

SOAP Procedure

Using PROC SOAP with Transport Layer Security (TLS)

TLS and Data Encryption

Transport Layer Security (TLS) enables web browsers and web servers to communicate over a secured connection by encrypting data. Both browsers and servers encrypt data before the data is transmitted. The receiving browser or server then decrypts the data before it is processed. Because PROC SOAP invokes a web service using the Java Native Interface, JREOPTIONS need to be specified either on the command line or in a configuration file to configure a TLS connection.

Note: All discussion of TLS is also applicable to the predecessor protocol, Secure Sockets Layer (SSL).

When you require client authentication, TLS has a renegotiation feature that prevents unauthorized text from being added to the beginning or end of an encrypted data stream. This feature is specifically used by certificate-based client authentication. This feature disables TLS renegotiation in the Java Secure Sockets Extension (JSSE) by default. As a result, when you attempt to access a web resource that requires certificate-based client authentication through the interception proxy, the following Java TLS error message is generated:

(javax.net.ssl.SSLException): HelloRequest followed by
an unexpected handshake message

However, it is still possible to enable the TLS renegotiation in Java by setting the following system property to true before the JSSE library is initialized.



```
sun.security.ssl.allowUnsafeRenegotiation
```

Making PROC SOAP Calls By Using the HTTPS Protocol

In order to make PROC SOAP calls using the HTTPS protocol, you must have a truststore that contains the certificates for the services that you trust. This truststore must be provided to the SAS session by setting Java system option Djavax.net.ssl.trustStore.


Clients must ensure that the CA that signed the certificate has been added to their truststore. You can provide the path to the truststore on the SAS command line or in a SAS configuration file using JREOPTIONS.

 `-jreoptions (-Djavax.net.ssl.trustStore=full-path-to-the-trust-store)`

Note: As a best practice, anytime that you are using passwords in configuration files, use file system permissions that allow write only access by the owner of the SAS or SAS Viya deployment. By default, the owner is "sas".


Here is an example using the SAS command line. The example uses the Windows operating environment.

Note: Add the following entry on one line.

 `"C:\Program Files\SAS\SASFoundation\9.4\sas.exe" -CONFIG "C:\Program Files\SAS\SASFoundation\9.4\nls\en\SASV9.CFG" -jreoptions (-Djavax.net.ssl.trustStore=C:\Documents and Settings\mydir\.keystore)`

Here is an example of how to specify the JREOPTIONS in the sasv9.cfg file.

Note: Add the system option and value on one line.

 `-JREOPTIONS (-Djavax.net.ssl.trustStore=
!SASHOME/./config/etc/SASSecurityCertificateFramework/
cacerts/trustedcerts.jks)`