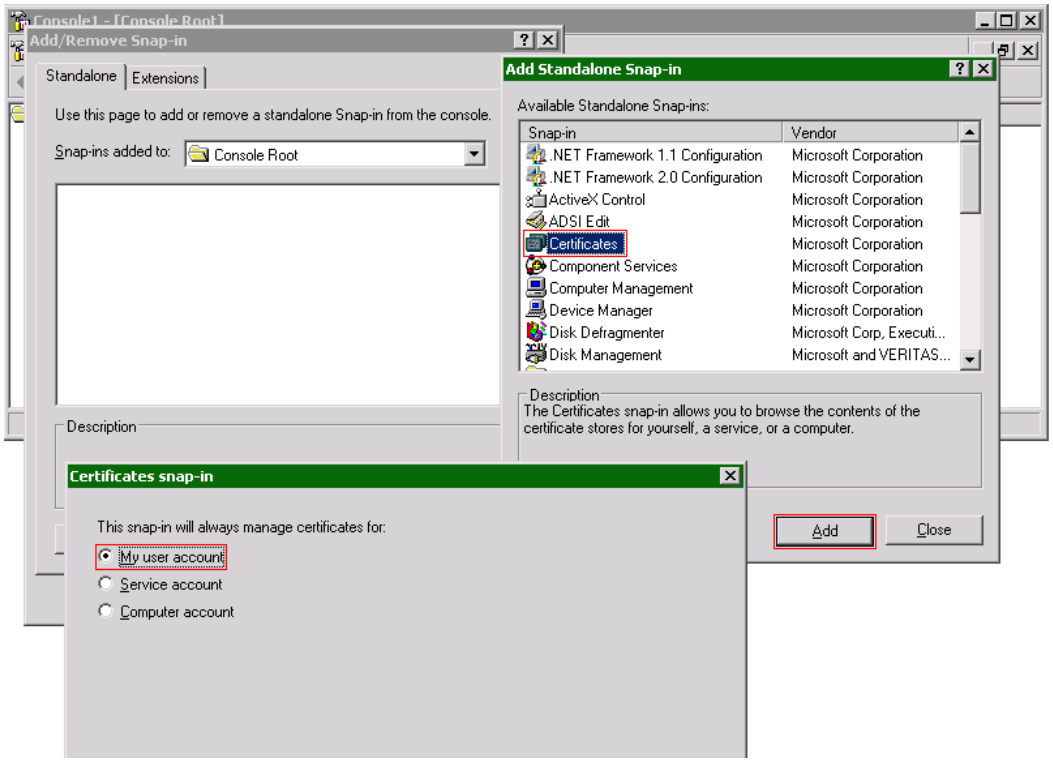

Microsoft Certificate Services - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.2.150/certsrv/
Go Links
Google Go Bookmarks 0 blocked Check AutoLink Settings
Microsoft Certificate Services -- COMTARSIA ROOT CA 10
Home
Welcome
Use this Web site to request a certificate for your Web browser, e-mail client, or other program. By using a certificate, you can verify your identity to people you communicate with over the Web, sign and encrypt messages, and, depending upon the type of certificate you request, perform other security tasks.
You can also use this Web site to download a certificate authority (CA) certificate, certificate chain, or certificate revocation list (CRL), or to view the status of a pending request.
For more information about Certificate Services, see Certificate Services Documentation.
Select a task:
Request a certificate
View the status of a pending certificate request
Download a CA certificate, certificate chain, or CRL
Local intranet

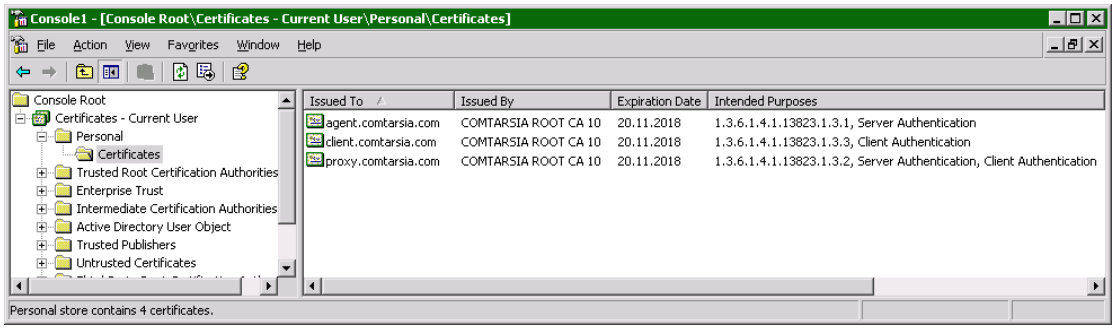

Microsoft Certificate Services - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://192.168.2.150/certsrv/certckpn.asp
Go Links
Google Go Bookmarks 0 blocked Check AutoLink Settings
Microsoft Certificate Services -- COMTARSIA ROOT CA 10
Home
View the Status of a Pending Certificate Request
Select the certificate request you want to view:
User-EKU (1.3.6.1.4.1.13823.1.3.2 1.3.6.1.5.5.7.3.1 1.3.6.1.5.5.7.3.2) Certificate (Donnerstag 20. November 2008 13:48:05)
User-EKU (1.3.6.1.4.1.13823.1.3.2 1.3.6.1.5.5.7.3.1) Certificate (Donnerstag 20. November 2008 13:49:27)
User-EKU (1.3.6.1.4.1.13823.1.3.2 1.3.6.1.5.5.7.3.1 1.3.6.1.5.5.7.3.2) Certificate (Donnerstag 20. November 2008 13:53:50)
Done
Local intranet



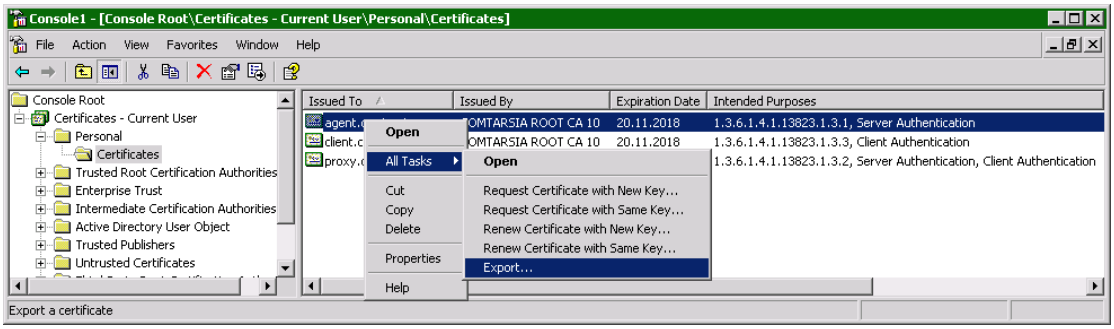

The „Installation"-process transfers the certificate and private key to the local certificate store of the current user.

Using the "Certificates" MMC snap-in the certificate and private key can be exported into a file.



The certificate can be exported with a right-click and then select „All Tasks“ -> „Export“.



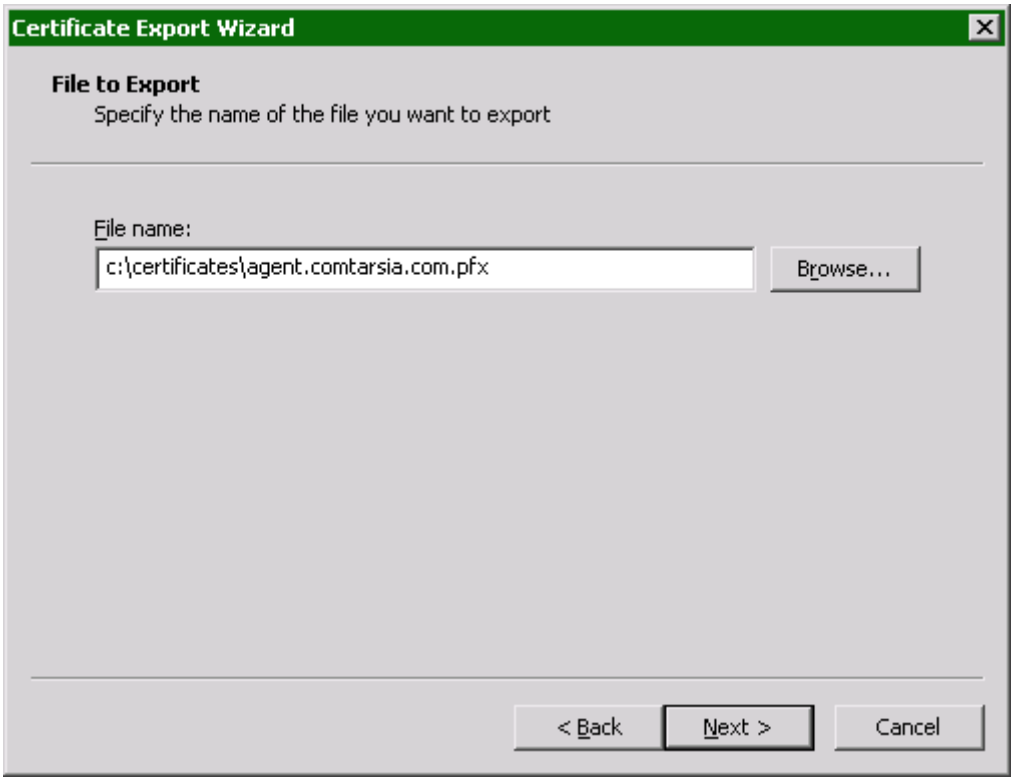


It is important to also export the private key!


Certificate Export Wizard
Export Private Key
You can choose to export the private key with the certificate.
Private keys are password protected. If you want to export the private key with the certificate, you must type a password on a later page.
Do you want to export the private key with the certificate?
Yes, export the private key
No, do not export the private key
< Back
Next >
Cancel

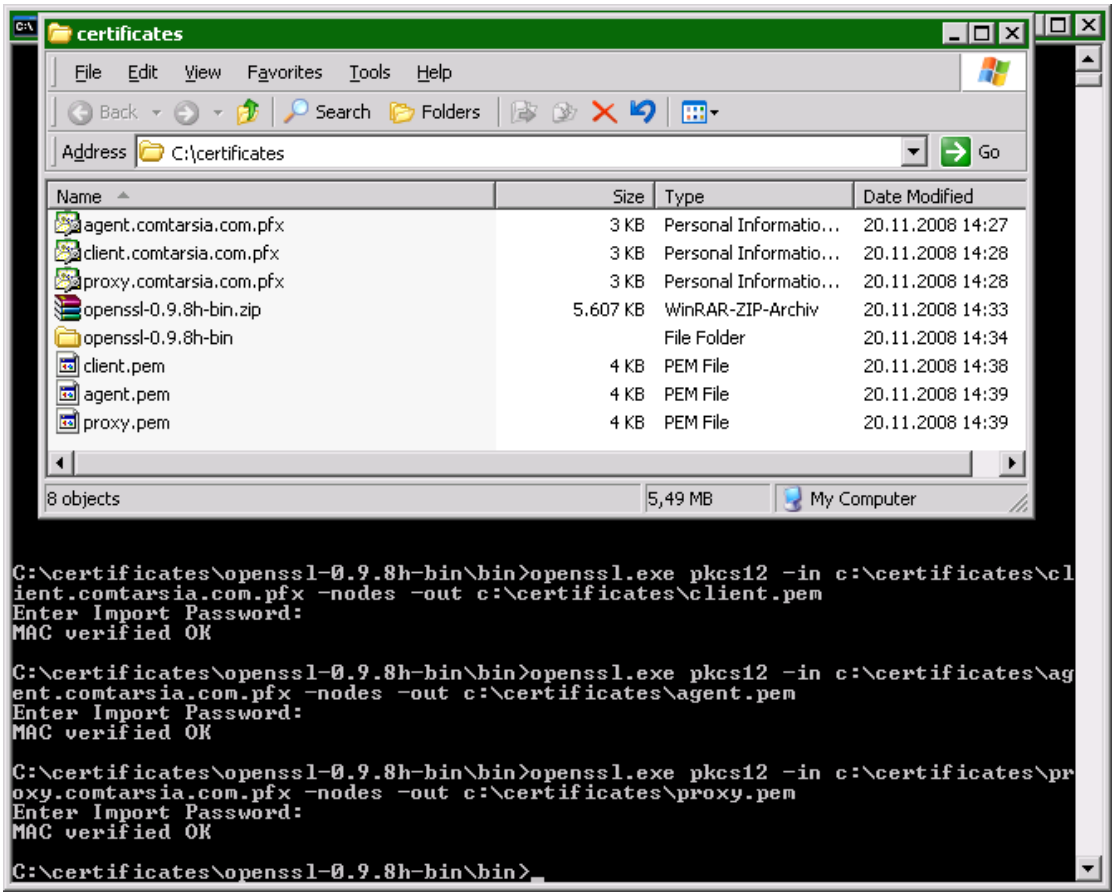

Certificate Export Wizard
Export File Format
Certificates can be exported in a variety of file formats.
Select the format you want to use:
DER encoded binary X.509 (.CER)
Base-64 encoded X.509 (.CER)
Cryptographic Message Syntax Standard - PKCS #7 Certificates (.P7B)
Include all certificates in the certification path if possible
Personal Information Exchange - PKCS #12 (.PFX)
Include all certificates in the certification path if possible
Enable strong protection (requires IE 5.0, NT 4.0 SP4 or above)
Delete the private key if the export is successful
< Back
Next >
Cancel


Certificate Export Wizard
Password
To maintain security, you must protect the private key by using a password.
Type and confirm a password.
Password:
Confirm password:
< Back
Next >
Cancel


Certificate Export Wizard
File to Export
Specify the name of the file you want to export
File name:
c:\certificates\agent.comtarsia.com.pfx
Browse...
< Back
Next >
Cancel


Afterward, the certificates must be converted to a different file format using OpenSSL.

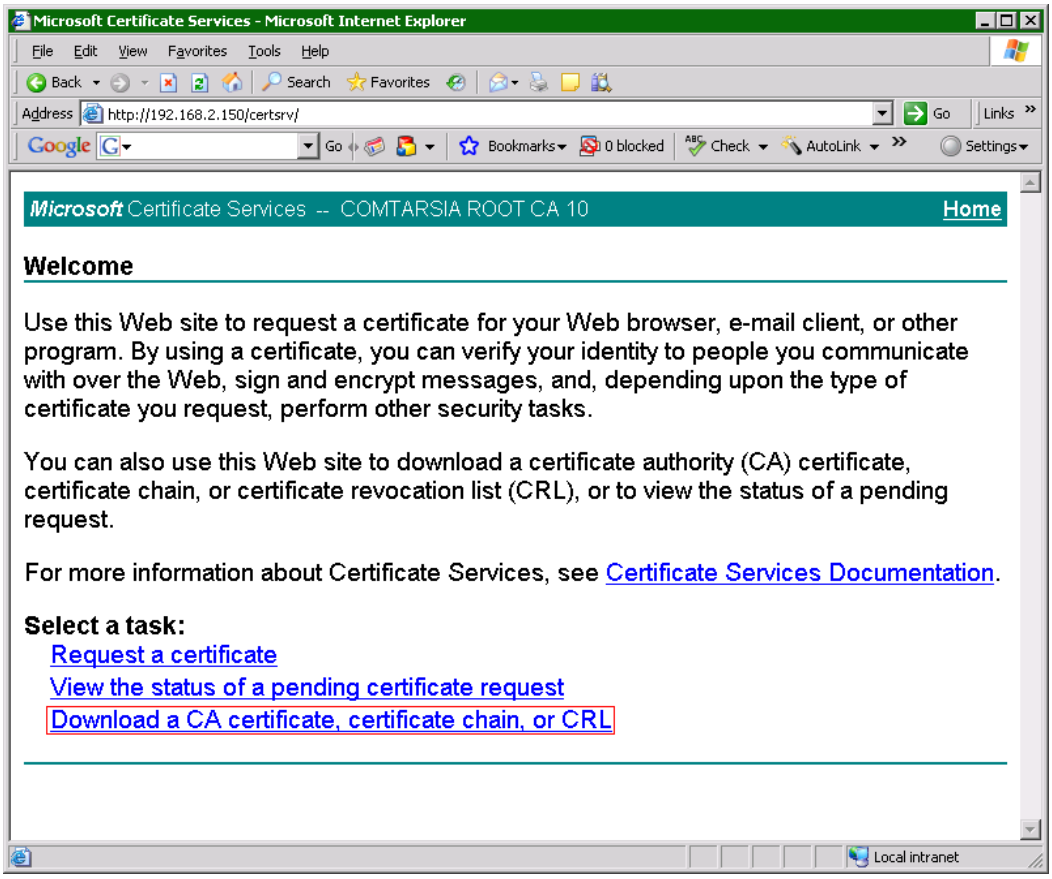
**openssl pkcs12 -in c:\certs\<*yourcert*>.pfx -nodes -out c:\certs\<*yourcert*>.pem**



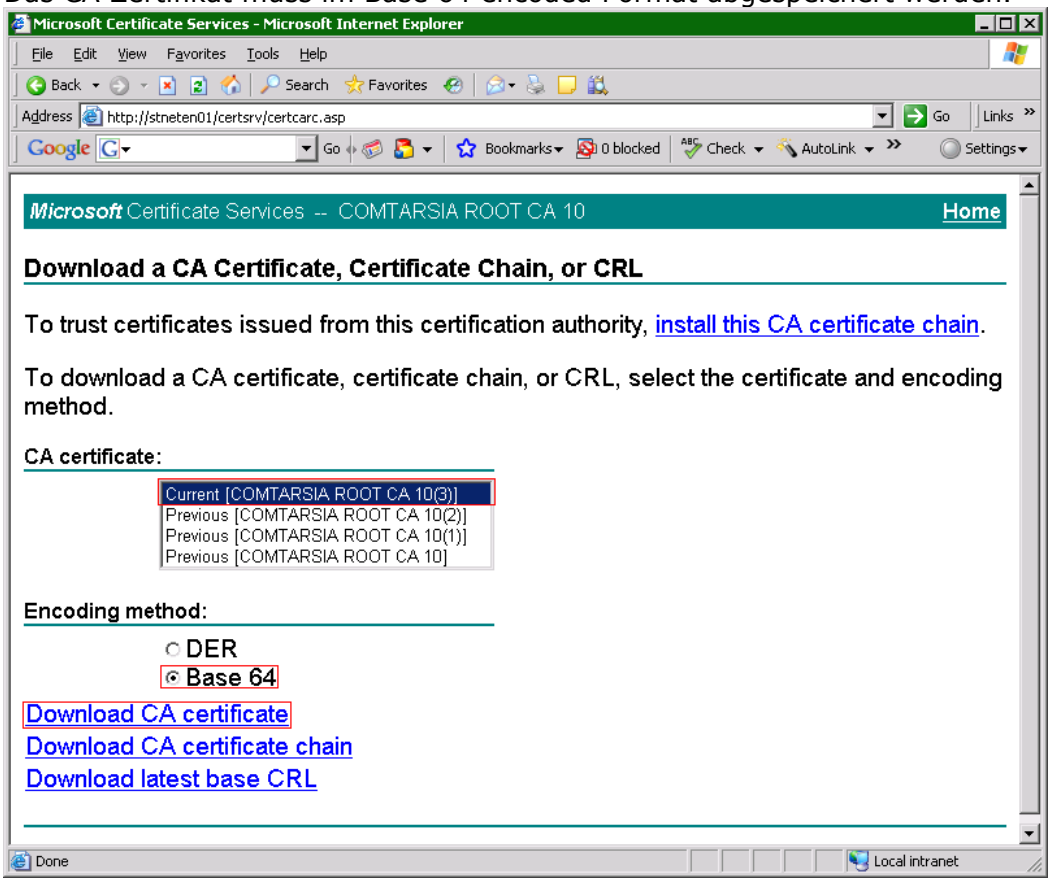

The so created „pem"-files contain the certificate as well as the private key and can be directly specified as key/certificate for Comtarsia products.

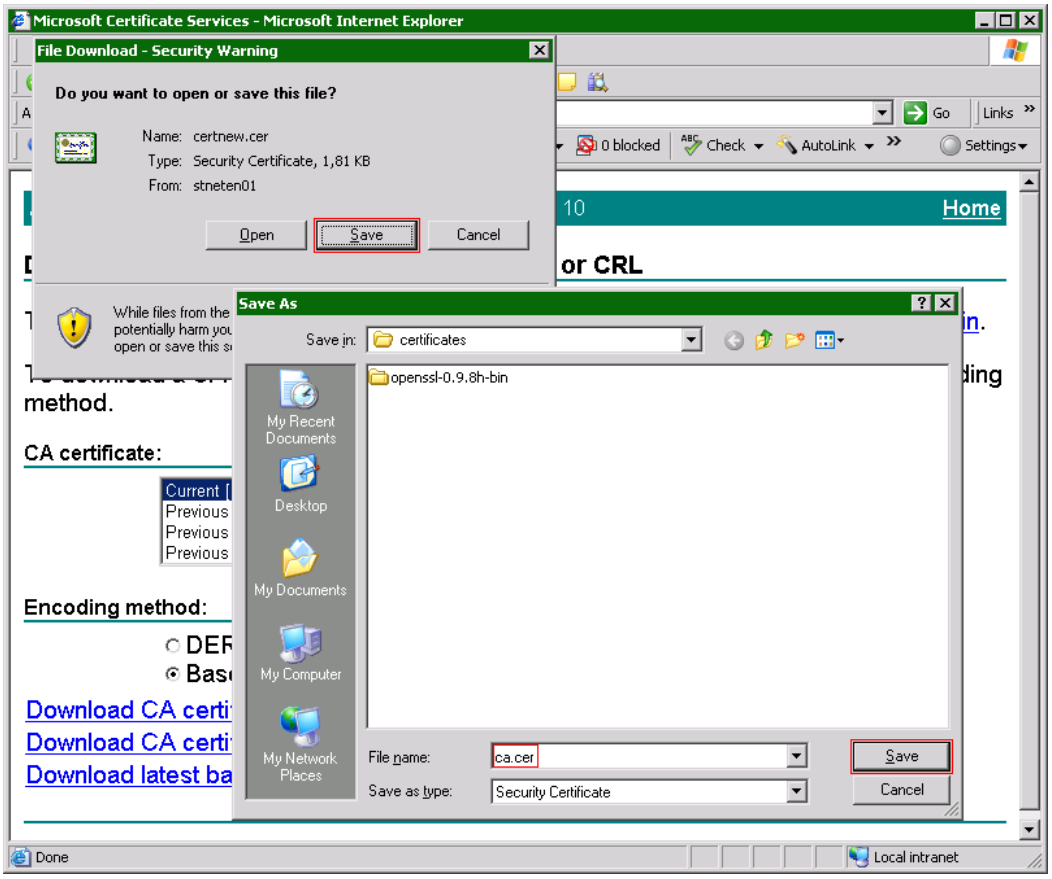
### 3.2.9 CA certificate

To enable each product to verify the validity of the peer certificates, each product also needs the CA certificate. This CA certificate can also be downloaded from the CA web interface.




Das CA Zertifikat muss im Base 64 encoded Format abgespeichert werden.




Microsoft Certificate Services - Microsoft Internet Explorer
File Download - Security Warning
Do you want to open or save this file?
Name: certnew.cer
Type: Security Certificate, 1,81 KB
From: stneten01
Open
Save
Cancel
While files from the
potentially harm you
open or save this s
Save As
Save in:
certificates
openssl-0.9.8h-bin
My Recent Documents
Desktop
My Documents
My Computer
My Network Places
File name:
ca.cer
Save as type:
Security Certificate
Save
Cancel
Home
or CRL
method.
CA certificate:
Current
Previous
Previous
Previous
Encoding method:
DER
Base
Download CA certi
Download CA certi
Download latest ba
Done
Local intranet

# 4. Configuration of the Comtarsia products

By default, every product checks that the certificate used by the peer is valid, which simplified means that it was issued by a trusted certificate authority and it is inside its validity period.

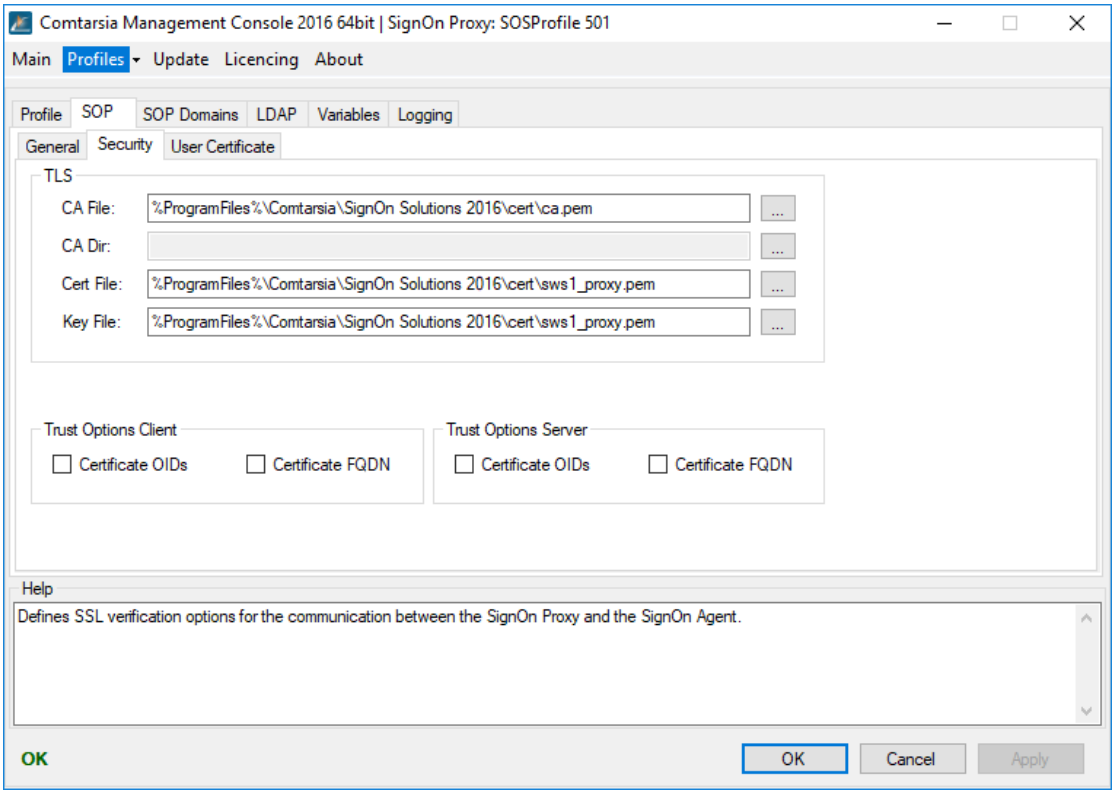
Additionally, all products offer further certificate checks, that can be enabled using the Comtarsia Management Console or the supplied configuration tool.

| | |
|---|---|
| No Check / Simple Mode | Only the above-mentioned checks are performed. |
| OID's | Additionally, the existence of the correct object identifier is checked. |
| FQDN | Additionally, a reverse lookup on the peer's IP address is performed. The resulting name must match the „CN" part of the certificate subject. |

In addition to the security measures mentioned above, the SignOn Agent offers the possibility to limit the allowed peers to specific IP addresses, which can the configured in the Management Console.

## 4.1 Comtarsia SignOn Proxy

All certificate trust options can be defined using the Management Console.

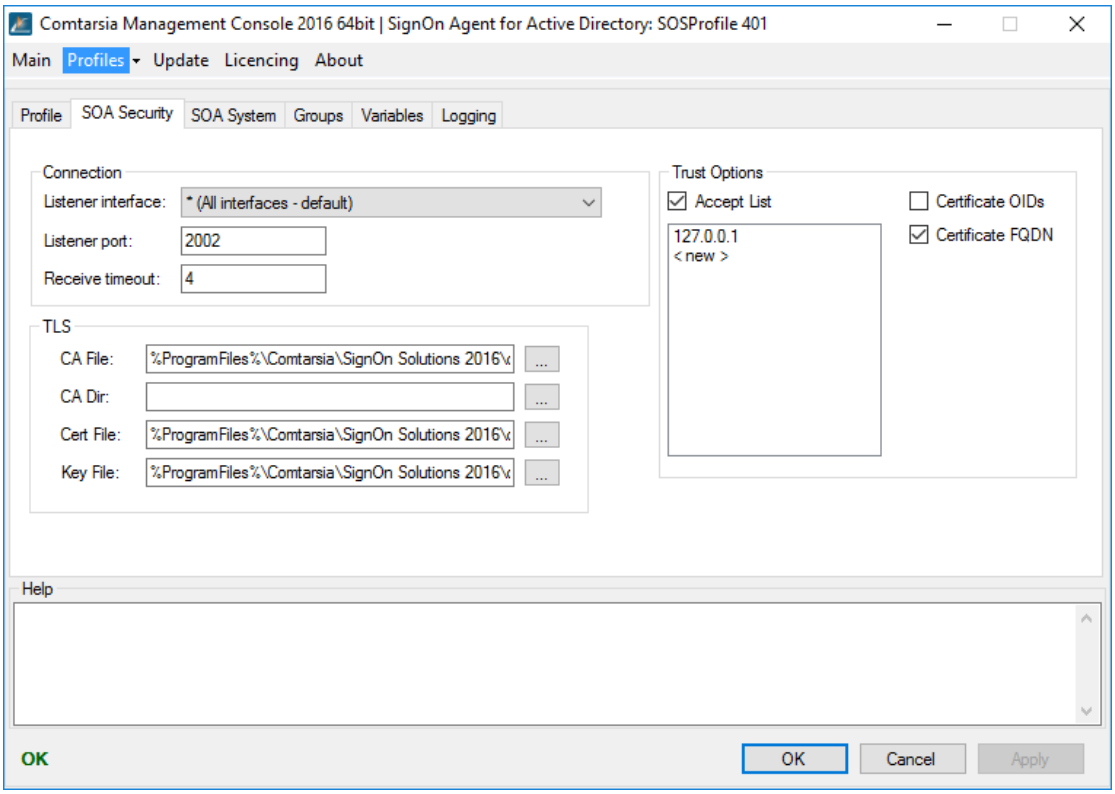



The most basic options that need to be set are the „CA File", which points to the certificate of the CA. Also the "Cert File" and the "Key File" must point to a certificate respective private key to use by the SignOn Proxy. It is possible to let "Cert File" and "Key File" point to the same file if this file contains the certificate and the key. If more than one CA certificate needs to be used, all certificates can be copied together in one file. The format for all files must be PEM encoded.

In the trust options section below the additional security option as described in 4 can be enabled. “Trust Options Client” are the options that are used to validate connections to clients (Logon Client, Web Gateway, LDAP Directory Replicator). “Trust Options Server” are the options that are used to validate connections to SignOn Agents.

## 4.2 Comtarsia SignOn Agent

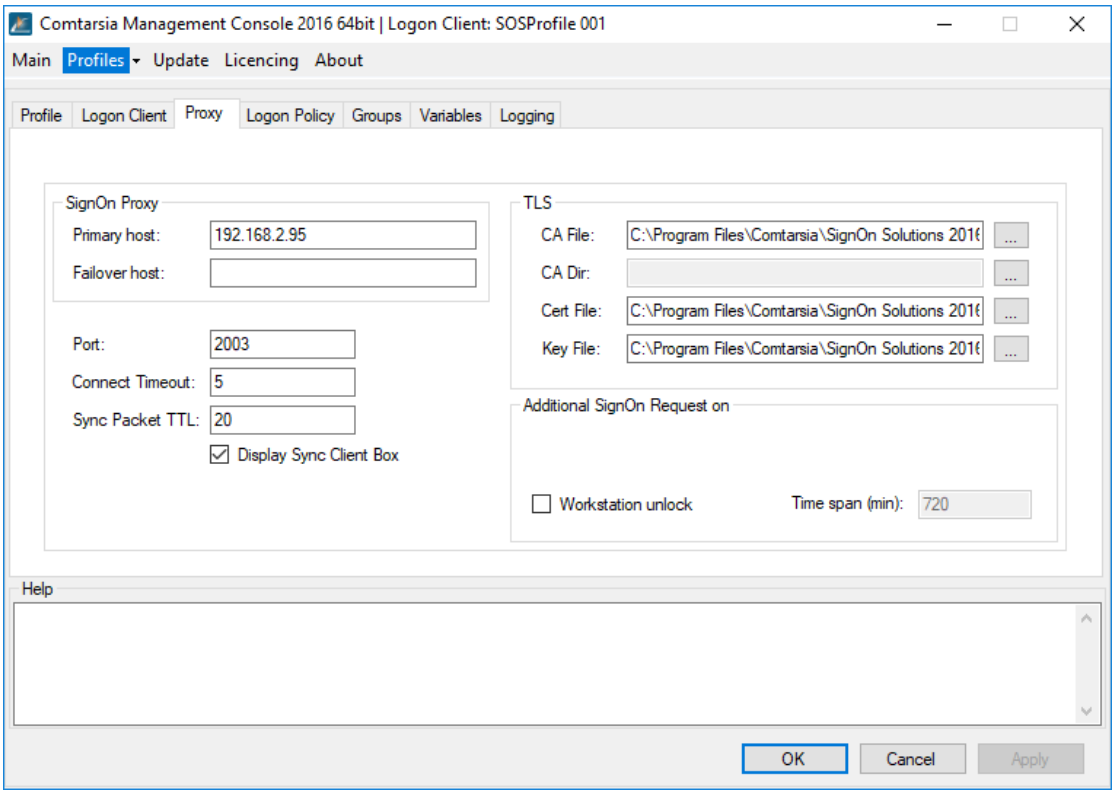
All certificate trust options can be defined using the Management Console.




The most basic options that need to be set are the „CA File", which points to the certificate of the CA. Also the "Cert File" and the "Key File" must point to a certificate respective private key to use by the SignOn Agent. It is possible to let "Cert File" and "Key File" point to the same file if this file contains the certificate and the key. If more than one CA certificate needs to be used, all certificates can be copied together in one file. The format for all files must be PEM encoded.
In the trust options section below the additional security option as described in 4 can be enabled. "Trust Options" are the options that are used to validate connections to the SignOn Proxy.

## 4.3 Comtarsia Logon Client

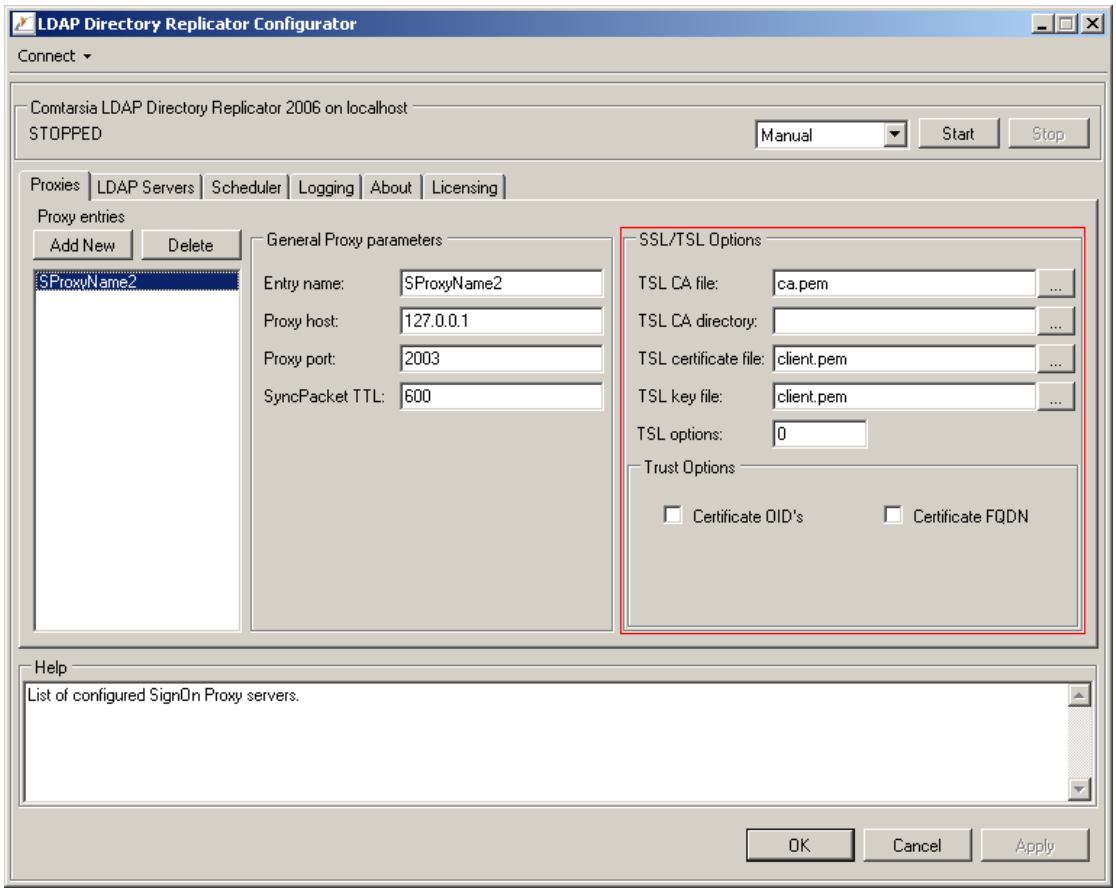
All certificate trust options can be defined using the Management Console.




The most basic options that need to be set are the „CA File", which points to the certificate of the CA. Also the "Cert File" and the "Key File" must point to a certificate respective private key to use by the Logon Client. It is possible to let "Cert File" and "Key File" point to the same file if this file contains the certificate and the key. If more than one CA certificate needs to be used, all certificates can be copied together in one file. The format for all files must be PEM encoded.
In the trust options section below the additional security option as described in 4 can be enabled. "Trust Options" are the options that are used to validate connections to the SignOn Proxy.

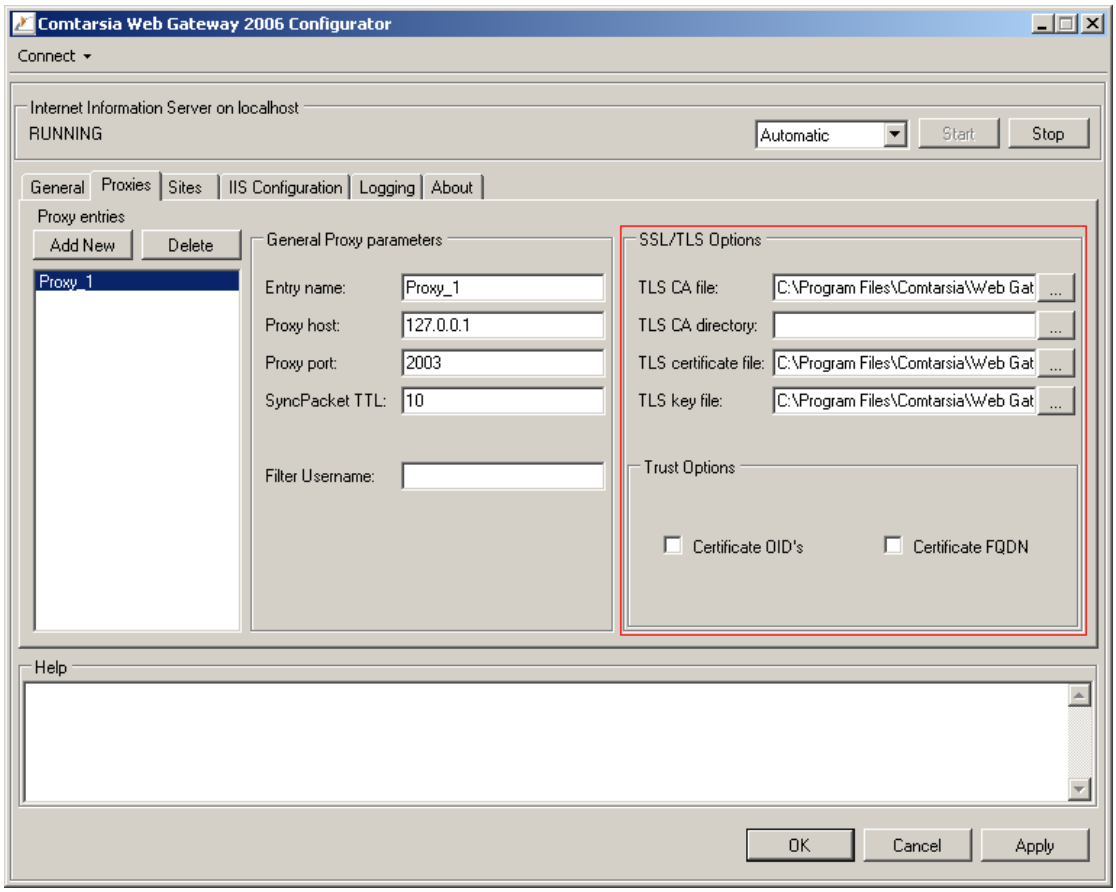
## 4.4 Comtarsia LDAP Directory Replicator

All parameters can be easily configured using the supplied configuration tool.



## 4.5 Comtarsia Web Gatewateway

All parameters can be easily configured using the supplied configuration tool.



# 5. Enhancing TLS Security

To further enhance security, lower TLS versions can be disabled or certain TLS cipher suites can be deactivated if not needed for older clients or servers. The following paragraphs will provide a guide on how to do this.
The Comtarsia SignOn Solution product are available for different platforms and employ different TLS stacks. Therefore, it can be slightly different for every product on how to configure the TLS settings.

## 5.1 Comtarsia internal TLS communication

### 5.1.1 Logon Client, SignOn Proxy and SignOn Agents

The OpenSSL library is used as a TLS implementation. Per default TLS Version 1.2 and up are enabled. If you need to support older Comtarsia products that do not support TLS 1.2 yet, the parameter REG_DWORD:"tlsMinProtocolVersion" can be used to set the minimum allowed protocol version. Possible values are hex 301 for TLS 1.0, hex 302 for TLS 1.1 and hex 303 for TLS 1.2, which is also the default value if this parameter is not set.

Per default, all by OpenSSL supported ciphers can be used for the different TLS protocols, excluding ciphers using the SHA1 hashing algorithm. This behavior can be changed using the parameter REG_SZ:" tlsCiphers". The default value for this parameter is „ALL:!SHA". To allow all supported ciphers for a TLS protocol, use „ALL". One or more cipher names can also be entered directly into this value to restrict the usage to these ciphers. Take care that both communicating parties need at least one common shared cipher. The syntax of this value is further described in this link: https://www.openssl.org/docs/man3.1/man1/openssl-ciphers.html

The registry location of the above mentioned parameters can be found under the following registry paths. The parameters have to be created if they don't exist yet.
For Logon Client:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 001\SyncClient

For SignOn Proxy:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 501\SignOnProxy

For SignOn Agent Active Directory:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 401\SignOnAgent

For SignOn Agent LDAP:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 411\SignOnAgent

For SignOn Agent System Windows:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 421\SignOnAgent

### 5.1.2 LDAP Gateway, RADIUSG Gateway and AuthSRVM under Windows

These products use the Windows Schannel librray for TLS communication. Using the parameter REG_DWORD:„enabledSslProtocols" the TLS protocols to use can

be set. This is a bitmask, so a comination of values is allowed. Possible values are 0 to let the operating sytsem choose the protocol, 192 for TLS 1.0, 768 for TLS 1.1, 3072 for TLS 1.2 and 12288 for TLS 1.3. Using TLS 1.3 also requires the underlying Schannel layer of the operating sytem to support TLS 1.3.

The registry location of the above mentioned parameter can be found under the following registry paths. The parameter has to be created if it doesn't exist yet.
For LDAP Gateway:
HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 121\LDAPGateway
"enabledSslProtocolsLDAPServer"
"enabledSslProtocolsLDAPClient"
"enabledSslProtocolsSyncClient"

HKEY_LOCAL_MACHINE\SOFTWARE\Comtarsia\SOSProfile 141\RADIUSGateway
"enabledSslProtocolsSyncClient"

If a specific cipher algorithms should be disabled, this can only be done on a system wide level. For example, to disable all ciphers that use SHA1 as hashing algorithm, set the following registry value:
REG_DWORD:HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Hashes\SHA\Enabled=0

A discription of all possible values can be found here:
https://learn.microsoft.com/en-us/troubleshoot/windows-server/windows-security/restrict-cryptographic-algorithms-protocols-schannel

## 5.2 LDAP Gateway LDAP communication under Windows

The LDAP Gateway uses for its socket listener and its client sockets the Windows Schannel for TLS communication. Using the parameter REG_DWORD:„enabledSslProtocols" the TLS protocols to use can be set. This is a bitmask, so a comination of values is allowed. Possible values are 0 to let the operating sytsem choose the protocol, 192 for TLS 1.0, 768 for TLS 1.1, 3072 for TLS 1.2 and 12288 for TLS 1.3. Using TLS 1.3 also requires the underlying Schannel layer of the operating sytem to support TLS 1.3.

If a specific cipher algorithms should be disabled, this can only be done on a system wide level. For example, to disable all ciphers that use SHA1 as hashing algorithm, set the following registry value:
REG_DWORD:HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Hashes\SHA\Enabled=0

A discription of all possible values can be found here:
https://learn.microsoft.com/en-us/troubleshoot/windows-server/windows-security/restrict-cryptographic-algorithms-protocols-schannel

## 5.3 Logon Client and SignOn Proxy LDAP communication under Windows

These products also use the Windows Schannel for TLS communication. Here, the minimum supported TLS versions currently cannot be configured and should be set on the server if required. Specific cipher algorithms can be disabled on a

system wide level. For example, to disable all ciphers that use SHA1 as hashing algorithm, set the following registry value:
REG_DWORD:HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Hashes\SHA\Enabled=0

A discription of all possible values can be found here:
https://learn.microsoft.com/en-us/troubleshoot/windows-server/windows-security/restrict-cryptographic-algorithms-protocols-schannel