12 SSL Client Certificate Authentication

12.1 Description

This tutorial demonstrates combining SSL client certificate authentication using a Layer 7 Federated Identity Provider with existing HTTP Basic authentication. The FIP acts as a sophisticated certificate trust store that can map users and groups to specific client certificates or virtual groups to signing certificates, and that can perform certificate validation based on a direct match or a trust chain of signing certificates to a trust anchor. Users can configure multiple FIPs for partitioning of different categories of certificates (e.g. partners, customers, etc.).

12.2 Prerequisites

12.2.1 Environment

1. Layer 7 SecureSpan Gateway *(this tutorial was designed using a version 7.0 gateway; it may or may not work with earlier versions; it should work with later versions)*
2. Layer 7 Policy Manager *(this tutorial uses the Policy Manager software installation; the software installation version must match the gateway version; alternatively, users can use the Policy Manager browser-based version which always matches the gateway version that is connected to)*
3. soapUI *(this tutorial was designed using the free soapUI version 4.5.1; it may or may not work with other versions of soapUI; other clients can be used for this and other tutorials, but specific steps will not be provided for those other clients)*

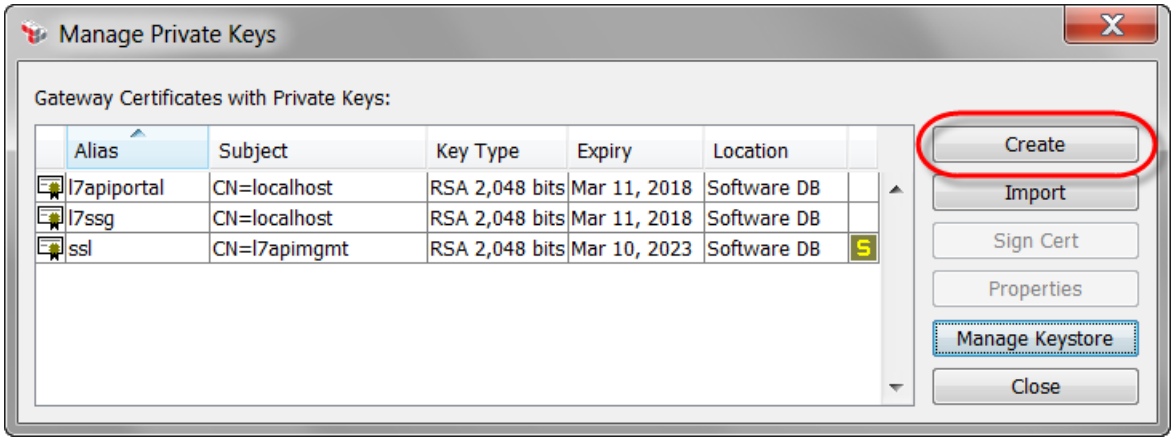
12.2.2 Tutorials

1. Layer 7 Tutorials - Getting Started
2. Tutorial 1 - Deploy Tutorial Services
3. Tutorial 3 - Test Tutorial REST Service
4. Tutorial 5 - Publish REST Service
5. Tutorial 6 - Basic Authentication

12.3 Tutorial Steps

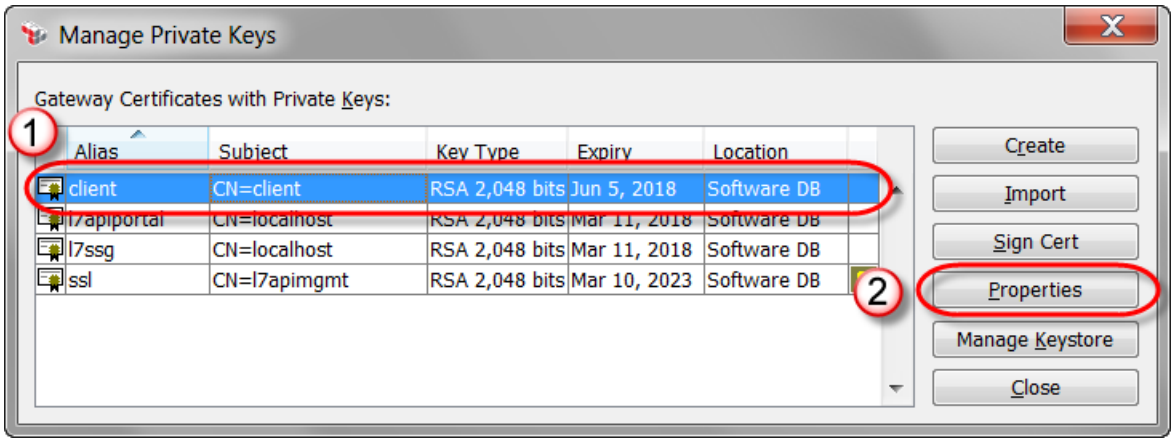
1. Connect to your gateway using Policy Manager (see tutorial ***Layer 7 Tutorials - Getting Started***).
2. Per ***Layer 7 Tutorials - Getting Started/Basic Policy Concepts/Policy Authoring/Policy Revisions***, set the active policy version of the ***Warehouse REST Tutorials*** service to the version that has been commented with, **Tutorial 6 Complete**.
3. This tutorial requires a client key and certificate. You can use a client key and certificate that you already have, one provided together with this tutorial (in the file ***client.p12*** in the ***Tutorial 12 - SSL Client Certificate Authentication*** subfolder of the tutorials package folder), or you can perform the following steps to create a new client key and certificate. If you select one of the first two options, then skip to step 17.
4. In Policy Manager, select the ***Tasks/Manage Private Keys*** menu item.

5. In the Manage Private Keys dialog, click the ***Create*** button.

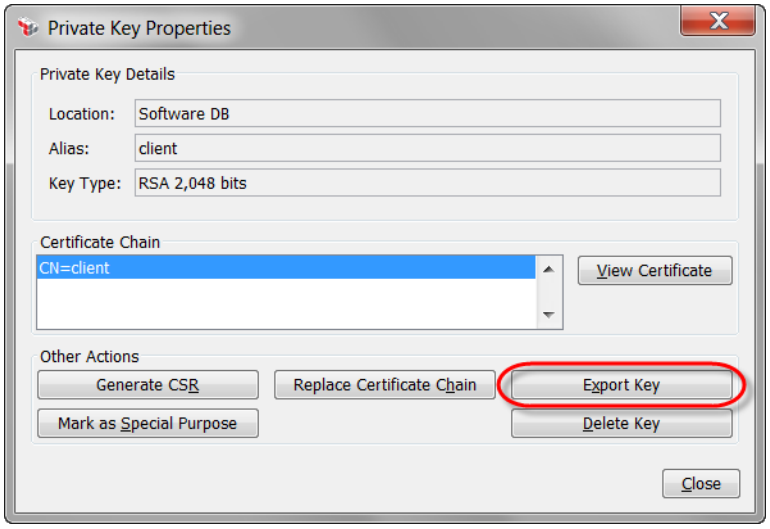


6. In the Create Private Key dialog, in the Alias field, enter ***client***, and click the ***OK*** button.

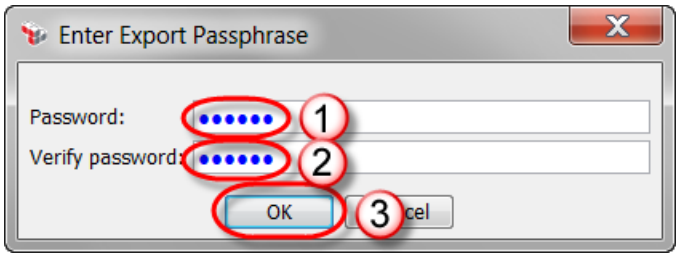
7. In the Manage Private Keys dialog, select the new ***client*** key, and click the ***Properties*** button.



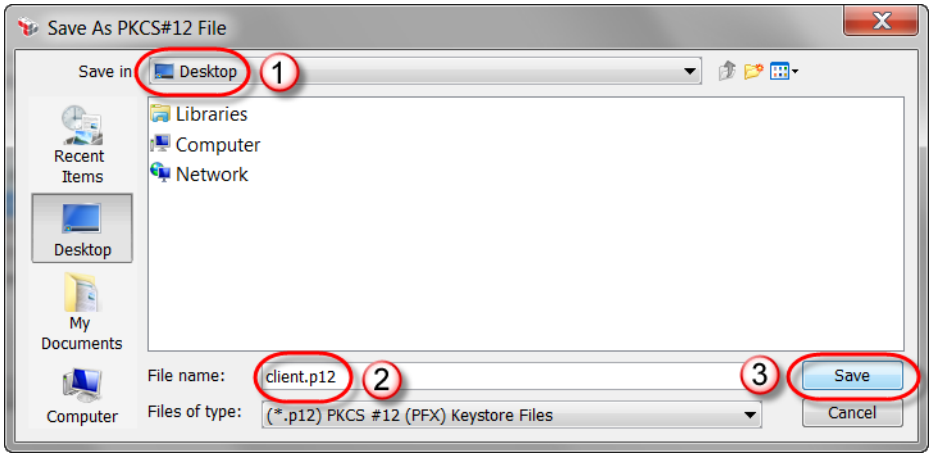
8. In the Private Key Properties dialog, click the ***Export*** button.



9. In the Enter Export Passphrase dialog, in both the Password and Verify Password field, enter ***7layer***, and click the ***OK*** button.

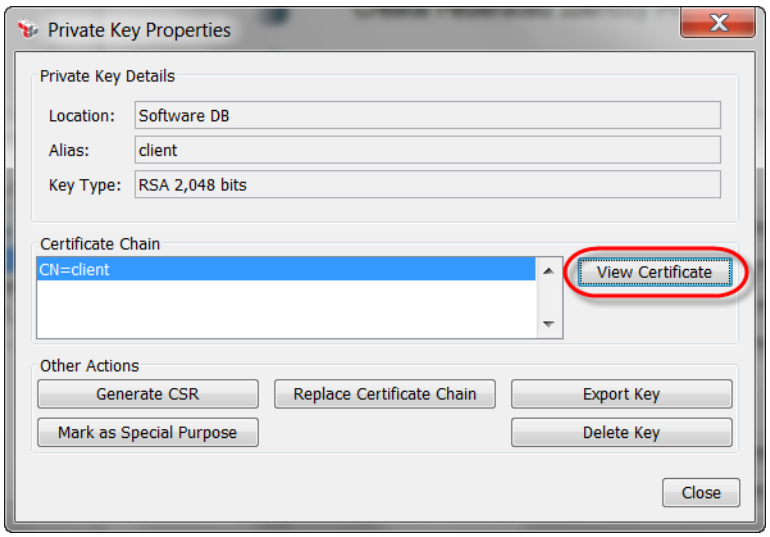


10. In the Save As PKCS#12 File dialog, select a location to save your key file (e.g. your Desktop), in the File Name field enter ***client.p12***, and click the ***Save*** button.

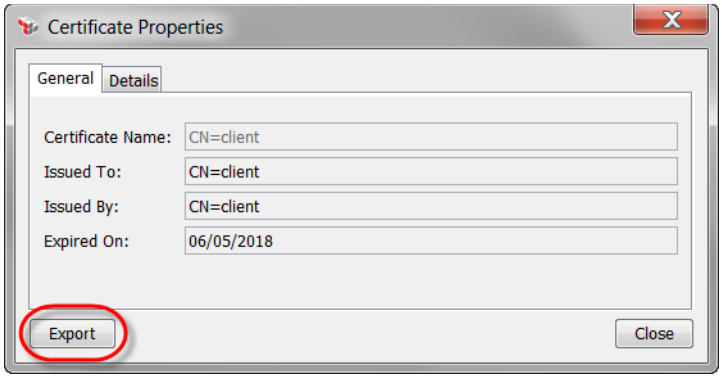


Note: This keystore file includes both the key and certificate. It will be used by soapUI during the client portion of its the mutual auth SSL handshake with the gateway.

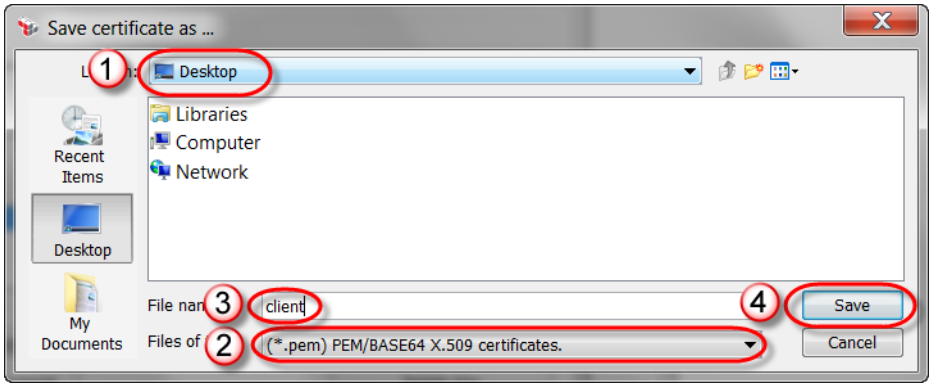
11. We will also export the client certificate separately. In the Private Key Properties dialog, click the ***View Certificate*** button.



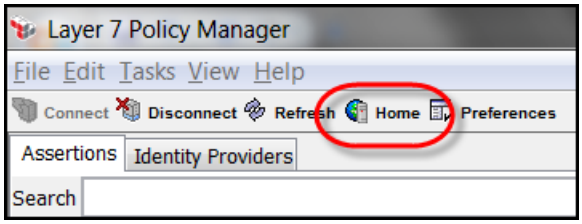
12. In the Certificate Properties dialog, click the ***Export*** button.



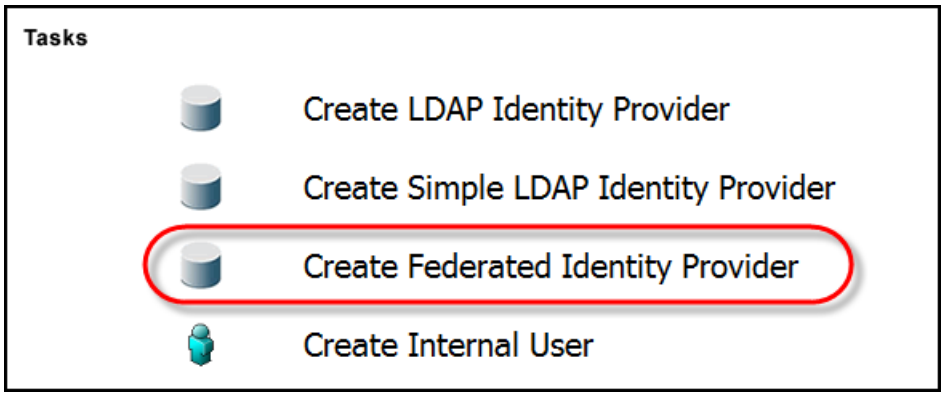
13. In the Save certificate as... dialog, select a location to save your key file (e.g. your Desktop), in the Files of Type field select ***(*.pem)...,*** in the File Name field enter ***client***, and click the ***Save*** button.



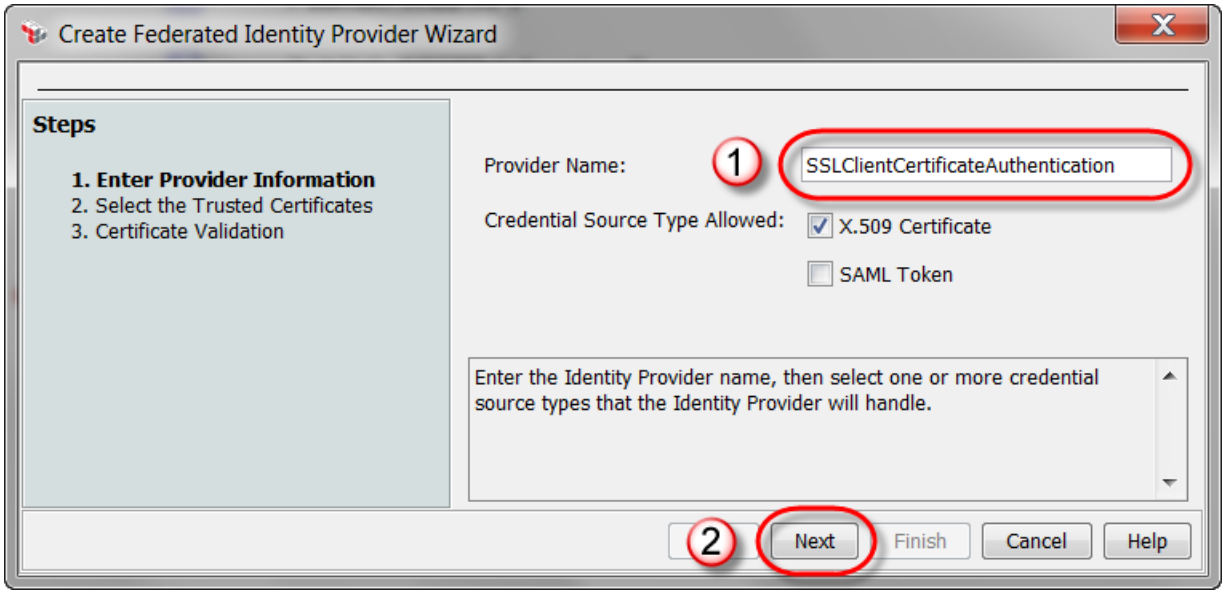
14. In the Certificate Properties dialog, click the ***Close*** button
15. In the Private Key Properties dialog, click the ***Close*** button.
16. In the Manage Private Keys dialog, click the ***Close*** button.
17. In Policy Manager, on the toolbar, click the ***Home*** button.



18. In the Policy Manager Home window, click the ***Create Federated Identity Provider*** button.



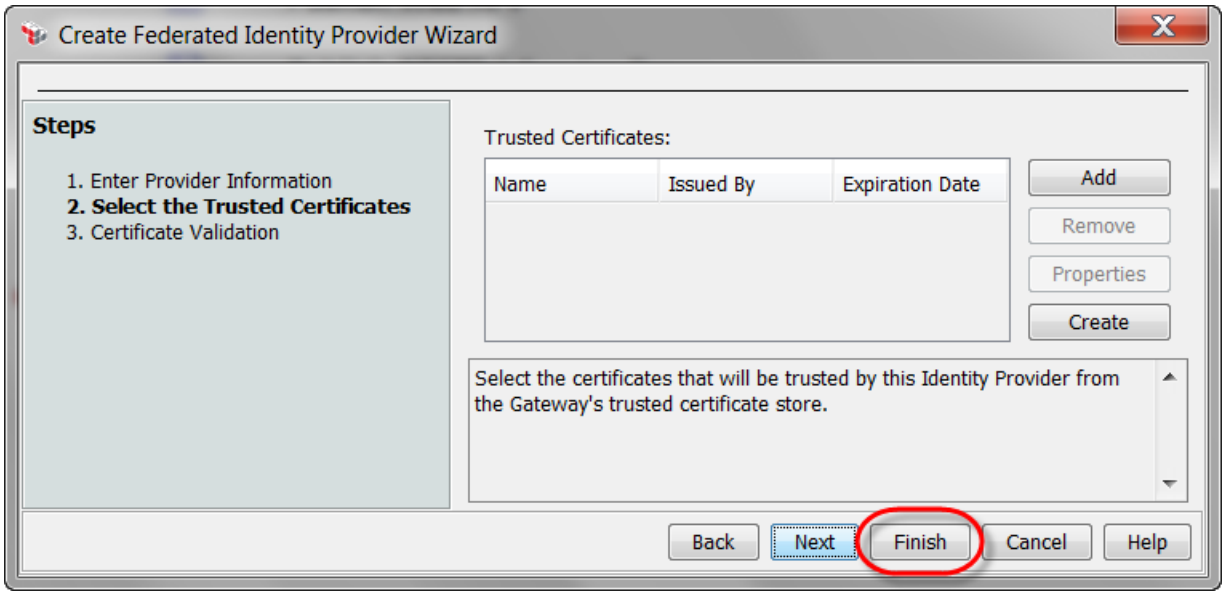
19. In the Create Federated Identity Provider Wizard dialog, on step 1, in the Provider Name field, enter ***SSLClientCertificateAuthentication***, and click the ***Next*** button.



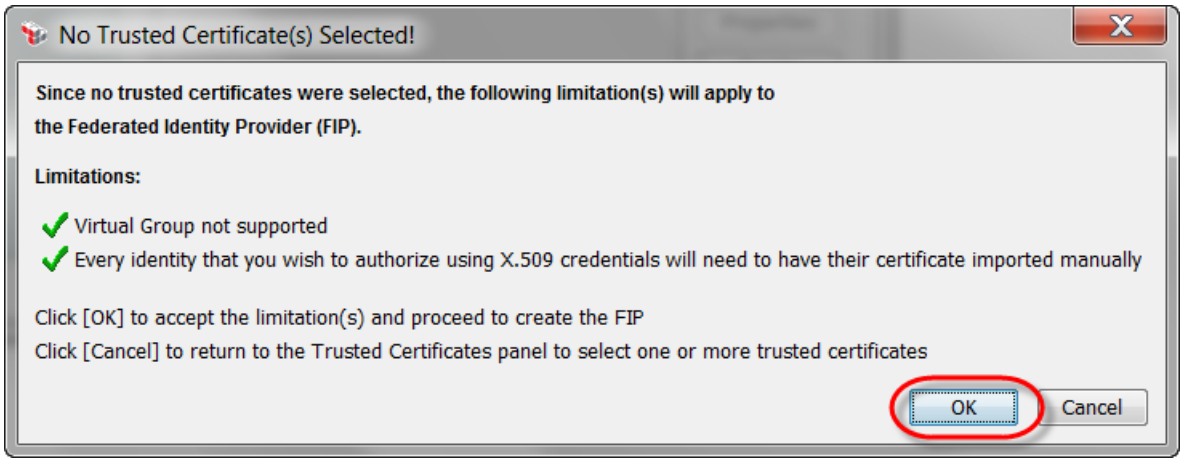
Note: FIPs can support certificate validation for x.509 and/or SAML workflows. FIPs are configured differently depending on which workflow you intend to use the FIP for, and whether

you're working with self-signed or signed certificates. Chapter 9 of the online Policy Manager help discusses these works flows and the corresponding FIP configuration in great detail. This tutorial is going to configure the FIP to support x.509 certificate validation of self-signed certificates in the most simple way possible.

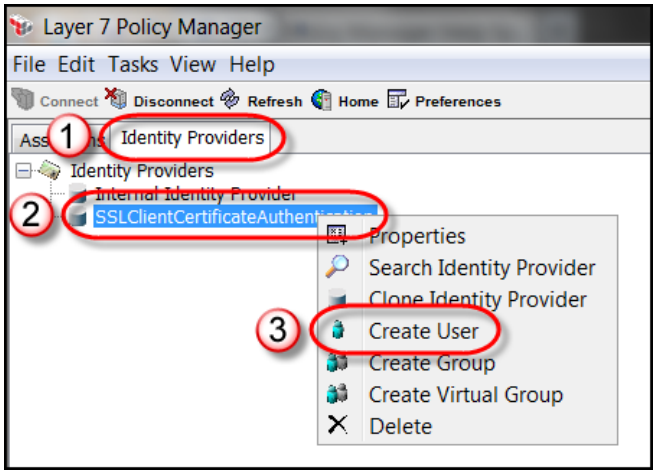
20. In the Create Federated Identity Provider Wizard dialog, on step 2, in the Provider Name field, click the ***Finish*** button.



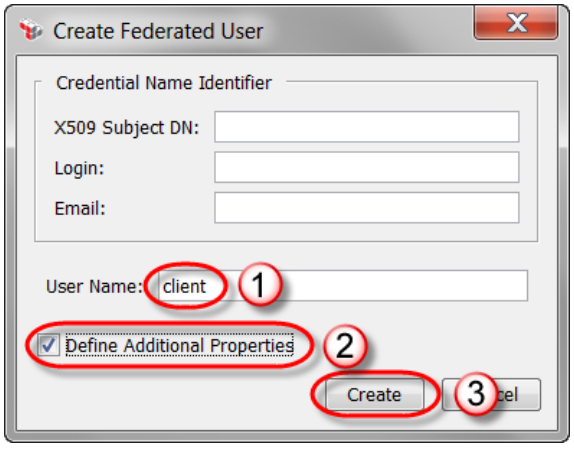
21. In the No Trusted Certificate(s) Selected! dialog, click the ***OK*** button.



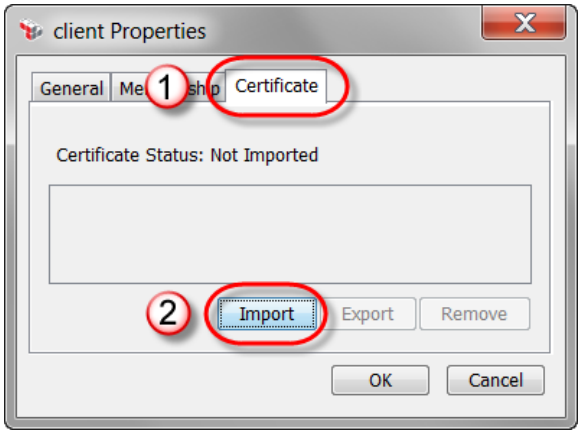
22. We will now add a user mapped to a trusted client certificate to the new FIP. Click the ***Identity Provider*** tab, right click the new ***SSLClientCertificateAuthentication*** FIP, and click the ***Create User*** context menu item.



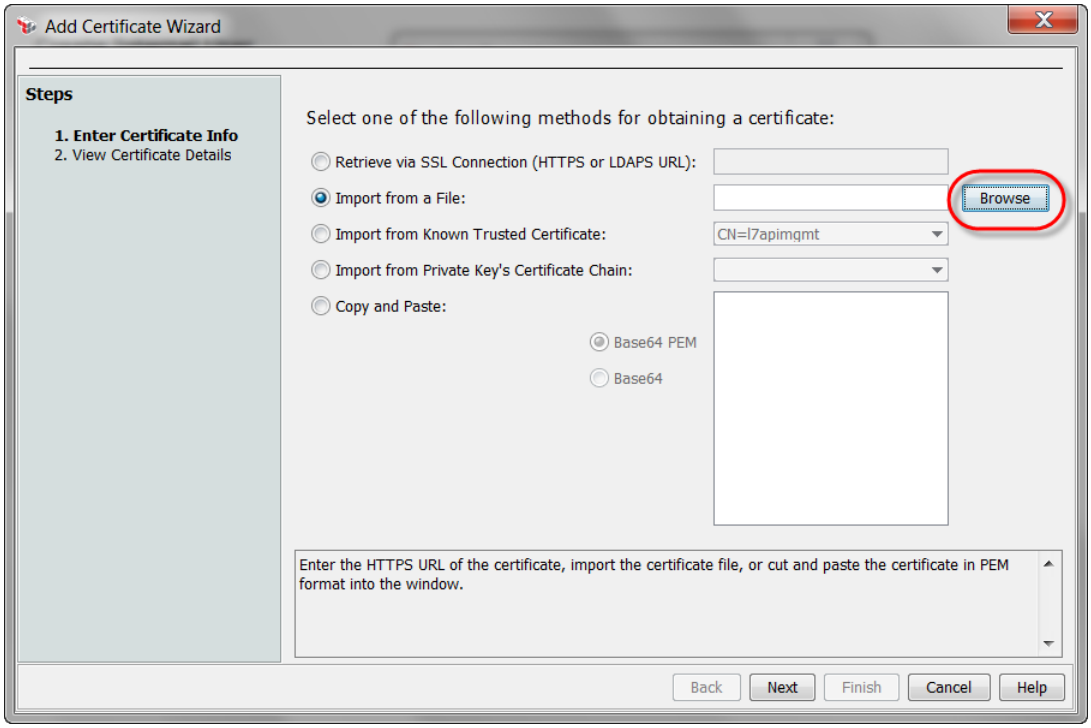
23. In the Create Federated User dialog, in the User Name field enter ***client***, select the ***Define Additional Properties*** option, and click the ***Create*** button.



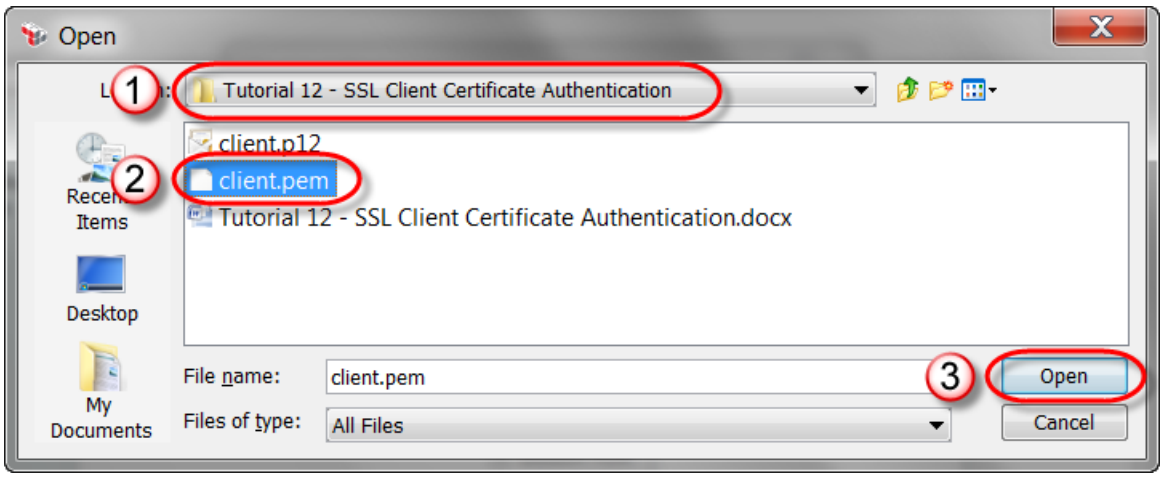
24. In the client Properties dialog, select the ***Certificate*** tab, and click the ***Import*** button.



25. In the Add Certificate Wizard dialog, on step 1, click the ***Browse*** button.

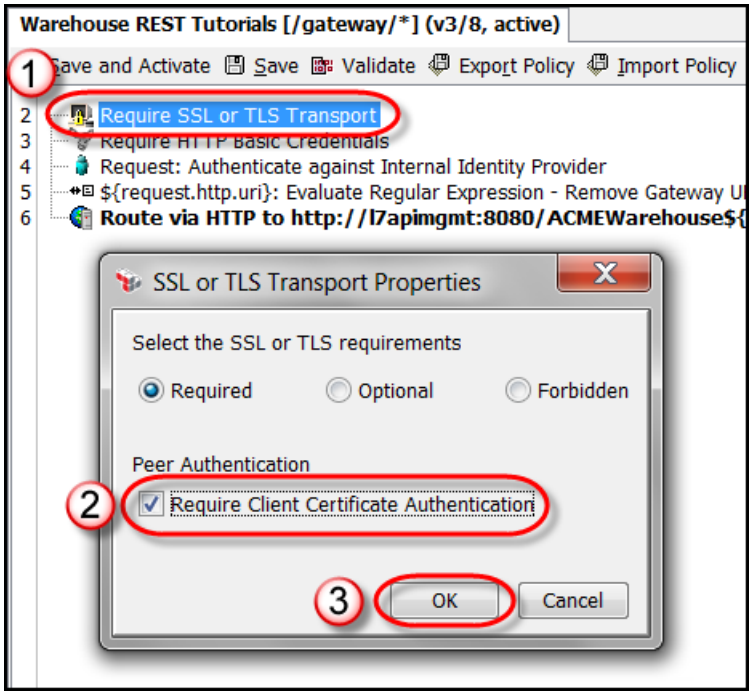


26. In the Open dialog, navigate to and select the client certificate (e.g. ***client.pem***) file that you intend to use during the tutorial, and click the ***Open*** button.

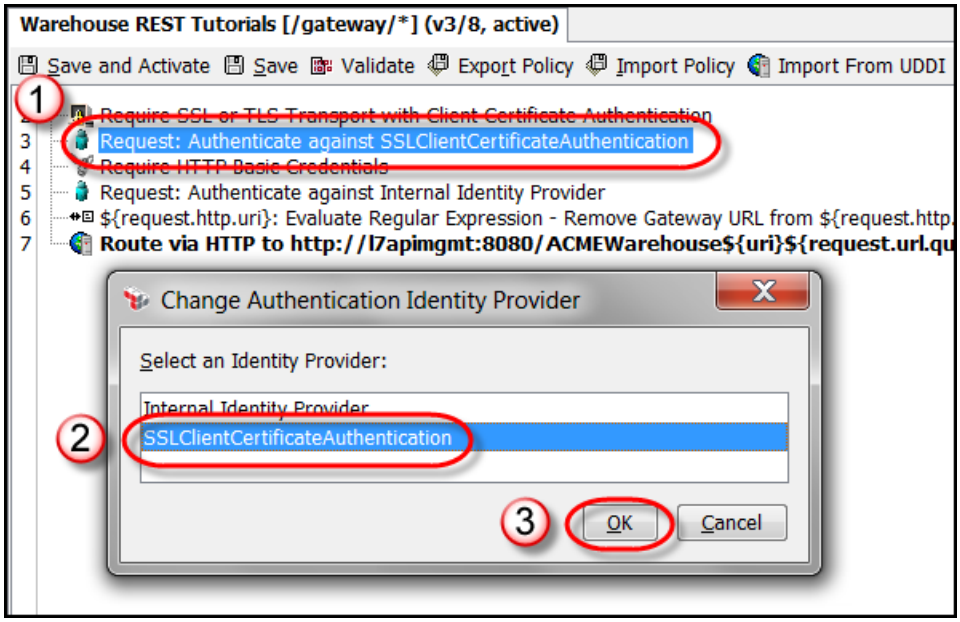


27. In the Add Certificate Wizard dialog, on step 1, click the ***Next*** button.
28. In the Add Certificate Wizard dialog, on step 2, click the ***Finish*** button.
29. In the client Properties dialog, click the ***OK*** button.
30. In the policies and services tree, find and double click the ***Warehouse REST Tutorials*** service to load its active policy into the policy editor.

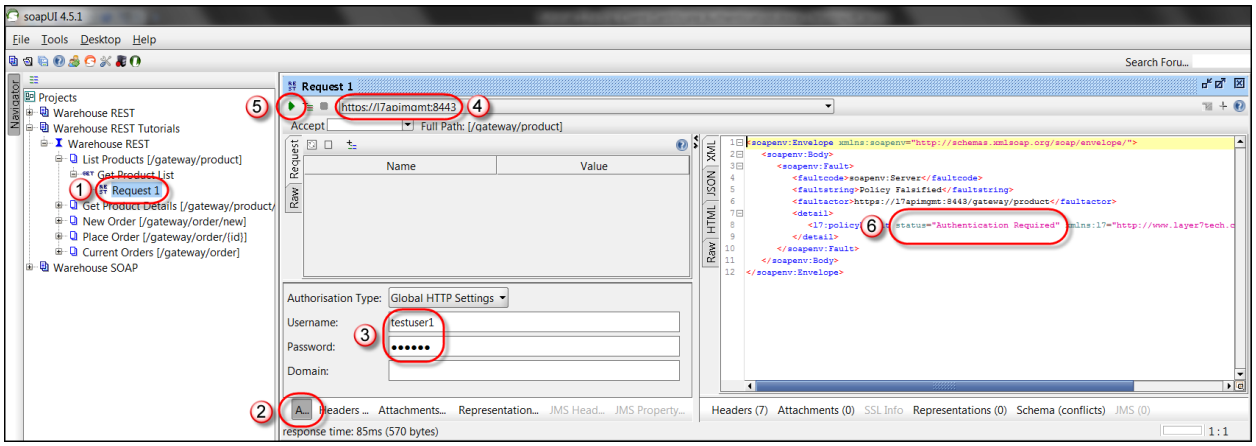
31. In the policy editor, double click on assertion #2, ***Require SSL or TLS Transport***, and in the SSL or TLS Transport Properties dialog, select the ***Require Client Certificate Authentication*** option, and click the ***OK*** button.



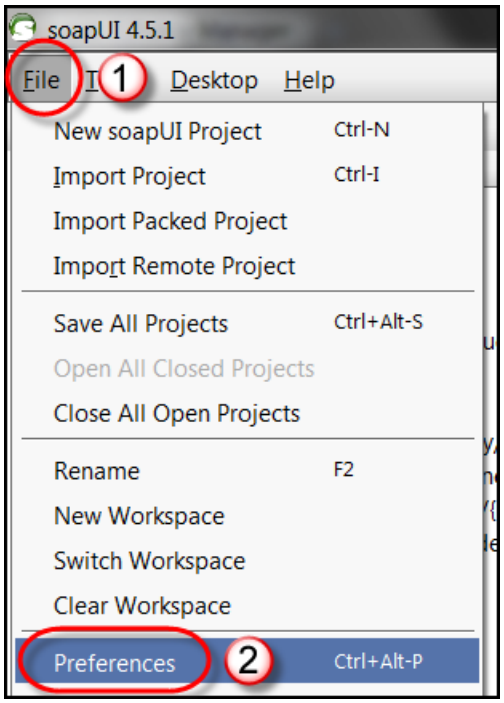
32. From the policy assertion tree, drag and drop the ***Policy Assertions/Access Control/Authenticate Against Identity Provider*** assertion so that it's assertion #3 in the ***Warehouse REST Tutorials*** service policy in the policy editor, and in the Change Authentication Identity Provider dialog, select the new ***SSLClientCertificateAuthentication*** FIP, and click the ***OK*** button.



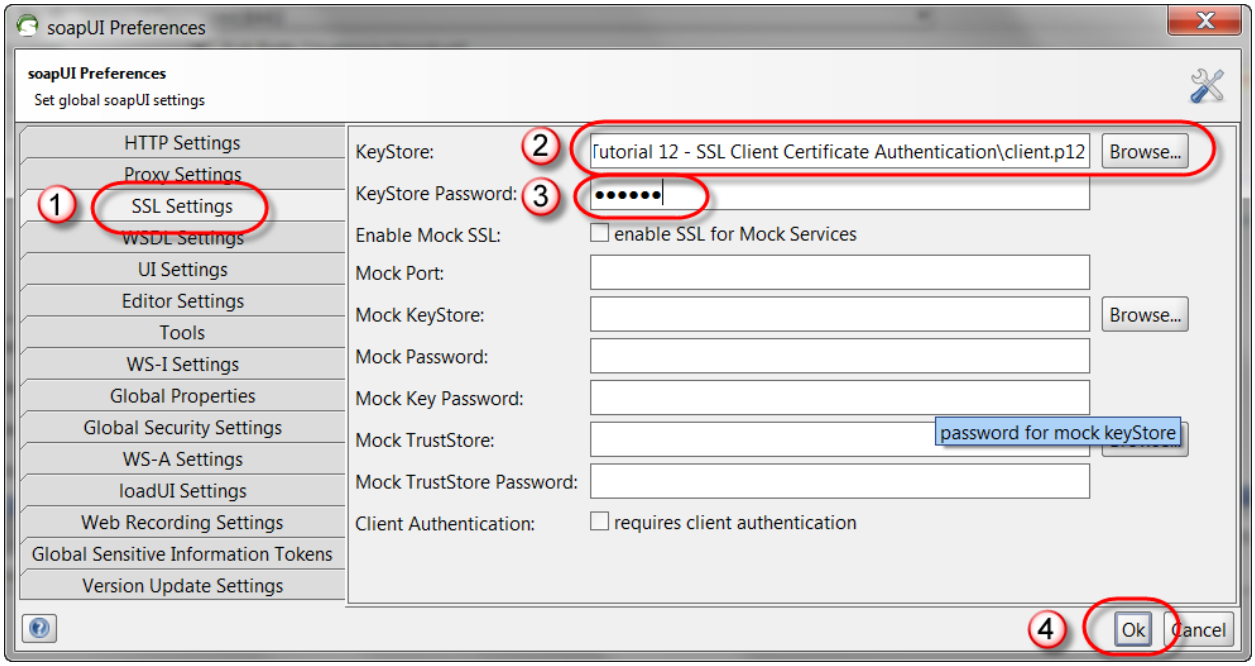
33. On the policy editor toolbar, click the ***Save and Activate*** button.
34. Open soapUI, and use the ***Warehouse REST Tutorials*** project used in the ***Tutorial 6 - Basic Authentication*** tutorial to test this tutorial's policy.
35. Open the ***Get Product List*** request, click the ***Auth*** tab, ensure that the Username filed is set to ***testuser1*** and the Password field is set to ***7layer***, select the HTTPS endpoint of the tutorial service on the gateway, click the ***green arrow*** button to send your request, and notice that you get back and ***Authentication Required*** error message.



36. Select the ***File/Preferences*** menu item.

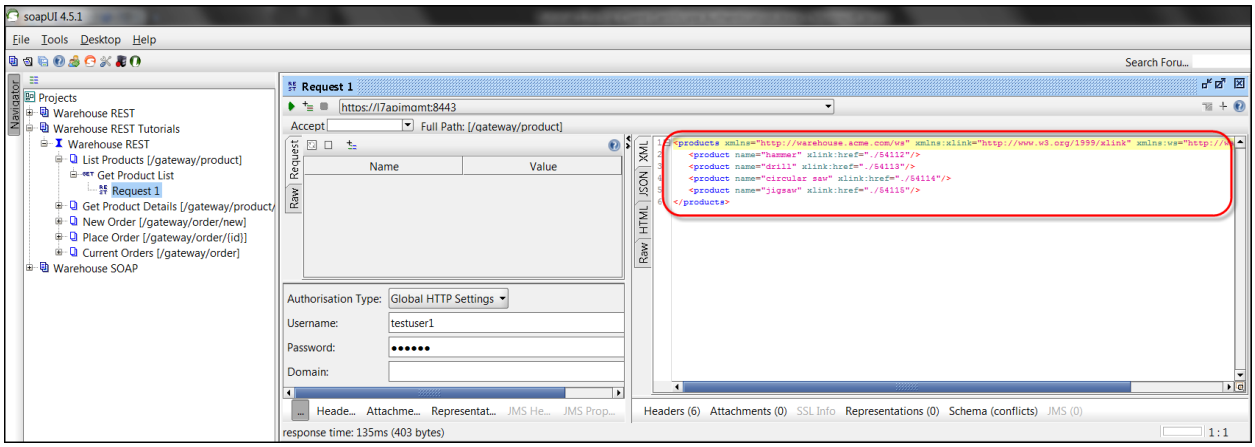


37. In the soapUI Preferences dialog, click the ***SSL Settings*** tab, browser for the ***client.p12*** keystore file that you're using for this tutorial, enter the password ***7layer***, and click the ***OK*** button.



Note: This is the easiest way to specify a client certificate that soapUI should use for mutual auth SSL connection. However, this specifies one client certificate used for all such connections in soapUI. There is a more advanced way of specifying keystores at the soapUI project level for more selectively using different client certificates for different requests in a project.

38. Resend the soapUI request from step 35. You should now get a successful response.



39. Per ***Layer 7 Tutorials - Getting Started/Basic Policy Concepts/Policy Authoring/Policy Revisions***, and as demonstrated at the end of ***Tutorial 1 - Deploy Tutorial Services***, comment the active policy revision of the ***Warehouse REST Tutorials*** service with the comment, **Tutorial 12 Complete**.
40. You are done with this tutorial.

12.4 Additional Context

None