

Technical Note: Model 1200 SecureSync

Windows HTTPS Certificate error pop-up message or Need to create a stronger HTTPS Certificate for network security requirements

Purpose: The purpose of this document is to provide information about replacing the Spectracom SecureSync's factory default HTTPS Certificate with either:

- 1) A "self-signed" (custom/locally created) HTTPS Certificate.
- 2) A purchased SSL Certificate.

This process may need to be performed due to three primary reasons:

- 1) To allow a web browser (such as FireFox, Internet Explorer, Edge or Chrome) to only need to accept the NTP server's HTTPS Certificate once, the first time that you connect to it.
- 2) Your network security requirements dictate that even stronger HTTPS encryption algorithms or longer keys need to be used for the HTTPS connection to the NTP server.
- 3) To prevent network security scans from reporting potential vulnerabilities due to a "self-signed" or a "too weak" Certificate being installed in the NTP server.

Note: Refer also to the online Model 1200 SecureSync user guide for additional information pertaining to topics such as the HTTPS certificate (this online document provides the most up-to-date information about the SecureSync, and can be easily searched). This information pertaining to the HTTPS certificate can be found, starting at: https://safran-navigation-timing.com/manuals/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_cert_formats.htm

Table of Contents

Generating a replacement SSL Certificate for HTTPS	5
Certificate Format types accepted/supported by the SecureSync	7
RFC-2818 Compliancy/Subject Alternate Names (UCC / SANS fields for Common Names).....	7
Generate Certificate Request (Software versions 5.8.0 and above).....	8
Generate Certificate Request (Software versions 5.1.2 through 5.7.1)	14
Import an external, locally generated, or purchased Certificate (all software versions 5.1.2 and higher)	19
Viewing/Installing an HTTPS Certificate with Internet Explorer	22
Viewing/Installing an HTTPS Certificate with Mozilla FireFox	26
Need to reset a custom HTTPS Certificate back to the factory default Certificate (defcert or defcert -sha1 CLI commands)	30
Safran Technical Support.....	31

Q. Windows Certificate error – Is there a step-by-step procedure for us to create then install the Windows Certificate? We get the Certificate error every time we try to enter the SecureSync’s configuration page.

A. A non-SecureSync default Certificate can be accepted and installed into the Windows PC to prevent the Security Alert message from being displayed during each login. By default, there is an available default SecureSync “self-signed” Certificate that is generic for all Safran time servers. You can tell Windows to trust this Certificate by installing it using the methods below.

This step will clear the first yellow exclamation point message in the Security Alert pop-up message. However, the default DNS hostname is “Spectracom” and the Certificate is generated around this being the “login” name. Unless you are using DNS on the network and are logging in by entering this assigned factory default DNS name into the address line (instead of entering the IP address of the SecureSync server), the Certificate will still be considered by Windows to be invalid.

For the Certificate to be fully valid (to keep the Security Alert message from being displayed) you need to use DNS to login by entering “Spectracom” into the address line instead of the IP address of the time server. Or, if you desire to enter the IP address of the time server (or a different DNS name besides the default of “Spectracom” has been configured), a new self-signed Certificate based on the IP address or the new DNS Host name needs to be created via the web browser.

For all three conditions to be met, to prevent the Security Alert message from being displayed, the Spectracom default HTTPS Certificate may need to be replaced with a user-created Certificate that has the correct credentials based on the desired login of the NTP server. Once the new self-signed Certificate has been created, this new self-signed Certificate needs to be accepted/installed into the Windows PC.

This same procedure can also be used to replace the default HTTPS Certificate, if an even stronger HTTPS Certificate is required to meet your network security requirements. Different Signature algorithms and Key bit lengths for the encryption algorithms can be selected when creating a new HTTPS Certificate.

The first step for all scenarios is to first create a new Certificate Request (CR)/Certificate Signing Request (CSR) by filling in all the fields in the “**Certificate Request Parameters**” section of the HTTPS Certificate page of the browser interface. With these fields all filled-in, a new Certificate Request (which is based upon these values) can be generated to replace the Spectracom default Certificate Request (CR/CSR).

With a new CR/CSR now created in the NTP server, and if it’s desired to either purchase an SSL Certificate or to have a local CA generate the new Certificate, the new CR/CSR can now be extracted from the NTP server and have it be used to generate the new HTTPS Certificate. Once the new Certificate has been either purchased or created by a local/private “Certificate Authority”, the new Certificate can then be uploaded into the NTP server. Otherwise, if it’s desired to simply use a self-signed Certificate generated by the NTP server, the CR/CSR does not need to be exported.

Terms associated with Certificates for HTTPS security:

Self-signed certificate: an identity **certificate** that is **signed** by the same entity whose identity it certifies.

Certificate Authority (CA): an entity that issues [digital certificates](#).

Certificate Request (CR) / Certificate Signing Request (CSR): a message sent from an applicant to a [certificate authority](#) in order to apply for a [digital identity certificate](#). It usually contains the public key for which the certificate should be issued, identifying information (such as a domain name) and integrity protection (e.g., a digital signature).

X.509: a standard that defines the format of public key certificates

Various ways to present certificates and their components:

PEM (Privacy Enhanced Mail): a container format that may include just the public certificate (such as with Apache installs, and CA certificate files `/etc/ssl/certs`) or may include an entire certificate chain including public key, private key, and root certificates.

DER: (Distinguished Encoding Representation): The parent format of PEM. It is the binary form of the certificate (and does not contain the "BEGIN **CERTIFICATE**/END **CERTIFICATE**" statements, like PEM certificates contain)

PKCS#7: An open standard used by Java and supported by Windows. Does not contain private key material.

Generating a replacement SSL Certificate for HTTPS

IMPORTANT INFORMATION

SecureSyncs contain a “factory default” SSL certificate, allowing HTTPS connections “out-of-the-box”. For security purposes and for computers to consider the Certificate “valid”, this default certificate needs to be replaced with one generated external to the SecureSync.

- **In software versions 5.6.0 and above**, the SSL certificate is generated by default using the **SHA256** encryption cypher (though a user can choose to create a new certificate using a “less secure” cipher, if necessary, such as when using Internet Explorer versions 8 and below)
- **In earlier software versions 5.5.1 and below**, the SSL certificate is generated using the **SHA1** encryption cypher (now considered a “less secure” cipher, than the SHA256 cipher)

NOTE: SecureSyncs **do not support wildcard SSL certificates!** The SecureSync’s unique private key (“private.key”) file **must** be generated within/remain in the SecureSync (for security reasons, the SecureSync’s private key cannot be exported out, or replaced by externally generating/importing one into the SecureSync).

The SecureSync’s web browser is used to generate a new, applicable Certificate Signature Request (CR/CSR) for just this one particular SecureSync. After being generated by a user filling in several fields in the browser, the new CSR is copy/pasted off the SecureSync and made into a file having a file extension of “.csr”. This CSR file needs to be provided to either a local “CA” (Certificate Authority), or it can be optionally sent to an organization such as “Verisign” to purchase a new Certificate.

The “CA” **MUST** use the provided, unique CSR (generated in and copy/pasted out of each SecureSync) to **create the new Certificate**. This process “pairs” the generated “public certificate” with the unique “private key” stored in only the one SecureSync that the particular CSR was obtained from. The generated public certificate provided by the CA can then be either copy/pasted or uploaded (depending on the file type of the provided Certificate) into that same SecureSync via its browser.

If a replacement certificate (which was not created using the CR/CSR copy/pasted out of that same SecureSync is placed into that, or any other SecureSync) the certificate/login will fail when trying to login to the SecureSync via HTTPS. The installed certificate will need to be deleted/removed from the SecureSync, to restore connection to the browser. This can be performed via the following CLI command (via an SSH connection): `defcert<enter>`. Note there will be no response to this command. Then you will be able to reconnect to the browser.

As the CSR generated within the SecureSync is unique to just that one SecureSync (its paired with its internal private key) and as a replacement private key cannot be uploaded into the SecureSync, this same process described above needs to be performed with each SecureSync that you wish to replace its HTTPS certificate.

Please Note if desired to replace the Certificate on more than one SecureSync: If there is more than one SecureSync you wish to replace the HTTPS Certificate on, a different CR/CSR and a different certificate needs to be created for EACH SecureSync. A Certificate is only compatible with the single SecureSync in which the corresponding Certificate Request (CR/CSR) was generated in/exported from.

Due to each SecureSync having its own internal unique private key in which the Certificate (the public key) has to match to, a Certificate is only compatible with the one SecureSync in which the associated Certificate Request was generated on. A certificate for this one SecureSync will not be compatible with any other SecureSync. So, the same process to generate a new Certificate Request (CR) and Certificate on one SecureSync will need to be repeated for each SecureSync.

A new HTTPS Certificates can be created by the NTP server (referred to as a “self-signed” Certificate). Or, a Certificate Request can be generated and exported from the SecureSync in order for a trusted Certificate Authority (CA) of either an Internet company (such as “<http://www.verisign.com/>” or “<http://www.godaddy.com/>“, for two examples) or a local “Certificate Authority” to be able to generate a new Certificate, which can then be imported into the NTP time server. The first method of having the NTP server create its own “self-signed” Certificate is the more common method of generating a new Certificate, as it allows a Windows PC to be able to accept the Certificate.

A list of several trusted Certificate Authority organizations can be found at: www.sslshopper.com/certificate-authority-reviews.html. Note that Verisign was purchased by Symantec, which is also included in the list at this site.

Keep in mind that a network security scanner may report this configuration of a locally created “self signed” certificate as potential vulnerabilities. In this case, a new SSL Certificate may need to be purchased and installed into the time server, to alleviate these security scan reports.

For more information on generating/installing replacement HTTPS certificates, please refer to the online Model 1200 SecureSync user guide at:
http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_CertRequ.htm

Certificate Format types accepted/supported by the SecureSync

Note: For more info on supported Certificate formats, refer to the online Model 1200 SecureSync User guide:
https://safran-navigation-timing.com/manuals/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_cert_formats.htm

In summary, SecureSync supports:

- **X.509 PEM** and **X.509 DER formatted** Certificates.
- **PKCS#7 PEM** and **PKCS#7 DER** formatted Certificates.

X.509 PEM certificates (a file named **cert.pem**) are clear-text ASCII files, which its contents can either be copy/pasted into a field of the SecureSync's browser, or the entire certificate file can be uploaded (as desired) into the unit via its web browser.

All other certificate formats (having file names such as the ones listed below) need to instead have the Certificate file uploaded using the web browser (these files can't simply be copy/pasted into the browser, as can be done with just an X.509 PEM certificate).

cert.der (for an **x509 DER** certificate)

certpem.p7c (for a **PKCS7 PEM** certificate)

certder.p7c (for a **PKCS7 DER** certificate)

RFC-2818 Compliancy/Subject Alternate Names (UCC / SANS fields for Common Names)

Software update **versions 5.8.0** (May, 2018) and above, adds support for RFC compliancy/SANS fields for Common Names.

If RFC-2818 is a requirement and the current software version, as reported in the **Tools -> Upgrade/Backup** page of the web browser (third value down, under "**System Configuration**") is software versions 5.8.0 or above is required. If the current version is prior to version 5.8.0, the software must be updated to a more recent version to support this functionality.

To obtain software update information and to download the most recent Model 1200 SecureSync software upgrade bundle, please visit us at: <https://safran-navigation-timing.com/portal/public-downloads/latest-securesync-and-netclock-9400-update-files/>

Generate Certificate Request (Software versions 5.8.0 and above)

Note: Software update version 5.8.0 (and above) added support for UCC/SANS (Subject Alternate Names).

For more info on HTTPS Subject Alternate Names, please refer to the online SecureSync user guide at:
http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPSubjAltName.htm

To replace the default Spectracom Certificate Request/Certificate with a “self-signed” locally generated, or a purchased Certificate that is based on the desired IP address OR Host name login.

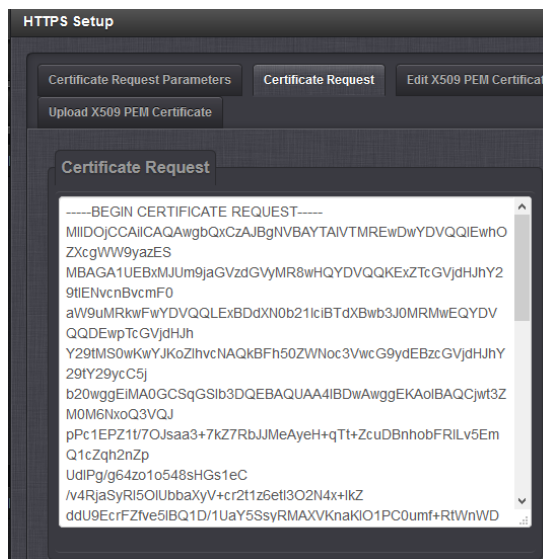
For additional info on creating/exporting a replacement Certificate Request (CR), refer to the online User Manual at:

http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_CertRequ.htm

In summary of this section, to either create a self-signed Certificate, or to create a new Certificate Request (CR) which can then be exported for the creation of a new certificate:

A) (Optional Step) It can be difficult to tell at a glance that a new CR/CSR has replaced the existing CR/CSR, once the new one has been created.

An Optional Step to be able to confirm a replacement Certificate Request/internal Certificate has been created at the end of this process: Go to the **Certificate Request** tab (**Management -> HTTPS Setup** page of the browser) to view the current Certificate Request. Either screenshot it, as shown below, or copy/paste the contents to a Word document for a side-by-side comparison of the new CSR.



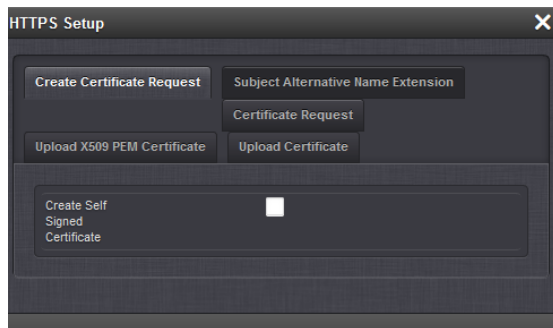
B) If required, add new Subject Alternate Names (for SAN/UCC certificates) (note this must be done before creating/generating a new Certificate Request)

Note: Refer also to the online SecureSync user guide at: https://safran-navigation-timing.com/manuals/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPSubjAltName.htm

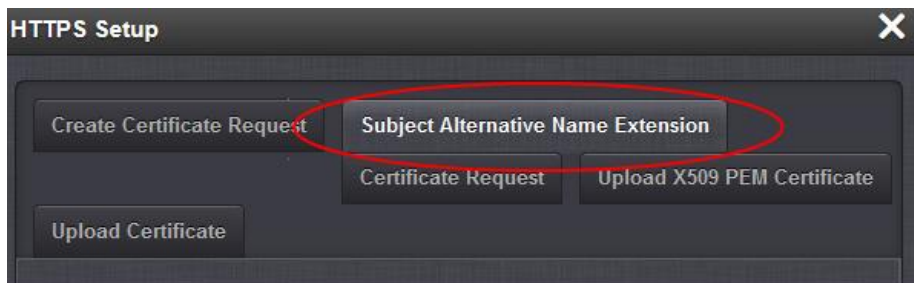
Any **additional** “Subject Alternate Names” (besides the name/address to be entered into the HTTPS certificate Request’s “Common Name field”, which will be automatically added to the “**Subject Alternate Names**” list once the new Certificate Request is generated at the end of this process) need to be added first.

Caution: Additional Subject Alternative Names must be added before a new Certificate Request is generated. Otherwise, the Certificate Request will have to be created again to include the Subject Alternative Names. Any information entered into the “Create Certificate Request” tab that has not been submitted will be lost by adding, deleting, or editing Subject Alternative Names.

1. Browse to the “**Management**” -> “**HTTPS Setup**” page of the web browser.

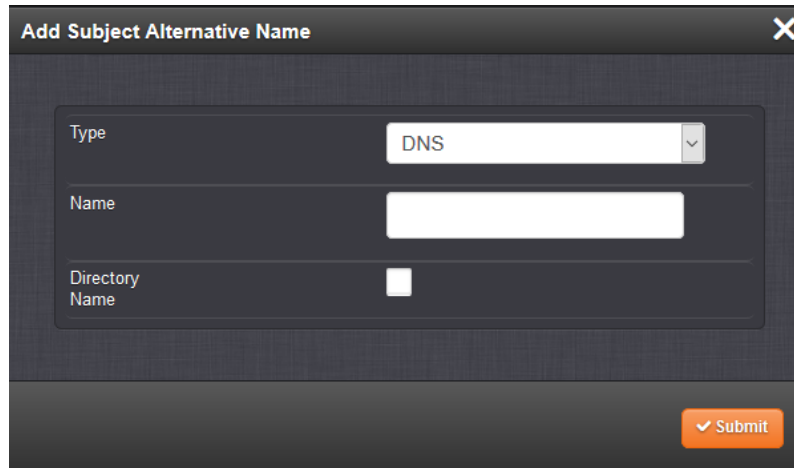


2. Select the “**Subject Alternative Name Extension**” tab



Repeat the following steps for each additional Subject Alternate Name (except the one name to be used in the “HTTPS certificate Request” form)

3. Select the plus (“+”) icon to access the “Add Subject Alternative Name” pop-up window



Fill in the available fields:

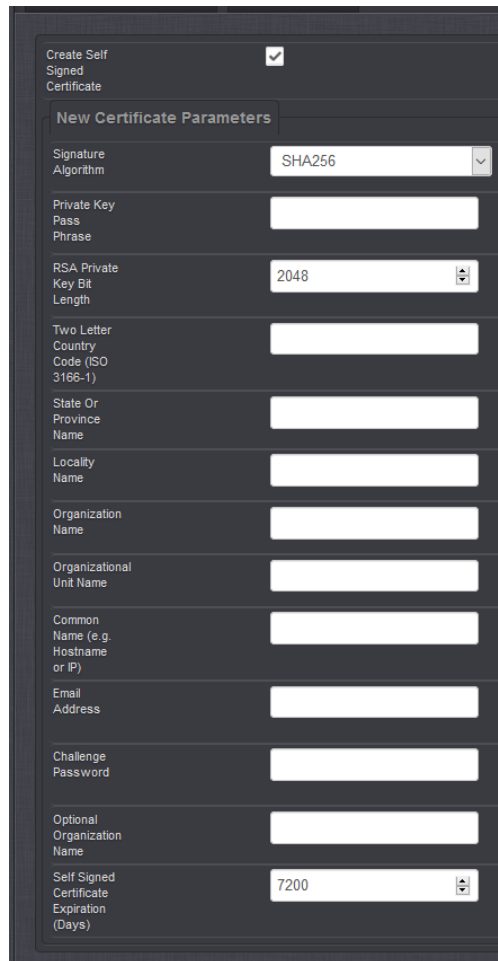
- **Type** [DNS, IP, email, URI, RID, dirName]
- **Name**
- **Directory Name** Checkbox: for Directory Subject Alternative Names (dirName), check the Directory. **Directory Name Fields** will now be displayed, and additional optional fields will be available:
 - **Two Letter Country Code**: must match ISO-3166-1 value.
 - **Organization name**: name of organization creating certificate.
 - **Organizational Unit Name**: The applicable subdivision of the organization creating the certificate.
 - **Common name**: The name of the host being authenticated. The Common Name field in the X.509 Certificate must match the host name, IP address, or URL used to reach the host via HTTPS.
- After completing and submitting the form, view the Subject Alternative Name tab to see existing entries. Existing Subject Alternative Names can be edited or deleted from this window.

C) Create/Generate a new Certificate Request

Note: Refer also to the online SecureSync user guide at:

http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_CertRequ.htm

1. Select “**Create Certificate Request**” button at the top of the page.
2. Select the “**Create Self Signed Certificate**” checkbox” (whether creating a self-signed Certificate or importing an externally generated Certificate).



Note: For more info on the various fields in this tab, see the list of fields further below.

3. Fill in each of the custom fields below the “Create Self Signed Certificate...” checkbox.

Note: Below is some additional information about these various fields in the tab:

Signature Algorithm: The default selected “Signature Algorithm” is “SHA1”. Other available encryption algorithms are available for selection, if desired (note that “MD5” is not considered to be as secure as SHA1”).

Private Key Pass Phrase: This field can normally be any value. Unless you are having a Certificate Authority (CA) create the Certificate for you (this is rare), this field is just used to help generate the new key and does not need to be “remembered” for later use.

RSA Private Key Bit Length: The “Private Key Bit Length” is a numerical value used to help determine the “strength” of the encryption of the Certificate. The larger the BIT Length number, the greater the security.

Selection of the bit length is based on the application of the equipment. RSA recommends using **1024** for corporate use or **2048** for extremely secure requirements (such as using a Certificate Authority to generate the Certificate). The available range is 512 to 4096. Also, the larger the Bit Length, the longer it takes for the new Certificate to be created.

Two Letter Country Code (ISO 3166-1): A field to enter a two-character value to indicate the country (such as “US” for the United States, “UK” for the United Kingdom, or “FR” for France).

State or Province Name: Enter a custom value.

Locality Name: Enter a custom value.

Organization Name: Enter a custom value.

Organizational Unit Name: Enter a custom value.

Note: As of at least software update versions 5.8.0 and below, the time server does not currently support **multiple Organizational Units (OU)**. However, we are considering this capability to potentially be added in a future software update.

Common Name (e.g. Hostname or IP): The “Common Name” field must contain either the IP address or the Host name that is entered into the address line of the browser. In this field, either enter the IP address of the NTP server if you normally type the IP address of the NTP server into the browser address field OR enter the Host name if you use DNS instead of the IP address to login to the browser (This field must match the value entered into the address field of the browser in order for the Certificate to be valid (If you type the IP address into the address field, enter the IP address of the unit into this field. If you enter a DNS name, enter the DNS name in the field). Then hit Submit.

Keep in mind that if you enter the IP address instead of the Host name, and the IP address is dynamic (assigned by a DHCP server), if the DHCP server changes the IP address of the NTP server, the Certificate will need to be re-created each time the address changes. It is recommended to change the IP address to a static address unless DHCP and DNS are linked together. In this case, enter into the field and login with the Host Name and not the IP address.

Email address: Enter a custom value.

Challenge Password: Can normally be any value. Unless you are having a “Certificate Authority” create the Certificate for you (this is rare), this field is just used to help generate the new key and does not need to be “remembered” for later use.

Optional Organization Name: Enter a custom value.

Self Signed Certificate Expiration (Days): Enter the desired number of days before the HTTPS Certificate expires (such as “365” for the Certificate to be valid for 1 year or “3650” for the Certificate to be valid for 10 years). Once the HTTPS Certificate expires, a new Certificate will need to be created, before HTTPS connection to the web browser will be available again.

Note: If the current Certificate expires before a new Certificate has been created, please refer to: [Need to reset a custom HTTPS Certificate back to the factory default Certificate](#) in order to access the browser again.

Once all the fields have been filled in with the appropriate data, press the Submit button at the bottom of this page to create a new Certificate Request (CR).

Important Notes:

- 1) It can take a few minutes for the new Certificate Request (CR/CSR) to be generated, after filling in all fields and pressing Submit. Be cautious not to export an existing Certificate Request before the new Certificate Request has been generated. The text field should change before exporting the Request.
- 2) Make sure web page caching is not enabled in your web browser.
- 3) **Copy/pasting the entire contents of the new Certificate Request (CR/CSR) from the “Certificate Request tab” into a new file named “cert.csr” is required. This file must be provided to the Certificate Authority (CA) and used for the development of the Certificate file. Wildcard SSL certificates are not supported.**

Depending on the value entered in the “RSA Private key bit length” field, within a few moments to a few minutes of hitting the Submit button, the new HTTPS Certificate Request will be created. When the new Certificate Request has been successfully created, you will likely be automatically exited out of the web browser connection and will have to log back into the unit (to accept the newly generated Certificate).

Also, the “scrambled” section of letters and symbols in the “**Certificate Request**” tab (the “**Certificate Request**” or “CR” field) will now be updated with different random numbers and characters. The replacement CR can be viewed or copy/pasted out of the SecureSync via the **Certificate Request** tab (**Management** -> **HTTPS Setup** page of the browser) as shown below:

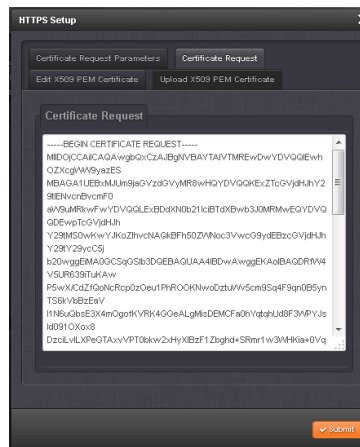


Figure 1: “Certificate Request”

Copy/paste the entire contents of the newly generated Certificate Request (CR/CSR) to a new text file named “**cert.csr**”. Provide this file to the Certificate Authority (CA) when letting them know which type certificate (such as either an “x509” or “PKCS7”, and such as either a “.pem” or “.der” certificate file) wish to receive.

Now refer to: [Import an external, locally generated, or purchased Certificate](#)

Generate Certificate Request (Software versions 5.1.2 through 5.7.1)

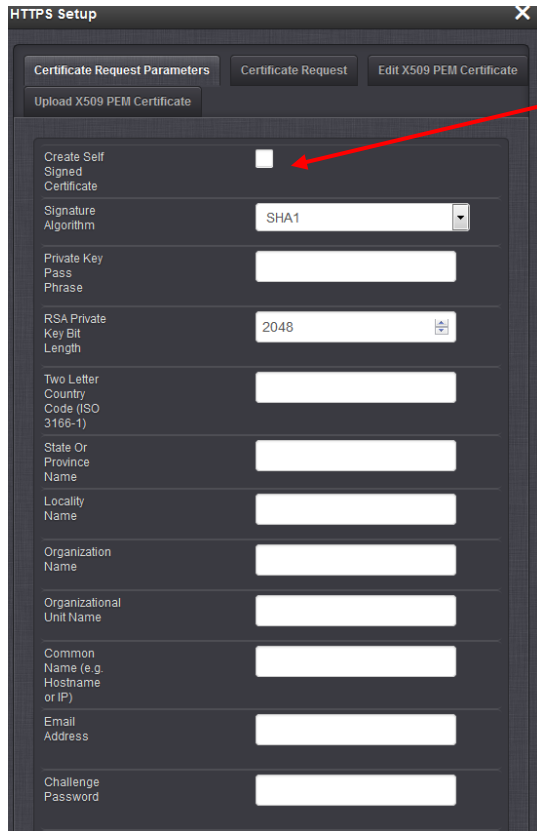
To replace the default Spectracom Certificate Request/Certificate with a “self-signed” locally generated, or a purchased Certificate that is based on the desired IP address OR Host name login.

Note: much earlier software versions 5.7.1 and below do not support SANS certificates (having more than one “Common Name”).

For additional info on creating/exporting a replacement Certificate Request (CR), refer to the online User Manual at: http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/HTTPS_CertRequ.htm

In summary of this section, to either create a self-signed Certificate, or to create a new Certificate Request (CR) which can then be exported for the creation of a new certificate:

1. Browse to the “**Management**” -> “**HTTPS Setup**” page of the web browser.
2. (Optional Step) It can be difficult to tell at a glance that a new CR/CSR has replaced the existing CR/CSR, once the new one has been created. An Optional Step to be able to confirm a replacement Certificate Request/internal Certificate has been created at the end of this process: Go to the **Certificate Request** tab (**Management** -> **HTTPS Setup** page of the browser) to view the current certificate. Either screenshot it, as shown below, or copy/paste the contents to a Word document for a side-by-side comparison of the new CSR.
3. Select the “**Certificate Request Parameters**” tab at the top of the page, as shown below.



The screenshot shows the 'Certificate Request Parameters' tab in the 'HTTPS Setup' window. At the top, there are three tabs: 'Certificate Request Parameters', 'Certificate Request', and 'Edit X509 PEM Certificate'. Below these is a section for 'Upload X509 PEM Certificate'. The main area contains several fields: 'Create Self Signed Certificate' (a checkbox, highlighted with a red arrow), 'Signature Algorithm' (a dropdown menu set to 'SHA1'), 'Private Key Pass Phrase' (a text field), 'RSA Private Key Bit Length' (a text field set to '2048'), 'Two Letter Country Code (ISO 3166-1)' (a text field), 'State Or Province Name' (a text field), 'Locality Name' (a text field), 'Organization Name' (a text field), 'Organizational Unit Name' (a text field), 'Common Name (e.g. Hostname or IP)' (a text field), 'Email Address' (a text field), and 'Challenge Password' (a text field).

Please Note (versions 5.6.0 and below): Make sure to always select the “**Create Self Signed Certificate**” checkbox at the top of the page (whether creating a new self-signed certificate, or importing a Certificate, before pressing Submit). Otherwise, the default Certificate Request (CR) will remain unchanged (an external Certificate will be created using an undesired CR).

Note: For more info on the various fields in this tab, see the list of fields further below.

Figure 2: “Certificate Request Parameters” tab

4. Select the “**Create Self Signed Certificate**” checkbox” (whether creating a self-signed Certificate or importing an externally generated Certificate). Otherwise, the default Certificate Request will remain unchanged- instead of being regenerated based on all the custom fields.
5. Fill in each of the custom fields below the “Create Self Signed Certificate...” checkbox.

Note: Below is some additional information about these various fields in the tab:

Signature Algorithm: The default selected “Signature Algorithm” is “SHA1”. Other available encryption algorithms are available for selection, if desired (note that “MD5” is not considered to be as secure as SHA1”).

Private Key Pass Phrase: This field can normally be any value. Unless you are having a Certificate Authority (CA) create the Certificate for you (this is rare), this field is just used to help generate the new key and does not need to be “remembered” for later use.

RSA Private Key Bit Length: The “Private Key Bit Length” is a numerical value used to help determine the “strength” of the encryption of the Certificate. The larger the BIT Length number, the greater the security. Selection of the bit length is based on the application of the equipment. RSA recommends using **1024** for corporate use or **2048** for extremely secure requirements (such as using a Certificate Authority to generate the Certificate). The available range is 512 to 4096. Also, the larger the Bit Length, the longer it takes for the new Certificate to be created.

Two Letter Country Code (ISO 3166-1): A field to enter a two-character value to indicate the country (such as “US” for the United States, “UK” for the United Kingdom, or “FR” for France).

State or Province Name: Enter a custom value.

Organization Name: Enter a custom value.

Organizational Unit Name: Enter a custom value.

Note: As of at least software update versions 5.7.1 and below, the time server does not currently support **multiple Organizational Units (OU)**. However, we are considering this capability to potentially be added in a future software update.

Common Name (e.g. Hostname or IP): The “Common Name” field must contain either the IP address or the Host name that is entered into the address line of the browser. In this field, either enter the IP address of the NTP server if you normally type the IP address of the NTP server into the browser address field OR enter the Host name if you use DNS instead of the IP address to login to the browser (This field must match the value entered into the address field of the browser in order for the Certificate to be valid (If you type the IP address into the address s field, enter the IP address of the unit into this field. If you enter a DNS name, enter the DNS name in the field). Then hit Submit.

Keep in mind that if you enter the IP address instead of the Host name, and the IP address is dynamic (assigned by a DHCP server), if the DHCP server changes the IP address of the NTP server, the Certificate will need to be re-created each time the address changes. It is recommended to change the IP address to a static address unless DHCP and DNS are linked together. In this case, enter into the field and login with the Host Name and not the IP address.

Note about RFC 2818: Subject Alternate Name (SAN) Certificates, (also referred to as “DNS SAN” Certificates or UCC certificates): As of at least software update versions 5.7.1 and below, the SecureSync does not currently support Subject Alternate Name (SAN) Certificates, or DNS SAN Certificates. However, this capability was added, starting in software version 5.8.0.

Email address: Enter a custom value.

Challenge Password: Can normally be any value. Unless you are having a “Certificate Authority” create the Certificate for you (this is rare), this field is just used to help generate the new key and does not need to be “remembered” for later use.

Optional Company Name: Enter a custom value.

Self Signed Certificate Expiration (Days): Enter the desired number of days before the HTTPS Certificate expires (such as “365” for the Certificate to be valid for 1 year or “3650” for the Certificate to be valid for 10 years). Once the HTTPS Certificate expires, a new Certificate will need to be created, before HTTPS connection to the web browser will be available again.

Note: If the current Certificate expires before a new Certificate has been created, please refer to: [Need to reset a custom HTTPS Certificate back to the factory default Certificate](#) in order to access the browser again.

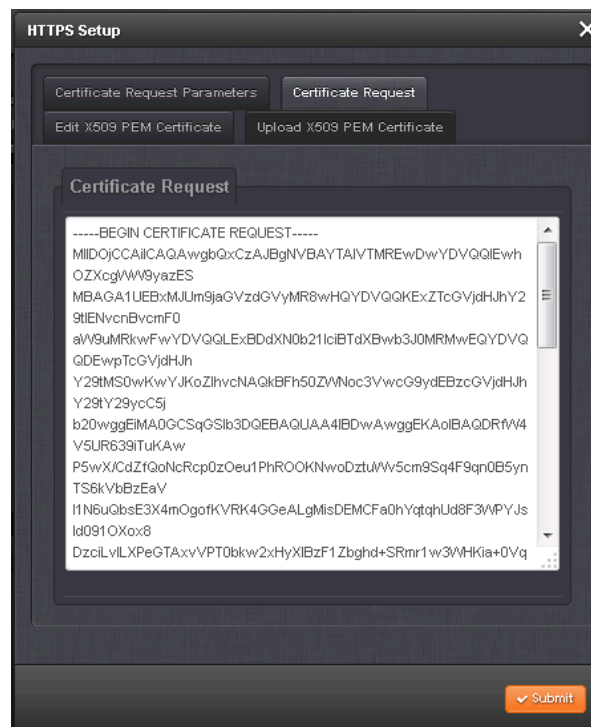
Once all the fields have been filled in with the appropriate data, press the Submit button at the bottom of this page to create a new Certificate Request (CR).

Important Notes:

- 1) It can take a few minutes for the new Certificate Request to be generated, after filling in all fields and pressing Submit. Be cautious not to export an existing Certificate Request before the new Certificate Request has been generated. The text field should change before exporting the Request.
- 2) Make sure web page caching is not enabled in your web browser.
- 3) Copy/pasting the entire contents of the new Certificate Request (CR/CSR) from the “Certificate Request tab” into a new file named “**cert.csr**” is required. This file must be provided to the Certificate Authority (CA) and used for the development of the Certificate file. Wildcard SSL certificates are not supported.

Depending on the value entered in the “RSA Private key bit length” field, within a few moments to a few minutes of hitting the Submit button, the new HTTPS Certificate Request will be created. When the new Certificate Request has been successfully created, you will likely be automatically exited out of the web browser and will have to log back into the unit (to accept the newly generated Certificate).

Also, the scrambled section of letters and symbols in the “**Certificate Request**” tab (the “**Certificate Request**” or “CR” field) will now be updated with random numbers and characters. The replacement CR can be viewed or copy/pasted out of the SecureSync via the **Certificate Request** tab (**Management** -> **HTTPS Setup** page of the browser) as shown below:



Copy/paste the entire contents of the newly generated Certificate Request (CR/CSR) to a new text file named “**cert.csr**”. Provide this file to the Certificate Authority (CA) when letting them know which type certificate (such as either an x509 or PKCS7, and such as either a .pem or .der certificate file)

Now refer to: [Import an external, locally generated, or purchased Certificate](#)

Import an external, locally generated, or purchased Certificate (all software versions 5.1.2 and higher)

A) To import an x509 PEM certificate

1) X509 PEM (plain text) certs can be simply copy/pasted into the web browser

- Refer to the online SecureSync User Manual:
http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/UploadX509Cert.htm

D) For importing Certificate formats other than PEM (such as DER and PKCS), or to upload a PEM certificate (instead of simply copy/pasting the contents of the PEM certificate)

- Unless it is a X.509 PEM-format Certificate, once the HTTPS Certificate has been issued by your Certificate Authority, you need to **upload** the Certificate file into the SecureSync.
- Refer to the online SecureSync User Manual:
http://manuals.spectracom.com/SS/Content/NC_and_SS/Com/Topics/SETUP/LoadNonX509cert.htm

Once the new Certificate Request (CR) has been created (as previously described), the new Certificate that was just created internally can now start to be used. Or instead, the newly created Certificate Request (CR) can now be exported from the SecureSync, for a local or Internet “Certificate Authority” to be able to generate the new Certificate for you (from the CR that was exported) Then, the new Certificate can then be imported into the SecureSync for use by the web server.

A list of several trusted Certificate Authority organizations can be found at: www.sslshopper.com/certificate-authority-reviews.html. Note that Verisign was purchased by Symantec, which is also included in the list at this site.

Note: With SSL installed in a Windows PC, a locally-created self-signed certificate can be created (keep in mind that security / vulnerability scanners may report self-signed certificates as a potential vulnerability).

For more information about **being your own Certificate Authority (CA)** to create a local self-signed certificate, refer to websites such as <http://www.g-loaded.eu/2005/11/10/be-your-own-ca/> which discuss creating a self-signed certificate for Apache.

Note: Per the link directly above, the openssl package will need to be installed in the machine being used to create/manage the Certificates.

To replace the default “self signed” Spectracom Certificate with either a locally “CA-signed” or a purchased SSL Certificate, once the new Certificate Request has been generated, perform the following to generate the new, appropriate HTTPS Certificate:

Once the Local Certificate Authority has generated the new Certificate (in x509 PEM, x509DER, PKCS7PEM or PKCS7DER format), or once a new SSL Certificate has been purchased/received, this new Certificate needs to be placed into the SecureSync, using either of two different methods, described below:

1) Import the PEM Certificate via the web browser (copy/paste the new certificate into a field)

Note: Applicable to **X509 PEM** certificates only. Refer to the next step further below for all other certificate formats.

- A. Select the “**Upload X509 PEM Certificate**” tab. Paste the entire text contents of the new x509 Certificate into the “**Update X509 PEM Certificate File**” field in this tab. Then press Submit.

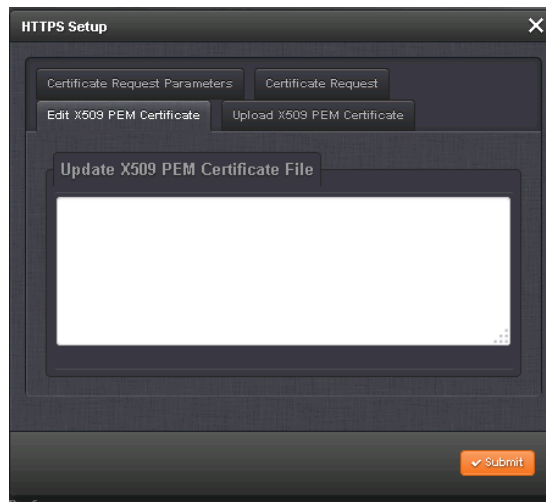


Figure 3: “X509 Certificate Upload”

- B. This will immediately “exit” the current HTTPS connection. Close out of the browser and re-open it to establish a new HTTPS connection using the new certificate.

2) Transfer the Certificate File into the SecureSync, using the web browser over a network connection

Note: applicable to x509 PEM, x509DER, PKCS7PEM or PKCS7DER certificates.

- A. Navigate to the **Management -> HTTPS Setup** page of the browser and select the “**Upload Certificate**” tab.
- B. In the “**Certificate Type**” drop-down box, select the format type of the Certificate being transferred into the SecureSync (such as PEM, DER, PKCS7 PEM, or PKCS7 DER)

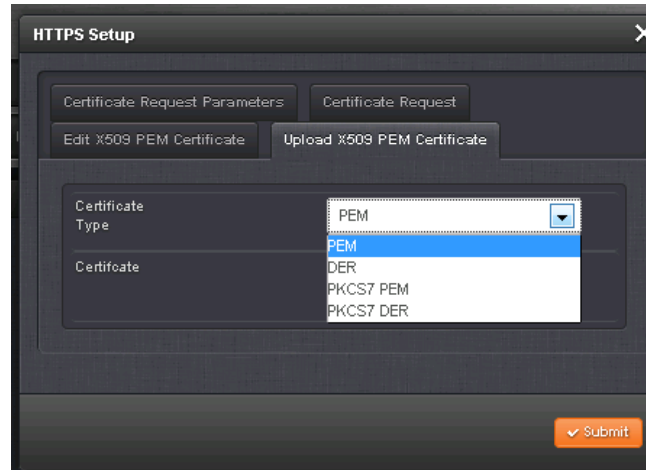


Figure 4: “PEM, DER or PKCS7 Certificate Upload”

- C. In the “**Certificate**” field, “browse” to the location of where the Certificate is located on the PC. Then press Submit.
- D. This will immediately “exit” the current connection. Close out of the browser and re-open it to establish a new connection, using the new certificate.

If the SecureSync’s browser is inaccessible after installing a replacement certificate

If the web browser is not accessible via HTTPS after copy/pasting or uploading a new certificate, it’s highly likely that the replacement certificate just installed wasn’t created using the CR/CSR from that particular SecureSync.

A replacement certificate (which was not created using the CR/CSR copy/pasted out of that same SecureSync is placed into that, or any other SecureSync) the certificate/login will fail when trying to login to the SecureSync. The installed certificate will need to be deleted/removed from the SecureSync, to restore connection to the browser. This can be performed via the following CLI command (via an SSH connection): `defcert<enter>`. Note there will be no response to this command. Then you will be able to reconnect to the browser.

For more info on performing the defcert CLI command, refer to (in this same document): [“Need to reset a custom HTTPS Certificate back to the factory default Certificate \(defcert or defcert -sha1 CLI commands\)”](#)

Viewing/Installing an HTTPS Certificate with Internet Explorer

Until you install the Certificate on the PC, each time you log into the web browser using Internet Explorer, a “Security Alert” window request window (like the one shown below) will open-up. You can either choose to proceed by just accepting the Certificate each time, or you can install the Certificate. Installing a valid Certificate will remove the need for this window to be displayed each time to log into the web browser.



Figure 5: “Security Alert” window

To verify the HTTPS Certificate, open Internet Explorer and enter the IP address of the NTP Time Server (just like you would normally do to open the web browser). Internet Explorer should open a “Security Alert” window, like the one shown below (If you don’t see the Security Alert, try logging in from a different PC). In this window, click on “View Certificate”



Figure 6: “Security Alert” window

A “Certificate” window will now be displayed. Click on the “Details” tab. The information about the Certificate will now be displayed (such as the Certificate issuer, when the Certificate expires, the encryption algorithm that was used to generate the Certificate, etc). The “Signature Algorithm” displays the encryption Algorithm that was used to create this Certificate.

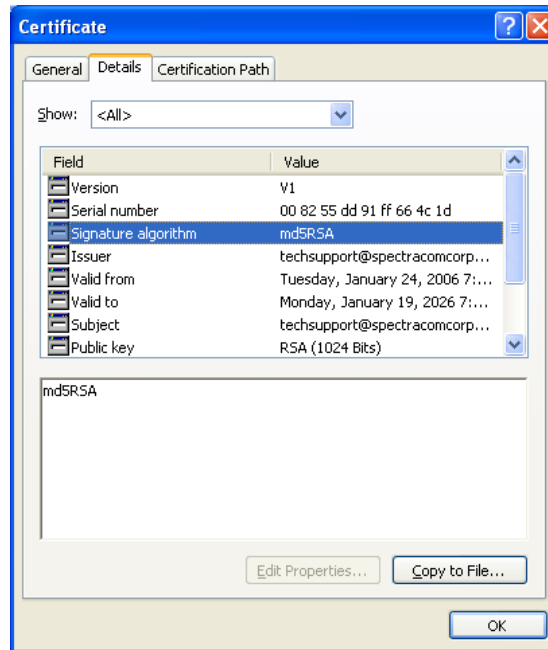


Figure 7: “Details” tab

The following list describes the values for the Certificate (as shown in Certificate window):

Serial Number: Used to uniquely identify the Certificate.

Subject: The person or entity identified.

Signature Algorithm: The algorithm used to create the signature.

Issuer: The entity that verified the information and issued the Certificate.

Valid-From: The date the Certificate is first valid from.

Valid-To: The expiration date.

Key-Usage: Purpose of the public key (e.g. encipherment, signature, Certificate signing...).

Public Key: The public key to encrypt a message to the named subject or to verify a signature from the named subject.

Thumbprint Algorithm: The algorithm used to [hash](#) the Certificate.

Thumbprint: The hash itself to ensure that the Certificate has not been tampered with.

To install the SSL Certificate, click on View Certificate. Then, a window will open that allows to you install the Certificate. In the “General” tab, click on “Install Certificate”. Keep clicking “next” until the “import was successful”.

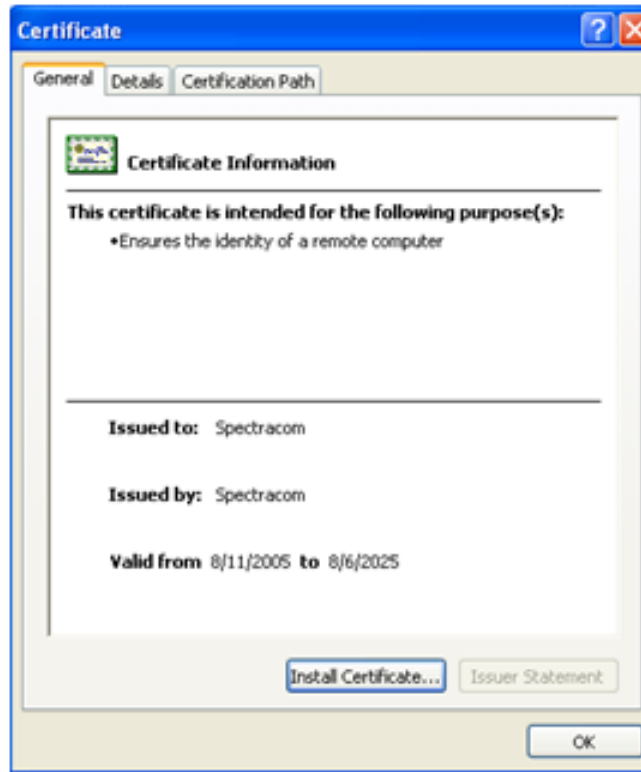


Figure 8: “General” tab

To view the Certificates currently installed on this PC, go to Tools/Internet Options and then select the “Content” tab. Next, click on “Certificates” to view all currently installed Certificates.

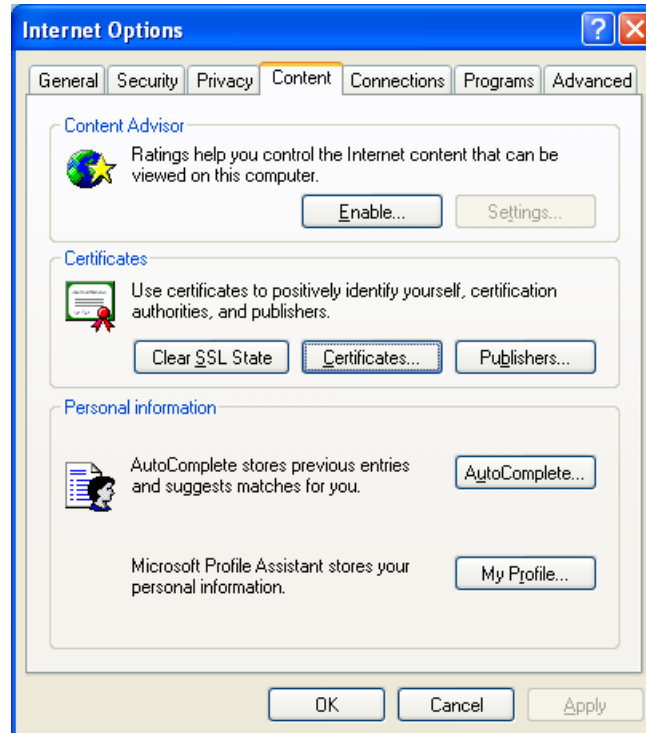


Figure 9: “Content” tab

Viewing/Installing an HTTPS Certificate with Mozilla FireFox

Until you install the Certificate on each PC, each time that you log into the time server using Mozilla FireFox, a “Security Alert” window request window (like the one shown below) will open-up. You can either choose to proceed by just accepting the Certificate, or you can install the Certificate. Installing a valid Certificate will remove the need for this window to be displayed each time to log into the web browser.

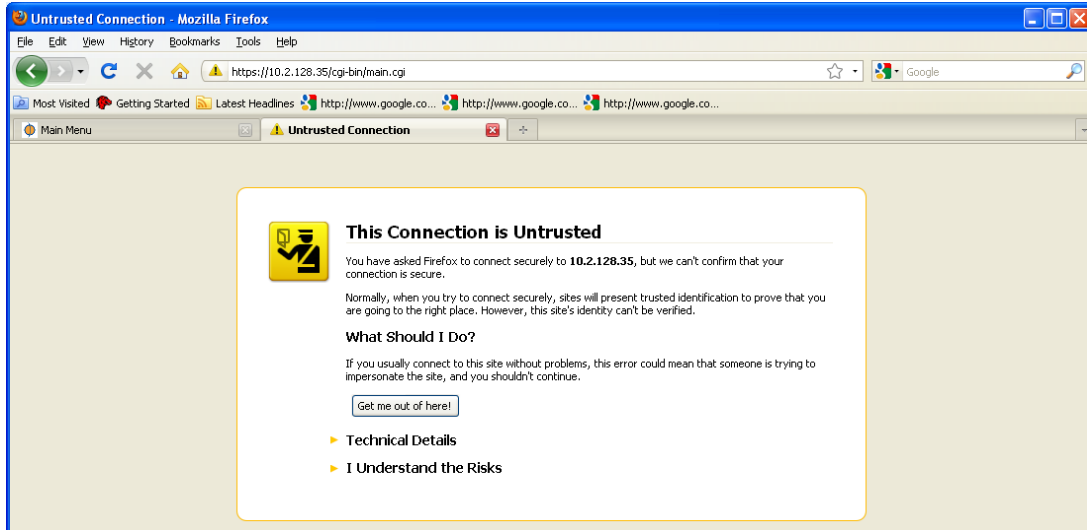


Figure 10: “Windows Security” pop-up

Click on “I Understand the Risks” and then click on “Add Exception”.

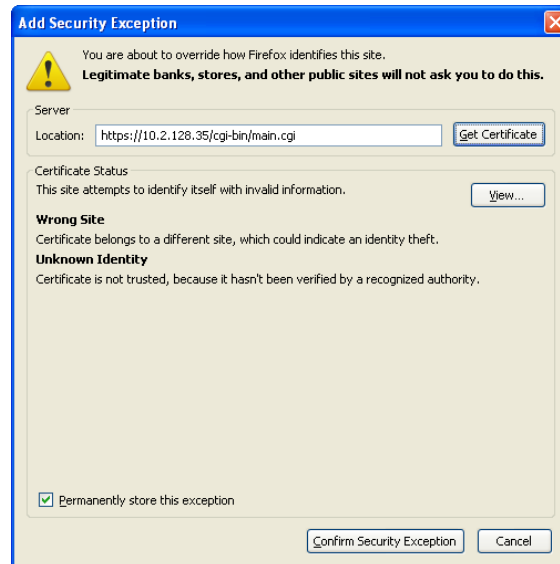


Figure 11: “Adding Security Exception” window

To verify the HTTPS Certificate, click “View”. A “Certificate Viewer” window will now be displayed. The “General” tab will contain the information about the Certificate (such as the Certificate issuer, when the Certificate expires, the encryption algorithm that was used to generate the Certificate, etc).

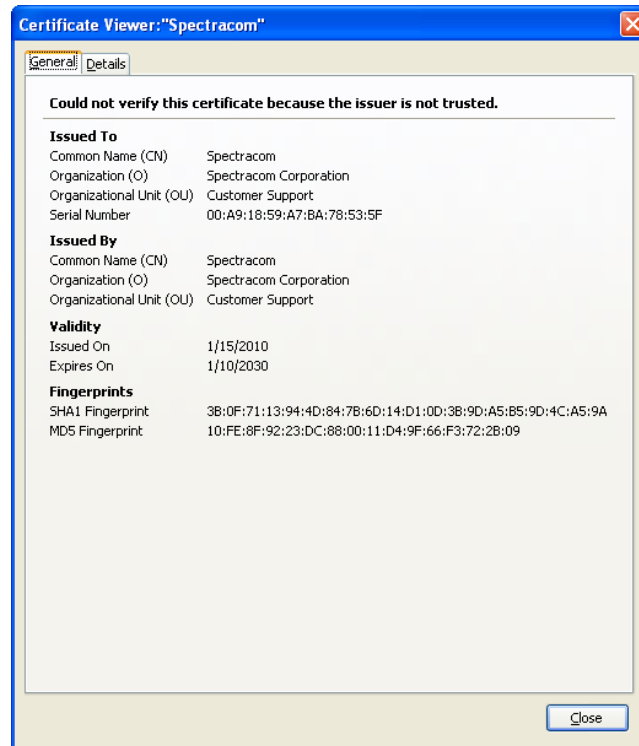


Figure 12: “General” tab

To find additional details about the Certificate, click on the “Details” tab. In the Certificate Fields section of this window, click on the desired field that you wish to obtain additional information about, and the “Field Value” section of the form will provide additional information about that field. The example below displays additional information about the encryption algorithm used to generate the Certificate:

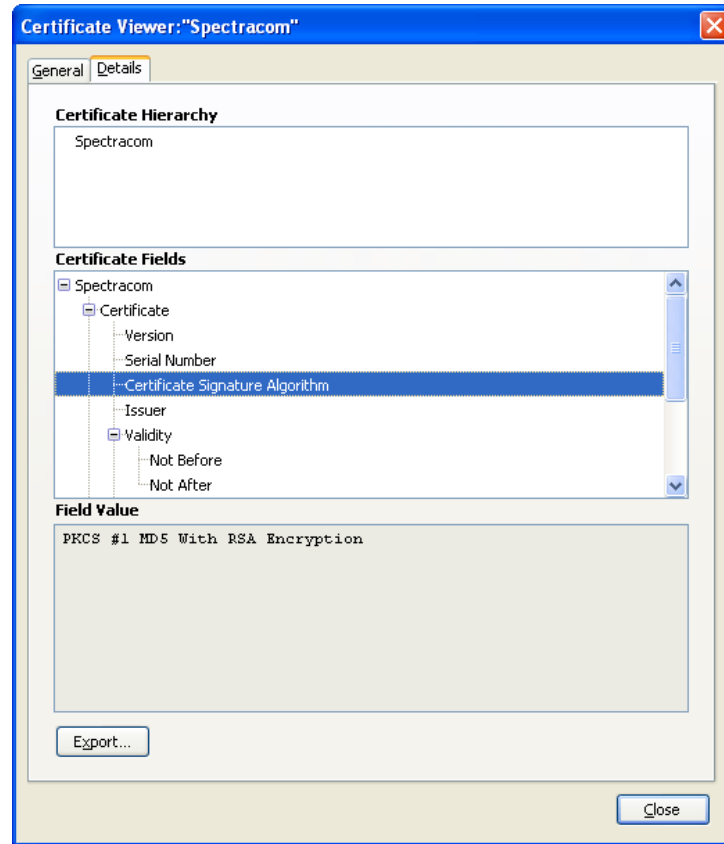


Figure 13: “Details” tab

The following list describes the values for the Certificate:

Version:

Serial Number: Used to uniquely identify the Certificate.

Certificate Signature Algorithm: The algorithm used to create the signature.

Issuer: The entity that verified the information and issued the Certificate (such as Spectracom)

Validity: The dates the Certificate is valid between.

Subject: The person, or entity identified.

Key-Usage: Purpose of the public key (e.g. encipherment, signature, Certificate signing...).

Subject Public Key info: The public key to encrypt a message to the named subject or to verify a signature from the named subject.

Certificate Signature Algorithm: The algorithm used to [hash](#) the Certificate.

Certificate Signature Value: The hash itself to ensure that the Certificate has not been tampered with.

To install the SSL Certificate on this PC, click on “Confirm Security exception: Then, the web browser login window will be displayed, allowing you to login to the browser if you wish. The Certificates that are currently installed on this PC can be displayed in Tools/Options. In the “Encryption” tab, click on “View Certificates”.

Need to reset a custom HTTPS Certificate back to the factory default Certificate (defcert or defcert -sha1 CLI commands)

It may be necessary at some point to have to reset a custom HTTPS Certificate back to the factory default HTTPS Certificate. Examples of this scenario is if a custom HTTPS Certificate is corrupted or has since expired, preventing you from being able to access the NTP server using HTTPS (if HTTP has also been disabled in the NTP server for security reasons, you won't be able to access the NTP server's web browser, without first resetting the HTTPS Certificate).

The NTP server provides the ability to reset the HTTPS Certificate without having to restore the entire unit back to the factory default settings. This Certificate reset can be performed via the CLI interface (using either a telnet or ssh session over an ethernet connection, or via an RS-232 direct connection to the front panel (this method sends the CLI command into the NTP server from a PC running a terminal emulator program).

To reset the current certificate using the front panel RS-232 Serial port (instead of using a telnet/ssh session over an ethernet connection) a PC running HyperTerminal (or other terminal emulator program) should be connected to the NTP server using a standard DB9M to DB9F serial cable that is pinned straight-thru (at least pin 2 to 2, pin 3 to 3 and pin 5 to 5 for the minimum configuration). If a serial com port is not available on your PC, a standard USB to Serial adapter will be needed and its driver installed.

Note: For additional information on the HyperTerminal connection (including port settings) please refer to the “Using HyperTerminal” PDF document on our website:
<http://www.spectracomcorp.com/Support/HowCanWeHelpYou/Library/tabid/59/Default.aspx?EntryId=315&Page=2>.

The CLI command (using either an ethernet or serial connection to the NTP server) to reset the current certificate is **defcert**. Depending on the software version currently installed in the time server, the **defcert** CLI command will create its new default certificate using either a **SHA1** encryption cipher, or a **SHA256** encryption cipher.

With software **versions 5.6.0 and above** installed, the more secure SHA256 cipher will be used to generate its new certificate (with software version 5.5.1 and below installed, the defcert command will instead use the SHA1 cipher). If you aren't sure what version of software is currently installed, the version can be obtained using the following CLI command: **version**

The issue with the SHA256 cipher used by default in software versions 5.6.0 and above is **Internet Explorer versions 8** and below doesn't support this cipher. With software versions 5.6.0 and above installed, and when using IE8 or below to access the NTP server, instead of using the CLI command of **defcert** to replace the existing certificate, issue instead the following CLI command to generate a new certificate using the SHA1 cipher: **defcert -sha** <enter>.

The **defcert** (or **defcert -sha1**) CLI command should then reset the HTTPS Certificate back to a factory default certificate (there should be no need to reboot/power cycle after issuing the command). Login to the browser and you will then be prompted to use the new Certificate. After accepting the default HTTPS Certificate, you can then create a new HTTPS Certificate as desired, using the information provided in this document.

Safran Technical Support

<https://safran-navigation-timing.com/>

Please contact one of the global Safran Electronics and Defense (SED) Technical Support centers for assistance:

USA Safran Trusted 4D, Inc

Timingsupport@safran-navigation-timing.com

45 Becker Rd Suite A W. Henrietta, NY 14623 (585) 321-5800

FRANCE Safran Trusted 4D SAS

Timingsupport@safran-navigation-timing.com

3 Avenue du Canada | 91974 Les Ulis, Cedex | +33 (0)1 64 53 39 80

Rev P, 18 May, 2023