HP Select Federation

For the Windows® operating system

Software Version: 7.00

Windows Connector Guide

Document Release Date: August 2007
Software Release Date: September 2007

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

HP Select Federation includes software developed by third parties. The software in Select Federation includes:

- Apache Derby, Apache Xalan Library, Apache Xerces Library, and Apache XML Dsig Library.
- Software developed by the Waveset Technologies, Inc. (www.waveset.com).
- Software developed by the University Corporation for Advanced Internet Development <http://www.ucaid.edu>Internet2 Project.

Trademark Notices

- Java™ and all Java based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.
- Microsoft®, Windows®, and Windows XP® are U.S. registered trademarks of Microsoft Corporation.
- Oracle® is a registered trademark of Oracle Corporation. Various product and service names referenced herein may be trademarks of Oracle Corporation.

Documentation Updates

This manual's title page contains the following identifying information:

- Software Version number, which indicates the software version
- Document Release Date, which changes each time the document is updated
- Software Release Date, which indicates the release date of this version of the software

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

You can visit the HP Software support web site at:

http://www.hp.com/go/hpsoftwaresupport

HP Software online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels and HP Passport, go to:

http://h20229.www2.hp.com/passport-registration.html

Contents

1 Introduction

The Windows connector provides two integration modes that can be used with Select Federation: outbound-integration for an Authority (IDP) Site and inbound-integration for an Application (SP) Site.

- Outbound-integration, called Integrated Windows Authentication (IWA), seamlessly integrates windows users as federation-capable users at a Select Federation IDP site.
- Inbound-integration, called Inbound Windows Integration (IWI), seamlessly integrates federated users at a Select Federation SP site to applications hosted on the Windows environment.

This chapter includes the following topics:

Prerequisites

This document assumes you have knowledge of the following:

- HP Select Federation (installation, configuration, concepts, and so on)
- HP Select Federation IIS Filter and IIS Configuration UI components (installation, configuration, concepts, and so on)
- Web application servers: Select Federation's built-in server (Tomcat 5.5), WebLogic 8.1 and 9.1, and WebSphere 6.0.2 (installation, configuration, concepts, and so on)
- Applications: Any application that can leverage Impersonation Tokens (installation, configuration, concepts, and so on) — for example, Citrix 4.5, see Appendix B, Integrating Select Federation with Citrix.

How IWA Works

Integrated Windows Authentication (IWA) allows Select Federation to leverage a user's Windows logon credentials to seamlessly authenticate the user and transfer the user to a trusted federation partner site. By using IWA, any authenticated user already logged on to their Windows machine, who attempts to log on to a protected Application Portal page, can visit any other third-party service providers from that page.

IWA consists of the following:

- Authentication Plugin to track sessions and set cookies.
- Filter-Support Service (FSS), a servlet component, which allows trusted programs to inject a Windows-authenticated user-id into an IDP session.
- ASP (Active Server Page) page to capture NTLM (NT LAN Manager) credentials and send them to the FSS over a secure channel.

Prerequisites

To provide Integrated Windows Authentication (IWA), the users must meet the following prerequisites:

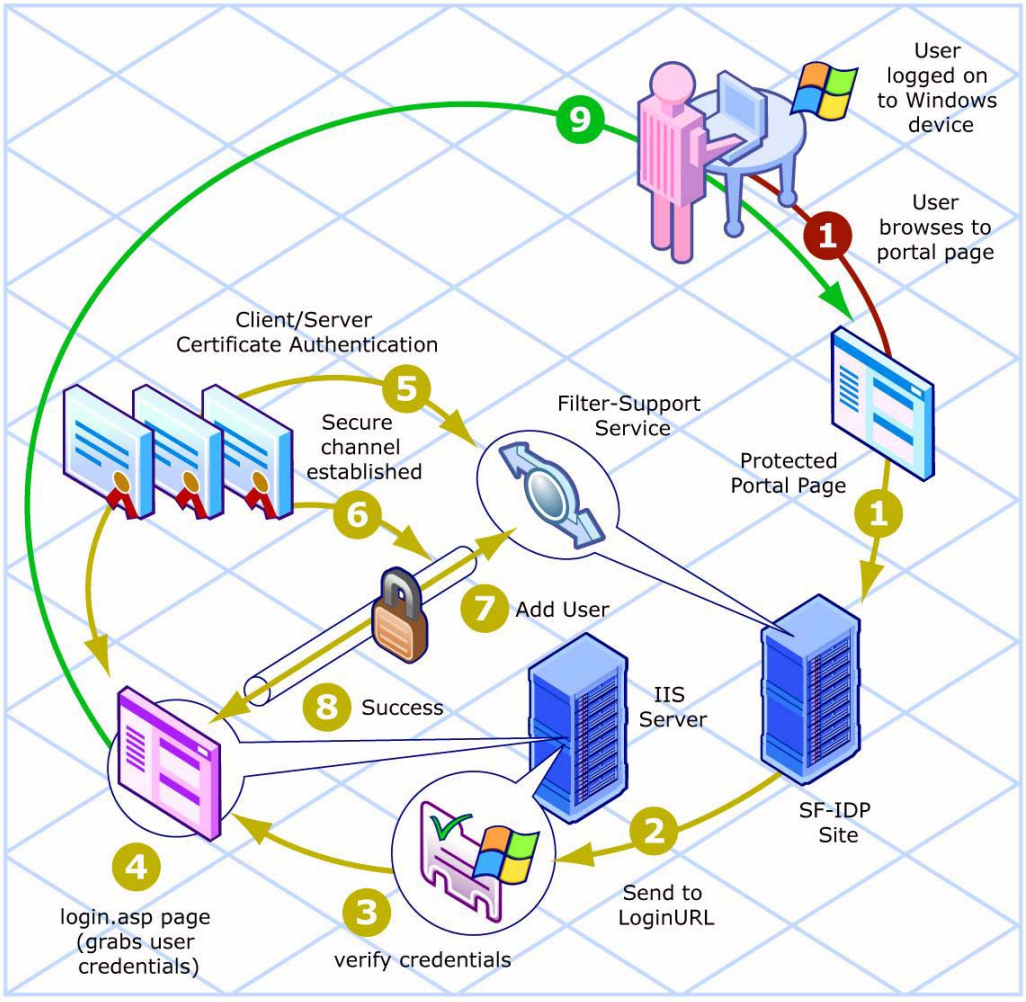
- Users must be logged on to their Windows machines.
- User browsers (such as Internet Explorer (IE)) must be enabled to use their Windows credentials to perform automatic logons.

> Users cannot use international characters for user names (double-byte characters) when using the Internet Explorer 6.0 browser to access a virtual directory on which Integrated Windows Authentication has been enabled. This problem does not occur when you use Internet Explorer 7 to access the same virtual directory while using international double-byte characters. For more information about using international characters, see “Localizing Select Federation” in the “Customizing Select Federation” chapter of the *HP Select Federation Configuration and Administration Guide*.

IWA Scenario

Figure 1 illustrates a typical workflow when using IWA.

Figure 1 Typical IWA Workflow



Following is a step-by-step explanation of this diagram:

1 A user (who is logged on to their Windows machine), attempts to log on to a protected Application Portal Page, and is sent to an SF-IDP (Select Federation IDP).

The Application Portal Page has been set up to point to the SF-IDP.

2 The Authentication Plugin at the SF-IDP redirects the user to the configured loginURL, which is an NTLM-protected page hosted on your IIS server.

3 The IIS server verifies that the credentials are valid and lets the user through to the `login.asp(ASP)` page.

4 The ASP login page captures the user's default windows credentials from the user's browser session with IIS.

5 The ASP page performs server and client certificate authentication with the Filter-Support Service (FSS), which is a component of the SF-IDP. The following certificates must be accepted for the successful establishment of a secure connection:

- Server certificate used by the FSS (server side) is accepted by the ASP making the call (client side).
- CA certificate that was used to sign the Client Certificate (used by the ASP client-side) is accepted by SF-IDP (server-side).

- Client certificate used by the ASP making the call (client-side) is validated by FSS running SF-IDP (server-side).

6 If the certificates are valid, a secure connection is established.

7 The ASP invokes the FSS (over the secure connection) to add and create a session for the user with the SF-IDP.

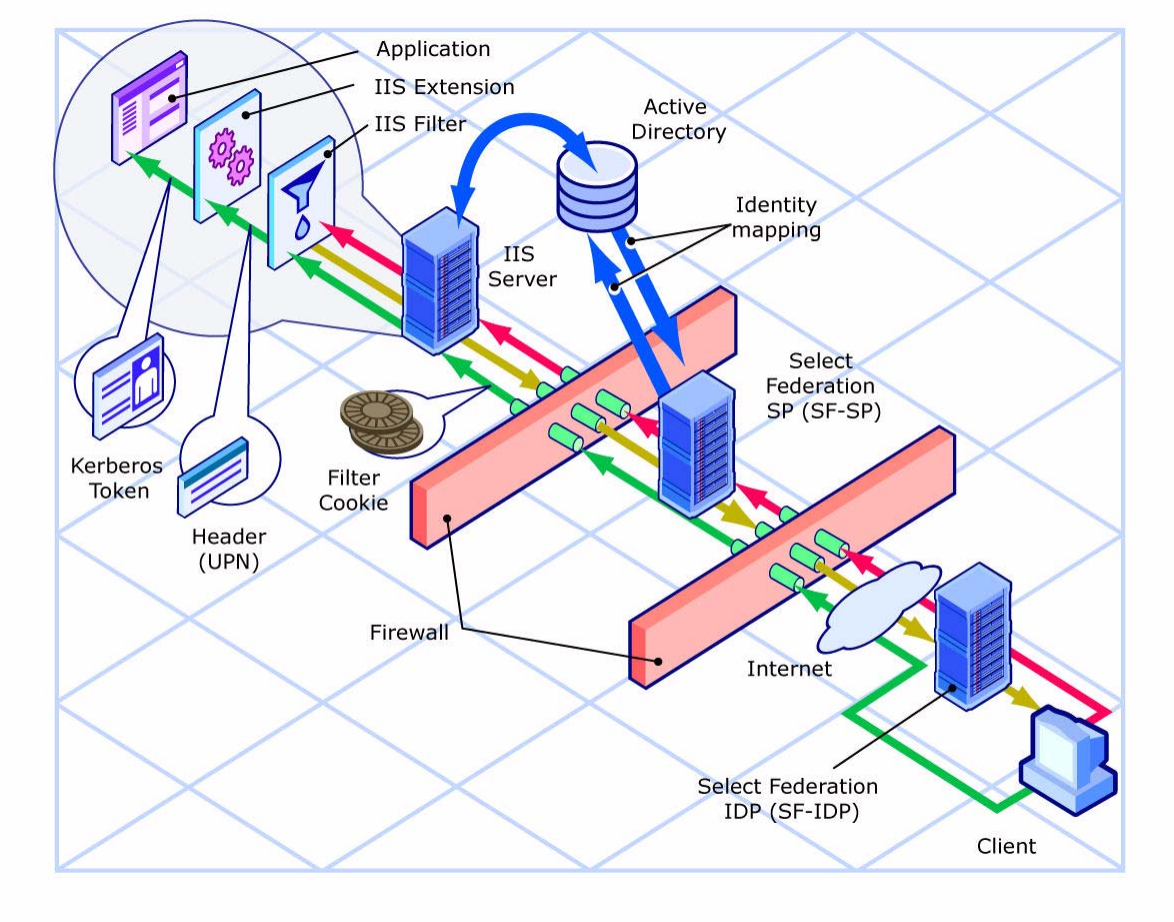
8 The FSS logs the user in to the SF-IDP and notifies the ASP page of its success.

9 The ASP page redirects the user back to the Application Portal Page that the user had initially attempted to access.

How IWI Works

Figure 2 illustrates a common use case where a successfully federated user at the SP site is integrated into the Windows environment.

Figure 2 Integrating a Federated User at an SP Site into a Windows Environment



Following is a step-by-step explanation of this diagram:

1 An unauthenticated user (the red line) tries to access a resource (protected URL) on the IIS server and is intercepted by the IIS Filter.

2 The IIS Filter redirects the unauthenticated user to the SF-SP (yellow line) and the user goes through the process of SP-initiated SSO.

3. From the SF-SP, the user is sent to an IDP where the user is authenticated.
4. The authenticated user is then redirected back to the SF-SP (green line) as a federated user (a Federation Assertion serves as proof).
5. If the assertion is valid, the SF-SP needs to know exactly who this user is going to be in the Windows domain. This process of mapping the federated user to a user in Windows is called **Identity Mapping**. Identity Mapping occurs between the SF-SP and the Active Directory.

   Identity Mapping is different for each customer. Therefore, Select Federation provides an Activate URL EventPlugin as a hook into the Select Federation workflow. See Chapter 3, IWI - Inbound Windows Integration for information on configuring the Activate URL EventPlugin.

   The process of Identity Mapping is explained as follows:

   a. At the SF-SP, some of the event plugins are responsible for activation. The end result should be that the Local User ID is set as the value of a UPN in the Active Directory.

   b. Then further down the chain of event plugins, the Select Federation Filter EventPlugin is executed and the cookie that allows the user to go past the IIS Filter is set.
6. After the Event Plugins at the SF-SP are completed, the user is sent to the URL that was originally requested. This time, the IIS Filter lets the user through to the IIS server.
7. The IIS Extension (`SFExtension.dll` file) retrieves the federated user's request and does the following:

   a. Uses the information provided by the SF-SP to acquire a Kerberos token.

   b. Stores the original token and sets the Kerberos token before allowing the request to be processed by the web application.

      In this example, the Kerberos Authentication-aware application acquires the token in an ASP.NET application running in the IIS server. The ASP.NET application does whatever is needed under the security context of that token.

   c. Resets the original token and cleans up the Kerberos token, after the main request processing has been handled by the IIS server.

   d. After the cleanup, the IIS Extension lets the IIS server return the response to the user.

Related Documentation

Select Federation includes the following documentation, which includes additional information on how to use Select Federation. The documents are located in the `<cd-base-directory>\docs` directory:

- *HP Select Federation Certificate Management User's Guide* — `cert.pdf`
- *HP Select Federation Configuration and Administration User's Guide* — `config.pdf`
- *HP Select Federation Deployment Concepts Guide* — `deployment.pdf`
- *HP Select Federation Installation Guide* — `install.pdf`
- *HP Select Federation Quick Start Guide* — `quickstart.pdf`

2 IWA - Integrated Windows Authentication

This chapter provides installation and configuration information for IWA in the following topics:

Installation

IWA has no components that require installing.

System Requirements

The following software is required to run IWA:

- Select Federation 7.00 — already installed. See the *HP Select Federation Installation Guide* for installation instructions.
- Windows 2003

Configuring IWA

The machine hosting the IIS server and the machine hosting SF-IDP need to have the same domain in their URLs for cookies to be functional and IWA to work. For example, if the IIS machine can be accessed over the URL `iis.hp.com`, then the SF-IDP machine should be accessible through a URL such as `idp.hp.com` where the common domain is `.hp.com`.

To finish configuring IWA, complete the following tasks:

Task 1: Add a Port with Server and Client Authentication

Task 2: Set up the Filter-Support Service (FSS) component

Task 3: Set up the ASP pages

Task 4: Set up the Authentication Plugin

Task 5: Set up Server Authentication

Task 6: Set up Client Authentication

Task 7: Restart SF-IDP for the configuration additions and changes to take effect.

Task 1: Add a Port with Server and Client Authentication

IWA requires a highly secure channel of communication for allowing the introduction of users into SF-IDP from Windows. Although your application server should already have a port that employs Server Authentication, you need an additional new port that uses both Server and Client Authentication.

> You SHOULD NOT reuse the pre-existing port because many other functions of Select Federation do not and should not require client authentication.

Adding a Port with Server and Client Authentication for the Built-in Application Server

To configure Server and Client Authentication for the built-in application server, perform the following steps:

1. Open the `<sf-home>/conf/server.xml` file to edit.
2. Add a line similar to the following example:

```
<!-- Define an SSL HTTP/1.1 Connector on port 9443 -->
<Connector port="9443" maxHttpHeaderSize="8192" maxThreads="150"
minSpareThreads="25" maxSpareThreads="75" enableLookups="false"
disableUploadTimeout="true" acceptCount="100" scheme="https" secure="true"
clientAuth="true" sslProtocol="TLS" keystoreFile="<sf-home>/conf/
sslkeystore.jks" keystorePass="<yourPassword>" />
```

   You may reuse the above example if you replace the values in the following parameters with values that correspond to your SF-IDP environment:

```
# Replace 9443 with the port that you wish to use
# for establishing all server/client TLS connections
Connector port="9443"

# Replace <sf-home> with the correct path, OR
# Replace <sf-home>/conf/sslkeystore.jks with the path to
# the keystore you use for establishing ssl connections
keystoreFile="<sf-home>/conf/sslkeystore.jks"

# Replace <yourPassword> with the password for your
# keystore
keystorePass="<yourPassword>"
```

Adding a Port with Server and Client Authentication for WebSphere

For instructions on how to add and configure a port with server and client authentication for WebSphere, see the “Select Federation SSL Deployment on WebSphere 6.02 with TLS Client Authentication Enabled” section in the “Certificate Management” chapter of the *HP Select Federation Configuration and Administration Guide*. For more details, see the WebSphere documentation.

Adding a Port with Server and Client Authentication for WebLogic

For instructions on how to add and configure a port with server and client authentication for WebLogic, see the WebLogic documentation. Once you have configured the port, perform the following step:

- Add `-Dsf.wlsClientAuth=true` to the `JAVA_OPTIONS` in the classpath variable of `startWeblogic.cmd` (Windows) or `startWeblogic.sh` (UNIX) scripts.
  - For WebLogic 8.1, the startup script path is as follows:

Windows:
```
$DOMAIN_HOME\startWebLogic.cmd
set JAVA_OPTIONS=-Dsf.wlsClientAuth=true;%JAVA_OPTIONS%;
```

UNIX:
```
$DOMAIN_HOME/startWebLogic.sh
JAVA_OPTIONS=-Dsf.wlsClientAuth=true:$JAVA_OPTIONS:
```

— For WebLogic 9.1, the startup script path is as follows:

Windows:
```
$DOMAIN_HOME\bin\startWebLogic.cmd
set JAVA_OPTIONS=-Dsf.wlsClientAuth=true;%JAVA_OPTIONS%;
```

UNIX:
```
$DOMAIN_HOME/bin/startWebLogic.sh
JAVA_OPTIONS=-Dsf.wlsClientAuth=true:$JAVA_OPTIONS:
```

Task 2: Set up the Filter-Support Service (FSS) component

The FSS component should already be deployed on any application server where Select Federation has been properly installed.

Perform the following steps to modify the `<sf-home>/conf/tfsconfig.properties` file of your SF-IDP (Select Federation IDP) to set up the FSS component:

1 Specify the `fssURL`.

The `securePortNumber` should be the same as the one you decided to use when you enabled Server and Client Authentication in your application server as part of Task 1.

```
# SF will send this information to the IIS machine,
# which will then use it to introduce users into SF.
# hostname: the SF-IDP server where your tfs-fs war is deployed
# securePortNumber: the SF-IDP server's port number that has been
# added to support server and client authentication
fssURL=https://<hostname>:<securePortNumber>/tfs-fs/FSS
```

2 Optionally, you may set the `requireDN` flag to `true` if you want the full user DN to be registered with the SF-IDP when a user logs in through IWA:

```
# This value can be explicitly set but it will default to true if
# DirPlugin_ADS or DirPlugin_LDAP is in use *and* ldapUserBaseDN is absent
# or empty because that means that a fullDN is NEEDED.
# boolean 0 (false) or 1 (true)
fss.requireDN=1
```

3 Optionally, you may set the `requireTLS` flag to `true`:

```
# For the "IDP-like" functionality of FSS (such as
# allowing IWA to register a user with the SF-IDP),
# the Client TLS Authentication is always required,
# no matter what the value of this tfsconfig option is.
# boolean 0 (false) or 1 (true)
fss.requireTLS=1
```

Task 3: Set up the ASP pages

Perform the following steps to set up the ASP pages:

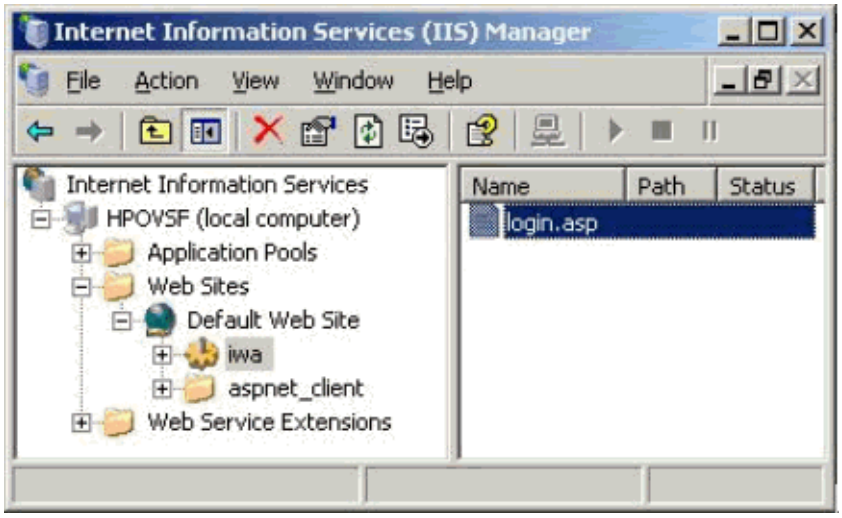
1 Copy the `login.asp` file (and other ASP files) from `<sf-cd-base-directory>\connectors\windows\iwa` to the machine where your IIS server is located.

2 Specify the location of the `login.asp` file in a virtual directory or web site that makes sense for your IIS server(s).

3 Set at least `Read` and `Run scripts` access permissions to the virtual directory or web site that contains the `login.asp` file.

The following figure shows a generic (non-production-specific) example of the end result on a test server.

Figure 3 Generic (Non-Production-Specific) Example of login.asp on a Test Server



4 Be sure that the **Status** of the **Active Server Pages** (under **Web Service Extensions**) is set to **Allowed**.

5 Right-click on the location you specified for hosting the `login.asp` file and select **Properties**.

6 Select **Directory Security** → **Authentication and access control** → **Edit**.

The Authentication Methods Dialog opens.

Figure 4 Authentication Methods Dialog

7 Select **Integrated Windows authentication** so that the location where you placed `login.asp` is NTLM-authentication enabled. Make sure to de-select **Enable anonymous access**.

8 For the duration of this setup, uncheck the **Show friendly HTTP error messages** option in your **Internet Explorer Tools** → **Internet Options** → **Advanced** tab.

9 Restart IE for the setting to take effect.

10 Test that you have followed the instructions correctly so far, by browsing to the `login.asp` ASP page using IE.

For example, based on the figure of the generic (non-production-specific) example above, you would test your install by browsing to: **http://localhost/iwa/login.asp**.

If you turned off friendly error messages and receive an error about a line in the `login.asp` file, you have followed the instructions correctly.

For web-based (browser-based) access to work “seamlessly” in any solution, it is imperative that the user's browser is enabled to present the user’s credentials by default. In the Internet Explorer (IE) browser, you can control this through the Security Settings for your users’ browsers depending on which security zone makes sense for your enterprise. It is up to you to decide how you want to put such a policy in effect enterprise-wide.

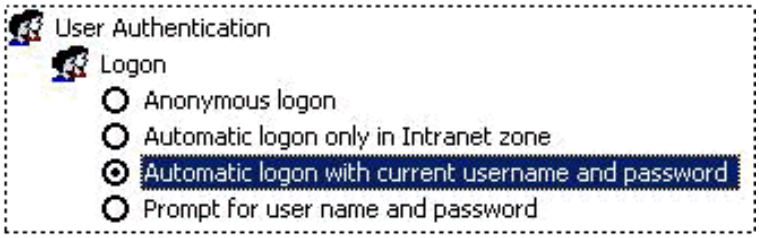
The following steps provide an example of how to enable the user’s browser manually for Internet Explorer (IE):

a Select **Tools** → **Internet Options** → **Security** and click the **Trusted Sites** icon (a Web Content Zone).

b Click the **Custom Level** button and the Security Settings window opens.

c Scroll to the bottom and select **User Authentication** → **Logon** → **Automatic logon with current username and password**.

Figure 5 User Authentication Setting in the Security Settings Window



d Click **OK** and then **Yes**.

e Click the **Sites** button, type in the URL to the IIS machine which hosts the login page (for example: **http://hpovsf.domain.com**), and click **Add** then **Close**.

f Click **OK** in the **Security Settings** window.

11 Test that you have followed the instructions correctly so far, by browsing to the `login.asp` page using the IE browser.

For example, you would test your install based on the following:

- The snapshots of the generic (non-production-specific) example above.
- The example for IIS machine name that you added to trusted sites above, you would test your install by browsing to: **http://hpovsf.domain.com/iwa/login.asp**.

If you are not prompted for your Windows credentials and receive an error about a line in the `login.asp` file, then it means that you have followed the instructions correctly.

If you are prompted for your Windows credentials and you did not receive an error about a line in the `login.asp` file, then it can mean one of the following occurred:

- You mis-configured enabling the browser.

OR

- Your IIS server has determined that the initial credentials provided by the browser are invalid according to whichever database or directory it employs for authentication and authorization of the NTLM credentials.
  - If this is the case then you can try to provide a set of credentials that you know to be genuine and see if you receive an error about a line in the `login.asp` file. If so, then you have configured properly so far.
  - As a suggestion, you may wish to consider if you need to expand your backend to include the tree or forest or database table that would include your user. This would enable IIS to allow you to seamlessly access the `login.asp` page and facilitate IWA in general.

12 Now that you have deployed the ASP page, modify the `$SF_HOME/conf/tfsconfig.properties` file of your SF-IDP (Select Federation IDP) as follows:

```
# If the tfsconfig.properties file already contains lines for loginURL,
# those lines should be commented out.
#
# For IWA, the loginURL should point to the login.asp location
# as accessed BY the end users' browsers. Meaning the user should
# actually be able to access that URL.
# For example, if: loginURL=http://hpovsf.domain.com/iwa/login.asp
# then the users should be able to get to it via their browsers
loginURL=http://<IISServerName>:<portNumber>/<somePath>/login.asp
```

Task 4: Set up the Authentication Plugin

Modify the `$SF_HOME\conf\tfsconfig.properties` file of your SF-IDP as follows:

```
# Specify the plugin: You may comment out whatever
# value previously existed
idpAuthnPlugin=com.trustgenix.tfsIDP.util.IDPAuthnPlugin_FSS
```

Task 5: Set up Server Authentication

IWA requires that you use an HTTPS install. If you chose HTTP rather than HTTPS during the Select Federation installation, the built-in application server's server certificate file, `tomcat.cer`, will not be available. To change from HTTP to HTTPS you can do one of the following:

- Manually convert your install from HTTP to HTTPS.
- Re-install Select Federation and select HTTPS.

Perform the following steps to set up the server authentication:

1. Copy the SF-IDP's `$SF_HOME/conf/tomcat.cer` file to your IIS machine.

   This is the Server Certificate that is presented over a secure connection by your SF-IDP. You must add this certificate to the list of trusted certificates in the local machine store at your IIS machine.

2. Select **Start** → **Run** and enter **`mmc`** to open the Microsoft Management Console (MMC).
3. Select **File** → **Add/Remove Snap-in**.

   The Add/Remove Snap-in dialog opens.

4. Click the **Standalone** tab and click the **Add** button.

   The Add Standalone Snap-in dialog opens, which displays the list of plug-ins available.

5. Select **Certificates** and click the **Add** button.

   The Certificates snap-in dialog opens.

6. Select **Computer account** to create the plug-in to manage certificates for the Computer Account.
7. Click the **Next** button.

   The Select Computer dialog opens.

8. Select **Local computer: [the computer this console is running on]** and click the **Finish** button.
9. Click the **Close** button to close the Add Standalone Snap-in window, then click **OK**.

   You are returned to the Console window with all the certificates on your local computer.

10. Expand the Certificates (Local Computer) node to view the certificates.
11. Click the **File** → **Save** menu options and **Save** this view as you will need it later.
12. Right-click on the **Trusted Root Certification Authorities** folder and select **All tasks** → **Import**.

    The goal is to import the server certificate file (that you copied from SF-IDP to your IIS machine) to **Certificates (Local Computer)** → **Trusted Root Certification Authorities**.

    The Certificate Import Wizard opens.

13. Click **Next** and browse to the location of your SF-IDP's server certificate file that you had copied to the IIS machine.
14. Open the file and Click the **Next** button.

The Certificate Store page opens.

15 Select **Place all certificates in the following store**.

By default **Trusted Root Certification Authorities** should be selected in the text box below this option.

16 Click **Next**.

The Completing the Certificate Import Wizard page opens.

17 Review your selections and make sure that you have all the right information.

18 Click **Finish**.

A confirmation dialog opens to confirm the successful import.

19 Click **OK**.

Task 6: Set up Client Authentication

Since the ASP page that invokes the FSS, represents the client-side of your secure connection, you need a `pkcs12` / `pfx` file that has both the certificate and the private key pair in it.

Another major requirement for the client certificate is that its Subject's `cn` field must match the IIS server machine's name that you specified in the loginURL.

As an example, if `loginURL=http://hpovsf.domain.com/iwa/login.asp` or `loginURL=http://hpovsf.domain.com:80/iwa/login.asp` then you need to make sure that the subject of your client certificate includes `cn=hpovsf.domain.com`.

- If you have a certificate that is just for your IIS server (that you think is appropriate for use in client authentication) and it matches the above requirements, then you can continue at Importing the Client Certificate (pkcs/pfx format file) on page 21.
- If you do not have a certificate that is just for your IIS server (that you think is appropriate for use in client authentication), or it does not match the above requirements, then you can create a certificate using any tool or utility of your preference to do the following:
  - Create a CSR (certificate signing request) that matches the above said requirements and your security constraints and then get it signed by the CA of your choice.
  - Create a self-signed certificate that matches the above requirements and your security constraints.

  Or

  - You may use the Certificate Management Tool (CMT), which Select Federation has provided. Follow the steps in the next section Creating a Certificate. (See the *HP Select Federation Certificate Management User's Guide* for more information.)

Creating a Certificate

You can use the Certificate Management Tool provided with your Select Federation installation to generate and export a self-signed certificate into a `pkcs12 /pfx` file format with both the certificate and the private key pair.

Perform the following steps to create a certificate using the Certificate Management Tool:

1 Start the CMT tool by running one of the following commands on the operating system where your SF-IDP is installed:

On Windows: **`<sf-home>/tools/cmt/cmt.cmd`**

On UNIX: **`<sf-home>/tools/cmt/cmt.sh`**

2 Select the **File** → **New KeyStore** menu options to create a new keystore, which saves the key-pair you will generate.

The Create New KeyStore dialog opens.

3 Provide a keystore file location and password for the new keystore and click **Create**.

The Certificate Management Tool console opens.

4 Select the **Edit** → **Create New Entry** menu options to create a new entry in the keystore to generate a key-pair.

The Create New Entry dialog opens.

5 Fill in all the required fields and any optional ones.

- Keep in mind the rule that the Subject's `cn` field must match the IIS server machine's name that you specified in the loginURL.
- Fill in the values for the following fields: **Alias**, **Password** and **Common Name**.

6 Click the **Self-Signed-Certificate** button to generate the certificate.

The Certificate Management Tool console now has the new entry listed under the **Keys & Certificates** node.

7 Click on the **Self-Signed certificate** (key-pair) under the **Keys & Certificates** node, that you want to use with the ASP page.

8 Select the **Export** → **Export to PKCS#12** file menu options.

The Export PKCS # 12 File dialog opens.

9 Provide the following information:

- Password for the key-pair you are exporting.
- New password for the `pkcs/pfx` file that will be exported.
- Name and location of that file.

10 Click the **Export** button.

If you get an error while exporting, it may be due to the length of the password you specified for the `PKCS` file. If you get an error, you can choose to do one of the following:

- Limit the password to be less than 8 characters in length.
- Download “Unlimited Strength” Jurisdiction Policy Files and install with the JRE/JDK used for running this tool. To download the JCE Unlimited Strength Jurisdiction Policy Files, go to the “Downloading the ‘Unlimited Strength’ Jurisdiction Policy Files” section from the following web site:

  http://java.sun.com/products/jce/index-14.html.

  You will need to restart the Certificate Management Tool.

11 Select **File** → **Save KeyStore**.

12 Select **File** → **Close KeyStore**.

Importing the Client Certificate (pkcs/pfx format file)

Perform the following steps to import the `pkcs/pfx` format file into your local store at the IIS machine if you have not done so already:

1 Select **Start** → **Run** and enter **`mmc`** to open the MMC console and open the view that you had saved when you were setting up server authentication.

If you never closed that view then you can continue using it now.

2 Right-click on **Personal** and select **All Tasks -> Import** to import the key-pair to use for client authentication.

The Certificate Import Wizard opens.

3 Click **Next** and provide the path to your `pkcs/pfx` format file.

4 Click **Next** and provide the password that you used to protect the file.

5 Click the **Next** button.

The Certificate Store page opens.

6 Select **Place all certificates in the following store** option and the **Personal** certificate store should be selected by default in the text box below it. Click **Next**.

The Completing the Certificate Import Wizard page opens.

7 Review your selections and make sure that you have all the right information. Then click **Finish**

8 A confirmation dialog opens to confirm the import. Click **OK**.

Granting Access to the Client Certificate (pkcs/pfx format file)

1 Locate the file that you imported into the **Personal** certificate store.

- The location of the certificates that you added to the local store using `mmc` is:

  ```
  <windows_install_directory>\Documents and Settings\All
  Users\Application Data\Microsoft\Crypto\RSA\MachineKeys
  ```

  The `<windows_install_directory>` can be drive letters such as `c`, `d` or `e`.

- Go to the appropriate location on your machine and sort the files under the `MachineKeys` folder by **Date Modified**.

  The most recently time-stamped file is the one you just imported into the **Personal** certificate store.

2 Right-click on the file you imported and select **Properties**.

3 In the Security tab, click the **Add** button

4 Enter **Users** in the text field and click the **Check Names** button.

5 If the check does not object, then simply click the **OK** button.

6 Click **Apply** and then **OK**.

Adding your Client Certificate's signing CA to Your SF-IDP's Truststore

To establish a secure connection, the certificate for the CA that signed your client certificate needs to be added to your SF-IDP's Truststore.

If the client certificate happens to be self-signed, then it itself needs to be added to your SF-IDP's Truststore.

1 If you have a proper CA that signed your certificate, then copy the certificate for that CA to your IIS machine and continue to step 3.

2 If you do not have a proper CA, you can use the CMT to export a certificate out of the keystore in which you had created a key-pair entry earlier, as follows:

a Open the keystore you had created earlier for generating your self-signed certificate.

b Provide the password that you had set for that keystore.

c Select the self-signed certificate (key-pair) under Keys & Certificates.

d Select the **Export** → **Export Certificate** menu options.

e Browse to the location where you want to save the exported certificate and provide a name for the file. Click **Save**.

f Click **Export** to export the certificate file to your local machine.

3 Import the certificate using the CMT into the `cacerts` (Truststore) file of the JVM that is used by your SF-IDP:

a Start CMT and select the **File** → **Open KeyStore** menu options to open the keystore.

The Load KeyStore dialog opens.

b Provide the location and password for the keystore and click the **Open** button.

— For the built-in application server the keystore is located at `<sf-home>\jre\lib\security` with the default password **`changeit`**.

— For WebSphere and WebLogic, refer to their respective manuals to find the location of their truststore.

c Once the truststore is open, select **Trusted Certificates**.

d Select the **Import** → **Import Certificate** menu options.

The Import Certificate dialog opens.

e Provide an alias and the location of the certificate file.

f Click the **Import** button.

The certificate is added to your truststore's list of trusted certificates.

g Select **File** → **Save KeyStore** menu options to save the changes.

h Select **File** → **Close KeyStore** menu options to close your keystore.

Adding the Client Certificate to Your SF-IDP's FSS Keystore

Since Certificate Authorities sign client certificates for any number of requestors, simply adding the CA certificate to the truststore is not enough. This is because doing so would mean that if any other party's client certificate is signed by the same CA, they can also connect to our FSS service.

Therefore, you also need to make your client certificate available to SF-IDP so that when a secure channel is established, FSS can check and make sure that the client certificate being used is indeed the one that is authorized for use with it.

Even if you are using a self-signed certificate, you will still need to add it once more.

1 If you want to create a separate keystore to manage your IWA client certificates, do the following:

a Create a keystore on the SF-IDP machine and add your CA-signed client certificate or your self-signed client certificate to it.

b Edit the `tfsconfig.properties` file on the SF-IDP to add the following:

```
###
## When absent, the following default to the top level keystore.
## In other words the admin could import the client certificates
## used by FSS into the default SF keystore.
```

```
#
# If fss.keystorePath is present but empty then default
# JVM truststore for SSL will be used, typically this is the
# JAVA_HOME/lib/security/cacerts file
fss.keystorePath=
fss.keystoreType=
fss.keystorePassword=
```

2 If you prefer to use the default keystore used by your SF-IDP, do the following:

a Import the client certificate to the default keystore. For the built-in application server, the default keystore is usually located at `$SF-HOME/conf/sslkeystore.jks`.

b You can find the location of the default keystore in your `tfsconfig.properties` file. Check the values for the following parameters:

```
# Keystore configuration
keystorePath=
keystoreType=
keystorePassword=
```

Task 7: Restart SF-IDP for the configuration additions and changes to take effect.

IWA Log File

For error messages, use the Select Federation application server log file. See Appendix A, Troubleshooting for the most common errors that would appear due to a broken workflow.

3 IWI - Inbound Windows Integration

This chapter provides IWI installation and configuration instructions and information in the following topics:

- Installation
- Configuring IWI
- IWI Log Files

Installation

To use IWI, the IIS Extension and Configuration Interface components need to be installed as described in the following sections:

- System Requirements
- Installing the IIS Extension
- Installing the Configuration Interface
- Uninstalling the IIS Extension
- Uninstalling the Configuration Interface

System Requirements

You need to think about which virtual directories or web sites have applications capable of leveraging Kerberos Tokens to perform their desired set of functions. It is these web sites or virtual directories that can make use of the IWI connector.

Software Requirements

The following software must be installed and configured:

- Select Federation 7.00 — already installed. See the *HP Select Federation Installation Guide* for installation instructions.
- Filter-Support — See “Filter-Support” in Chapter 6, “Enabling Applications” in the *HP Select Federation Configuration and Administration Guide* for instructions on how to configure and use the Filter-Support application.
- IIS Filter Configuration Interface — already installed. See “IIS Filter” in Chapter 6, “Enabling Applications” in the *HP Select Federation Configuration and Administration Guide* for instructions on how to install and configure the IIS Filter Configuration interface.

- IIS Filter provided by Select Federation — already installed. See “IIS Filter” in Chapter 6, “Enabling Applications” in the *HP Select Federation Configuration and Administration Guide* for instructions on how to install and configure the IIS filter.
- IIS Extension — needs to be installed and configured for identity mapping. See Installing the IIS Extension on page 26 for installation instructions and Configuring Select Federation to Perform Identity Mapping on page 32 for configuration instructions.

Supported by IWI

IWI supports Web server IIS 6.x and IIS applications that can leverage Impersonation tokens on Windows 2003 — for example, Citrix 4.5, see Appendix B, Integrating Select Federation with Citrix.

Installing the IIS Extension

Complete the following tasks to install the IIS Extension using the IIS console:

Task 1: Set up the IIS Extension file.

1. Save the `<cd-base-directory>/connectors/windows/iwi/SFExtension.dll` anywhere on your local hard drive.
2. Navigate to the `SFExtension.dll` file on your local hard drive, right-click on the file and select **Properties**.
3. Select the **Security** tab.
4. Click the **Add...** button.

   The Select users, Computers, or Groups dialog opens.
5. Enter **`Authenticated Users`** in the **Enter the object names to select** text field.
6. Click the **Check Names** button.

   If the text in the text field is automatically underlined, you can proceed to the next step. Otherwise, contact your local Active Directory administrator to debug the issue.
7. Click **OK**.
8. In the **Permissions for Authenticated Users** list in the Security tab, make sure that the only permissions allowed are the following:
   - **Read & Execute**
   - **Read**
9. Click **OK**.

Task 2: Open the IIS console.

1. From the **Start** menu click **Run**.

   The Run dialog opens.
2. Enter **`inetmgr`** in the **Open** text box, and click **OK**.

   The Internet Information Services (IIS) Manager dialog opens.

Task 3: Set up the Application Pool

The Application Pool in which your web site or virtual directory is running, needs to have its security account set to the predefined Local System account. To set up the Application Pool, you need to do the following:

- Find which Application Pool you are in
- Configure the Application Pool
- Or, you can create a new Application Pool, then find it and configure it.

Find which Application Pool you are in

Perform the following steps to find your Application Pool:

1. Select the web site or virtual directory that you want to protect.
2. Right-click and select **Properties**.

   The Properties dialog opens.

   - If you right-clicked on a web site, then select the **Home Directory** tab.
   - If you right-clicked on a virtual directory, then select the **Virtual Directory** tab.
3. Note the value of the Application Pool you are currently using, in the **Application Pool** dropdown list in the last field.

Configure the Application Pool

Perform the following steps to configure your Application Pool.

1. Expand the Application Pools node in the left pane of IIS Manager.
2. Right-click on your Application Pool and select **Properties**.

   The Properties dialog box opens.
3. Select the **Identity** tab.
4. Select the Predefined bullet and the **Local System** in the dropdown list.
5. Click **OK**.

Create a New Application Pool

If you think that editing a particular Application Pool offers too many applications with an elevated privilege level, then you can create a new Application Pool to assign to your web site or virtual directory. Then assign this as the pool of choice at the Application Pool dropdown list (see step 3 in Find which Application Pool you are in).

Perform the following steps to create a new Application Pool:

1. Right-Click the Application Pools node in the left pane of the IIS Manager.
2. Select **New** → **Application Pool**.
3. Provide an ID for the pool, such as: **SFAppPool** and click **OK**.
4. Follow the instructions in Configure the Application Pool on page 27.
5. Select the new Application Pool you have created in the **Application Pool** dropdown list in the last field.
6. Click **OK**.

Task 4: Add the IIS Extension file.

Perform the following steps to add the IIS Extension file:

1. Select the web site or virtual directory that you want to protect.
2. Right-click and select **Properties**.

   The Properties dialog opens.

   - If you right-clicked on a web site, then select the **Home Directory** tab.
   - If you right-clicked on a virtual directory, then select the **Virtual Directory** tab.
3. Click the **Configuration...** button in the Application settings pane.

   The Application Configuration dialog opens.
4. Select the **Mappings** tab.
5. Click the **Insert...** button.

   The Add/Edit Application Extension Mapping dialog opens.
6. Enter the path name or browse for the `SFExtension.dll` file that you saved on your hard drive.

   If the path to the DLL file contains spaces then put quotes around it.
7. Be sure the **Verify that file exists** check box is checked and click **OK**.

   The newly added path should now be listed under the Wildcard application maps list.

   Ideally, you would only have the path to the `SFExtension.dll` in the Wildcard application maps list. If this is the case, then continue to the next step and ignore the rest of this step. Otherwise, if you have multiple entries in the Wildcard application maps list, you need to consider the following:

   - Do any of the other extensions perform token manipulation or impersonation? If so, then you run a high risk of breaking the code and failure in functionality in any or all of the extensions.
   - Do any of the other extensions rely on knowing the security context of the incoming user rather than simply providing Anonymous access? If so, then move the path to `SFExtension.dll` to be above any such other extensions, using the **Move Up** button.
8. Click **OK** in the **Application Configuration** dialog, then click **OK** in the **Properties** dialog.

Task 5: Set the IIS Extension file permission to run on the IIS console.

Perform the following steps to set the IIS Extension file permission to run on the IIS console:

1. Right-click on the Web Service Extension node in the left panel.
2. Select the **Add a new Web service extension...** option.

   The New Web Service Extension dialog opens.
3. Enter a value in the Extension name field.

   For example: **SFExtension**.
4. Check the **Set extension status to Allowed** check box.
5. Click on the **Add...** button.

   The Add file dialog opens.

6 Enter the path name or browse for the `SFExtension.dll` file that you saved on your hard drive.

7 Click **OK**.

The path displays in the **Required** files list in the **New Web Service Extension** dialog.

8 Click **OK**.

The extension should now be listed in the Web Service Extension list in the right panel with the **Status** set to **Allowed**.

The SF Extension is installed.

9 Reboot the machine after installing the SF Extension.

If you do not reboot the machine, the extension may not load properly.

Installing the Configuration Interface

Be sure you have installed the IIS Extension as described in Installing the IIS Extension on page 26.

Install the Configuration Interface

Perform the following steps to install the Configuration interface:

1 Double-click on the **install.cmd** executable in `\windows\iwi\conf\`.

The Open File - Security Warning dialog opens.

2 Click on the **Run** button and enter **`yes`** to any pop-ups.

Configure the Configuration Interface

Perform the following steps to configure the Configuration interface:

1 Open the IIS Console as follows:

- From the Start menu click **Run**.
- In the Open text box, enter **inetmgr** and click **OK**.

2 Select the virtual directory where you installed `SFExtension`.

3 Right-click and select **Properties**.

4 Select the **Select Federation Extension Configuration** tab.

The Select Federation Extension Configuration page opens with the Option and Value fields.

5 Configure each of the following options as described in the following subsections:

```
SFExtLogFile
SFExtDebug
SFExtErrorURL
```

Configure these options before you start the IIS server. Configuring while the IIS server is running will have no functional effect and will require a restart for the changes to take effect.

Configure SFExtLogFile

If you want SFExtension to provide logging, you need to configure the SFExtLogFile option. This option is not necessary to run SFExtension.

Perform the following steps to configure SFExtLogFile:

1 Select SFExtLogFile in the **Option** dropdown list.

2 Provide the path and the name of the log file in the **Value** field.

3 Click **Apply**.

> When specifying the path name, be sure you provide a valid path since Select Federation will only create the file you specify but not the path to the file.

Configure SFExtDebug

By default, debug logging is set to **off**. Configure SFExtDebug when you want to see extensive DEBUG logging to diagnose an issue.

SFExtension contains three levels of logging — ERROR, INFO and DEBUG.

- ERROR statements are always logged if a value for **SFExtLogFile** is specified in the SFExtension's Configuration interface.
- INFO statements are the default level of logging and are always logged if a value for SFExtLogFile is specified in the SFExtension's Configuration interface.
- DEBUG statements can be turned ON or OFF by specifying TRUE or FALSE for SFExtDebug in the SFExtension's Configuration interface. For this to work, the value for SFExtLogFile must be specified as well.

Perform the following steps to configure SFExtDebug:

1 Select **SFExtDebug** in the **Option** dropdown list.

2 Select **TRUE** or **FALSE** in the **Value** dropdown list.

3 Click **Apply**.

Configure SFExtErrorURL

By default, in the case of an error, Select Federation provides the end users with a hard-coded error message telling them to contact an Administrator. As an alternative, configuring the error URL allows you to provide your end users with the following:

- Intelligent information about what went wrong or steps they can take to reconcile.
- Pages that are localized.

Perform the following steps to configure SFExtErrorURL:

1 Select **SFExtErrorURL** in the **Option** dropdown list.

2 Provide the relative path to the error URL page in the **Value** field.

This is the error page you want users to see when they encounter an error.

For example, if users encounter an error when trying to access:

https://www.enterprise.com/protected/resource.asp

And you want them to see the error page:

https://www.enterprise.com/common/error.asp

You need to configure the Value as:

\common\error.asp

If the error URL points to a page in another Virtual Directory (VD), then those two VDs must share the same application pool. Otherwise your users will see an IIS-generated error page about unauthorized access rather than the error page you wanted the users to see.

3 Click **Apply**.

Uninstalling the IIS Extension

Perform the following steps to uninstall the IIS Extension:

1 Open the IIS console:

a From the **Start** menu click **Run**.

The Run dialog opens.

b In the **Open** text box, enter **inetmgr** and click **OK**.

The Internet Information Services (IIS) Manager dialog opens.

2 Select **Web Service Extensions** in the left panel.

3 Select the name you entered for the `SFExtension.dll` file in the right panel.

4 Click the **Delete** key.

5 Select any web site or web sites or virtual directories where you added mapping for the extension.

6 Right-click and select **Properties**.

The Properties dialog opens.

- If you right-clicked on a web site, then select the **Home Directory** tab.
- If you right-clicked on a virtual directory, then select the **Virtual Directory** tab.

7 Click the **Configuration...** button in the Application settings pane.

The Application Configuration dialog opens.

8 Select the **Mappings** tab.

9 Select the path to the `SFExtension.dll` file and click the **Remove** button.

10 Click **OK**.

11 Click **OK** in the **Application Configuration** dialog, then click **OK** in the **Properties** dialog.

Uninstalling the Configuration Interface

Perform the following steps to uninstall the Configuration interface:

1 Double-click on the **uninstall.cmd** executable in `\windows\iwi\conf\`.

The Open File - Security Warning dialog opens.

2 Click on the **Run** button and enter **yes** to any pop-ups.

Configuring IWI

This section describes how to configure IWI in the following topics:

- Configuring Select Federation to Perform Identity Mapping
- Configuration Parameter Descriptions

Configuring Select Federation to Perform Identity Mapping

Based on your mapping requirements, there are three ways to configure Select Federation to set up a unique identity mapping between incoming federated users and the users in your LDAP directory server:

- Configuring the Activate LDAP EventPlugin — successfully maps a user, if an attribute from the incoming federated user's profile can be used to locate a unique user in the directory server.
- Configuring the Activate URL EventPlugin — redirects the user to a configurable URL of your choice so that you may do the Identity Mapping and activate the user.
- Configuring the Activate LDAP and Activate URL EventPlugins — if a user cannot be mapped successfully by the Activate LDAP plugin, then the Activate URL plugin redirects the user to a configurable URL. At this URL, you can own the Identity Mapping and activate the user as you wish.

When using the Activate URL EventPlugin, the mapped account that is provided as the result of activation, must be properly formatted. For example:

username@domain.com

domain\username

All three ways require that you edit your SP's `$SF_HOME\conf\tfsconfig.properties` file. When you edit the `tfsconfig.properties` file, be sure to do the following:

1. Make a backup copy of the `tfsconfig.properties` file before editing it.
2. Edit the `tfsconfig.properties` file in the configuration directory of the application server — the directory in which the configuration files were copied.

The following sections provide detailed instructions on how to use and configure each EventPlugin with sample configuration models. The instructions include code snippets that you can copy and paste into your `tfsconfig.properties` file. For descriptions of each parameter, see Configuration Parameter Descriptions on page 36.

Configuring the Activate LDAP EventPlugin

You need to configure the Activate LDAP EventPlugin by providing values for the Activate LDAP EventPlugin parameters that make sense for your deployment.

Perform the following steps to configure the Activate LDAP EventPlugin to set up a unique Identity Mapping:

1. Add the following required parameters if they are not already in the `tfsconfig.properties` file:

```
#############
### REQUIRED
#############
```

```
##
## Your LDAP directory information
ldapURL=
ldapPrincipal=
ldapPassword=
ldapUserBaseDN=
```

2 Specify the required parameter values that make sense for your deployment.

3 (Optionally) copy and paste the following lines if they are not in the file and uncomment the parameters.

```
#############
### OPTIONAL
#############
##
## You do not need to specify the type of dirPlugin
## because SPEventPlugin_ActivateLDAP already assumes
## that it must use the DirPlugin_LDAP
## HPSF Plugin Configuration.
#dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_LDAP
##
## Set to "person" by default. You can change
## this to some other objectClass such as "user".
## You can use this to quicken the directory search.
#ldapUserObjectClass=person
```

4 Configure the plugin by copying and pasting the following lines into the `tfsconfig.properties` file, if they are not there:

```
#############
### REQUIRED
#############
##
## You do not want any random IDs.
filterSupport.spAutoGenerateLocalUserId=0
##
## Add the plugin as the first one to run
## in the (space-separated) list of the SP's Event Plugins.
spEventPlugin=com.trustgenix.tfsSP.util.SPEventPlugin_ActivateLDAP
com.hp.ov.selectfederation.filters.support.FilterSupportPlugin
##
## SF looks for the UPN in the LDAP once SF finds a unique
## identity mapping. Currently, only the UPN is supported for
## the windows inbound connector.
ldapUserAttr=userPrincipalName

#############
### OPTIONAL
#############
##
## The value for this flag can be: 0(FALSE) or 1(TRUE).
## It decides if the user should be successfully activated
## before proceeding forward.
## This flag should always be set to TRUE(1) for the
## last Activation Plugin in an Event Plugin chain.
## By default this is set to 1(TRUE)
#SPEventPlugin_ActivateLDAP.requireActivationSuccess=1
```

5 Perform additional configuration, depending on which particular profile attribute from the user's profile needs to map to the existing attributes in your Directory Server.

For example, if the value of the `personal_mail` attribute from the incoming federated users' profile needs to uniquely map to the value of the `userPrincipalName` attribute in your Active Directory Server, then you would edit the `tfsconfig.properties` file as follows:

a Add the following lines:

```
#############
### REQUIRED
#############
##
## The name of the attribute whose value should be
## picked up from the user's profile attributes.
SPEventPlugin_ActivateLDAP.userProfileAttr=personal_email
```

b Update the `personal_email.ldapAttr` required parameter with the attribute name in your LDAP directory that has a unique match for the value received from the user's profile attributes.

Change: `personal_email.ldapAttr=mail`

To: `personal_email.ldapAttr=userPrincipalName`

Configuring the Activate URL EventPlugin

Using the Activate URL EventPlugin, results in a redirect to a URL specified by you, thus hooking you into the workflow. This allows you to present and/or execute your own identity-mapping logic to the user.

Add Required Parameters

To configure the Activate URL EventPlugin, add the following required parameters if they are not already in the `tfsconfig.properties` file:

```
#############
### REQUIRED
#############
##
## You do not want any random IDs.
filterSupport.spAutoGenerateLocalUserId=0
##
## The URL that you wish to redirect to:
SPEventPlugin_ActivateURL.spActivateURL=
##
## Add the plugin as the first one to run
## in the (space-separated) list of the SP's Event Plugins.
spEventPlugin=com.trustgenix.tfsSP.util.SPEventPlugin_ActivateURL
com.hp.ov.selectfederation.filters.support.FilterSupportPlugin
```

Demo Activation Page for Testing

A demo activation page has been bundled for testing purposes. It shows how an activation page fulfills its responsibilities by mapping the user's identity. In this particular case the user is prompted to provide the user name and password. The user name can be in the `samAccountName` format or the user `PrincipalName` format.

Following is the sample configuration:

```
#######################
### DEMO PURPOSES ONLY
#######################
##
## You can set the following values at your SP for testing purposes.
## The activate-demo.jsp page and its dependant configuration
## settings are ***NOT MEANT FOR PRODUCTION USE***
SPEventPlugin_ActivateURL.spActivateURL=<providerId>/sf-demo/activate-demo.jsp
##
## The dirPlugin that will be used by
## the activate-demo.jsp page
dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_ADS
##
## Your Active Directory related information
ldapURL=
ldapPrincipal=
ldapPassword=
DirPlugin_ADS.useSAMAccNameOnly=0
##
## You do not want any random IDs.
filterSupport.spAutoGenerateLocalUserId=0
##
## Add the plugin as the first one to run
## in the (space-separated) list of the SP's Event Plugins.
spEventPlugin=com.trustgenix.tfsSP.util.SPEventPlugin_ActivateURL
com.hp.ov.selectfederation.filters.support.FilterSupportPlugin
```

Configuring the Activate LDAP and Activate URL EventPlugins

You can configure the plugins so that if the Activate LDAP EventPlugin was to fail to map the identity, the user would be redirected to a URL where you would have a workflow to address the identity-mapping.

Perform the following steps:

1. Follow all the instructions in Configuring the Activate LDAP EventPlugin on page 32.
2. Perform the following edits and additions to the `tfsconfig.properties` file:.

You may need to make further changes as required by the custom Activate URL page that you have developed for your own purposes.

```
#############
### REQUIRED
#############
##
## This flag should always be set to TRUE(1) for the
## last Activation Plugin in an Event Plugin chain.
## Since you need to get past the SPEventPlugin_ActivateLDAP
## plugin to use the SPEventPlugin_ActivateURL plugin
## as a fall back, you can ease the requirements for the
## SPEventPlugin_ActivateLDAP plugin.
SPEventPlugin_ActivateLDAP.requireActivationSuccess=0
##
## The URL that you wish to redirect to
SPEventPlugin_ActivateURL.spActivateURL=
##
## Add the plugin as the second one to run, in
## the (space-separated) list of the SP's Event Plugins
```

```
spEventPlugin=com.trustgenix.tfsSP.util.SPEventPlugin_ActivateLDAP
com.trustgenix.tfsSP.util.SPEventPlugin_ActivateURL
com.hp.ov.selectfederation.filters.support.FilterSupportPlugin
```

3 If you are configuring for testing purposes and you want to use a demo activation page for testing, then perform the following edits on top of what you have done so far:

```
######################
### DEMO PURPOSES ONLY
######################
##
## You can set the following values at your SP for testing purposes.
## The activate-demo.jsp page and its dependant configuration
## settings are ***NOT MEANT FOR PRODUCTION USE***
SPEventPlugin_ActivateURL.spActivateURL=<providerId>/sf-demo/
activate-demo.jsp
##
## The dirPlugin that will be used by
## the activate-demo.jsp page
dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_ADS
##
## Your Active Directory related information
ldapURL=
ldapPrincipal=
ldapPassword=
DirPlugin_ADS.useSAMAccNameOnly=0
##
## You do not want any random IDs.
filterSupport.spAutoGenerateLocalUserId=0
```

Configuration Parameter Descriptions

The following table lists the required and optional IWI connector parameters that can be used.

Table 1 IWI Connector Parameters

Name	Type	Default (if not reqd)	Description
`ldapURL`	String	Required	LDAP URL to use, by default, for connections to the directory.
`ldapPrincipal`	String	Required	LDAP user to use, by default, for connections to the directory.
`ldapPassword`	String	Required	LDAP password to use, by default, for connections to the directory.
`ldapUserBaseDN`	String	Required	Base DN to use in constructing user DN from user name and `ldapUserAttr`. User DN looks like the following: `<ldapUserAttr>=<username>,<ldapUserBaseDN>`.

Name	Type	Default (if not reqd)	Description
ldapUserObjectClass	String	person	ObjectClass used for performing lookups on the directory. You could change this to some other value, for example, user.
dirPlugin	String	null	DirPlugin implementation class. If non-null, the class will be instantiated and called to perform directory operations such as verifying passwords and fetching user profile attributes.
ldapUserAttr	String	Required	LDAP RDN user attribute to use for user name in constructing user DN.
SPEventPlugin	String	null	A list of one or more (space-separated) SPEventPlugin implementation class names. If non-null, the class will be instantiated and called for login and logout events. If installation is used to host applications that are to be protected by HP Select Access, this should be set to com.hp.ov.selectfederation.HPSA_SPEventPlugin.
SPEventPlugin_ActivateLDAP.requireActivationSuccess	String	1	The value for this flag can be: 0(FALSE) or 1(TRUE). It decides if the user should be successfully activated before proceeding forward. It is recommended to always set it to 1(TRUE) for the last Activation Plugin in an Event Plugin chain. By default this is set to 1(TRUE). This is in context of the SPEventlugin_ActivateLDAP plugin.
SPEventPlugin_ActivateLDAP.userProfileAttr	String	Required	The name of the attribute whose value should be picked up from the user's profile attributes. This is in context of the SPEventlugin_ActivateLDAP plugin.
SPEventPlugin_ActivateURL.requireActivationSuccess	String	1	The value for this flag can be: 0 (FALSE) or 1 (TRUE). It decides if the user should be successfully activated before proceeding. By default this is set to 1(TRUE). This is in context of the SPEventlugin_ActivateLDAP plugin.
SPEventPlugin_ActivateURL.spActivateURL	URL	Required	The URL to which you wish to redirect the user. This is in context of the SPEventlugin_ActivateLDAP plugin.

IWI Log Files

Select Federation includes two types of log files for IWI:

- IIS Filter log, which is a user-created log.

  You need to create a location on your system where you feel it is appropriate for the IIS filter to log its messages. For example:

  `c:\Program Files\Select Federation\logs\SFFilter.log`.

  You must specify the log file location since the log file is important for reconfirming the configuration used by the filter, understanding the filter functionality and gathering information to debug any issues.

  If you are using a virtualization software to test the filter, be aware that the filter is not able to place the file at the specified location every time. You need to know to which directory the filter can actually write the file.

- IIS Extension log, which needs to be configured if you want to log errors.

  You need to configure the `SFExtLogFile` option in the IIS console. See Configure SFExtLogFile on page 30 for details.

See Chapter 4, Error Messages for further details.

4 Error Messages

This chapter lists reported error messages.

IWI Error Messages

IWI reports the following types of error messages:

- Event Plugin Errors
- Directory Plugin Errors
- ISAPI Filter Level Errors
- ISAPI Extension Level Errors

Event Plugin Errors

Any Event Plugin Errors

Error Message

```
error locating module class:...
```

Problem

This could be due to any of the following issues:

- The module name was incorrect.
- If you have a custom module, the jar file may not be in your application server's classpath.
- If you listed multiple EventPlugins in the `tfsconfig.properties` file, they may not all be separated by spaces.

Solution

- Try specifying the fully qualified module name or names instead of just the module name. For example:

  Instead of: `spEventPlugin=SPEventModuleName`

  Try: `spEventPlugin=com.hp.full.package.info.SPEventModuleName`

- If you have a custom module, make sure that the jar file is in your application server's classpath. Or instead of putting it in the classpath, specify the location of the jar and the name of the class file using Select Federation syntax. For example:

  ```
  spEventPlugin=myImaginaryModule

  myImaginaryModule.class=com.my.company.ImaginaryMod
  ```

```
myImaginaryModule.jar=/path/to/myModule.jar
```

- If you have multiple EventPlugins, be sure your list of EventPlugins has a space between each EventPlugin.

Error Message

```
error loading config parameter: …
```

Problem

This could be due to any of the following issues:

- All the required parameters for the EventPlugins you are using may not have been specified in the `tfsconfig.properties` file.
- You might not have restarted the server after editing the `tfsconfig.properties` file.

Solution

- Make sure that all the required parameters for the EventPlugins you are using are specified in the `tfsconfig.properties` file.
- Be sure to restart the server after editing the `tfsconfig.properties` file.

SPEvenPlugin_ActivateLDAP Errors

Error Message

```
Could not map unique-identity based on userProfileAttr=… and
userProfileAttrValue=…
```

Problem

Select Federation could not find your user at all, or a unique user in the Active Directory.

Solution

- You may need to configure/populate your LDAP with the right set of attributes.
- You may need to adjust the values for your EventPlugin's configuration parameters.
- If you cannot find a solution, it is good practice to first turn on DEBUG logging and find out if there are any exceptions that point to the source of the problem.

SPEventPlugin_ActivateURL Errors

Error Message

```
"Error redirecting user: "…
```

Problem

You may not have configured the `spActivateURL` parameter correctly.

Solution

- Be sure you configure the `spActivateURL` parameter correctly.
- If you cannot find a solution, it is good practice to first turn on DEBUG logging and find out if there are any exceptions that point to the source of the problem.

Directory Plugin Errors

Error Message

```
Could not map unique-identity based on directoryAttrName=… and
userProfileAttrValue=…
```

Problem

This could be due to any of the following issues:

- The directory-related configuration parameters may have had the wrong values.
- The plugin failed to search the directory due to some exceptions.

Solution

- First, be sure that all the directory-related configuration parameters have the correct values.
- Look at the log files to find out what exceptions may have caused the plugin to fail to search the directory.

ISAPI Filter Level Errors

For troubleshooting information on filters, See “Troubleshooting Filters” in the “Troubleshooting” appendix of the *HP Select Federation Configuration and Administration Guide*.

ISAPI Extension Level Errors

Error

```
Could not allocate enough heap space.
```

Problem

Your machine does not have enough space.

A Troubleshooting

This appendix lists troubleshooting information.

Troubleshooting IWA

Use the Select Federation application server log file to view logged messages. This section lists the most common errors that would appear due to a broken workflow for IWA. Scenarios and suggested solutions are provided to help you find the root of the problem.

Error Message

```
msxml3.dll error '80072efd'
A connection with the server could not be established
```

Problem

The URL that you provided as the value for `fssURL` in your ASP pages is actually unreachable from the machine (on which your ASP pages are installed) due to network issues.

Solution

Make sure that you can actually reach the URL provided as the value for `fssURL` in your ASP pages. Make sure that the address for the `fssURL` can be resolved properly by the machine on which your ASP pages are installed.

Error Message

```
Unable to access Active Directory.

Please make sure that you have specified reasonable values for
activeDirectoryHost and that Active Directory service is running at '…'
```

Problem A

You forgot to configure the location of your Active Directory Server in ASPs

Solution

`activeDirectoryHost` - This should point to Active Directory Server, which the ASP pages can query to get a DN for the specified user.

Problem B

The location that you have provided is actually invalid or cannot be accessed from the given machine.

Solution

Try accessing the address that you provide by directly entering into the IE browser's address bar on the machine where your ASP file is being hosted. More than likely it will also not work, in which case you need to determine the correct location of your AD Server or at least how to get it from your machine.

Problem C

You provided an accurate location but wrote the letters of the protocol in all lowercase (`ldap://`) when specifying the location in the ASP file, rather than uppercase (`LDAP://`)

Solution

Try changing the characters that specify the LDAP protocol to all uppercase in your ASP file and see if it resolves the issue. For example, `LDAP://localhost:389`

Error Message

```
No response received from IDP-FSS.
```

```
Please make sure that you have specified reasonable values for fssURL and that
IDP-FSS service is running at '…'
```

Problem

You forgot to configure the location of your IDPFSS in the ASPs.

Solution

`fssURL` – This should be the URL used to make calls to the IDPFSS web service.

Error Message

```
msxml3.dll error '80072ee7'
```

```
The server name or address could not be resolved
```

Problem

Your IIS machine cannot resolve the host name of the SF-IDP machine provided in the fssURL.

Solution

Make sure that you can resolve the host name to the actual IP of the SF-IDP machine (at your IIS machine).

B Integrating Select Federation with Citrix

This appendix is for Citrix users who are integrating Select Federation IWI only with Citrix. This appendix provides integration information in the following sections:

- Requirements
- Integration Instructions
- Configuring Servers for Constrained Delegation
- Optimizing Citrix Deployments
- Troubleshooting Citrix

Since Citrix is an application, the steps given in this appendix (at least for Task 1) needs to be addressed before installing or configuring IWI.

Requirements

- Citrix versions: CWI 4.5 + CPS 4.5 with Hotfix Rollup 3.
- Citrix Web Interface machine must be a member of the same domain as the Citrix Presentation Server machine(s) (or a trusted domain).
- CWI and CPS machines must be configured for delegation.
- CPS must be version 4.5 with Hotfix Rollup Pack #2 or above.
- XML Service on the CPS machine must be hosted by IIS and must be defined in the CWI site configuration as a host name (NETBIOS), not an FQDN or an IP address.

Integration Instructions

Perform the following tasks to integrate Select Federation with Citrix.

Task 1: Configure federation-capable sites in Citrix Web Interface.

CWI 4.5 allows you to create a special site type that leverages Citrix's new advanced Kerberos support.

To create this site, perform the following steps:

1. Use the `sitemgr` command line tool that should be in the directory where you installed CWI.
2. Open a command prompt and create the site by including the `Federated=Yes` parameter in the site definition string that you will supply to `sitemgr`. For example:

```
sitemgr -c "WIDest=1:/your/virtual/
directory,Config=Local,XMLService=YourCPSMachineName:80,Federated=Yes"
```

For detailed instructions on using the `sitemgr` tool, see the Citrix documentation.

Once the site is created using `sitemgr`, you can manage it as usual with the Access Management Console.

Task 2: Configure the Domain functional level for delegation.

1. Log on to the machine that serves as your Domain Controller.
2. Select the domain name from the MMC Active Directory Users and Computers snap-in.
3. Click **Properties** on the **Action** menu.
4. If the domain is not at the Windows Server 2003 functional level, select the domain name and select **Raise domain functional level**.

   To raise the domain level, all domain controllers in the domain must be running Windows Server 2003.

Task 3: Configure delegation for the Citrix Web Interface machine(s)

XML Broker or XML Service is located on the CPS machine.

1. Log on to the machine that serves as your Domain Controller.
2. Make sure that **Advanced Features** is checked in the MMC **Active Directory Users and Computers** snap-in's **View** menu.
3. Select the machine that is running the Web Interface in the `Computers` folder under the domain name.

   The Web Interface must be a member of the domain, for you to do this.
4. Click **Properties** on the **Action** menu.
5. Select the **Delegation** tab and click **Trust this computer for delegation to specified services only** and **Use any authentication protocol.** Then click **Add**.

   Repeat the following process for each machine in the farm that is running the CPS XML Broker that the selected Web Interface is configured to contact.
6. Click the **Users or Computers** button on the **Add Services** screen.
7. On the **Select Users or Computers** screen, enter the name of the server running the CPS XML Broker in the **Enter the object names to select** text box and then click **OK**.
8. Select the http service type from the list and click **OK**.
9. Select the **Delegation** tab and verify that the http service type for the server running the XML Service appears in the **Services to which this account can present delegated credentials** list.
10. Click **OK**.

Task 4: Configure delegation for the Citrix Presentation Server machines

XML Broker or XML Service is located on the CPS machine.

1. Log on to the machine that serves as your Domain Controller.

2 Make sure that **Advanced Features** is checked in the MMC **Active Directory Users and Computers** snap-in's **View** menu.

Repeat the following process for each machine in the farm that is running the CPS XML Broker that the Web Interface is configured to contact

3 Select the machine that is running the CPS XML Broker, which the Web Interface is configured to contact, in the `Computers` folder under the domain name.

4 Click **Properties** on the **Action** menu.

5 Click **Trust this computer for delegation to specified services only** and **Use Kerberos only** on the **Delegation** tab. Then click **Add**.

If you are running CPS and CWI on the same machine (collapsed environment), you need to select **Use any authentication protocol** instead of **Use Kerberos Only**. This is not ideal because it leads to less security, but you need to do this for the collapsed environment that you decided to use.

6 Click the **Users or Computers** button on the **Add Services** screen.

7 On the **Select Users or Computers** screen, enter the name of the server running the XML Service in the **Enter the object names to select** text box and then click **OK**.

8 Select the **HOST** service type from the list and click **OK**.

9 Select the **Delegation** tab and verify that the **HOST** service type for the server running the XML Service appears in the **Services to which this account can present delegated credentials** list.

10 Click **OK**.

Task 5: Configure delegation for resources accessible from the Citrix Presentation Server machines.

1 Log on to the machine that serves as your Domain Controller

2 Make sure that **Advanced Features** is checked in the MMC **Active Directory Users and Computers** snap-in's **View** menu

If you are using multiserver farms, repeat the following procedure for each machine that is running CPS.

3 Select the machine that is running CPS, in the `Computers` folder under the domain name.

4 Click **Properties** on the **Action** menu.

5 Click **Trust this computer for delegation to specified services only** and **Use Kerberos only** on the **Delegation** tab. Then click **Add.**

If you are running CPS and CWI on the same machine (collapsed environment), you need to select **Use any authentication protocol** instead of **Use Kerberos Only**. This is not ideal because it leads to less security, but you need to do this for the collapsed environment that you are using.

6 Click the **Users or Computers** button on the **Add Services** screen.

7 On the **Select Users or Computers** screen, enter the name of the domain controller in the **Enter the object names to select** text box and then click **OK**.

8 Select the **cifs** and **ldap** service types from the list, for the domain controller and click **OK**.

If two choices appear for the ldap service, then select the one that matches the FQDN of your domain controller.

9 Select the **Delegation** tab and verify that the **cifs** and **ldap** service types for the domain controller appears in the **Services to which this account can present delegated credentials** list.

10 Click **OK**.

Configuring Servers for Constrained Delegation

For security reasons, you must configure all servers running the Presentation Server for constrained delegation. To provide users with access to resources on those servers you must add the relevant services to the Services list using the MMC **Active Directory Users and Computers** snap-in.

For example, to enable users to authenticate to a Web server on host `foo`, add the http service for server `foo`. To enable users to authenticate to an SQL server on host `bar`, add the MSSQLSvc service for server `bar`.

For more detailed information, see the "Service Principal Names and Delegation" white paper in the Citrix Knowledge Base.

Optimizing Citrix Deployments

Manually Configure the IIS Filter

After creating Citrix's federation capable site, you may have turned on the Filter protection by enabling the **Enable Select Federation Protection** check box on your virtual directory. However, you can save some processing power by manually configuring the IIS Filter.

Perform the following steps to manually configure the IIS Filter:

1 Open the IIS Console as follows:

a From the Start menu click **Run**.

b In the Open text box, enter **inetmgr** and click **OK**.

2 Select IIS →Server name (local computer) →Right click →**Properties**.

The Web Site Properties dialog opens.

3 Select the **Select Federation Filter - Site Configuration** tab.

4 Provide specific information in the **Protected directories** text box.

For example, if you had used the following command to install Citrix's federation-capable web interface:

```
sitemgr -c "WIDest=1:/Federated/
AccessPlatform,Config=Local,XMLService=YourCPSMachineName:80,Federated=Yes"
```

You can enter:

```
/Federated/AccessPlatform/auth/:;;SF-LocalUserId,SF-UserSessionId,;
```

This allows the Filter to focus on the `\Federated\AccessPlatform\auth\` directory rather that the complete `\Federated\AccessPlatform\` directory.

5 Click **Apply**.

Maintain Least-privilege Security Segregation with Multiple Citrix Sites

If you have two VDs deployed on your CWI's IIS server such as the following VDs, you may run them both under the same Application Pool (AP) by default:

- Generic Citrix VD
- Federation-Capable Citrix VD (see Integration Instructions on page 45)

Since `SFExtension` for the Federation-Capable Citrix VD requires elevated privileges, you can do the following to accomplish a least-privilege security segregation of the two VDs:

- Create a new AP with the old Citrix AP as the base.
- Elevate the new AP's privileges to Local System, and
- Configure the Federation-Capable Citrix VD to use the newly created AP which should have all the right configurations from the default Citrix AP, plus the right set of privileges for `SFExtension` to function.

Troubleshooting Citrix

Problem

A warning or error appears regarding this message:

```
Allow logon through terminal services. You don't get seamless SSO.
```

The machine might have been added to the domain after CPS was installed.

Solution

Do the following:

- Check who is in your **Remote Desktop Users** group. Since it is a local group, you should see it by going to your CPS machine and looking in **Computer Management** → **Local Users and Groups** → **Groups** ....
- You need to have **Domain Users** in this group. Add it if this entry does not already exist.
- Usually, the CPS installer adds the Domain Users entry to all CPS machine local groups.

Problem

In the event logs, you see the following:

```
A request for a launch ticket was denied since this server does not trust
requests sent to the XML service.
```

Solution

There is a link that needs to be secured with IPSec. The user must tell CPS that they have secured the link. Refer to the Citrix Knowledgebase.

To bypass the security link, perform the following steps:

1. Go to your CPS machine, and open the Access Management Console.
2. Expand **Citrix Resources** → **Presentation Server** → **<yourFarm>** → **Servers** → **<yourServer>**.
3. Right-click and select **Properties**.

4 Select **Xml Service** in the left pane.

5 Check the box **Trust requests sent to the XML Service** and click **OK.**

Glossary

Access Control

The authorization policies and conditions that regulate identity access to resources with a goal towards preventing unauthorized use or use in an unauthorized manner.

Access Management

The process of authentication and authorization.

Activation

Process of setting up mapping from a federated name identifier to a local user ID.

Active Directory Federation Services (ADFS) (WS-Federation 1.0)

A feature of Microsoft Windows 2003 Server R2, which allows a federation with Active Directory-based users, by using the WS-Federation 1.0 protocol.

Active Server Pages (ASP)

Microsoft pages, which log users in by invoking the IDP-FSS over a secure channel. See also Identity Provider Filter-Support Service (IDP-FSS).

ADFS

See Active Directory Federation Services (ADFS) (WS-Federation 1.0).

Administrator

An identity with full permission to manage Select Federation.

API

See Application Program Interface (API).

Application Helper

Select Federation component that helps you configure URLs in your application for seamless navigation to the Service Provider (SAML Consumer) sites or for authentication through the Identity Provider (SAML Producer) sites.

Application Program Interface (API)

An interface that enables programmatic access to an application.

Application Site Role

An Application Site (also called a Service Provider (SP) Site), which is a Trusted Partner site that participates in a federation to provide a service or application to common users and relies on an authority site to provide authoritative user authentication and other information. For example, in a federation of an extranet with partners' corporate portals, the site hosting the extranet is the Application Site.

Artifact Binding

Specifies that the browser should be redirected from the Authority Site (IDP) to the Application Site (SP) using a random string known as the "artifact" and that string should then be used by the SP over a SOAP call to retrieve the actual protocol message.

ASP

See Active Server Pages (ASP).

Attribute

One or more characteristics that are part of an identity profile. For each identity, an attribute has a corresponding value. For example, an attribute called "Department" may be assigned the values of, "IT", "Sales", or "Support". These attributes are interpreted and assigned appropriately to profiles in different applications (LDAP-compliant directories, databases, SAPs, and so on) based on the mapping rules defined for that application.

Authentication

The act of verifying the credentials of an identity and matching them with an identity profile. The evaluation of credentials ensures that the identity is truly who or what they claim to be.

Authority Site Role

An Authority Site (also called an Identity Provider (IDP) Site), which is a Trusted Partner site that participates in a federation to authenticate users and provide other authoritative user information to other sites. For example, in a federation of an extranet with partners' corporate portals, the portals act as the Authority Site.

Authorization

The process of defining and enforcing the entitlements of an identity. Checking whether the entitlements of an authenticated principal permit the principal to perform the requested operation. Authentication is a prerequisite for authorization. See Access Control and Authentication.

Bindings

Possible ways in which messages can be conveyed in the context of a browser-based user transaction between an Authority Site (IDP) and an Application Site (SP).

CA

Certificate Authority

CardSpace

An active client software protocol that manages the release of identity information to Service Providers (SP). Identity information is organized into "cards" on the end user's computer. Each computer contains a set of "claims" or identity attributes, such as name or email

address. Each time the user is required to authenticate to an SP, the user selects one of these cards, which determines the set of claims that will be sent.

Certificate Revocation Checking

Verifies the validity of certificates against a certification authority's published list of revoked certificates. Select Federation provides a simple means of enabling certificate revocation checking via Certificate Revocation Lists or CRLs.

Context

A Select Identity concept that defines a logical grouping of users that can access a Service.

CSR

Certificate Service Request

Delegated Administrator

An identity that has been added by the root administrator. The delegated administrator can perform all functions that the root administrator performs except admin-related functions such as add and remove admins and change admin passwords. When Select Federation is running in Standalone mode, the delegated administrator also cannot view the Admin Audit log. But when Select Federation is integrated with Select Access, then the delegated administrator can view the Admin Audit log. See Root Administrator.

DS

Discover Service

DST

(Data Services Template) DST-based services such as the Personal Profile service (ID-PP) and the Employee Profile service (ID-EP).

Edge Router

A Federation Router that is located at the edge of an enterprise where employees of that enterprise use applications offered by partners of the enterprise. Those applications request authentication of users (employees) of the Federation Router, and the Federation Router "routes" that authentication request to the appropriate departmental authority. See Federation Router.

Event

Federation activity such as **Logged In**, **Received Logout Request**, **Logged Out**, and so on. Select Federation logs server events (operational activities of enabled users) and administrator events (all the federated identity activities of each administrator).

Event Plugin Chain

A set of plugins that are called in order whenever an event occurs. A chain may contain one or more Event Plugins. See Event.

Federation

The combination of business and technology practices to enable identities to span systems, networks and domains in a secure and trustworthy fashion. This is analogous to how passports are used to assert our identity as we travel between countries.

Federation Router

A Select Federation installation that simplifies trust relationships between Authority Sites (IDPs) and Application Sites (SPs). The Federation Router acts as an intermediary for multiple organizational entities.

Filter-Support

A dedicated Java web application, which integrates Select Federation with the filters provided for the corresponding web servers: IIS, Apache 2.0 and Java Servlet Containers. Filter-Support also integrates Select Federation with web servers that cannot access the Select Federation databases, which are normally kept behind a firewall.

Filter-Support Service (FSS)

A servlet component that exposes Select Federation functionality to non-java applications, which can make web requests through xml messages. FSS exposes two main pieces of functionality: a) allowing trusted programs to inject a Windows-authenticated `user-id` into an IDP session, and b) allowing trusted programs to query for user attributes.

FSS

See Filter-Support Service (FSS).

GMT

See Greenwich Mean Time (GMT).

Greenwich Mean Time (GMT)

Standard time used throughout the world based on the mean solar time of the meridian of Greenwich. See Universal Coordinated Time (UTC).

Group

For Select Federation, a Group shares a common set of policies. All groups and partners within that Group inherit those policies. An administrator may override the Group setting for a particular partner within that Group.

Identity Mapping

The process of determining a local user ID against which to map an incoming federated name identifier. Two common techniques for identity mapping are either generating a random local user ID based on the federated name identifier or using any attributes available to determine a local user ID.

Identity Provider Filter-Support Service (IDP-FSS)

A servlet component of the Integrated Windows Authentication (IWA). The IDP-FSS enables a trusted program to add a Windows-authenticated user ID into an IDP session.

Identity Provider (IDP)

An Authority organization or web site that asserts the identity of users to the Service Providers or SPs in a federated network. The assertion of the user identity is done using standard protocols such as SAML and Liberty.

Identity Web Services Framework (ID-WSF)

Liberty Identity Web Services Framework security mechanism, which is a federated web service protocol. ID-WSF is used to build federated (identity-based) web services.

IDP

See Identity Provider (IDP).

IDP-FSS

See Identity Provider Filter-Support Service (IDP-FSS).

ID-WSF

See Identity Web Services Framework (ID-WSF).

IE

Internet Explorer

IIS

See Internet Information Server (IIS).

Impersonation Token

Any token that allows actions to be carried out on the user's behalf. For example, in Windows, tokens issued through Kerberos are often used for impersonating users. Various technologies running on Windows have APIs defined that take an impersonation token and apply them to threads and/or processes that can then leverage them for whatever actions they need to perform on behalf of the users.

Inbound Windows Integration (IWI)

Inbound-integration that seamlessly integrates federated users at a Select Federation Application (SP) site to applications hosted on the Windows environment.

Integrated Windows Authentication (IWA)

Outbound integration that allows Select Federation to leverage a user's Windows logon credentials to seamlessly authenticate the user and transfer the user to a Trusted Federation Partner site.

Internet Information Server (IIS)

The web server that is bundled with the Windows 2003 Server.

IWA

See Integrated Windows Authentication (IWA).

IWI

See Inbound Windows Integration (IWI).

JAVA

Object-oriented programming language.

JVM

Java Virtual Machine. A platform independent execution environment that converts Java bytecore into machine language then executes it.

Keystore

A database of keys. The private keys are associated with a certificate chain, which authenticates the corresponding public key. The keystore also contains certificates from trusted entities. By generating the keystore, you add another layer of security to the data that is exchanged in the Select Federation system.

LDAP

See Lightweight Directory Access Protocol (LDAP).

LECP

Liberty Enabled Client/Proxy Service.

Liberty Identity-based Web Services Framework (ID-WSF)

A protocol that provides standards for discovering and invoking identity-based web services.

Liberty Identity Federation Framework (ID-FF)

An open standard federation standard protocol that provides basic single sign-on capabilities.

Lightweight Directory Access Protocol (LDAP)

A set of open protocols for accessing information directories. LDAP can make the physical network topology and protocols transparent so that a network identity can access any resource without knowing where or how it is physically connected.

Metadata

Online exact description of a Trusted Partner site in a federation. The metadata describes the various URLs at which its site services (such as Single Sign-On, Single Logout) are available. It also describes the public key certificates so that sites receiving messages from these Trusted Partner sites can confirm that the messages are signed correctly and have not been tampered with. See Single Sign-On (SSO) and Single Logout (SLO).

Microsoft Management Console (MMC)

MMC is used to set up server authentication and to import the `pkcs` / `pfx` format file into your local store on the IIS machine.

MMC

See Microsoft Management Console (MMC).

NTLM (NT LAN Manager)

Default network authentication protocol for Windows NT 4.0.

OCSP

See Online Certificate Status Protocol (OCSP).

Online Certificate Status Protocol (OCSP)

OCSP support exists in JDK 1.5. OCSP support is available for the Built-in application server (Tomcat 5.5.23) and WebLogic 8.1 and 9.1.

Partner

For Select Federation, the main entity in a federation trust relationship. A partner is described in terms of its protocol metadata, various descriptive attributes, and policy information. Select Federation allows partners to be grouped together in "Groups."

Passive URLs

Passive URLs are for resources where users' personalized content is not critical for the application. Users are allowed to access these URLs even though they cannot be authenticated without being prompted. However, if the user is already logged in at the IDP, has a federation session with Select Federation, or can be authenticated without being prompted, the user's identity and attribute information is presented in the federation session to the application.

PDC

Primary Domain Controller

Plugin

Compiled code that can interact with the core product to provide additional functionality, without replacing parts of the core product. In the context of Select Federation, the "compiled code" can be thought of as Java compiled code that is packaged in JARs and the "core product" can be thought of as any Select Federation install.

POST Binding

Specifies that the protocol message is to be delivered to an SP from an IDP through an auto-posted HTML form.

Presence Service

A service that informs the WSC if a user is online, available, and so on. See Web Service Consumer (WSC).

Privacy Manager

End-user visible component of Select Federation. Its visibility allows extensive customizing.

Protected URLs

Protected URLs require users to be authenticated to allow access to these URLs. If a user is not authenticated, the filter redirects the user to Select Federation for authentication. The Select Federation installation may authenticate the user locally or initiate federated login at another Authority (IDP).

Protocol

A set of rules that controls or enables communication between two endpoints. In the context of Select Federation, an endpoint is software that is capable of using any one of the many protocols that Select Federation supports.

Root Administrator

The “super user” administrator who has complete entitlement to all functionality in the Select Federation Administration Console. The root administrator’s login is always **`admin`**. Only the root administrator can add and remove delegated administrators and change administrators’ passwords.

SAML

Security Assertion Markup Language open standard federation protocol. Identity federation standard that was created by the OASIS Security Services Technical Committee (SSTC).

Secure Sockets Layer (SSL)

A handshake protocol, which supports server and client authentication.

Service Provider (SP)

An application that allows authenticated access based on an authentication performed by an IDP using a federated identity protocol such as Liberty or SAML.

Single Logout (SLO)

Permits a user to do a global log out from all active sites.

Single Sign-On (SSO)

Session/authentication process that permits a user to enter one set of credentials (such as name/password, secureId, fingerprint, and so on) to access multiple applications. A Web SSO is a specialized SSO system for web applications.

Site Role

Type of web site in a federation. Typically, you and your Trusted Partner agree in advance on how to set up the federation. Generally, one site hosts the application, while the other provides the authentication for end users to seamlessly access the application. When you deploy Select Federation in your site, you must set the site role as one of the following: (1) an Authority Site, (2) an Application Site, (3) both an Authority and Application Site, or (4) a Federation Router. See also Service Provider (SP), Identity Provider (IDP), and Federation Router.

SOAP

Simple Object Access Protocol is a fundamental web services standard for XML-based communication between web service providers and consumers.

SP

See Service Provider (SP).

SSC

Self Signed Certificate

SSL

See Secure Sockets Layer (SSL).

SSO

See Single Sign-On (SSO).

TLS

Transport Layer Security

Universal Coordinated Time (UTC)

Standard time used throughout the world based on the mean solar time of the meridian of Greenwich. Formerly known as Greenwich Mean Time (GMT).

Unprotected URLs

Unprotected URLs allow users access to these URLs without being authenticated. Typically, special URLs such as the login URL and logout URL are unprotected URLs.

UPN

User Principal Name

UTC

See Universal Coordinated Time (UTC).

Web Service Consumer (WSC)

An application that uses web services. It may not be a web service in itself, but uses XML and typically SOAP-based communication with a web service to perform some of its functions.

Web Service Provider (WSP)

A web service application that services requests it receives based on XML and typically SOAP-based communication.

WSC

See Web Service Consumer (WSC).

WSP

See Web Service Provider (WSP).

Index

D

E

F

G

I

K

L

M

O

P

R

S

T

U

W


invent


Java™
COMPATIBLE