HP OpenView Select Federation

For the HP-UX, Linux, Solaris and Windows® Operating Systems

Software Version: 6.60

Configuration and Administration Guide

Document Release Date: November 2006

Software Release Date: November 2006

Legal Notices

Warranty

Restricted Rights Legend

Copyright Notices

HP OpenView 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 University Corporation for Advanced Internet Development <http://www.ucaid.edu>Internet2 Project.

Trademark Notices

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

Please visit the HP OpenView support web site at:

http://www.hp.com/managementsoftware/support

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter 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 log in. Many also require a support contract.

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

http://www.managementsoftware.hp.com/passport-registration.html

Contents

1 Introduction

This *HP OpenView Select Federation Configuration and Administration Guide* describes how to configure HP OpenView Select Federation, and how to perform basic administration tasks once Select Federation has been installed.

This chapter provides a brief overview of the capabilities of Select Federation. The operating and configuration instructions are given in their respective chapters, which also provide detailed descriptions of Select Federation's various functions.

Prerequisites

This guide assumes a general knowledge about installation and configuration of web servers, databases, and so on for the target operating environment.

This guide also assumes a working knowledge of:

- Identity Management
- Federated Identity
- HP OpenView Select Access

What Does Select Federation Do?

Federated Identity or Identity Federation is a new approach to solving the single sign-on problem through a secure exchange of identity information among cooperating organizations, whether within a company or between companies using open standards. Select Federation helps companies to achieve cross-domain single sign on quickly and easily.

Users typically have a web account that they use regularly such as their corporate account. They also have many independent accounts at one or more web sites that they use less frequently. Once these accounts are federated, users can access all the federated web sites through their most frequently used account without having to log in each time.

Built on the latest federated identity standards, Select Federation does not require any radical changes to the existing technology infrastructure. It provides a de-centralized approach to cross-domain single sign-on, provisioning and privilege management across identity domains.

If required, Select Federation can be used in Standalone mode or together with HP OpenView Select Access. With Select Access, Select Federation adds standards-based cross-domain single sign-on capabilities to Select Access, the HP OpenView product for centralized access management.

Setting Your Site Role When Deploying Select Federation

When you deploy Select Federation at your site, you need to set your site to one of the following site roles:

- Authority site
- Application site
- Both an Authority and Application site

Typically, you and your Trusted Partner agree in advance how the federation is to be set up. One site hosts the application, while the other provides the authentication so that the end users can seamlessly access the application.

The following sections describe each type of site role.

Authority Site

An authority site (also called a SAML Producer or Identity Provider (IDP) Site) 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 a partners' corporate portals, the portals act as the Authority site.

Application Site

An application site (also called a SAML Consumer or Service Provider (SP) Site) 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. In the extranet example, the site hosting the extranet is the Application site.

Both Authority Site and Application Site

A single Select Federation instance can handle both the Application site and Authority site roles. For example, you may host an extranet for your partners' employees to access, in which case you are the Application site. However, your partners may also host applications that require your employees to authenticate at your site, in which case you are the Authority site.

What are the Select Federation's Key Features?

Select Federation is a J2EE-based server that can run on top of many J2EE servlet engines such as BEA WebLogic and IBM WebSphere. Select Federation also includes its own built-in application server.

Select Federation includes the following features:

- **Comprehensive federation features** including:
  - **Single Sign-On**: Provides seamless navigation to common applications
  - **Provisioning**: Provides instant activation of users at common applications
  - **Coarse-Grained Privilege Management**: With this feature, you can set the LDAP directory to control which end users have access to which common applications

- **Termination**: Also known as single logout, with this feature users terminated in home domain lose access to all the common applications

- **Multi-protocol support**. As detailed in the next section "Which Open Federation Standards Does Select Federation Support?," Select Federation supports all of the popular federation protocols, including SAML 2.0, Liberty Alliance ID-FF 1.1, Liberty Alliance ID-FF 1.2, Liberty Alliance ID-WSF, SAML 1.0, and SAML1.1.

- **Easy integration with LDAP directories:** Select Federation readily integrates with an LDAP directory for obtaining user-profile and authentication information. Using an LDAP directory is simply a matter of configuration, no code required. Select Federation connects to the LDAP directory through the HP Select Access Adapter. The Select Access Adaptor is a component of Select Federation that connects Select Federation with the Select Access Policy and your LDAP repository (see Figure 3 on page 24) to see how the Select Access Adaptor connects Select Federation with Select Access. Chapter 2, Select Federation Architecture has more details on the Select Access Adaptor.

- **Scalability and reliability:** Designed to be deployed on multiple servers, Select Federation can scale to handle large transaction loads.

How Do Select Federation's Features Work?

In a typical federation environment, multiple sites are seamlessly connected. It is important that sites share their online descriptions or *metadata* with each other. This ensures that a site knows where to send messages to another site in its federation, or a site knows that a message that it has received is guaranteed to be from a site that it trusts and has not been tampered with.

Metadata

Metadata in a federation is a description of the Trusted Partner site with which you want to federate. It is an online exact description of a site in a federation. Metadata describes the various URLs at which different site services (such as single sign-on and 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 Setting Up Partnerships on page 29 for more information on metadata and how to exchange metadata with your partners.

In the SAML 1.0, SAML 1.1, and Liberty ID-FF 1.1 protocols, the metadata specifications are either informal specifications or just a convention in the community about how to define the meta-data. Under the Liberty ID-FF 1.2 federation protocol, the metadata specification is formalized, and metadata using this protocol is conforming and has been certified under interoperability testing.

In Select Federation, site configuration is done using the Administration console. The Administration console enables an administrator to publish the site's metadata as well as import other sites' metadata. Select Federation simplified the process of getting your metadata for all the popular federation protocols. With one click, you can download your site information into any of the needed federation protocol formats. Alternatively, if your partner prefers the information in text format, all information is readily available on the web page so that you can copy and paste the text.

User Provisioning/Activation

Select Federation supports user-provisioning or “activation” across identity domains. The first time a new user accesses the Application Site, the user attributes that are needed by the Application Site are requested from the Authority Site to automatically activate this new user account.

The following steps show how user provisioning works, assuming both the Authority and Application sites use Select Federation:

1. A new user logs in to the user’s local “employee portal” or other internal application that enables the user to access a federated partner site for the first time.
2. Select Access at the user’s home site uses the local Identity Management System or LDAP repository to authenticate the user.
3. When the user clicks the external link, the user is redirected to the local instance of Select Federation. This generates a SAML or Liberty assertion for the user with a new “federated-identifier” – an opaque large number that represents the user at the partner site. The Select Federation at the local site then redirects the user to the Select Federation at the external Application Site and transmits the assertion message.
4. The Select Federation at the partner site receives the assertion message and obtains the “federated identifier” from the assertion message. If Select Federation does not recognize the federated identifier, it knows that the user is coming to the site for the first time and thus requires provisioning or activation.
5. Select Federation then triggers the request for the activation profile from the user’s home Select Federation.
6. The home Select Federation queries the LDAP repository for the user’s profile information.
7. The home Select Federation provides the required profile attributes to the requesting Select Federation (after verifying that the site policy allows such disclosure).
8. The Select Federation at the partner site then populates a new entry with all the profile attributes in the partner’s LDAP repository that is known to the Select Access installation there. It automatically assigns a local user ID for this new user.

Single Sign-On

Similar to the provisioning and activation feature, single sign-on works as follows:

1. The user clicks a link at an Application Site or an Authority Site and is redirected to the Select Federation instance at the Authority site. Such links can be generated using the Application Helper as described in Using the Application Helper on page 77.
2. To authenticate the user, Select Federation redirects the user to the Select Access Adapter. The Select Access Adapter provides the authenticated user information to Select Federation and redirects the user back to Select Federation.
3. Select Federation then generates a SAML or Liberty assertion for the user and redirects the user to the destination Application Site.
4. The Select Federation at the Application Site receives and verifies the assertion. It then identifies the local user.
5. Using the Select Access Adapter, the Select Federation at the Application Site sets a temporary cookie in the browser. The cookie allows the user to seamlessly navigate to any site protected by that Application Site’s Select Access instance.

6 The Select Federation at the Application Site then redirects the user to the final destination URL to which the user initially intended to go.

Single Logout or Termination

The user may click a “global logout” or “single logout” button at an application to indicate that the user wishes to logout from all active sites. Select Federation provides a “single logout URL” to which the application can redirect the user to do a global logout.

Select Federation redirects the user to the user’s Authority Site with a signed logout request. The Select Federation at the Authority Site keeps track of all the sites at which the user is logged in and sends signed messages to all such Application Sites, thus enabling a global Logout.

Name Federation Policy

Select Federation allows users to connect to Trusted Partner web sites in three ways: using the user’s local name which has identifiable user information, using a unique identifier that does not reveal the users’ identities to outside sites, or total anonymity. This feature is called the Name Federation Policy. Regardless of which federation protocol is being used between two sites, the authority site can determine a name-federation policy. This policy can be one of three values:

- **Local names:** The local user IDs at the authority site are revealed unmodified to the partner site. This is typically useful when two internal sites are using Select Federation to enable single sign-on between them.
- **Pseudonyms:** These are also identifiers or tokens that are generated to keep the user’s local identity unknown to the Service Provider. Select Federation automatically generates an opaque large random number mapping for a local user ID. Note that this pseudonym is unique to each partner site that the user is federated with. Thus, unlike the One Time Pseudonym, each time the user goes to the Trusted Partner site, the same identifier is presented. The Service Provider or Application site will know that this user’s activity at its site. This is an useful name ID policy in a typical business-to-business (B2B) scenario.
- **One-time pseudonyms:** These are anonymous identifiers or tokens (also an opaque large random number) that is generated each time the user accesses a Trusted site. Every time the same user visits the same partner site, Select Federation will generate a new pseudonym for the user. This provides complete anonymity for the user at the partner site and is useful in business-to-consumer (B2C) scenarios.

Attribute Exchange

Select Federation provides extensive support for exchanging personal information (user attributes) between Trusted Partner sites. Applications typically need attributes about the authenticated users. In a federated system, the most recent values for these user attributes are at the original source of the authentication, such as the Identity Provider or SAML Producer.

The administrator at the authority site can choose to allow, on a per partner site basis, certain attributes to be pushed along with an authentication assertion (SAML or Liberty) and certain attributes to be queried by a particular partner site. Attributes are configured in the Select Federation properties file and are fetched on every user authentication. Select Federation can further be configured to “push” attributes on every outbound user authentication when working as a SAML producer or IDP, further saving the overhead in fetching attributes about the user. See Chapter 7, Configuring Attributes for information on how to configure attribute exchange.

Privacy Manager

The Privacy Manager is available with Select Federation. It is the only component of Select Federation that is end user visible. Therefore, it also has extensive customizing abilities. The Privacy Manager resides at the relative URL: `/pm` within the Select Federation deployment. See Chapter 9, Configuring Privacy Manager for more information.

The Privacy Manager consists of two parts:

- Preference Setting service (`privacy`) — This is available only in HTML and is used like a regular application by the user to set the privacy preferences before any information about that user has been exchanged.
- Interaction/Consent service (`irs`) — This is available in HTML and WML and is invoked when the user has set a preference that the user should be asked before information about the user is disclosed to a partner site and the transaction that the user is undergoing results in the need for such disclosure.

When using the SAML protocols (2.0, 1.1 or 1.0), the Interaction / Consent screen is invoked only on attributes that are "pushed" with the authentication assertion. Since the SAML 1.1 or 1.0 protocols do not specify a mechanism to handle interactions during attribute queries, such queries fail if the user sets the preferences that the user needs to consent such information.

When using the Liberty protocols (1.1 or 1.2 and ID-WSF), the Interaction / Consent screen may be invoked either during an attribute "push" within the Liberty Authentication Response message or during a Liberty Profile Service query.

The Interaction / Consent service displays the details of the information that is about to be provided to the partner site.

Artifact Pickup Security Mechanisms

Select Federation provides extensive support for various security mechanisms between Trusted Partner sites for picking up SAML artifacts. The mechanisms supported by Select Federation are:

- **Signature:** The site requesting an artifact should digitally sign the request.
- **SSL/TLS Client Authentication:** The site requesting the artifact is required to provide a digital certificate to successfully complete an SSL or TLS client Authentication handshake before the artifact may be disclosed.
- **HTTP Basic Authentication:** The site requesting the artifact needs to authenticate with a user name and password. This option is provided mainly to support federation products that do not support the other two security mechanisms.

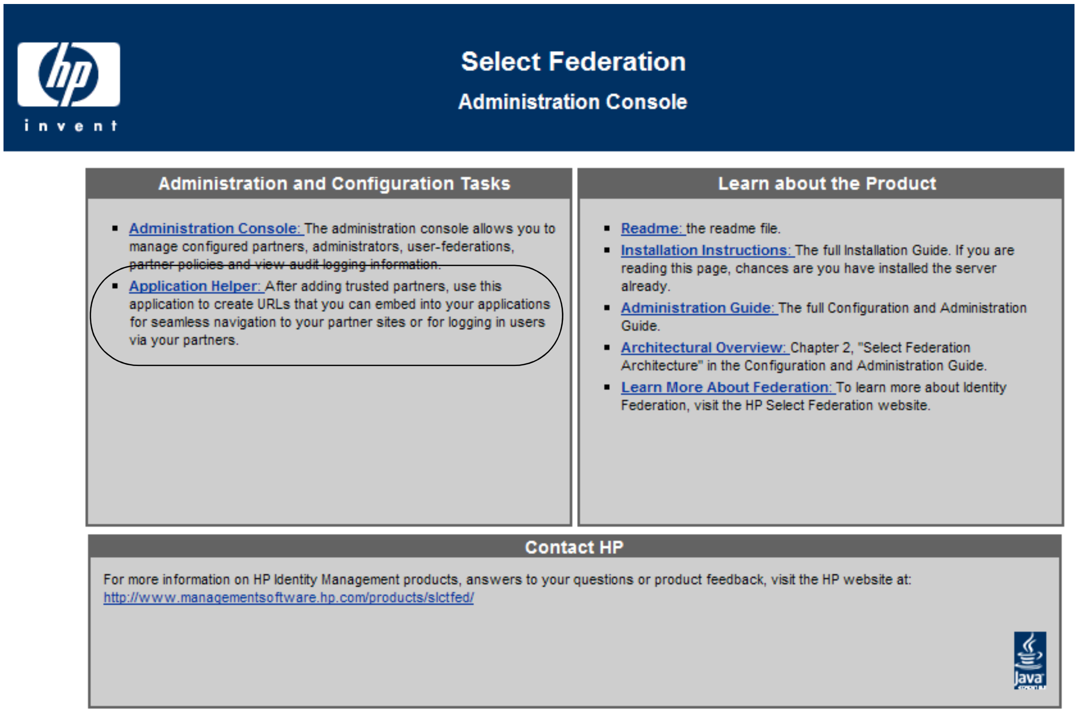
Additional Utilities for Application Integration and Testing

Select Federation provides two additional utilities for easing application integration and testing. They are the Application Helper and Demo Application. See Chapter 6, Enabling Applications for descriptions of both utilities.

Application Helper

The Application Helper is designed to enable web site administrators to obtain URLs that enable seamless navigation between federated sites or that enable an administrator to request federated login from a select Identity Provider or Authority Site. The Application Helper can be found on the Select Federation Landing page, as shown in Figure 1. (See Using the Application Helper on page 77 for more information.)

Figure 1 Application Helper Link on the Select Federation Landing Page



Select Federation

Administration Console

Administration and Configuration Tasks

- Administration Console: The administration console allows you to manage configured partners, administrators, user-federations, partner policies and view audit logging information.
- Application Helper: After adding trusted partners, use this application to create URLs that you can embed into your applications for seamless navigation to your partner sites or for logging in users via your partners.

Learn about the Product

- Readme: the readme file.
- Installation Instructions: The full Installation Guide. If you are reading this page, chances are you have installed the server already.
- Administration Guide: The full Configuration and Administration Guide.
- Architectural Overview: Chapter 2, "Select Federation Architecture" in the Configuration and Administration Guide.
- Learn More About Federation: To learn more about Identity Federation, visit the HP Select Federation website.

Contact HP

For more information on HP Identity Management products, answers to your questions or product feedback, visit the HP website at: http://www.managementsoftware.hp.com/products/slctfed/

Demo Application

The Demo Application is a regular J2EE application that uses the Select Access Servlet Enforcer. It demonstrates that normal Select Access applications can benefit from the seamless federated single sign-on and other capabilities provided by Select Federation. The Demo Application is bundled with Select Federation and can be found at `/sf-demo`. (See Using the Demonstration Program on page 79 for more information.)

Which Open Federation Standards Does Select Federation Support?

Select Federation is one of the most comprehensive federation protocol solutions. It supports multiple federated identity protocol standards which provides flexibility when connecting with multiple Trusted Partners with multiple federated identity protocol. Select Federation simultaneously supports different Trusted Partners in a federation that may be communicating using multiple federation protocols.

Select Federation supports all existing Liberty Alliance and Security Assertion Markup Language (SAML) protocols, including the following popular federation protocols:

- SAML 2.0
- Liberty Identity Federation Framework (ID-FF) 1.1
- Liberty Identity Federation Framework (ID-FF) 1.2
- Liberty Identity-based Web Services Framework (ID-WSF) 1.0 and (ID-WSF) 1.1
- SAML 1.0, SAML 1.1

SAML 2.0

SAML 2.0 is an identity federation standard that was created by the OASIS Security Services Technical Committee (SSTC). It represents the convergence of the Liberty ID-FF 1.2 standard, and the prior SAML 1.x standards. Select Federation has been certified interoperable by the Liberty Alliance for SAML 2.0. SAML 2.0 specifies a metadata format for SAML 2.0 protocols that is supported by Select Federation. The metadata specification is a conformant part of interoperability certification. The specification is available at:

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security#samlv20

Liberty Identity Federation Framework (ID-FF) 1.1

The first version of the Liberty Identity Federation Framework (ID-FF 1.1) provides basic single sign-on capabilities. Select Federation is a Liberty ID-FF 1.1 certified interoperable product, and hence has support for all features of Liberty ID-FF 1.1 specified by the interoperability specification, available at:

http://www.projectliberty.org/specs/liberty-idff-1.1-scr-v1.0.pdf

Liberty Identity Federation Framework (ID-FF) 1.2

The second version of the Liberty Identity Federation Framework (ID-FF 1.2), coupled with the Liberty Identity-based Web Services Framework (ID-WSF), extends the standards into identity-based web-services capabilities. Select Federation supports all capabilities of Liberty ID-FF 1.2 that are included in Liberty ID-FF 1.1. Select Federation also supports all of the new features, including the new meta-data format, publishing meta-data at URLs and so on.

Liberty Identity-based Web Services Framework (ID-WSF) 1.0

ID-WSF 1.0 provides the standards for discovering and invoking identity based web services. Identity Service Interface Specifications is a set of standard service interfaces specified to provide commonly required services. In the Liberty specifications, this is limited to a “personal profile” service and “employee profile” service. Select Federation provides a Liberty ID-WSF 1.0 compatible Discovery Service and a configurable number of Data Services Template (DST)-based services such as the Personal Profile service (ID-PP) and the Employee Profile service (ID-EP). In the ID-WSF security mechanisms, Select Federation supports the null, clientTLS and X509 security mechanisms.

SAML 1.0 and SAML 1.1

The SAML specification is an XML framework for exchanging authentication and authorization information. Select Federation provides comprehensive support for both the SAML 1.0 and SAML 1.1 standards, including the ability to create and consume signed SAML authentication and attribute assertions. Select Federation provides a SAML Authentication Authority and an Attribute Authority.

2 Select Federation Architecture

Select Federation was architected to support all the federation protocols, with scaleability and reliability in mind. Select Federation is a complete, easy-to-install federation solution that integrates seamlessly with your existing Select Access deployment.

What are the Components of Select Federation's Architecture?

Select Federation depends upon Select Access for authentication. Figure 3 provides an overview of Select Federation's architecture and details how its components integrate with Select Access through the Select Access Policy. The Select Access Policy is created using Select Access Policy Builder. For more information on creating the Select Access Policy, see "Authorize Entitlements with Access Policies" in the *Select Federation Installation Guide*.

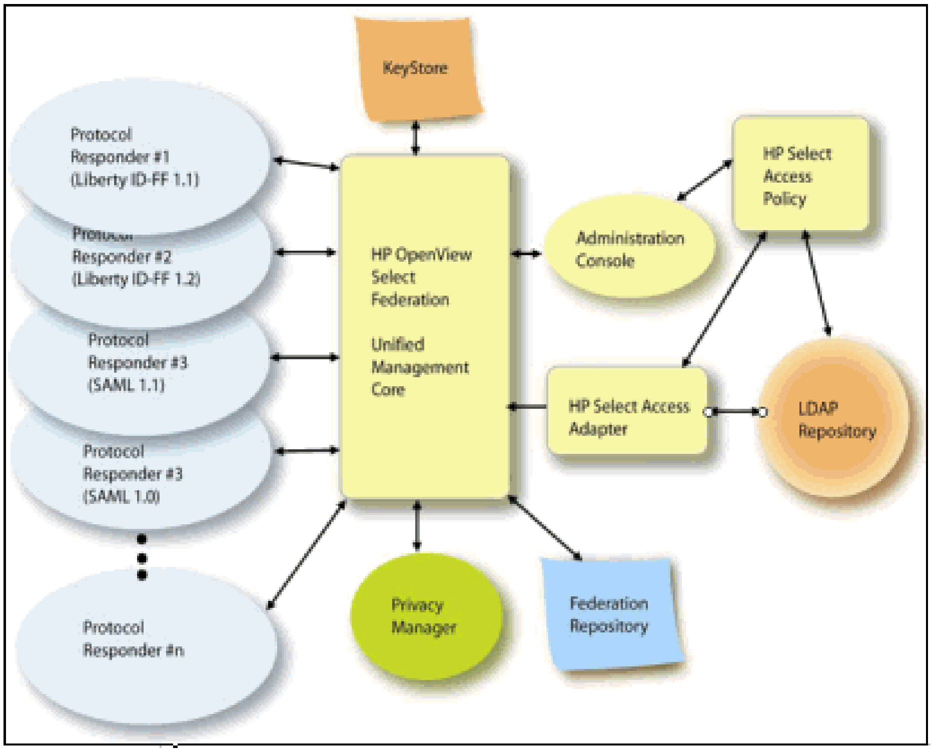
Select Federation is comprised of these key components which are detailed in this chapter:

- Protocol Responders
- Unified Federation Management Core
- Select Access Adapter
- Administration Console
- KeyStore
- Privacy Manager

Select Federation also connects to third-party software (an open source federation repository is bundled with Select Federation) for these components:

- Federation Repository
- LDAP Repository

Figure 2 Select Federation Architecture and How it Integrates with Select Access


KeyStore
Protocol Responder #1 (Liberty ID-FF 1.1)
Protocol Responder #2 (Liberty ID-FF 1.2)
Protocol Responder #3 (SAML 1.1)
Protocol Responder #3 (SAML 1.0)
Protocol Responder #n
HP OpenView Select Federation
Unified Management Core
Administration Console
HP Select Access Policy
HP Select Access Adapter
LDAP Repository
Privacy Manager
Federation Repository

Protocol Responders

Protocol responders are J2EE servlets that receive messages either on the front-channel (from a user-agent such as a browser) or on the back-channel (such as a SOAP message from another Select Federation server). The front-channel and back-channel servlets are packaged in separate “web-archives” or WARs so that they may be deployed on different servers or ports with independent firewall configurations. These protocol responders represent:

- **Front-channel URLs:** Examples of front-channel URLs include the SAML Single Sign-On Service URL or the Liberty Assertion Consumer Service URL. These URLs are available and are advertised in the meta-data.
- **Back-channel Web-Services:** Examples of back-channel Web-services include the Liberty Profile Service or the SAML Attribute Authority Service. These URLs may be advertised through the meta-data (as in the case of the Liberty 1.1 SOAP service), or through the Liberty Discovery Service which is also a part of the protocol responders.

Unified Federation Management Core

The Select Federation Unified Federation Management Core provides the basic infrastructure for Select Federation to work. It manages the user-federation-session information, federation mappings of user identities, circle-of-trust information of your Trusted Partner sites, and audit events. It uses the federation repository to store all this information.

Federation Repository

The federation repository is a relational database used by the Unified Federation Management Core to manage all internal data required for federation. Select Federation supports Oracle 9i and Apache Derby database servers. Apache Derby is bundled with Select Federation so there is no need to install it separately.

Select Access Adapter

The Select Access Adapter is a component of Select Federation that connects Select Federation with Select Access and an LDAP v3 compatible directory to provide integrated federated user authentication and user profile information. The Select Access Adapter obtains information about the location of the LDAP directory using a configuration file that is generated in the installation process. This configuration file can be manually edited for subsequent changes. The LDAP directory is referenced for profile attributes of the user as well as verifying membership for privileged access to external applications.

Administration Console

Select Federation provides an Administration Console that allows the root administrator to add additional administrators to Select Federation and to monitor the activities of all administrators and enabled end users. The Administration Console is a web front-end that connects into Select Access. It allows the administrator to monitor existing federations with Trusted Partners including capabilities such as defining trusted sites, manually deleting user federations, and viewing audit log.

KeyStore

The keystore is a Java Cryptography Architecture-compliant key store that can be realized in software or hardware. Select Federation requires the keystore to be co-resident with it on the same application server.

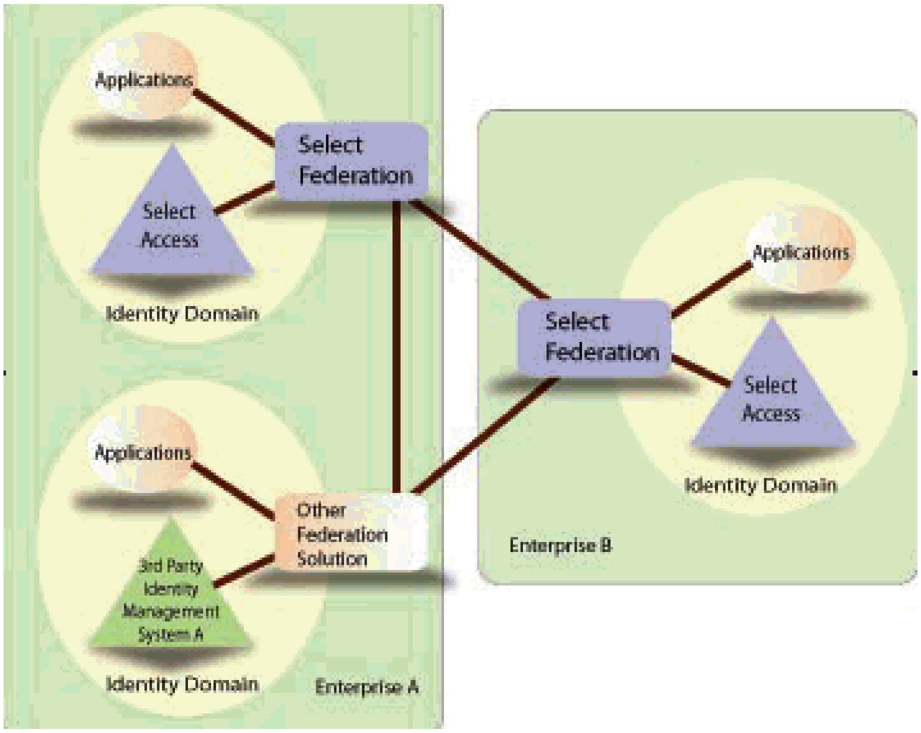
Privacy Manager

The Select Federation Privacy Manager is a unique component that empowers end users to control the exchange of their personal attributes and their preferences about exchanging such information between trusted sites. The Privacy Manager is provided with Select Federation. It enables end users to consent the personal information that may be exchanged between federated sites. This exchange of personal information typically occurs as a part of a Liberty Profile Service query or a SAML Attribute query.

How Does Select Federation Work with Select Access?

Select Federation is designed to complement an enterprise's existing Select Access deployment, adding the specialized function of federated identity management. Select Federation in effect extends the identity management capabilities of Select Access to disjoint domains which may or may not be in your organization. Select Federation's deployment model is described by the following concept diagram.

Figure 3 Select Federation Concept Diagram



Select Federation delivers the federation capabilities that allow you to connect to all your Trusted Partners regardless of which federation protocol and federation solution the partners selected. As shown in Figure 3, Select Federation is connected to Select Access and the associated applications that your end users need to access. Thus, with one instance of Select Federation, your Select Access users are able to connect to with all your trusted partners' applications without the user needing to login separately each time.

3 Getting Started

This chapter provides an overview of the Select Federation interface. Detailed procedures for each task are provided in subsequent chapters and in online help.

Select Federation provides an Administration console that allows the root administrator to add and configure additional delegated administrators to Select Federation, and to monitor the activities of these delegated administrators and end users. The Administration Console allows administrators to perform the following functions:

- Download Metadata — Download metadata from your site as a file, which can be uploaded into other sites that you trust. See Sending Your Metadata to Your Trusted Partner on page 31 for more information and instructions.
- Manage partners — Set up, add, remove and edit partners and groups See Chapter 4, Managing Partners for more information and instructions.
- Manage federations — View federated users and remove federated partners. See Managing Federations on page 74 for more information and instructions.
- Audit server and admin activities — Monitor the activities of all administrators and the enabled end users through viewing the server and admin audit logs. See Auditing on page 63 for more information and instructions.
- Manage administrators — All administrators can change their passwords. Only the root administrator can add and remove delegated administrators. See Managing Admins on page 71 for more information and instructions.

Running the Administration Console

You access the Administration Console through the Select Federation Administration Console startup page. The Select Federation Administration Console startup page is usually deployed at:

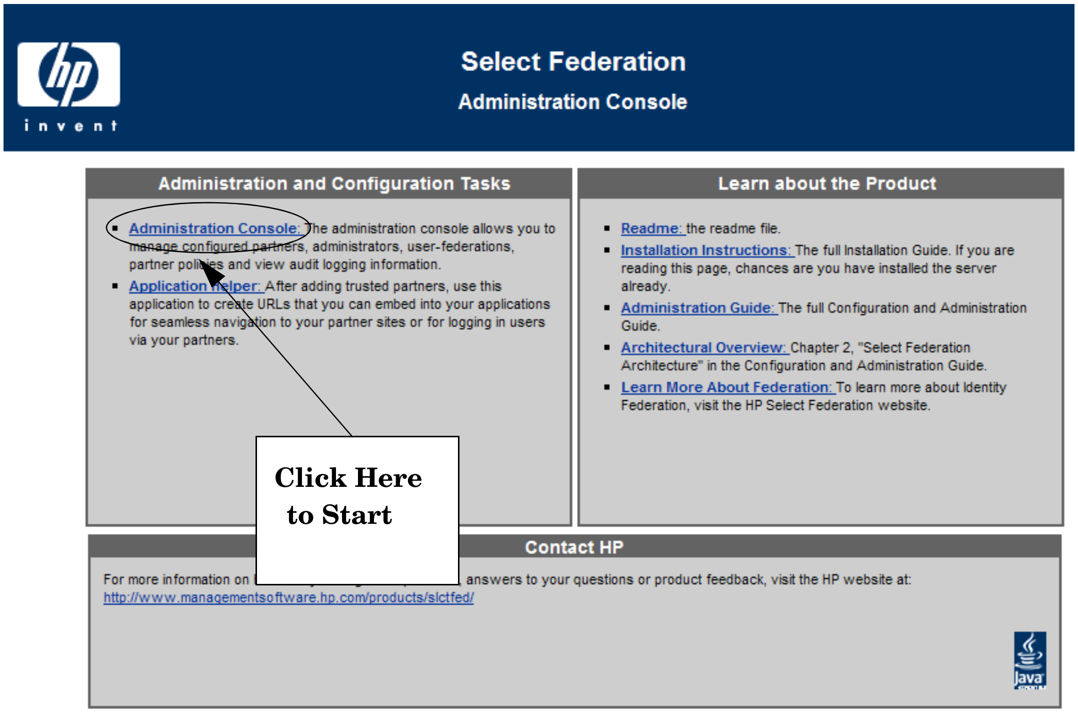
```
http://<base-url>/tfs-internal
```

<base-url> is the root of the application server on which you have deployed Select Federation.

➤ It is recommended that you deploy `tfs-internal` on a separate server for security reasons. In this case, you would use the URL for that server to access tfs-internal rather than the *<base-url>*.

After you deploy Select Federation, the Select Federation Administration Console startup page opens with links to the documentation and various resources. This page also includes a link to the Administration Console, as shown in Figure 4.

Figure 4 Select Federation Administration Console Startup Page



Select Federation

Administration Console

Administration and Configuration Tasks

- Administration Console: The administration console allows you to manage configured partners, administrators, user-federations, partner policies and view audit logging information.
- Application Helper: After adding trusted partners, use this application to create URLs that you can embed into your applications for seamless navigation to your partner sites or for logging in users via your partners.

Learn about the Product

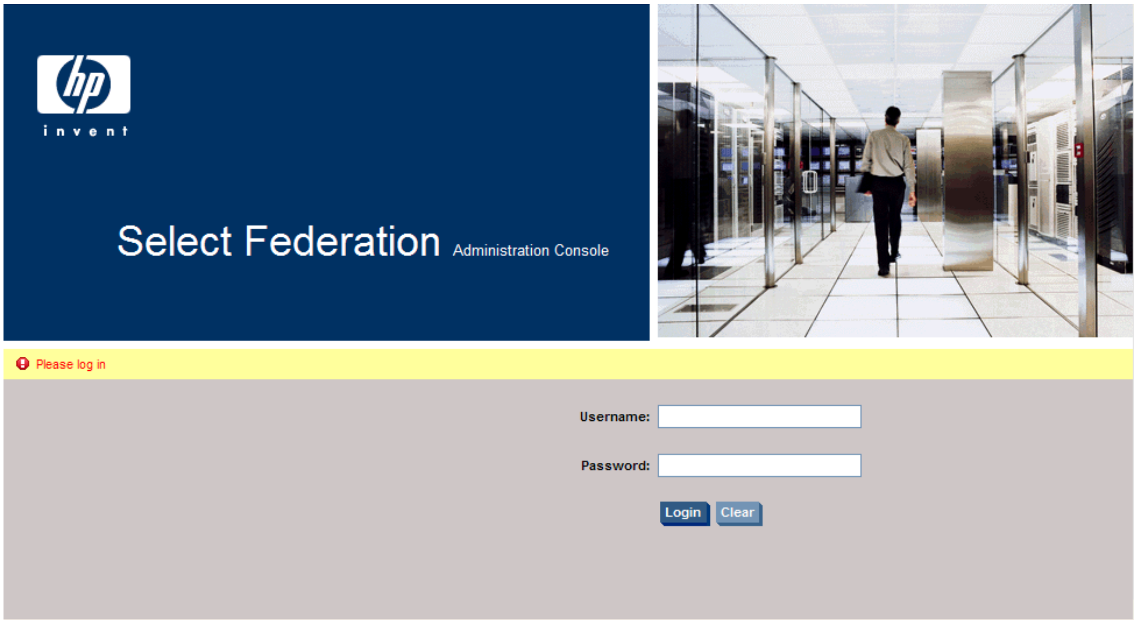
- Readme: the readme file.
- Installation Instructions: The full Installation Guide. If you are reading this page, chances are you have installed the server already.
- Administration Guide: The full Configuration and Administration Guide.
- Architectural Overview: Chapter 2, "Select Federation Architecture" in the Configuration and Administration Guide.
- Learn More About Federation: To learn more about Identity Federation, visit the HP Select Federation website.

Contact HP

For more information on answers to your questions or product feedback, visit the HP website at: http://www.managementsoftware.hp.com/products/slctfed/

Click on **Administration Console** to open the Administration Console login page.

Figure 5 Select Federation Administration Console Login Page



Select Federation Administration Console

Please log in

Username:

Password:

If Select Access is not used, the default Admin account is "`admin`" and the default password is "`tgadmin`".

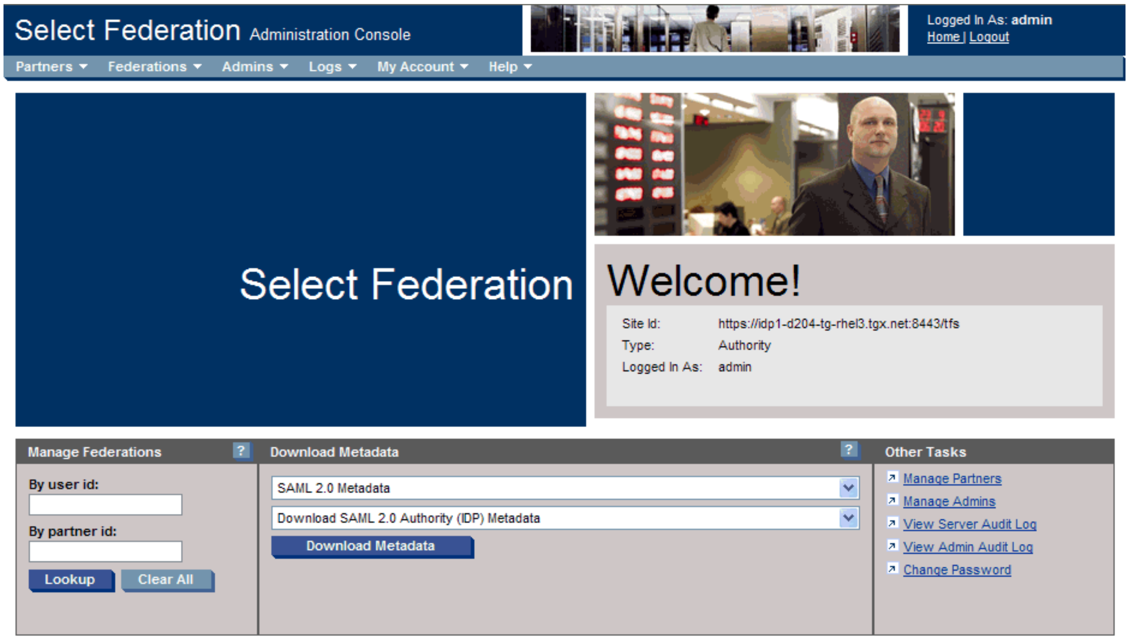
Change the default password immediately after installing Select Federation.

➤ If Select Access is used to protect the Select Federation installation, then Select Access will prompt for credentials, and the Select Federation login page will not be shown.

Navigating the Select Federation Administration Console

After you log in, the Administration Console Welcome page opens.

Figure 6 Administration Console Welcome Page



Site Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs

Type: Authority

Logged In As: admin

The Welcome page includes the following options:

Menus — Allow you to link to all the features of this product.

Home Page Panels — Also allow you to link to all the features.

You can view context-sensitive Help by clicking the question mark button on the upper right-hand corner of a panel.

Home – Clicking **Home** brings you back to the Administration Console Welcome page showing your site ID and type. This is the only page that allows you to download the metadata that you need to exchange with you partners. See Exchanging Metadata with Your Partners on page 30 for more information.

Menus

The menus link the administrator to all the features of this product. You can also use the corresponding links in the panel (see Home Page Panels on page 28 for more information).

Following is a brief summary of each menu. More details on how to use these features are in the corresponding chapters:

Partners → **Manage Partners** — Allows you to create or remove Partner Groups, to view the details of each Partner Group and Partners with which you have federations, to edit the details of each Group or Partner and remove and/or change the configuration. See Chapter 4, Managing Partners for more information and instructions.

Federations → Manage Federations — Allows you to see all of the users that are federated with your Partners. You can also perform a search with a single user name. See Managing Federations on page 74 for more information and instructions.

Admins → Manage Admins —Allows only the root administrator to add and remove delegated administrators. This menu is only available to the root administrator when Select Federation is running in Standalone mode. See Managing Admins on page 71 for more information and instructions.

Logs → View Server Audit Log — Allows any administrator to view all the operational activities of each enabled user. See Server Audit Log on page 64 for more information and instructions.

Logs → View Admin Audit Log — Allows only root administrators to view the federated identity activities of all administrators, when Select Federation is running in Standalone mode. When Select Federation is running in Select Access-Integrated mode, this menu option is available to anyone who is logged in. See Admin Audit Log on page 67 for more information and instructions.

My Account → Change Password — Allows any administrator to change their password. The root administrator can change the password of a delegated administrator. This menu option is only available when Select Federation is running in Standalone mode. See Changing the Admin Password on page 73 for more information and instructions.

My Account → View Details — Allows delegated administrators to view details about their accounts that were added by the root administrator. This menu option is only available when Select Federation is running in Standalone mode. See Viewing Delegated Admin Account Details on page 73 for more information and instructions.

Help – Provides online help for the following:

> **Contents and Index** – Allows you to see all available help in a Table of Contents or search for a key word using an Index.
>
> **Online Support** – Opens the HP Software Support page.
>
> **About Select Federation** – Provides the version and copyright information for the Select Federation product.

Home Page Panels

The following three panels on the Welcome page (see Figure 6) also link the administrator to all the features of this product:

Manage Federations – You can search for a specific federation or click the **Lookup** button to view all federations on the Federations page. This panel corresponds to the **Federations → Manage Federations** menu option.

Download Metadata – You may download the metadata from your site as a file so that it can be uploaded into other sites that you trust, or you may just view the parameters of your site. You can only download metadata from the Welcome page.

Other Tasks – Each link corresponds to a menu option. See Menus on page 27 for a brief description of each feature.

4 Managing Partners

This chapter describes how to manage your partners in the following topics:

- Setting Up Partnerships
- Adding Groups and Partners to Your Installation
- Removing Partners and Groups from Your Installation
- Editing Partner Parameter Settings

Setting Up Partnerships

A federated system depends upon trust between independent cooperating web sites to provide a seamless experience to its common users. Select Federation acting as a node in such a federated network needs to know about trusted partner sites that can act as either authorities which assert users that can seamlessly be provided access to local resources, or as application sites which can provide seamless access to local users authenticated locally by Select Federation, or both.

The main entity in Select Federation for expressing such a trust relationship is a “Partner”. A partner is described in terms of its protocol metadata, various descriptive attributes, and policy information. Select Federation also allows partners to be grouped in “Groups”. A group shares a common set of policies and all groups and partners within that group inherit those policies. An administrator may override the group setting for a particular partner within that group.

The basic advantage of a federation is that your enterprise can quickly provide the benefits of a centralized identity management system to a larger set of users than is possible with a centralized identity management system. This larger set of users can be within your enterprise and/or from other organizations. They can also be customers, users of your extranet, users of your supply chain, or other external users that you share with your partner companies.

This section explains how to create and send the metadata from your Select Federation installation to your Partners. For information on how to add Partners to your installation, see Adding Groups and Partners to Your Installation on page 34.

Using Open Standard Federation Protocols

The two most popular open standard federation standards today are Security Assertion Markup Language (SAML) and Liberty Alliance. To create a federated link with your partner, you need to decide which open standard protocol you and your partner will use. You can then exchange the appropriate metadata with each other.

To set up a federation, you first need to decide your site's role:

- Authority site role (also called SAML Producer or Identity Provider (IDP)) — As an Authority site, you authenticate users and allow them to seamlessly use other Application sites in your federation.
- Application site role (called SAML Consumer or Service Provider (SP)) — As an Application site, you host an application, but do not authenticate users.
- Both Authority and Application site roles - A single installation can also act in both roles at various times, towards various partners and/or for different users. However it is not very common for a single installation to play both an Application and an Authority role.

Once you have decided the role of your site, the first step is to download the metadata. Select Federation is unique in that it supports all of the popular open federation standards. This makes it easier to connect to multiple partners that may not have selected the same standards or conventional identity management solution.

Understanding the Impact of Metadata on Federation

Metadata in a federation is a description of the Trusted Partner site with which you want to link. Metadata is an online exact description of a site in a federation. The metadata describes the Trusted Partner site's public-key certificates and the various URLs at which its site services (such as single sign-on, logout) are available. Because of the metadata, sites receiving messages from that Trusted Partner site can confirm that those messages are signed by the Trusted Partner site and have not been tampered with.

In some federation standards such as Liberty 1.2 or SAML 2.0, the metadata specification is a conformant part of interoperability certification. In other specifications such as SAML 1.0, SAML 1.1, and Liberty 1.1, there is either just an informal metadata specification or a convention in the community about how to define the metadata. In Select Federation, the Administration Console enables an administrator to publish the site's metadata as well as import other sites' metadata.

Exchanging Metadata with Your Partners

To add Trusted Partner sites to your federation, both you and your Trusted Partner need to upload each other's metadata. Metadata exchange is mutual, so you need to ensure that the other site has added your metadata to its federation. The sections that follow describe how you can forward relevant data to your partner. For details on how to use metadata forwarded to you from a partner, see Chapter 4, Managing Partners.

You can download the metadata into a metadata file and send this file to your Trusted Partner, or send the partner the information for manual entry.

You need to know the protocol and protocol version that the Trusted Partner site is capable of using, so that you can select the type of federation you would like to set up. Following are the protocols and protocol versions you can select:

- For SAML 2.0, see Downloading Your Site's Metadata for a SAML 2.0 Federation on page 31
- For SAML 1.0 or 1.1, see Downloading your Site's Metadata for SAML 1.0 or SAML 1.1 on page 32
- For Liberty ID-FF 1.1 or 1.2, see Downloading Your Site's Metadata for a Liberty ID-FF 1.1 or ID-FF 1.2 Federation on page 33

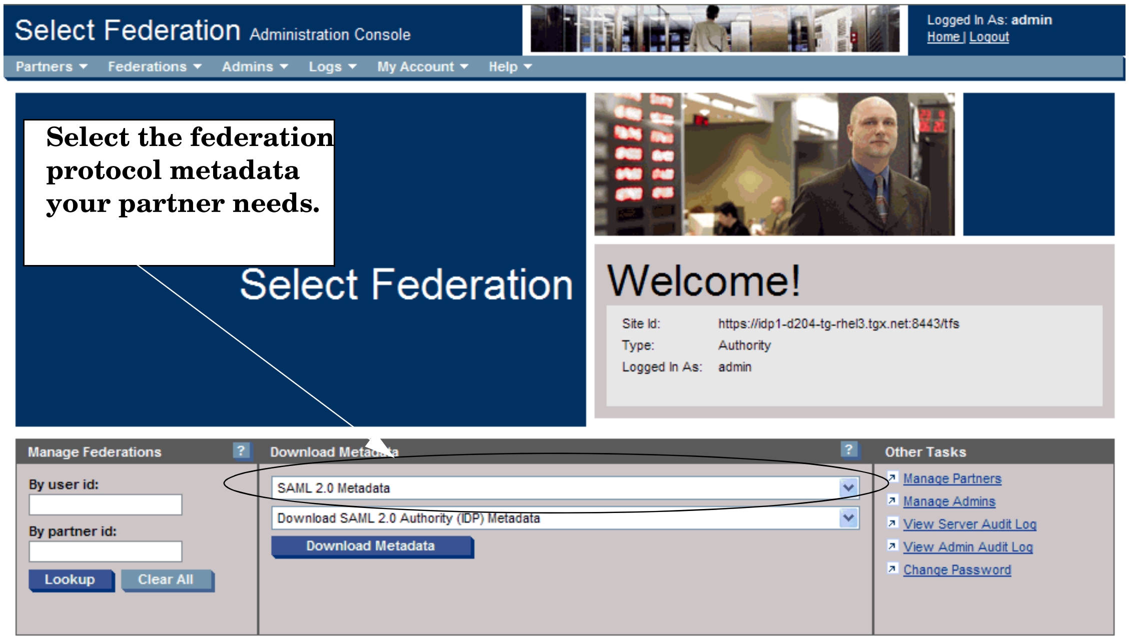
- For Active Directory Federation Services (ADFS)/WS-Federation 1.0, see Downloading Your Site's Metadata for Active Directory Federation Services (ADFS)/WS-Federation 1.0 on page 33

Sending Your Metadata to Your Trusted Partner

Select Federation has simplified the process of obtaining your metadata for all the popular federation protocols. With one click, you can download your site information into any of the supported formats. Alternatively, if your partner prefers the information in text format, you can open the file instead of saving it and cut-and-paste the file contents to a text file.

Following are the general steps you take to download the metadata and send it to your Trusted Partner:

1 Be sure you are on the Welcome page. If not, click the **Home** link in the upper-right panel.



Select the federation protocol metadata your partner needs.

Site Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs
Type: Authority
Logged In As: admin

1 Select a protocol format and your site role from the drop-down lists in the Download Metadata panel.

2 Click the **Download Metadata** button.

You are prompted to **open** the metadata file to view it, **save** the file on your hard drive, or **cancel** downloading the metadata file.

3 Click the **Save** button to save the metadata file on your hard drive.

4 Send this file to your partner to be uploaded into your partner's federation software. Follow instructions in Adding a Partner for which Metadata is Available on page 36 to upload your partner's metadata file.

Downloading Your Site's Metadata for a SAML 2.0 Federation

Select Federation uses metadata exchange with your Trusted Partner to set up the SAML federation.

Perform the following steps to create your metadata file to be sent to your Trusted Partner:

1 Select **SAML 2.0 protocol Metadata** from the first drop-down list in the **Download Metadata** panel on the Welcome page.

2 Do one of the following:

- If your site is to be the SAML Consumer or Application Site, select **Download SAML 2.0 Application (SP) Metadata** from the second drop-down list in the **Download Metadata** panel.
- If your site is to be the SAML Producer or Authority Site, select **Download SAML 2.0 Authority (IDP) Metadata** from the second drop-down list in the **Download Metadata** panel.
- To describe both the Authority Site and Application Site in one metadata file, select **Download Combined SAML 2.0 Application and Authority Metadata** from the second drop-down list in the **Download Metadata** panel.

3 Click the **Download Metadata** button in the **Download Metadata** panel.

4 Click the **Save** button to save the metadata file on your hard drive.

5 Send this file to your partner to be uploaded into your partner's federation software. Follow instructions in Adding a Partner for which Metadata is Available on page 36 to upload your partner's metadata file.

Downloading your Site's Metadata for SAML 1.0 or SAML 1.1

Federation software from other vendors that does not support SAML 2.0 typically does not recognize SAML 1.0 or 1.1. metadata. Only SAML2.0 specified a format for metadata for SAML protocols. Earlier versions of Select Federation supported a proprietary format for SAML 1.0 and 1.1. metadata that is still supported in Select Federation 6.60.

The SAML 1.X metadata format is specified by Oasis (see `http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security`). This format is based on the SAML 2.0 metadata specification.

Perform the following steps to create your metadata file to be sent to your Trusted Partner:

1 Select the SAML protocol that you and your partner have agreed to use, SAML 1.0 or SAML 1.1 from the first drop-down list in the **Download Metadata** panel on the Welcome page.

2 Do one of the following:

- If your site is to be the SAML Consumer or Application Site, select **Download SAML 1.x Application (SP) Metadata** from the second drop-down list in the **Download Metadata** panel.
- If your site is to be the SAML Producer or Authority Site, select **Download SAML 1.x Authority (IDP) Metadata** from the second drop-down list in the **Download Metadata** panel.
- To describe both the Authority Site and Application Site in one metadata file, select **Download Combined SAML 1.x Application and Authority Metadata** from the second drop-down list in the **Download Metadata** panel.

3 Click the **Download Metadata** button in the **Download Metadata** panel.

4 Click the **Save** button to save the metadata file on your hard drive.

5 Send this file to your partner to be uploaded into your partner's federation software. Follow instructions in Adding a Partner for which Metadata is Available on page 36 to upload your partner's metadata file.

Downloading Your Site's Metadata for a Liberty ID-FF 1.1 or ID-FF 1.2 Federation

Perform the following steps to create your Liberty ID-FF 1.1 or ID-FF 1.2 metadata file to be sent to your Trusted Partner:

1. Select the Liberty protocol that you and your partner have agreed to use, either **Liberty ID-FF 1.1** or **ID-FF 1.2** from the first drop-down list in the **Download Metadata** panel on the Welcome page.

   If your partner is also using Select Federation, it may be desirable to choose Liberty 1.2. However, all protocols will work between two instances of Select Federation.

2. Do one of the following:

   - If your site is to be the SAML Consumer or Application Site, select **Download Liberty 1.x Application (SP) Metadata** from the second drop-down list in the **Download Metadata** panel.

   - If your site is to be the SAML Producer or Authority Site, select **Download Liberty 1.x Authority (IDP) Metadata** from the second drop-down list in the **Download Metadata** panel.

   - To describe both the Authority Site and Application Site in one metadata file, select **Download Combined Liberty 1.2 Application and Authority Metadata** from the second drop-down list in the **Download Metadata** panel.

3. Click the **Download Metadata** button in the **Download Metadata** panel.

4. Click the **Save** button to save the metadata file on your hard drive.

5. Send this file to your partner to be uploaded into your partner's federation software. Follow instructions in Adding a Partner for which Metadata is Available on page 36 to upload your partner's metadata file.

Downloading Your Site's Metadata for Active Directory Federation Services (ADFS)/ WS-Federation 1.0

Perform the following steps to create your metadata file to be sent to your Trusted Partner:

1. Select **ADFS (WS-Federation 1.0) protocol Metadata** from the first drop-down list in the **Download Metadata** panel on the Welcome page.

2. Do one of the following:

   - If your site is to be the SAML Consumer or Application Site, select **Download ADFS (WS-Federation 1.0) Application (SP) Metadata** from the second drop-down list in the **Download Metadata** panel.

   - If your site is to be the SAML Producer or Authority Site, select **Download ADFS (WS-Federation 1.0) Authority (IDP) Metadata** from the second drop-down list in the **Download Metadata** panel.

   - To describe both the Authority Site and Application Site in one metadata file, select **Download Combined ADFS (WS-Federation 1.0) Application and Authority Metadata** from the second drop-down list in the **Download Metadata** panel.

3. Click the **Download Metadata** button in the **Download Metadata** panel.

4. Click the **Save** button to save the metadata file on your hard drive.

5. Send this file to your partner to be uploaded into your partner's federation software. Follow instructions in Adding a Partner for which Metadata is Available on page 36 to upload your partner's metadata file.

Metadata URLs

The metadata URL for your installation is its “ProviderId”, with a value of

```
https://<site-base-URL>/tfs
```

Note that the metadata format under this URL can be specified by the `defaultMetadata` system configuration entry, which can have one of the following values:

- `liberty12`
- `liberty11idp`
- `liberty11sp`
- `saml10`
- `saml11`
- `saml20`
- `wsfed10`

The default value is `saml20`.

Adding Groups and Partners to Your Installation

In Select Federation it is possible to create groups for your federation partners. A group is used to apply common federation policies such as protocol security mechanisms and attributes to be exchanged to a set of partners. You can always override the policy setting for a particular partner, even if that partner belongs to the group.

Organizing partners into groups gives you the benefit of better organization between different types of partners and it will also make partner configuration easier, since you can make the partner inherit its configuration from the parent group. You should note that the metadata is always partner-specific — groups do not have metadata of their own.

Before you begin to add groups and partners, you need to determine the partner organization:

- How many groups do you want and what will be the group names?
- Into which groups will the partners belong?
- Which partners will not be in groups?

➤ You cannot move partners to another group once they are added.

This section describes how to do the following:

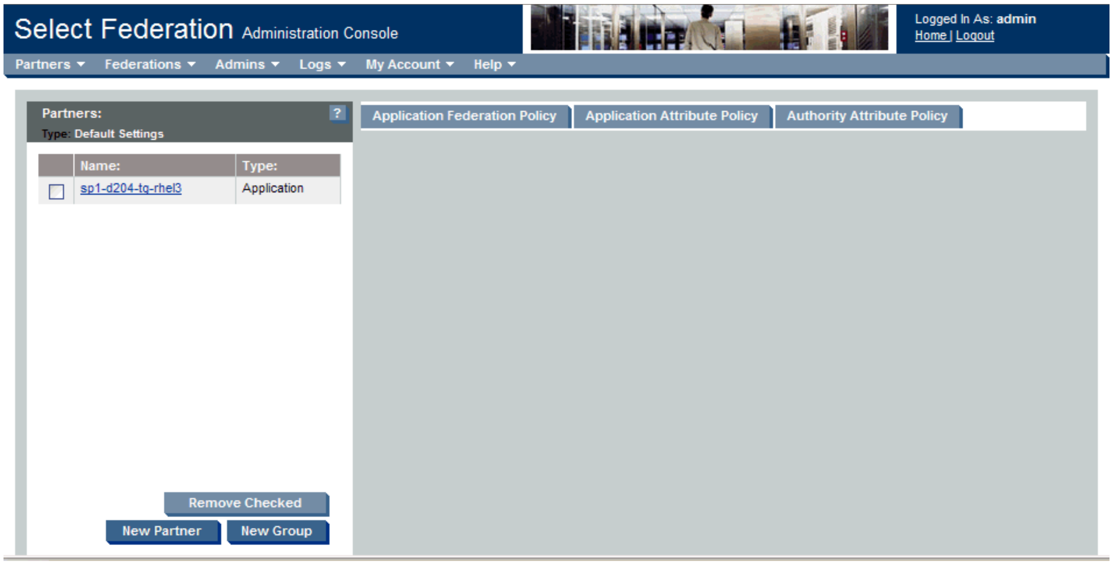
- Add Groups To Your Installation
- Add Partners To Your Installation

Add Groups To Your Installation

To add groups to your installation, perform the following steps:

1 Select the **Partners** → **Manage Partners** menu options, or click the **Manager Partners** link on the Welcome page.

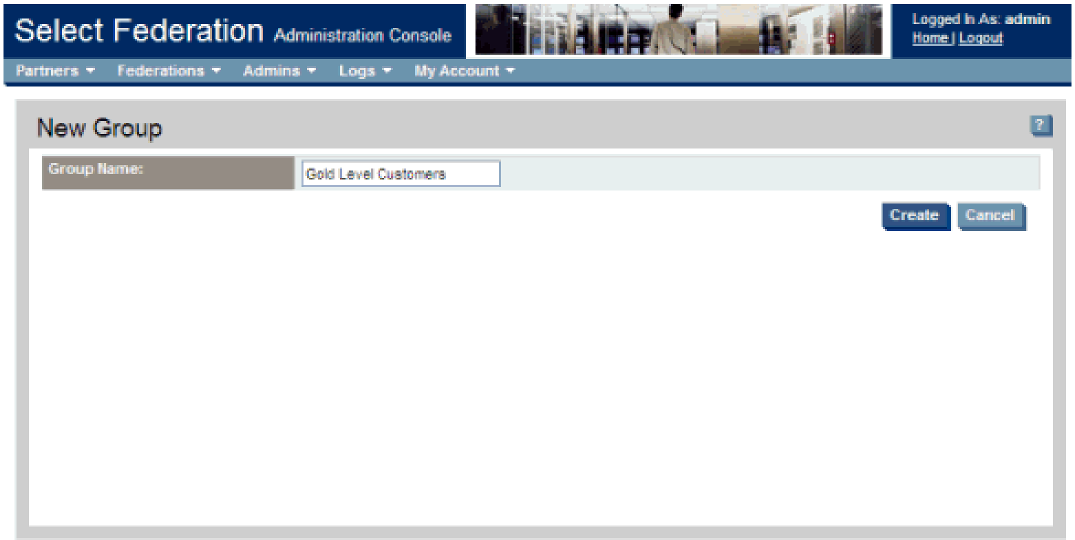
The Partners page opens.



	Name:	Type:
□	sp1-d204-tg-rhel3	Application

2 Click on the **New Group** button in the left panel.

The New Group page opens.



Group Name: Gold Level Customers

3 Enter a name for your Group in the Group Name field.

4 Click **Create** to finish.

The Display Info page opens. You can edit this page to add information specific to the group. See Configuring the Display Information on page 46 for instructions on filling in the display information.

➤ You can create subgroups easily by navigating to the Group of your choice and performing Steps 2 - 4 again.

Add Partners To Your Installation

To add Partners to your federation, you need to have the metadata for those Partners. Metadata exchange is mutual, so you need to ensure that the other Partner has added your metadata to its federation. For more information on metadata, see Understanding the Impact of Metadata on Federation on page 30.

There are two ways to obtain data from your partners:

- If the Partner’s metadata file is available, download it from a well-known URL or obtain the metadata securely from the administrator of the Partner. See Adding a Partner for which Metadata is Available on page 36.
- If a metadata file or download is NOT available, see Adding a Partner for Which Metadata is Not Available on page 39.

Select Federation can detect the protocol used by the Partner from the metadata provided by that Partner. However it is recommended that you and your Partner agree upon the protocol to use for the federation.

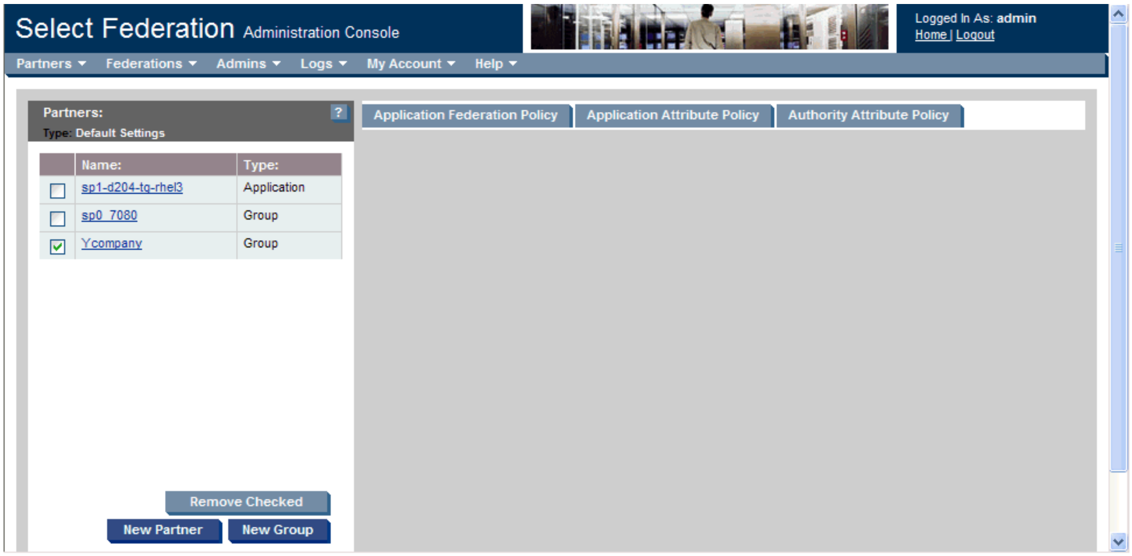
Adding a Partner for which Metadata is Available

To create the new federation, you need your partner’s metadata file so that you can upload this information into your Select Federation.

To add a partner for which metadata is available, perform the following steps:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.



	Name:	Type:
□	sp1-d204-tq-rhel3	Application
□	sp0_7080	Group
☑	Ycompany	Group

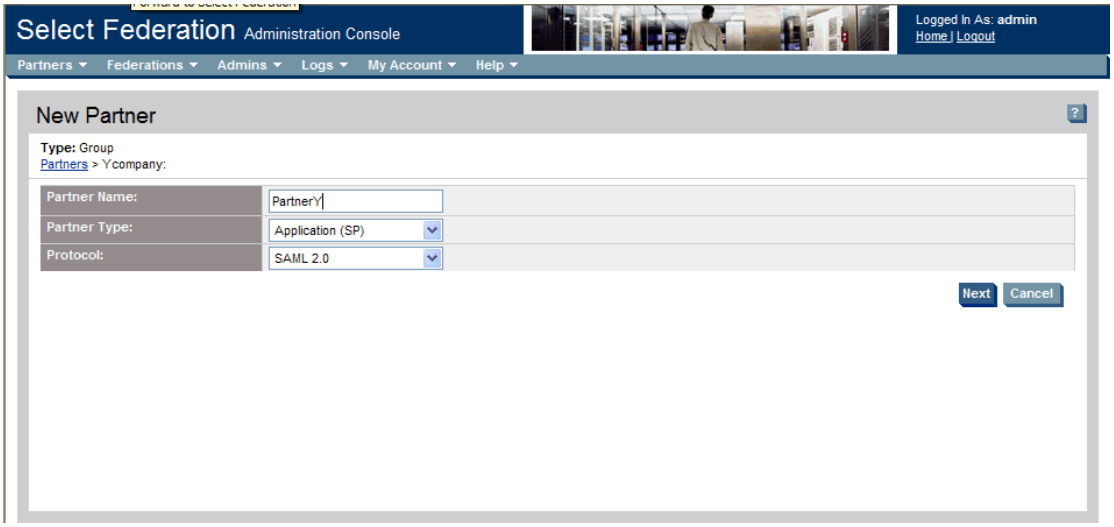
Notice “Partners” at the top of the left panel. This indicates the top-level group, which is the default. All other groups and partners are added underneath.

2 If you are adding a partner to a particular group, click in the check box next to the group to which you want to add the partner.

Otherwise, you can add a new partner to the default Partners group.

3 Click the **New Partner** button in the left panel.

The New Partner page opens.



New Partner

Type: Group
Partners > Ycompany:

Partner Name:	PartnerY
Partner Type:	Application (SP)
Protocol:	SAML 2.0

4 Enter a name for your Trusted Partner site in the **Partner Name** field.

You can assign any “friendly name” to describe your partner’s site on your HP Administration Console. Enter the **Partner Name** as you would like it to appear in your system.

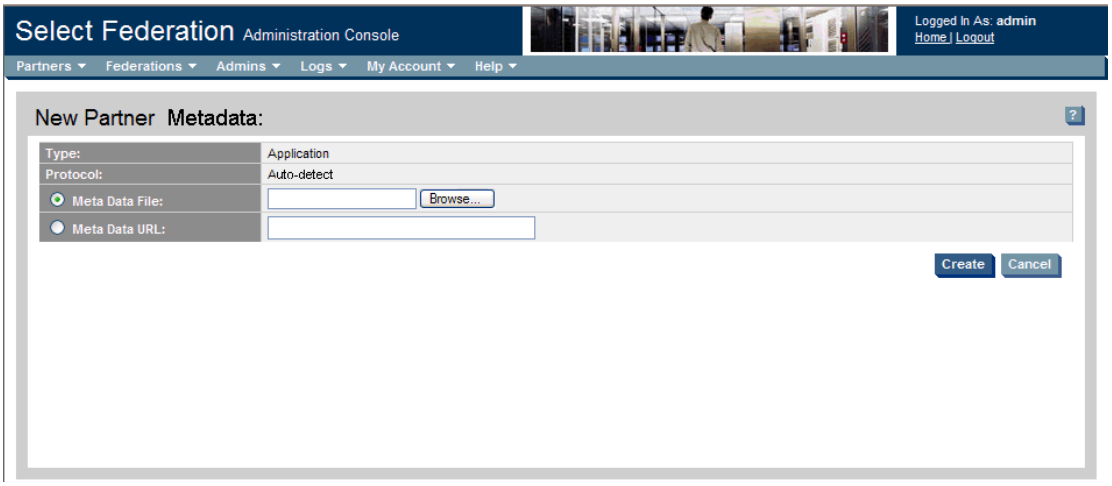
➤ You must enter data into this field. This name is also visible to the end users.

5 Select the available role of your Partner under the **Partner Type** – either an Application, Authority Site, or both.

- If your site is an authority site or Identity Provider, then your Partner is the application site or the Service Provider. **Select Application (SP)** as the Site Type for the Partner you are adding.
- If your site is an application site or Service Provider, then specify **Authority (IDP)** as the Site Type for the Partner you are adding.
- To import both authority site and application site data in the same metadata file, select **Application/Authority (SP/IDP)** for the Partner you are adding.

6 In the **Protocol** field, select either **Auto-detect** or the particular protocol that you would like to use with the partner. Click **Next.**

The New Partner Meta Data page opens. Clicking **Cancel** will cancel the creation of a new Partner.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
?
Browse...
Create
Cancel

New Partner Metadata:

Type:	Application
Protocol:	Auto-detect
● Meta Data File:	
○ Meta Data URL:	

7 You can either upload your Partner’s metadata file or get the information from a URL.

- **Metadata File**: Enter or browse to find the full path of the metadata file that you received from your partner.
- **Metadata URL**: Enter the URL where the metadata information from your partner is stored.

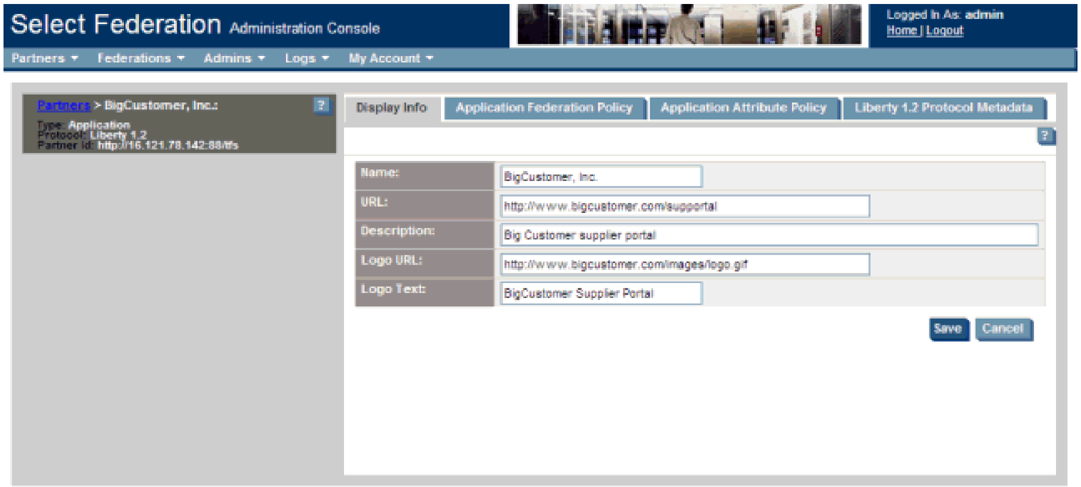
8 Click **Create** to complete the creation of your federation link.

Clicking **Cancel** will cancel the creation of a new Partner.

You will see a screen that shows the newly added Partner in the federation.

9 Click **Edit** to edit the details of the new Partner.

The editable Display Info page opens.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾
?
Display Info
Application Federation Policy
Application Attribute Policy
Liberty 1.2 Protocol Metadata
?
Save
Cancel

Partners > BigCustomer, Inc.:
Type: Application
Protocol: Liberty 1.2
Partner Id: http://16.121.78.142:88/tfs

Name:	BigCustomer, Inc.
URL:	http://www.bigcustomer.com/supportal
Description:	Big Customer supplier portal
Logo URL:	http://www.bigcustomer.com/images/logo.gif
Logo Text:	BigCustomer Supplier Portal

The **Name** field is mandatory, but the rest of the text fields are optional. However, filling in these optional fields help the look-and-feel of the applications that process your federation information. These optional fields allow you to import your Partner’s logo and link directly to your Partner’s web page.

The **URL** is the default application that the Partner makes available for single sign-on to users of your installation. This is mainly useful if you want to add a single sign-on link to that partner's application to a portal (see also Chapter 6, Enabling Applications).

This URL is NOT a link to the Partner's ProviderId.

10 Enter a one-line **Description** of the Partner site to which you are connecting. This is optional and can be left blank.

Logo URL is the logo that appears on your portal and represents the logo of the federation link you created. It can be your Partner's logo. This field is optional.

Logo Text is the text that appears in the bubble when you put your mouse over the Logo URL. This field is optional.

11 Click **Save** to save the changes you have made.

Clicking **Cancel** cancels the changes.

Adding a Partner for Which Metadata is Not Available

For partners using a protocol that does not specify a rigid metadata file format (such as SAML 1.0, SAML 1.1 and ADFS (WS-Federation 1.0)), Select Federation allows manual entry of metadata. The input fields for manual entry of metadata are dependent upon the protocol and upon whether the partner site is an Authority (IDP) site or an Application (SP) site.

The following sections describe how to manually add the protocol metadata:

- Manually Adding the SAML 1.x Authority Protocol Metadata
- Manually Adding the SAML 1.x Application Protocol Metadata
- Manually Adding the ADFS (WS-Federation 1.0) Authority Protocol Metadata
- Manually Adding the ADFS (WS-Federation 1.0) Application Protocol Metadata

Manually Adding the SAML 1.x Authority Protocol Metadata

Perform the following steps to manually add your SAML 1.x authority protocol metadata:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

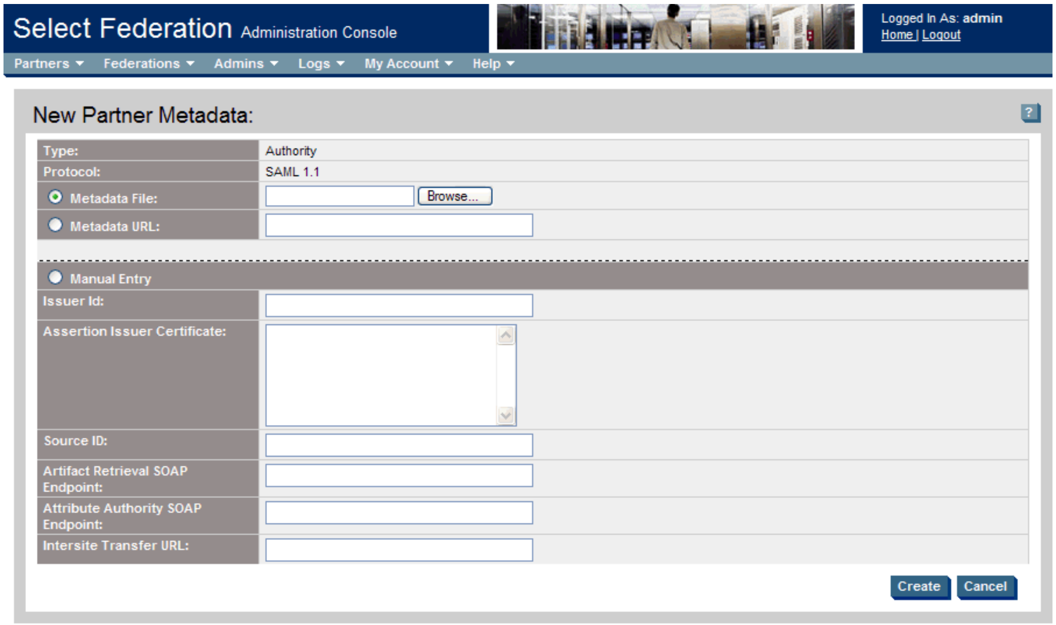
2 Click the **Name** of the Partner that you wish to update.

3 Select the **SAML 1.x Authority Protocol Metadata** tab.

The SAML Authority Protocol Metadata page opens.

4 Click the **Update** button.

The New Partner Metadata page opens with the necessary parameters.



New Partner Metadata:

Type:	Authority
Protocol:	SAML 1.1
● Metadata File:	
○ Metadata URL:	
○ Manual Entry	
Issuer Id:	
Assertion Issuer Certificate:	
Source ID:	
Artifact Retrieval SOAP Endpoint:	
Attribute Authority SOAP Endpoint:	
Intersite Transfer URL:	

5 Click **Manual Entry**.

6 Fill in the necessary fields:

- **Issuer Id**: The identifier that the partner site includes in the issuer field in the assertions that it generates.
- **Source Id**: A 20-byte hex encoded or base64 encoded binary value placed in artifacts that the partner site generates, when using the artifact profile. It is not something that can be chosen by the administrator adding the partner.
- **Artifact Retrieval SOAP Endpoint**: This is the location of the SAML responder’s SOAP service used for artifact pickup.
- **Attribute Authority SOAP Endpoint:** This is the URL where your site will invoke the partner's attribute authority service over SOAP for obtaining user attributes.
- **Intersite Transfer URL:** This field is optional. This is the SAML Inter-site Transfer Service URL used to navigate to your partner site in the federation.

7 Click **Create** to complete the creation of your federation link.

A screen opens showing the newly added site in the federation. Clicking **Cancel** cancels the creation of a new site. To edit the details of the new partner site, click **Edit**. See Changing Settings for a Partner on page 44 for details.

Manually Adding the SAML 1.x Application Protocol Metadata

Perform the following steps to manually add your SAML 1.x application protocol metadata:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

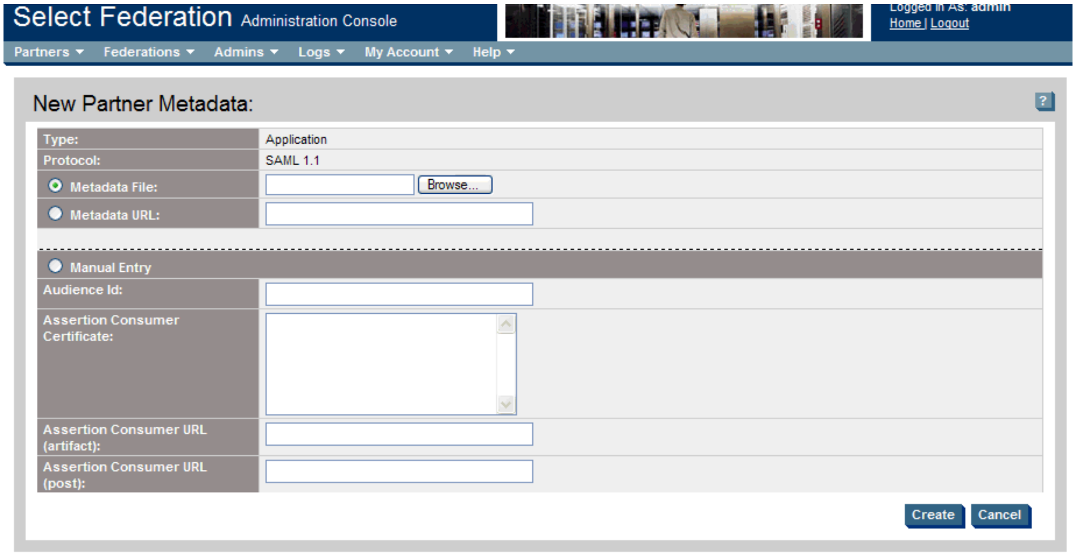
2 Click the **Name** of the Partner that you wish to update.

3 Select the **SAML 1.x Application Partner Protocol Metadata** tab.

The SAML Application Protocol Metadata page opens.

4 Click the **Update** button.

The New Partner Metadata page opens with the necessary parameters. The following figure is an example of the SAML 1.1 New Partner Metadata page.



New Partner Metadata:

Type:	Application
Protocol:	SAML 1.1
● Metadata File:	
○ Metadata URL:	
○ Manual Entry	
Audience Id:	
Assertion Consumer Certificate:	
Assertion Consumer URL (artifact):	
Assertion Consumer URL (post):	

5 Click **Manual Entry**.

6 Fill in the necessary fields:

- **Audience Id:** The identifier for the Partner site.
- **Assertion Consumer Certificate:** This certificate is used by the Assertion Consumer to authenticate to the SAML Producer for picking up the SAML Assertion Artifact. This certificate is required if the partner site signs the authentication requests that it issues, but not otherwise.
- **Assertion Consumer URL (artifact):** This is the URL to which your site sends assertion artifacts.
- **Assertion Consumer URL (post):** For SAML 1.1 only, this is the URL to which the user is redirected from your site to the SAML consumer site when using the SAML POST profile.

7 Click **Create** to complete the creation of your federation link.

A screen opens showing the newly added site in the federation. Clicking **Cancel** cancels the creation of a new site. To edit the details of the new partner site, click **Edit**. See Changing Settings for a Partner on page 44 for details.

Manually Adding the ADFS (WS-Federation 1.0) Authority Protocol Metadata

Perform the following steps to manually add your ADFS (WS-Federation 1.0) authority protocol metadata:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

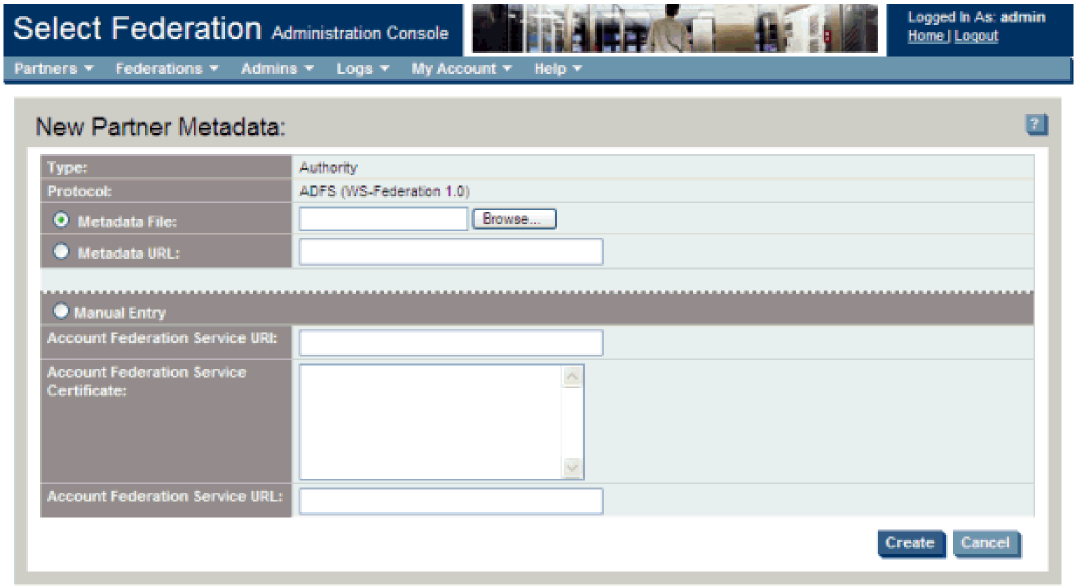
2 Click the **Name** of the Partner that you wish to update.

3 Select the **ADFS (WS-Federation 1.0) Authority Protocol Metadata** tab.

The ADF (WS-Federation 1.0) Authority Protocol Metadata page opens.

4 Click the **Update** button.

The New Partner Metadata page opens with the necessary parameters.



New Partner Metadata:

Type:	Authority
Protocol:	ADFS (WS-Federation 1.0)
● Metadata File:	
○ Metadata URL:	
○ Manual Entry	
Account Federation Service URI:	
Account Federation Service Certificate:	
Account Federation Service URL:	

5 Click **Manual Entry**.

6 Fill in the necessary fields:

- **Account Federation Service URI**: The identifier for the Partner site.
- **Account Federation Service Certificate**: The certificate with which the partner site signs the assertions that it issues.

  The certificate field uses a PEM encoded certificate. Therefore, the entry must start and end with the following:

  ```
  -----BEGIN CERTIFICATE-----

  ...

  -----END CERTIFICATE-----
  ```

- **Account Federation Service URL**: This is the URL to which the user is redirected from your site to the ADFS authority site to request authentication.

7 Click **Create** to complete the creation of your federation link.

A screen opens showing the newly added site in the federation. Clicking **Cancel** cancels the creation of a new site. To edit the details of the new partner site, click **Edit**. See Changing Settings for a Partner on page 44 for details.

Manually Adding the ADFS (WS-Federation 1.0) Application Protocol Metadata

Perform the following steps to manually add your ADFS (WS-Federation 1.0) application protocol metadata:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

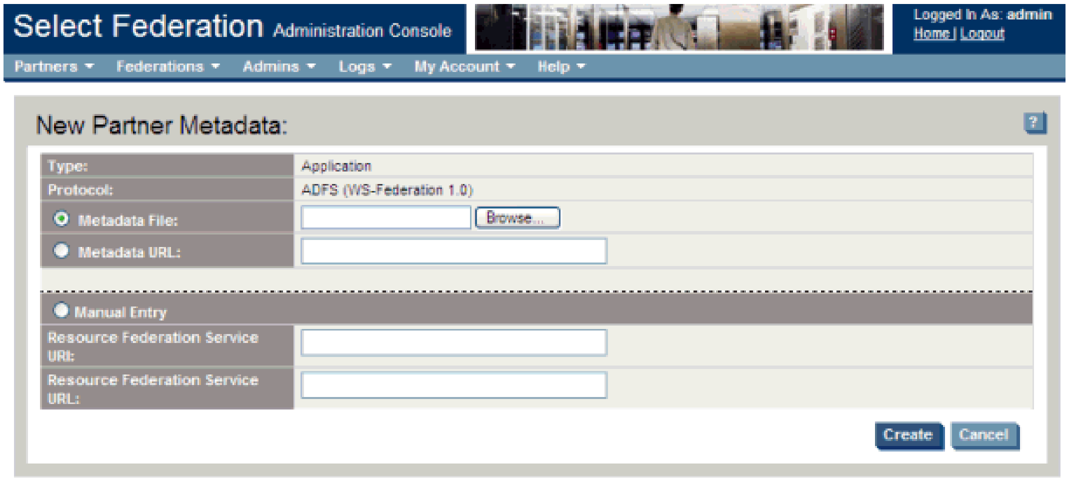
2 Click the **Name** of the Partner that you wish to update.

3 Select the **ADFS (WS-Federation 1.0) Application Protocol Metadata** tab.

The ADF (WS-Federation 1.0) Application Protocol Metadata page opens.

4 Click the **Update** button.

The New Partner Metadata page opens with the necessary parameters.



New Partner Metadata:

Type:	Application
Protocol:	ADFS (WS-Federation 1.0)
● Metadata File:	
○ Metadata URL:	
○ Manual Entry	
Resource Federation Service URI:	
Resource Federation Service URL:	

5 Click **Manual Entry**.

6 Fill in the necessary fields:

- **Resource Federation Service URI**: The identifier for the Partner site.
- **Resource Federation Service URL**: This is the URL to which the user is redirected from your site to the ADFS resource site with an authentication assertion.

7 Click **Create** to complete the creation of your federation link.

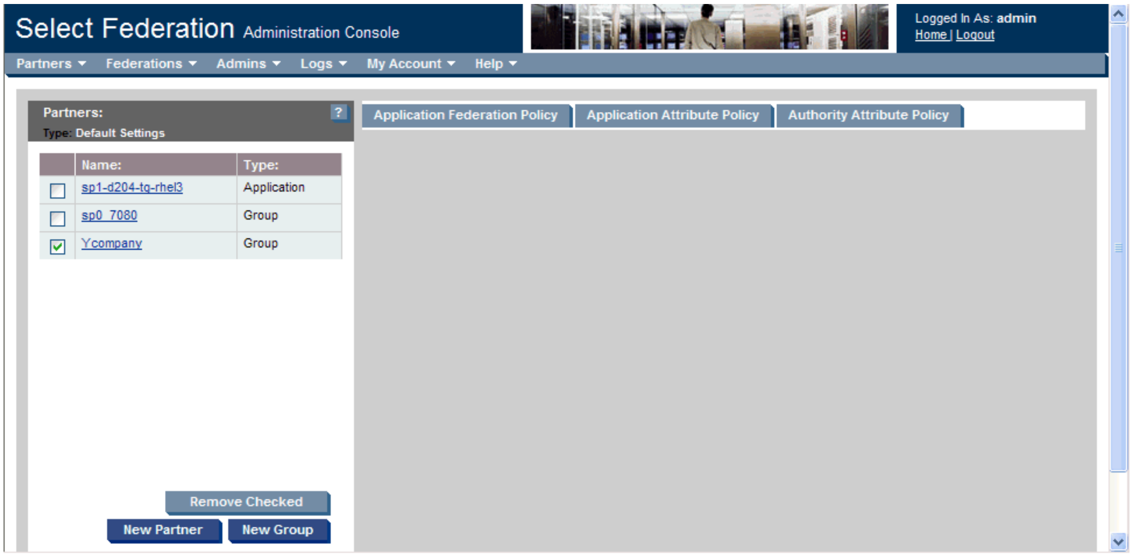
A screen opens showing the newly added site in the federation. Clicking **Cancel** cancels the creation of a new site. To edit the details of the new partner site, click **Edit**. See Changing Settings for a Partner on page 44 for details.

Removing Partners and Groups from Your Installation

Follow these steps to remove an existing Partner or Group:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Application Federation Policy
Application Attribute Policy
Authority Attribute Policy
Remove Checked
New Partner
New Group

Partners:
Type: Default Settings

	Name:	Type:
□	sp1-d204-tg-rhel3	Application
□	sp0_7080	Group
☑	Ycompany	Group

2 Click in the check box or check boxes next to the Partner and/or Group that you want to remove.

If you are removing a group that still contains partner entries, the partner entries are deleted along with the group.

3 Click **Remove Checked**.

A confirmation dialog opens asking if you want to remove existing federations.

4 Click **OK** to remove the selected partners and groups.

Editing Partner Parameter Settings

Select Federation allows you to make changes to all the parameters of your existing federations, including the following information:

- **Display Info**: This describes how your Partner appears in your system. For details, see Configuring the Display Information on page 46.
- **Federation Policy**: These are the rules that you and your partner agreed to use in communicating between your sites. For details, see Configuring the Federation Policy on page 48.
- **Attribute policy**: Select Federation allows you to configure and change the user profile attribute exchange. For details, see Configuring the Attribute Policy on page 51.
- **Protocol Policy**: These are the actual protocol parameters, also known as metadata, consisting of URLs for various protocol services, certificates, and so on. For details, see Configuring the Protocol Policy and Metadata on page 53.

Changing Settings for a Partner

You can edit any of the partner settings in the federations you have in your system, such as:

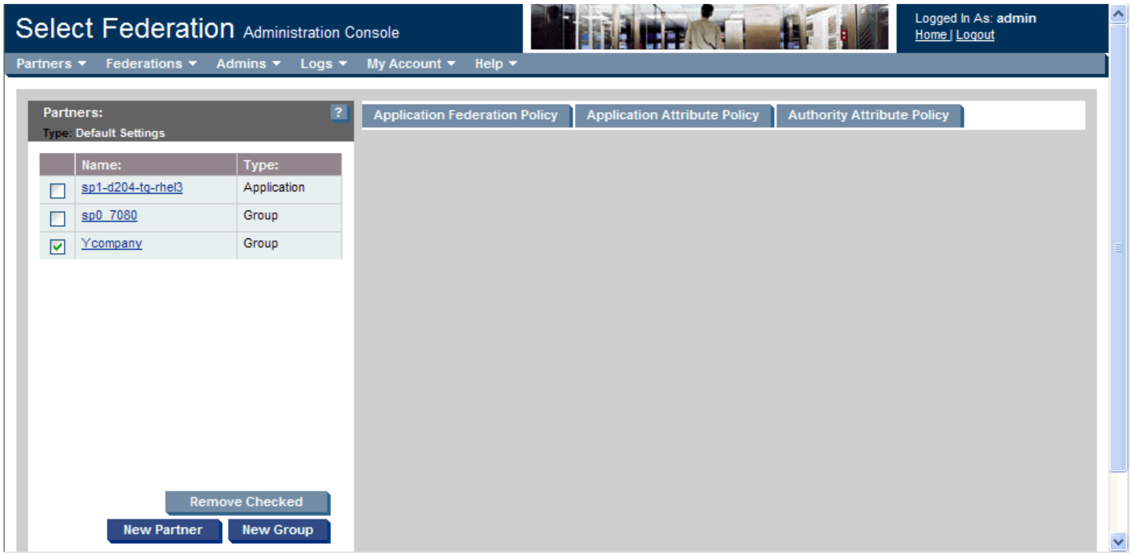
- View the details of each Partner with which you have a federation

- Edit the details of each site
- Remove, and/or change the configuration

To edit partner settings, perform the following steps:

1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

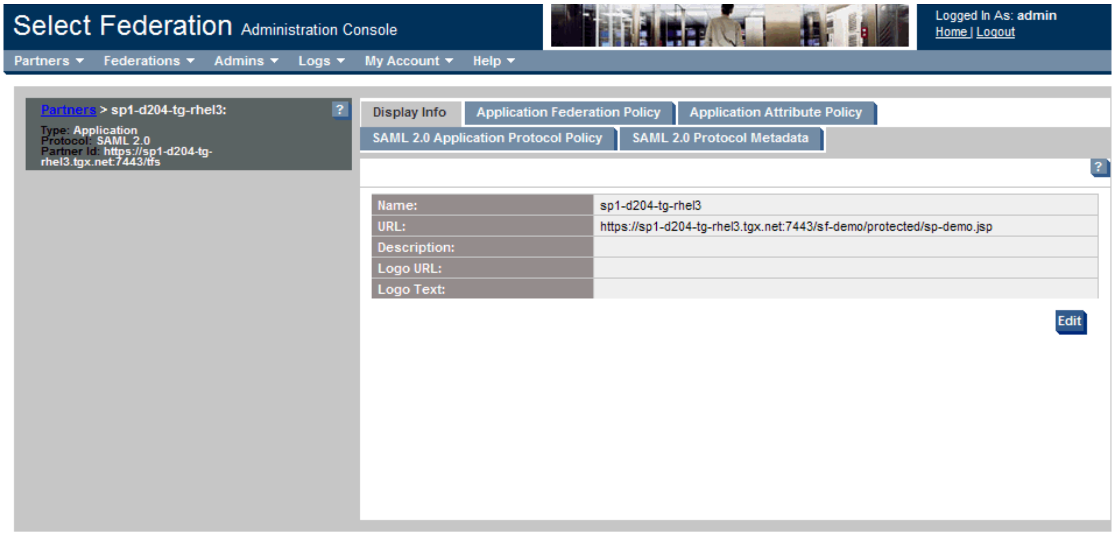

Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
Application Federation Policy
Application Attribute Policy
Authority Attribute Policy
Remove Checked
New Partner
New Group

Partners:
Type: Default Settings

	Name:	Type:
□	sp1-d204-tg-rhel3	Application
□	sp0_7080	Group
☑	Ycompany	Group

2 Click on the **Name** of the site that you want to edit.

The Display Info page opens with basic information related to this Partner.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
Edit

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Name:	sp1-d204-tg-rhel3
URL:	https://sp1-d204-tg-rhel3.tgx.net:7443/sf-demo/protected/sp-demo.jsp
Description:	
Logo URL:	
Logo Text:	

The following information displays in the left panel.

- **Partners:** Displays the Partner or Group name as a link. When you click on the name link, the first Partners left panel (select **Partners** → **Manage Partners**) displays.

For a subgroup, displays the parent Group name as the link after the Partners link. A link is added for each parent Group and subgroup. When you click on a parent Group name, the information for that Group displays in the right panel in the Display Info tab.

- **Type:**
  - For a Partner, displays the site role: Authority, Application, or both.
  - For a Group, always displays Group.
- **Protocol:** Displays the protocol that this federation used. It can be SAML 2.0, SAML 1.1, SAML 1.0, Liberty 1.2 or Liberty 1.1, ADFS (WS-Federation 1.0).
- **Partner Id:** Displays the URL for the Trusted site with which you have a federation.

3. Do one of the following:
   - Click the **Edit** button to edit Partner details. See Configuring the Display Information.
   - Select the (Application for an SP Partner or Authority for an IDP Partner) Federation Policy tab. See Configuring the Federation Policy on page 48.
   - Select the (Application for an SP Partner or Authority for an IDP Partner) Attribute Policy tab. See Configuring the Attribute Policy on page 51.
   - Select the Protocol Policy tab that is available for your selected Partner. For example, if your Partner is an SP, the tab may be SAML 2.0 Application Protocol Policy. See Configuring the Protocol Policy and Metadata on page 53.
   - Select the Protocol Metadata tab that is available for your selected Partner. For example, for either SP or IDP partners, the tab may be SAML 2.0 Protocol Metadata. See Configuring the Protocol Policy and Metadata on page 53.

Configuring the Display Information

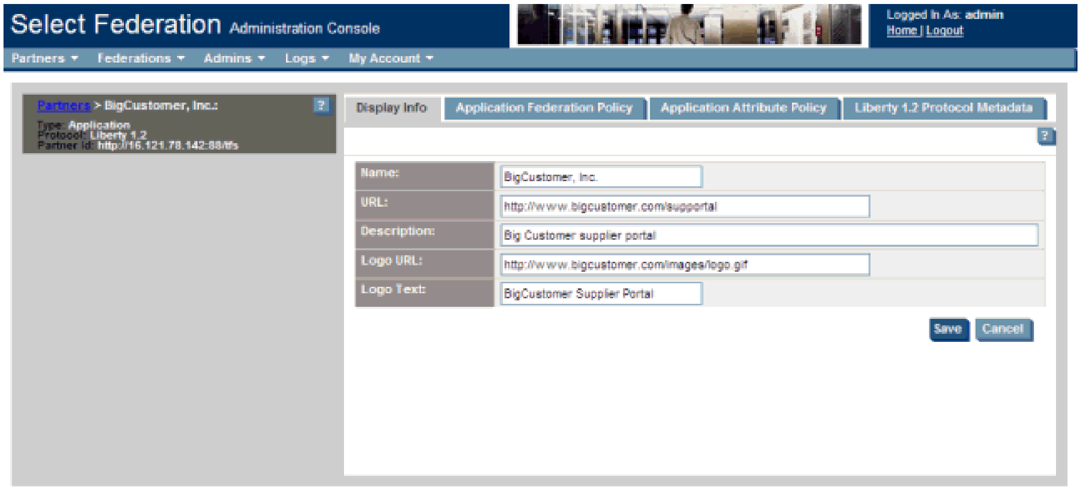
The Display Info page contains the data that determines how your Partner appears on your federation web page. This includes your Partner's name, URL, and/or logo. Only the Partner name is required. The information on this Display Info page can also be used for Groups.

The only required field is the **Name**. All other fields are optional.

To add or change your Partner's detail information, perform the following steps:

1. Perform step 1 through step 2 in Changing Settings for a Partner on page 44 to open the Display Info page.
2. Click **Edit** in the Display Info page.

   The editable Display Info page opens.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾
Partners > BigCustomer, Inc.:
?
Display Info
Application Federation Policy
Application Attribute Policy
Liberty 1.2 Protocol Metadata
?
Save
Cancel

Type: Application
Protocol: Liberty 1.2
Partner Id: http://16.121.78.142:88/tfs

Name:	BigCustomer, Inc.
URL:	http://www.bigcustomer.com/supportal
Description:	Big Customer supplier portal
Logo URL:	http://www.bigcustomer.com/images/logo.gif
Logo Text:	BigCustomer Supplier Portal

This page allows you to add or change any of the fields. Filling in the optional fields help the look-and-feel of the applications that process your federation information. These optional fields allow you to import your Partner's logo and link directly to your Partner's web page.

3 Optionally, change your Partner's site name as you would like it to appear in your system.

Your Partner's site name is automatically entered. The **Name** field must be filled in.

4 Optionally, fill in any of the following fields.

These fields are used to distinguish your Partner:

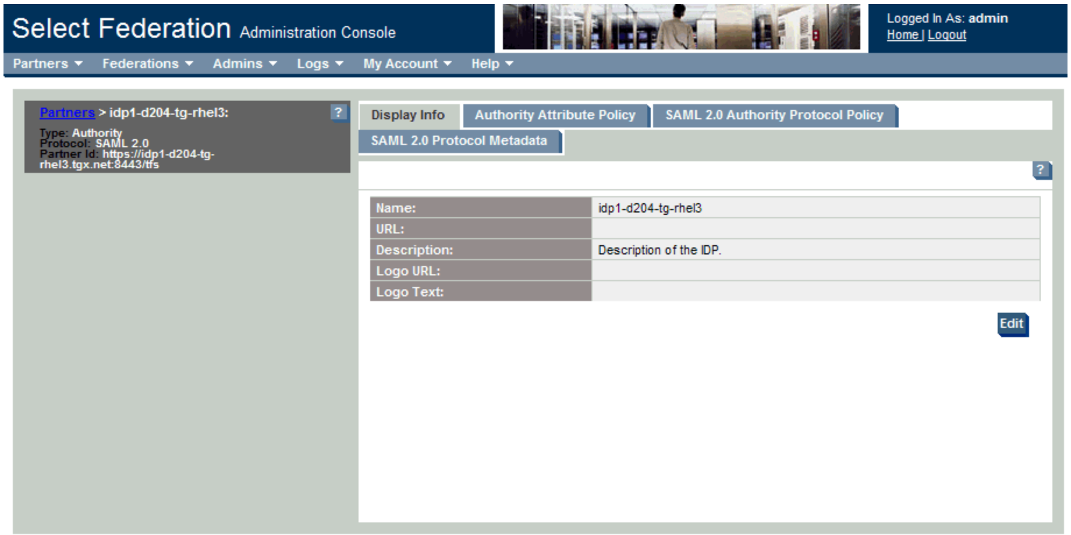
- **URL:** Your Partner's URL that users may use as a home page, if this site acts as an IDP portal.

➤ This URL is NOT a link to the Partner's ProviderId.

- **Description:** You can enter a one-line description of the Partner Site to which you are connecting.
- **Logo URL:** Logo that appears on your portal and represents the logo of the federation link you created. It can be your Partner's logo.
- **Logo Text:** Text that appears in the bubble when you put your mouse over the Logo URL.

5 Click **Save**.

The saved Display Information opens in an uneditable page.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Display Info
Authority Attribute Policy
SAML 2.0 Authority Protocol Policy
SAML 2.0 Protocol Metadata
?
Edit

Partners > idp1-d204-tg-rhel3:

Type: Authority
Protocol: SAML 2.0
Partner Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs

Name:	idp1-d204-tg-rhel3
URL:	
Description:	Description of the IDP.
Logo URL:	
Logo Text:	

Configuring the Federation Policy

The federation policy is the set of rules that both you and your Partner agreed to use in communicating between the sites. These rules are comprised of the following, which are described in detail in the topics that follow:

- Name Federation or the form of the user name
- User Consent

Name Federation

Select Federation allows users to connect to Partner web sites in three ways:

- Using the user's local name which has identifiable user information.
- Using a unique identifier that does not reveal the users' identities to outside sites.
- Total anonymity. This is accomplished through the following identifiers:
  - **Local Names** — names (or identifiers) that the users are known by at the Identity Provider or Authority site. The Authority Site may also elect to pass some user information to the Service Provider. One Time Pseudonyms and Pseudonyms are generated and used in place of the Local Name.
  - **Pseudonyms** — identifiers that are randomly generated to keep the user's local identity unknown to the Service Provider. However, unlike the One Time Pseudonym, each time the user goes to the Partner site, the same identifier is presented. The Service Provider or Application site will know that this user has been active at its site.
  - **One Time Pseudonyms** — identifiers that are randomly generated each time the user accesses a Trusted site, providing the strongest privacy to users.

➤ If you are using the one-time pseudonym name federation policy, you will not be able to link accounts on an IDP / Authority to an existing account on an SP / Application. This is because the linked account at the SP will refer to a federated identifier that will never be repeated by the IDP, so a user navigating to the SP for a second time will not be recognized as a returning user.

User Consent

The first time a user goes to a new Trusted Partner Site, the user has the option to consent to sending the user's federated name identifier to this Trusted Site. The type and level of user information that would be transmitted are determined by the Authority Site (or Identity Provider or SAML Producer). The administrator decides whether User Consent is **Required** or **Not Required**. To enable attribute consent, see Configuring the Use of End User Privacy Policies on page 142.

If your site is the Authority Site, you can edit the site federation policy for each Application Site in your federation by performing the following steps:

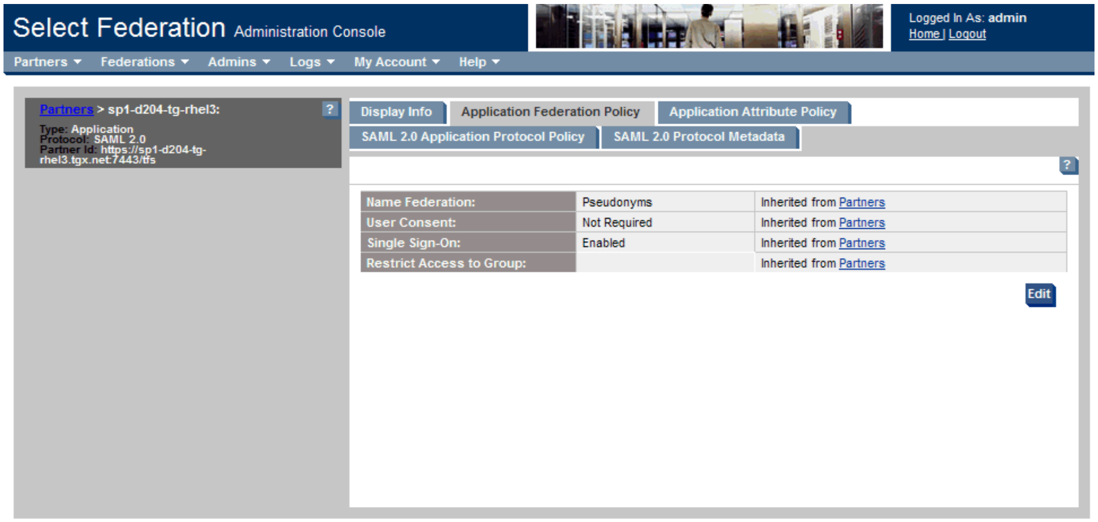
1. Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

   The Partners page opens.

2. Click on the **Name** of the site with which you have a federation that you want to edit.

3. Select the (Application/Authority) **Federation Policy** tab.

   In the following example, the Application Federation Policy page opens.

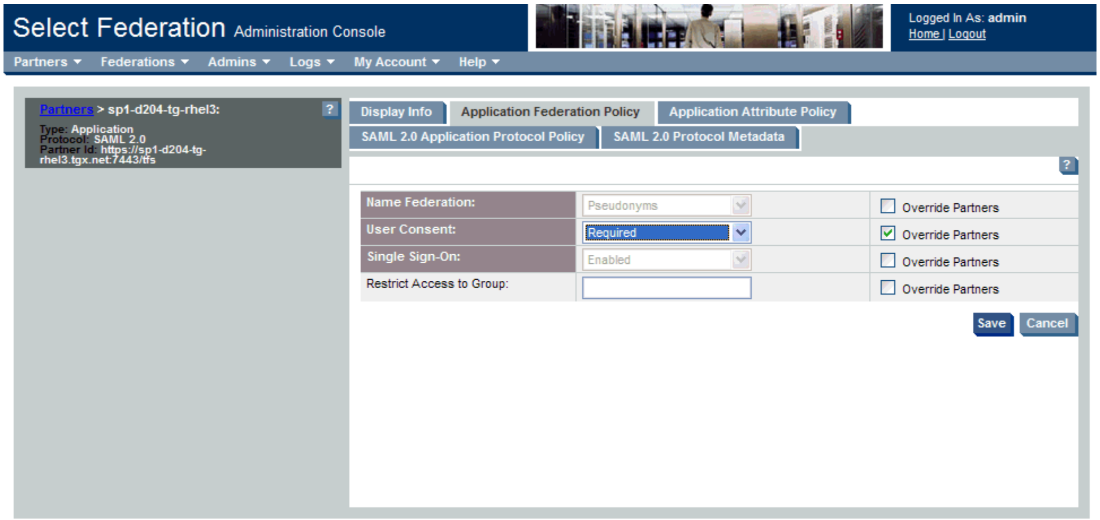

Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
?
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
?
Edit

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Name Federation:	Pseudonyms	Inherited from Partners
User Consent:	Not Required	Inherited from Partners
Single Sign-On:	Enabled	Inherited from Partners
Restrict Access to Group:		Inherited from Partners

4. Click **Edit** to change your Federation Policy.

   The editable Federation Policy page opens.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
?
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
Save
Cancel

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Name Federation:	Pseudonyms	□ Override Partners
User Consent:	Required	☑ Override Partners
Single Sign-On:	Enabled	□ Override Partners
Restrict Access to Group:		□ Override Partners

Initially the Application Federation Policy is inherited from the parent Group where the Partner resides. You can either edit the parent Group's Application Federation Policy to affect all Partner's under the Group in the same way or override the Group Policy and edit the Partner's Policy by itself.

5 Check the **Override** check box if you want to change your Partner's Application Federation Policy from the Group policy.

6 Make the needed changes to each of the fields.

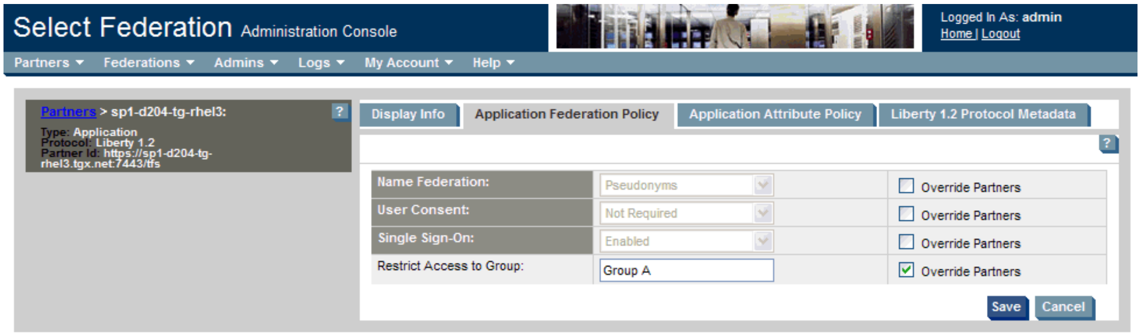
If you set **Single Sign-On** to **Disabled**, then the user must be re-authenticated each time the user logs in.

7 Click **Save** to save your settings.

Restrict Access to Group

A Select Federation administrator can choose which of the local users have access to certain partner sites by specifying a value to the **Restrict Access to Group** field in the **Application Federation Policy** tab as shown in the following figure.

Figure 7 Restrict Access to Group Field on the Application Federation Policy Page


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
?
Display Info
Application Federation Policy
Application Attribute Policy
Liberty 1.2 Protocol Metadata
Save
Cancel

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: Liberty 1.2
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Name Federation:	Pseudonyms	□ Override Partners
User Consent:	Not Required	□ Override Partners
Single Sign-On:	Enabled	□ Override Partners
Restrict Access to Group:	Group A	☑ Override Partners

The Group name depends upon the directory plugin being used, since this name is passed verbatim to the Directory Plugin's “is Member” method. Out-of-the-box Select Federation uses an LDAP directory plugin so the value of this field should be the LDAP Distinguished Name (DN) of the LDAP group of which the user must be a member in order to access that site.

If you have specified a custom Directory Plugin, this parameter is passed verbatim to the `DirPlugin:isMember` method of the Directory Plugin

➤ When using the Sun Directory Server, make sure that the type of group that you create in the directory is a “Role” and sets the `nsRole` attribute in the user object. Also change the value of the configuration parameter `ldapGroupMembershipAttr` to be `nsRole` in the `tfsconfig.properties` file.

Configuring the Attribute Policy

Applications typically need attributes about the authenticated users. In a federated system, the most recent values for these user attributes are at the original source of the authentication, that is, the Identity Provider or SAML Producer. The Profile Service is a module in Select Federation that allows you to transmit user attributes on every user authentication.

Introduction

Using the Liberty Profile Services (Personal Profile Service – ID-PP or Employee Profile Service – ID-EP) or the SAML Attribute Authority, Select Federation provides these user attributes to the application residing at the federated site, that is, the Service Provider or SAML Consumer. You and your Partner will need to agree on the attributes to be exchanged.

An Application must be configured with the attributes it “requests” from its Authority partners whereas an Authority must be configured with the attributes it is “willing to consider for release” to its Application partners. The set of attributes to request or release is called an Attribute Policy. The remainder of the chapter explains how to configure Attribute Policies.

Select Federation has a built-in profile service that uses the system configuration to determine the sources of attribute information. This is explained in more detail in Chapter 7, Configuring Attributes. An Application does not use the profile service of its own installation but requests attributes from its Authority partners. The release of profile attributes to applications can be subject to end user consent (see Chapter 9, Configuring Privacy Manager).

Configuring Your Attribute Policy

Select Federation allows you to configure and change the user profile attribute exchange for Groups and Partners. Partners can inherit their Application Attribute Policy from a Group if so desired. This can be helpful when many Partners share a similar configuration.

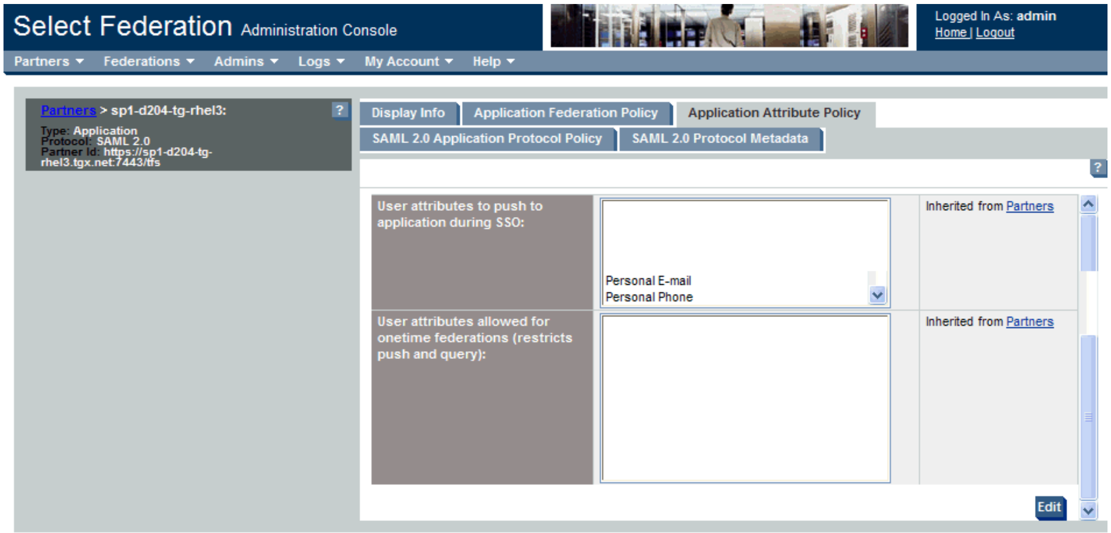
Perform the following steps to set or change your profile attribute policy:

1. Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

   The Partners page opens.

2. Click the **Name** of the Group or Partner that you wish to edit.
3. Select the (Application/Authority) **Attribute Policy** tab.

The Attribute Policy page opens, which delineates all the existing attribute policy parameters, if any, for this Liberty federation.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
?
Edit

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

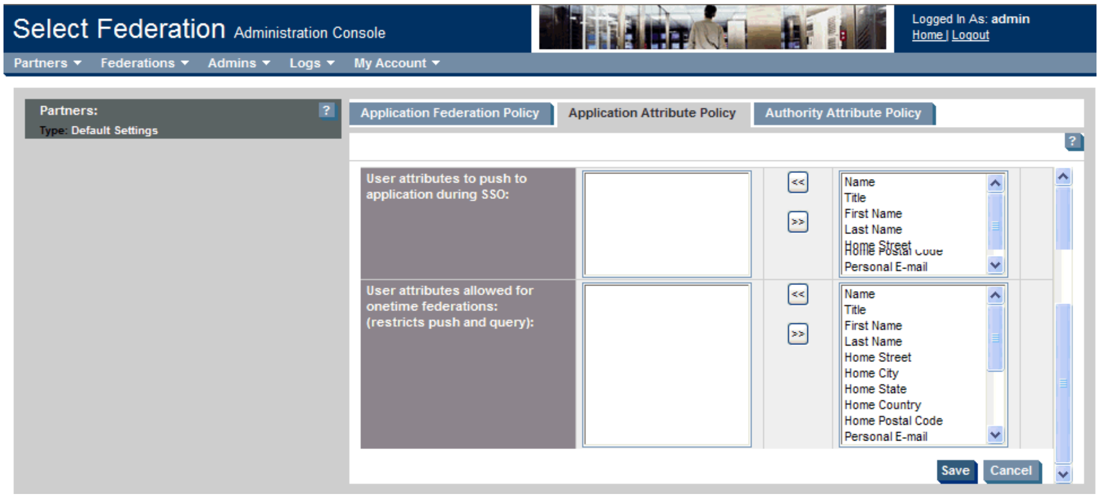
User attributes to push to application during SSO: Personal E-mail
Personal Phone — Inherited from Partners

User attributes allowed for onetime federations (restricts push and query): — Inherited from Partners

Initially the Application Attribute Policy is inherited from the parent Group where the Partner resides. You can either edit the parent Group's Application Attribute Policy to affect all Partner's under the Group in the same way or override the Group Policy and edit the Partner's Policy by itself.

4 Click the **Edit** button.

The Application Attribute Policy page opens to a page similar to the following figure with attribute choices in the right panel.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Application Federation Policy
Application Attribute Policy
Authority Attribute Policy
?
<<
>>
Save
Cancel

Partners:
Type: Default Settings

User attributes to push to application during SSO:
Name
Title
First Name
Last Name
Home Street
Home Postal Code
Personal E-mail

User attributes allowed for onetime federations: (restricts push and query):
Name
Title
First Name
Last Name
Home Street
Home City
Home State
Home Country
Home Postal Code
Personal E-mail

The attributes that may be conveyed at the time of single sign on from the Authority Partner (IDP or SAML Producer) to the Application Partner (SP or SAML Consumer) are the following:

- **Application Attribute Policy**
  - **User attributes to push to application during SSO:** The attributes that are pushed from the Authority Partner to the Application Partner each time users log in to the application.

- — **User attributes to allow application to query**: The additional attributes that the Application Partner is allowed to pull from the Authority Partner. They are attributes that were not pushed by the Authority Partner in the initial sign on. The Application Partner queries the Liberty Profile Service or SAML Attribute Authority for this information.
- — **User attributes allowed for one time federations (restricts push and query)**: If the Federation Policy is set for anonymous logins using the One Time Pseudonyms, you can set user attributes for the one-time logins, if desired.

- **Authority Attribute Policy**
  - — **User attributes to obtain from authority on each login:** Each time the user executes a transaction at the Application Partner this user information is retrieved from the Authority Partner.
  - — **Additional user attributes to obtain from authority on activation**: The first time a new user accesses the Application Partner, these are the user attributes that the Application Partner needs from the Authority Partner to activate the user account.

5. Select the user attributes in the right panel that you would like to pass for each login.
6. Click the left double arrows to add the attributes.

   You can select more than one attribute in each category to add at once by using the **<Ctrl>** key to select multiple options, or the **<Shift>** key to select a range of options.
7. Click **Save** when you are finished.

Configuring the Protocol Policy and Metadata

HP OpenView Select Federation makes it relatively easy to make changes to your Partner's protocol policy and/or protocol metadata. Due to the differences between the Liberty and SAML specifications, the way to update existing federation links differs as explained in the following sections:

- Updating Your SAML Application Protocol Metadata
- Updating Your SAML Application Protocol Policy
- Updating Your SAML Authority Protocol Metadata
- Updating Your SAML Authority Protocol Policy
- Updating Your Liberty Protocol Metadata
- Updating Your ADFS (WS-Federation 1.0) Protocol Metadata

➤ Changing your SAML Protocol parameters can be a two-step process if you need to make changes in both your Protocol Metadata and Protocol Policy. It is recommended that you first update your Protocol Metadata, and then update your Protocol Policy.

Updating Your SAML Application Protocol Metadata

Perform the following steps to update your SAML Application protocol metadata:

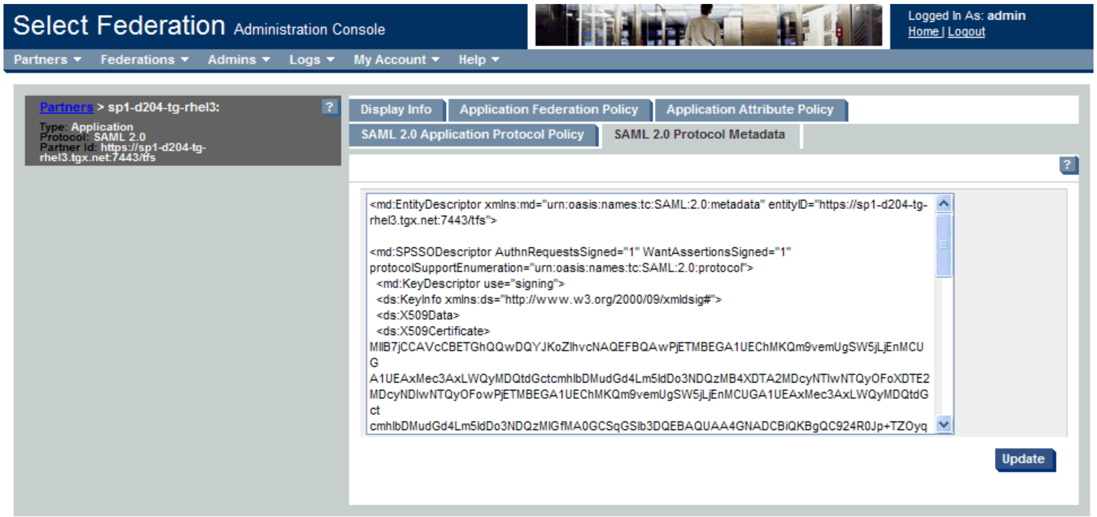
1. Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

   The Partners page opens.

2 Click the **Name** of the Partner that you wish to update.

3 Select the **SAML Application Protocol Metadata** tab.

The SAML Application Protocol Metadata page opens, which delineates all the certificate information, protocol policy, and URLs that are needed for this SAML Partner.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Partners > sp1-d204-tg-rhel3:
?
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
?
Update

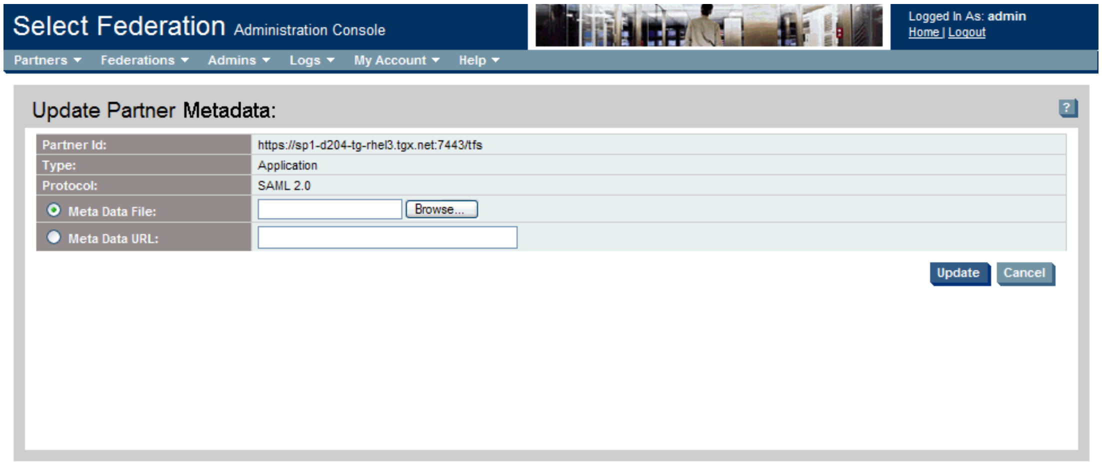
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

```
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata" entityID="https://sp1-d204-tg-rhel3.tgx.net:7443/tfs">

<md:SPSSODescriptor AuthnRequestsSigned="1" WantAssertionsSigned="1"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
  <md:KeyDescriptor use="signing">
  <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:X509Data>
  <ds:X509Certificate>
MIIB7jCCAVcCBETGhQQwDQYJKoZIhvcNAQEFBQAwPjETMBEGA1UEChMKQm9vemUgSW5jLjEnMCU
G
A1UEAxMec3AxLWQyMDQtdGctcmhlbDMudGd4Lm5ldDo3NDQzMB4XDTA2MDcyNTIwNTQyOFoXDTE2
MDcyNDIwNTQyOFowPjETMBEGA1UEChMKQm9vemUgSW5jLjEnMCUGA1UEAxMec3AxLWQyMDQtdG
ct
cmhlbDMudGd4Lm5ldDo3NDQzMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC924R0Jp+TZOyq
```

4 Click the **Update** button.

The Update Partner Metadata page opens, such as the following example.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
?
Browse...
Update
Cancel

Update Partner Metadata:

Partner Id:	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs
Type:	Application
Protocol:	SAML 2.0
● Meta Data File:	
○ Meta Data URL:	

You can update your Partner’s SAML Consumer (Application Partner or SP) policy information.

If your Trusted Partner uses SAML 1.X, it is possible that no metadata file or URL link is available. In this case you need to enter the updated information manually. The appropriate fields for manual editing of metadata are shown when you click the **Update** button. For descriptions of the fields see Manually Adding the SAML 1.x Authority Protocol Metadata on page 39 and Manually Adding the SAML 1.x Application Protocol Metadata on page 40.

5 Click the **Update** button to save your changes.

Updating Your SAML Application Protocol Policy

Perform the following steps to update your SAML Application Protocol policy:

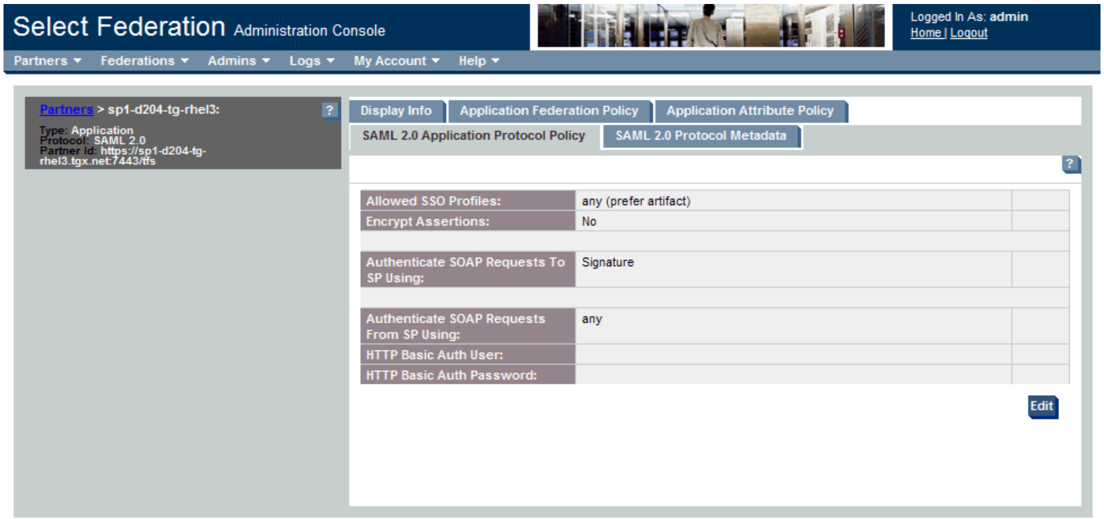
1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

2 Click the **Name** of the Group or Partner that you wish to update.

3 Select the **SAML** (Application/Authority) **Protocol Policy** tab.

The SAML Application Protocol Policy page opens. The following figure shows an example of an SP Partner's SAML Protocol Policy page.

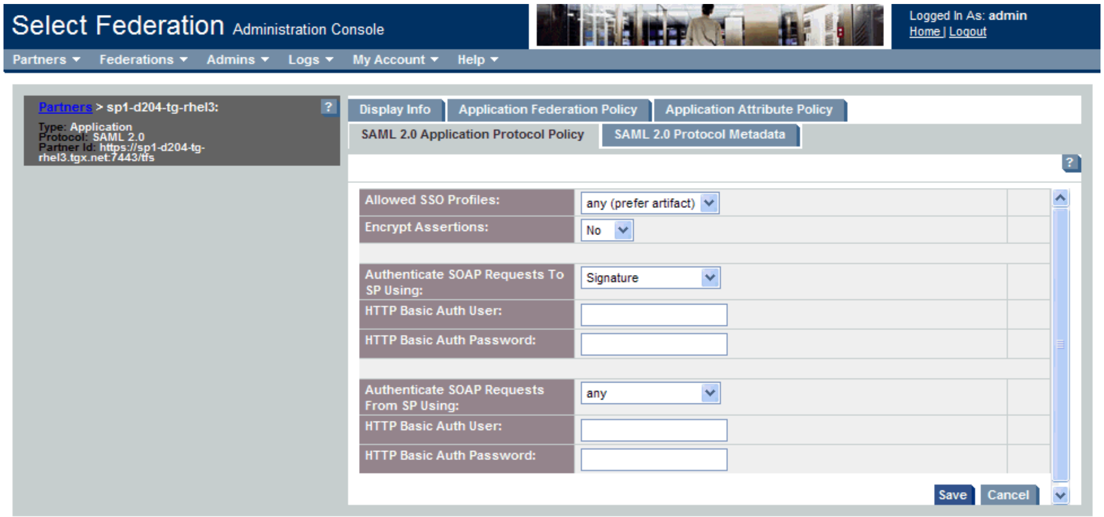

Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
Edit

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Allowed SSO Profiles:	any (prefer artifact)
Encrypt Assertions:	No
Authenticate SOAP Requests To SP Using:	Signature
Authenticate SOAP Requests From SP Using:	any
HTTP Basic Auth User:	
HTTP Basic Auth Password:	

4 Click the **Edit** button.

The editable SAML Application Protocol Policy page opens as shown in the following example.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Display Info
Application Federation Policy
Application Attribute Policy
SAML 2.0 Application Protocol Policy
SAML 2.0 Protocol Metadata
Save
Cancel

Partners > sp1-d204-tg-rhel3:
Type: Application
Protocol: SAML 2.0
Partner Id: https://sp1-d204-tg-rhel3.tgx.net:7443/tfs

Allowed SSO Profiles:	any (prefer artifact)
Encrypt Assertions:	No
Authenticate SOAP Requests To SP Using:	Signature
HTTP Basic Auth User:	
HTTP Basic Auth Password:	
Authenticate SOAP Requests From SP Using:	any
HTTP Basic Auth User:	
HTTP Basic Auth Password:	

5 (For SAML 1.1 and 2.0) Set the allowed SSO profile in your SAML SP Policy for an SP Partner.

If your site is the SAML Producer (Authority or IDP) and you will receive authentication requests from your SAML Consumer Partners (Application or SP), you need to set the single sign-on parameters in this section of the web page.

The **Allowed SSO Profiles** are:

- **Any (prefer artifact)**: Your system will accept any SSO profile, but will prefer artifact.
- **Any (prefer post)**: Your system will accept any SSO profile, but will prefer post.
- **Artifact**: Your system will only accept artifact profiles.
- **Post**: Your system will only accept post profiles.

6 (For SAML 2.0) Set the **Encrypt Assertions** to Yes or No.

7 (Set the SOAP authentication method in your SAML SP Policy for an SP Partner.

In the post profile, a SAML assertion is sent from the SAML Producer (Authority or IDP) to the SAML Consumer (Application or SP) through the browser only and SOAP requests are not used. In the artifact profile, a pointer is sent from the IDP to the SP through the browser and a SOAP call is set from the SAML Consumer (Application or SP) to the SAML Producer (Authority or IDP) using one of the four SOAP Authentication methods.

a (For SAML 2.0) Select one of the three SOAP Authentication methods that your site will use to **Authenticate SOAP Requests to SP Using**:

— **Signature**: SAML Digital Signature-based Authentication.

— **SSL/TLS Client Authentication**: Certificate-based authentication for authenticating the SAML consumer to the SAML producer. The SAML consumer presents an SSL client certificate to successfully establish a secure SSL / TLS channel for picking up the SAML artifact.

— **HTTP Basic Authentication**: Must have a user name and password.

If you select HTTP Basic Authentication, you need to enter the user name and password in the fields **HTTP Basic Auth User** and **HTTP Basic Auth Password**.

b (For all SAML versions) Select one of the following SOAP authentication methods that your site will use to **Authenticate SOAP Requests From SP Using**:

— **Signature**: SAML Digital Signature-based Authentication

The **Signature** authentication method does not work for SAML 1.0 due to a limitation in the protocol specification. Therefore, it is important to configure the authentication method for SAML 1.0 to be anything other than **Signature**.

— **SSL/TLS Client Authentication** Certificate-based authentication for authenticating the SAML consumer to the SAML producer. The SAML consumer presents an SSL client certificate to successfully establish a secure SSL/TLS channel for picking up the SAML artifact.

Using the client authentication method requires both partners to be using the HTTPS protocol.

— **HTTP Basic Authentication** using a username and password. If you select this authentication method, you need to enter the desired username and password in the fields **HTTP Basic Auth User** and **HTTP Basic Auth Password**.

— **Any** of the above.

8 Click the **Save** button to save your changes.

Updating Your SAML Authority Protocol Metadata

Perform the following steps to update your SAML Authority Partner protocol metadata:

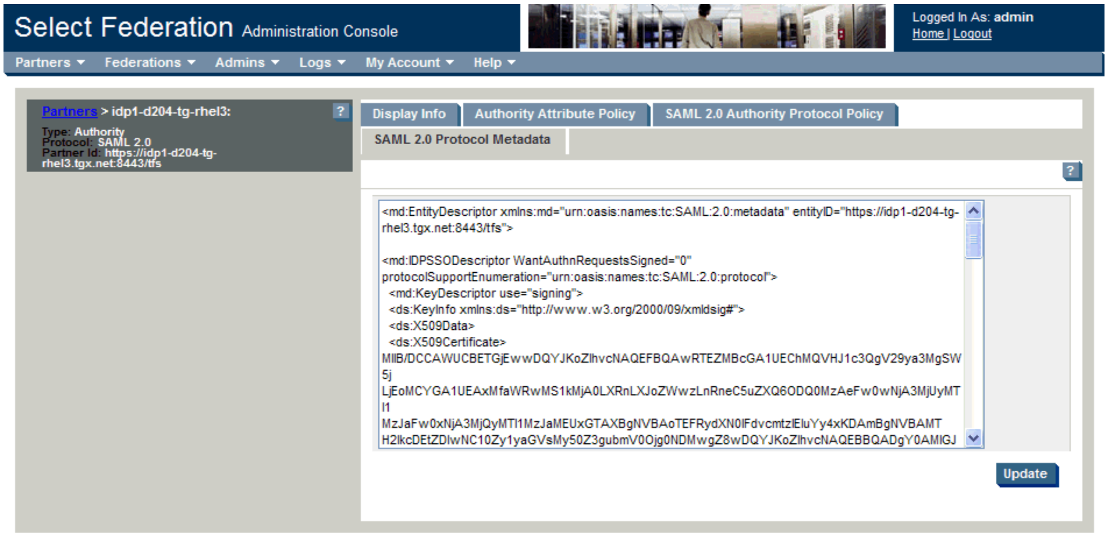
1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

2 Click the **Name** of the Partner that you wish to update.

3 Select the **SAML Protocol Metadata** tab.

The SAML Authority Partner Protocol Metadata page opens. This page delineates all the certificate information, protocol policy, and URLs that are needed for this SAML Partner.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Partners > idp1-d204-tg-rhel3:
?
Display Info
Authority Attribute Policy
SAML 2.0 Authority Protocol Policy
SAML 2.0 Protocol Metadata
Update

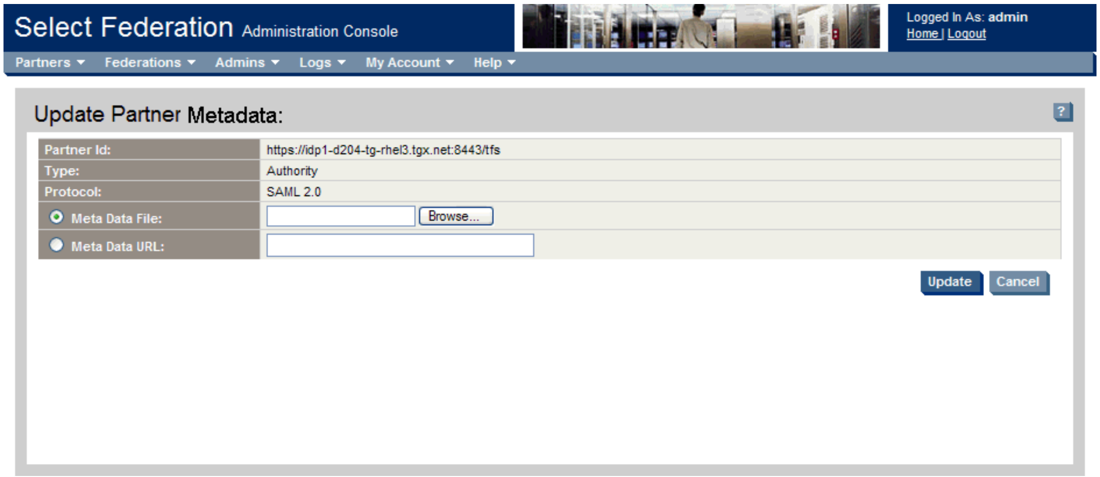
Type: Authority
Protocol: SAML 2.0
Partner Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs

```
<md:EntityDescriptor xmlns:md="urn:oasis:names:tc:SAML:2.0:metadata" entityID="https://idp1-d204-tg-
rhel3.tgx.net:8443/tfs">

<md:IDPSSODescriptor WantAuthnRequestsSigned="0"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
  <md:KeyDescriptor use="signing">
  <ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:X509Data>
  <ds:X509Certificate>
MIIB/DCCAWUCBETGjEwwDQYJKoZIhvcNAQEFBQAwRTEZMBcGA1UEChMQVHJ1c3QgV29ya3MgSW
5j
LjEoMCYGA1UEAxMfaWRwMS1kMjA0LXRnLXJoZWwzLnRneC5uZXQ6ODQ0MzAeFw0wNjA3MjUyMT
I1
MzJaFw0xNjA3MjQyMTI1MzJaMEUxGTAXBgNVBAoTEFRydXN0IFdvcmtzIEluYy4xKDAmBgNVBAMT
H2lkcDEtZDIwNC10Zy1yaGVsMy50Z3gubmV0Ojg0NDMwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJ
```

4 Click the **Update** button to update your Partner's SAML Producer (Authority or IDP) policy information.

The Update Partner Metadata page opens as shown in the following example.



Update Partner Metadata:

Partner Id:	https://idp1-d204-tg-rhel3.tgx.net:8443/tfs
Type:	Authority
Protocol:	SAML 2.0
● Meta Data File:	
○ Meta Data URL:	

If your Trusted Partner uses SAML 1.X, it is possible that no metadata file or a URL link is available. In this case you need to enter the updated information manually. See Manually Adding the SAML 1.x Authority Protocol Metadata on page 39 for instructions.

5 You can either upload your Partner's metadata file or get the information from a URL.

- **Metadata File**: Enter or browse to find the full path of the metadata file that you received from your partner.
- **Metadata URL**: Enter the URL where the metadata information from your partner is stored.

6. Click the **Update** button to save your changes.

Updating Your SAML Authority Protocol Policy

Perform the following steps to update your SAML Authority protocol policy:

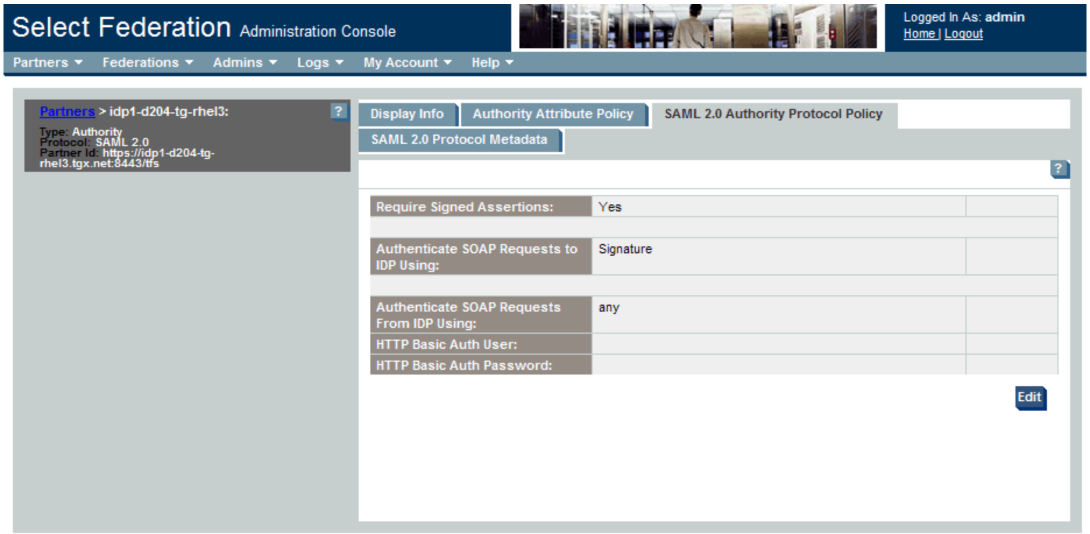
1. Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

   The Partners page opens.

2. Click the **Name** of the Partner that you wish to update.

3. Select the **SAML Authority Protocol Policy** tab.

   The SAML Authority Protocol Policy page opens.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
Display Info
Authority Attribute Policy
SAML 2.0 Authority Protocol Policy
SAML 2.0 Protocol Metadata
?
Edit

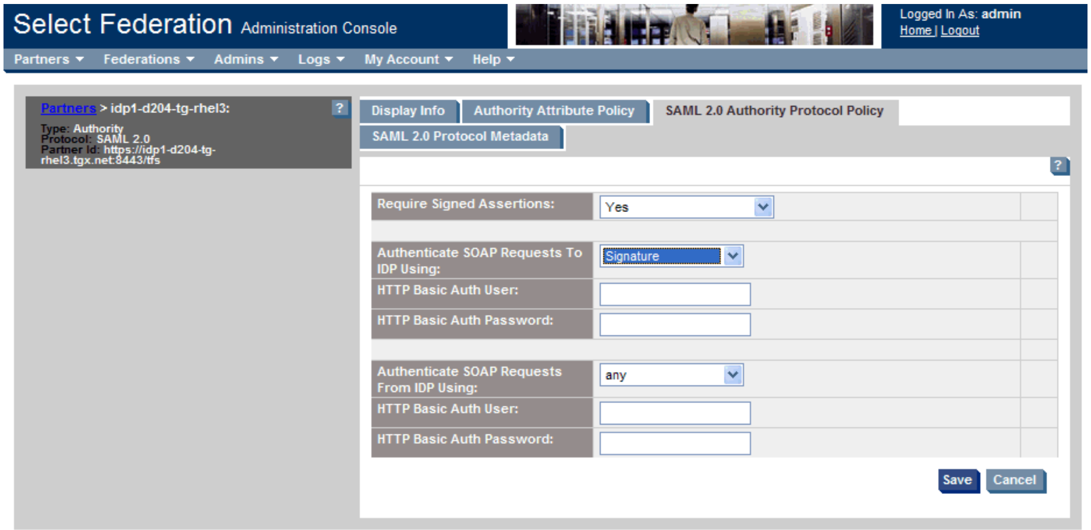
Partners > idp1-d204-tg-rhel3:

Type: Authority
Protocol: SAML 2.0
Partner Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs

Require Signed Assertions:	Yes
Authenticate SOAP Requests to IDP Using:	Signature
Authenticate SOAP Requests From IDP Using:	any
HTTP Basic Auth User:	
HTTP Basic Auth Password:	

4. Click the **Edit** button.

   The editable SAML Authority Protocol Policy page opens as shown in the following example.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Partners > idp1-d204-tg-rhel3:
Display Info
Authority Attribute Policy
SAML 2.0 Authority Protocol Policy
SAML 2.0 Protocol Metadata
Save
Cancel

Type: Authority
Protocol: SAML 2.0
Partner Id: https://idp1-d204-tg-rhel3.tgx.net:8443/tfs

Require Signed Assertions:	Yes
Authenticate SOAP Requests To IDP Using:	Signature
HTTP Basic Auth User:	
HTTP Basic Auth Password:	
Authenticate SOAP Requests From IDP Using:	any
HTTP Basic Auth User:	
HTTP Basic Auth Password:	

If your site is a SAML Consumer (Application or SP) and you will be sending artifacts to the SAML Producer (Authority or IDP), you need to set the single sign-on parameters on this web page.

5 (For SAML 1.1 and 1.2) Select Yes or No for the **Require Signed Assertions**.

6 (For all SAML versions) Select one of the three SOAP Authentication methods that your site will use to **Authenticate SOAP Requests to the IDP**.

— **Signature**: SAML Digital Signature-based Authentication.

➤ The **Signature** authentication method does not work for SAML 1.0 due to a limitation in the protocol specification. Therefore, it is important to configure the authentication method for SAML 1.0 to be anything other than **Signature**.

- **SSL/TLS Client Authentication**: Certificate-based authentication for authenticating the SAML consumer to the SAML producer. The SAML consumer presents an SSL client certificate to successfully establish a secure SSL / TLS channel for picking up the SAML artifact.
- **HTTP Basic Authentication:** Must have a user name and password.

  If you select HTTP Basic Authentication, you need to enter the user name and password in the fields **HTTP Basic Auth User** and **HTTP Basic Auth Password**.

7 (For SAML 2.0) Select one of the following SOAP Authentication methods that your site will use to **Authenticate SOAP Requests from the IDP Using**:

- **any**: Any of the methods can be used.
- **Signature**: SAML Digital Signature-based Authentication.
- **SSL/TLS Client Authentication**: Certificate-based authentication for authenticating the SAML consumer to the SAML producer. The SAML consumer presents an SSL client certificate to successfully establish a secure SSL/TLS channel for picking up the SAML artifact.
- **HTTP Basic Authentication**: Must have a user name and password.

  If you select HTTP Basic Authentication, you need to enter the user name and password in the fields **HTTP Basic Auth User** and **HTTP Basic Auth Password**.

8 Click the **Save** button to save your changes.

Updating Your Liberty Protocol Metadata

Similar to setting up a new partner that uses the Liberty protocol, updating an existing Liberty protocol metadata is relatively automated.

Perform the following steps to update your Liberty protocol metadata:

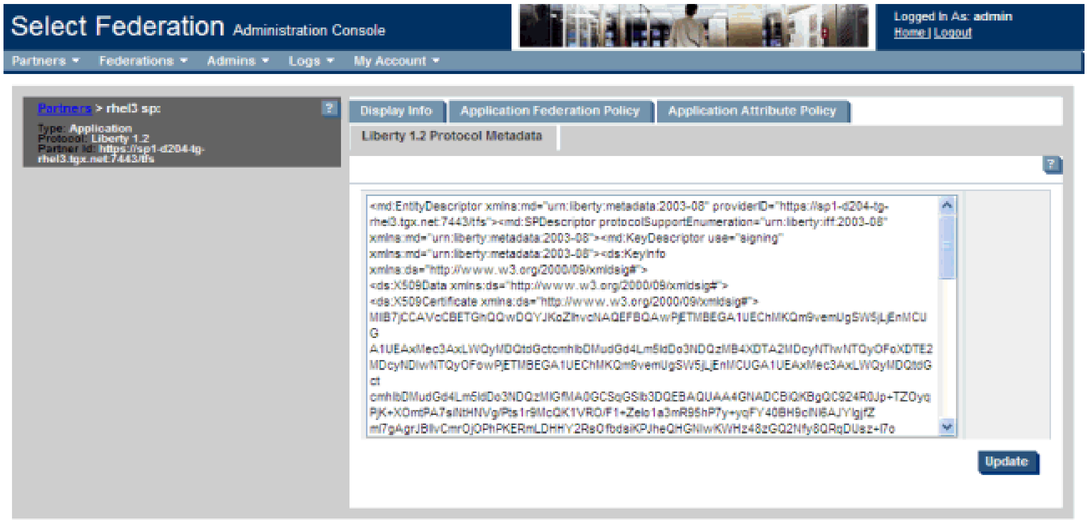
1. Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

   The Partners page opens.

2. Click the **Name** of the Partner that you wish to update.

3. Select the **Liberty Protocol Metadata** tab.

   The Liberty Protocol Metadata page opens. This page shows the XML document that contains all metadata information about the partner site.

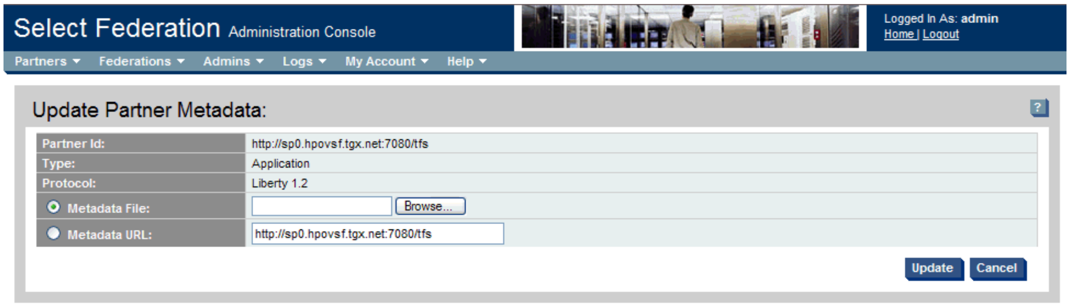

Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾
Display Info
Application Federation Policy
Application Attribute Policy
Liberty 1.2 Protocol Metadata
Update

Partners > rhel3 sp:
Type: Application
Protocol: Liberty 1.2
Partner Id: https://sp1-d2044g-rhel3.tgx.net:7443/tfs

```
<md:EntityDescriptor xmlns:md="urn:liberty:metadata:2003-08" providerID="https://sp1-d204-tg-
rhel3.tgx.net:7443/tfs"><md:SPDescriptor protocolSupportEnumeration="urn:liberty:iff:2003-08"
xmlns:md="urn:liberty:metadata:2003-08"><md:KeyDescriptor use="signing"
xmlns:md="urn:liberty:metadata:2003-08"><ds:KeyInfo
xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:X509Data xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:X509Certificate xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
MIIB7jCCAVcCBETGhQQwDQYJKoZIhvcNAQEFBQAwPjETMBEGA1UEChMKQm9vemUgSW5jLjEnMCU
G
A1UEAxMec3AxLWQyMDQtdGctcmhlbDMudGd4Lm5ldDo3NDQzMB4XDTA2MDcyNTIwNTQyOFoXDTE2
MDcyNDIwNTQyOFowPjETMBEGA1UEChMKQm9vemUgSW5jLjEnMCUGA1UEAxMec3AxLWQyMDQtdG
ct
cmhlbDMudGd4Lm5ldDo3NDQzMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC924R0Jp+TZOyq
PjK+XOmtPA7sINtHNVg/Pts1r9McQK1VRO/F1+Zelo1a3mR95hP7y+yqFY40BH9cIN6AJYlgjfZ
ml7gAgrJBllvCmrOjOPhPKERmLDHHY2Rs0fbdaiKPJheQHGNlwKWHz48zGQ2Nfy8QRqDUsz+l7o
```

Changing any information for a Liberty Partner is just a matter of uploading a new metadata file provided by your Partner.

4. Click the **Update** button.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Browse...
Update
Cancel

Update Partner Metadata:

Partner Id:	http://sp0.hpovsf.tgx.net:7080/tfs
Type:	Application
Protocol:	Liberty 1.2
● Metadata File:	
○ Metadata URL:	http://sp0.hpovsf.tgx.net:7080/tfs

5. Do one of the following:
   - Upload a new Liberty metadata file by entering or browsing for the file path name.
   - Enter the metadata URL.

6. Click the **Update** button when you are done.

Upon a successful update, the page displaying the new metadata opens.

Updating Your ADFS (WS-Federation 1.0) Protocol Metadata

Perform the following steps to update your ADFS (WS-Federation 1.0) protocol metadata:

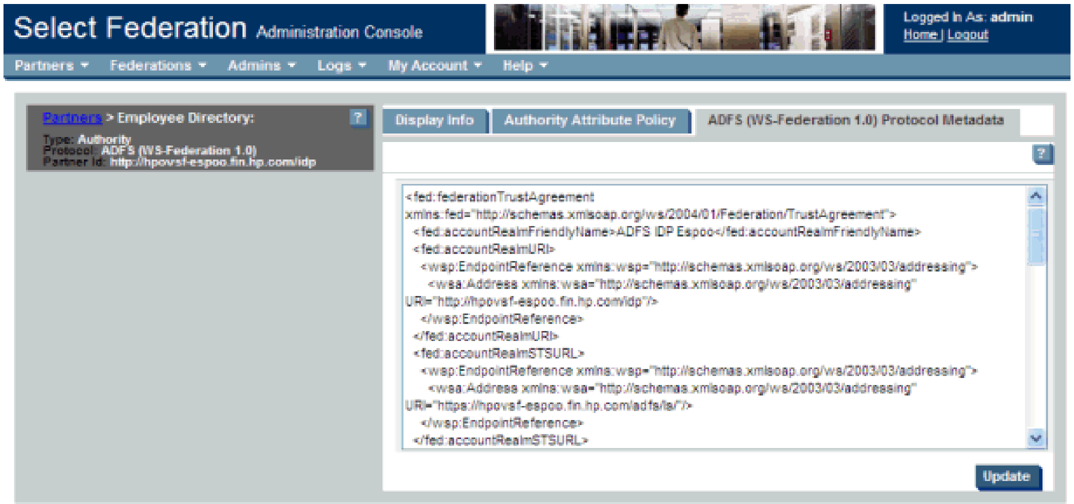
1 Select the **Partners** → **Manage Partners** menu options or click the **Manage Partners** link on the Welcome page.

The Partners page opens.

2 Click the **Name** of the Partner that you wish to update.

3 Select the **ADFS (WS-Federation 1.0) Protocol Metadata** tab.

The ADFS (WS-Federation 1.0) Protocol Metadata page opens. This page delineates all the certificate information, protocol policy, and URLs that are needed for this partner.

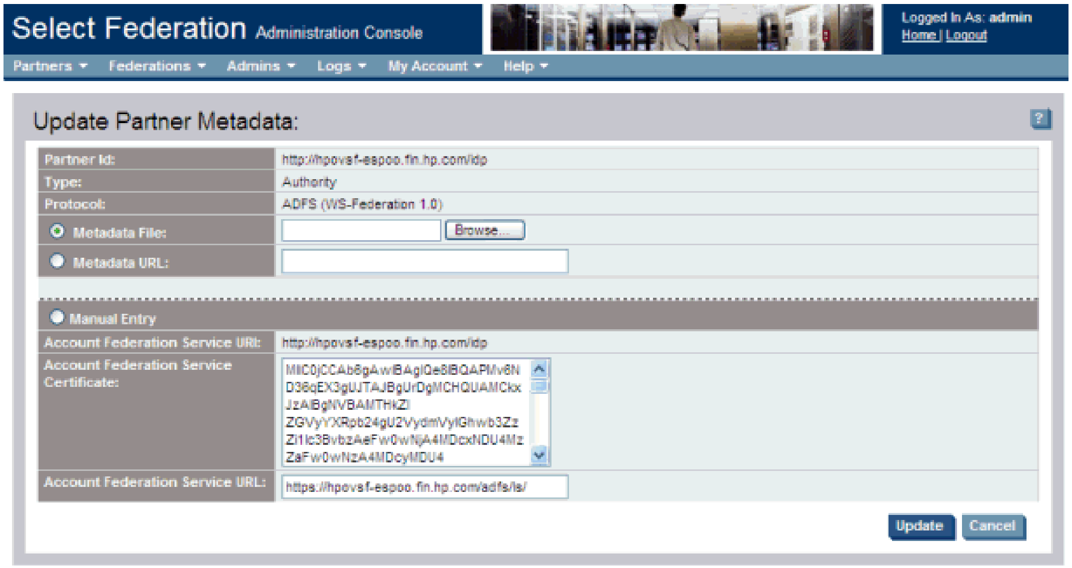

Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾
Federations ▾
Admins ▾
Logs ▾
My Account ▾
Help ▾
Display Info
Authority Attribute Policy
ADFS (WS-Federation 1.0) Protocol Metadata
?
Update

Partners > Employee Directory:
Type: Authority
Protocol: ADFS (WS-Federation 1.0)
Partner Id: http://hpovsf-espoo.fin.hp.com/idp

```
<fed:federationTrustAgreement
xmlns:fed="http://schemas.xmlsoap.org/ws/2004/01/Federation/TrustAgreement">
  <fed:accountRealmFriendlyName>ADFS IDP Espoo</fed:accountRealmFriendlyName>
  <fed:accountRealmURI>
    <wsp:EndpointReference xmlns:wsp="http://schemas.xmlsoap.org/ws/2003/03/addressing">
      <wsa:Address xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
URI="http://hpovsf-espoo.fin.hp.com/idp"/>
    </wsp:EndpointReference>
  </fed:accountRealmURI>
  <fed:accountRealmSTSURL>
    <wsp:EndpointReference xmlns:wsp="http://schemas.xmlsoap.org/ws/2003/03/addressing">
      <wsa:Address xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing"
URI="https://hpovsf-espoo.fin.hp.com/adfs/ls/"/>
    </wsp:EndpointReference>
  </fed:accountRealmSTSURL>
```

4 Click the **Update** button.

The Update Partner Metadata page opens.



Update Partner Metadata:

Partner Id:	http://hpovsf-espoo.fin.hp.com/idp
Type:	Authority
Protocol:	ADFS (WS-Federation 1.0)
● Metadata File:	
○ Metadata URL:	
○ Manual Entry	
Account Federation Service URI:	http://hpovsf-espoo.fin.hp.com/idp
Account Federation Service Certificate:	MIIC0jCCAb6gAwIBAgIQe8BQAPMv6N D36qEX3gUJTAJBgUrDgMCHQUAMCkx JzAlBgNVBAMTHkZl ZGVyYXRpb24gU2VydmVyIGhwb3Zz Zi1lc3BvbzAeFw0wNjA4MDcxNDU4Mz ZaFw0wNzA4MDcyMDU4
Account Federation Service URL:	https://hpovsf-espoo.fin.hp.com/adfs/ls/

You can update your Partner's metadata in one of the following ways:

- Upload a new ADFS (WS-Federation 1.0) metadata file.
- Download from an accessible URL.
- Manually enter the appropriate values into the manual entry fields.

5 Click the **Update** button to save your changes.

5 Other Admin Tasks

This chapter describes other administrative tasks you can perform on the Administration Console. The following topics describe these tasks:

- Auditing
- Managing Admins
- Managing Federations

Auditing

Select Federation includes the following helpful, simple-to-use auditing and administrative tools on the Administration Console:

- Server Audit Log — Logs operational activities of enabled users. All administrators can view the Server Audit Logs.
- Admin Audit Log — Logs all the federated identity activities of each administrator. When Select Federation is running in Standalone mode, only the root administrator can view the administrator audit logs. But when Select Federation is running in Select Access-Integrated mode, any administrator can view the administrator audit logs.

Select Federation tracks all federation logins and logouts in the database, in which you can search by most of its parameters.

You can access these logs through the **Logs** menu on the Administration Console, depending on in which mode Select Federation is running:

- When Select Federation is running in Standalone mode, the **Logs** → **View Server Audit Log** menu option is available to any administrator. However, the **Logs** → **View Admin Audit Log** menu option is only available to the root administrator.
- When Select Federation is running in Select Access-Integrated mode, both the **Logs** menu options are available to anyone who is logged in.

Audit Log Event Timestamps

All the machines involved in an installation must have synchronized system time, which is required by the federation protocols. If the machine system times are not synchronized, it can result in audit log event timestamps not matching with the protocol messages.

It is strongly recommended that you run all systems in the GMT (Greenwich Meridian Time) time zone, especially for installations that operate across time zones. Also, the database software must be in the same time zone as the machine on which it is running and therefore, in the same time zone as the Select Federation machine.

Select Federation does all of its computations in GMT. Therefore, its basic operation is not affected by the change in the daylight savings time scheme.

Server Audit Log

All administrators can view the server log for their department or region to see the operational activities of their enabled users. The root administrator can also see the operational activities of the delegated administrators' enabled users.

The following topics describe how to view the Server Audit log:

- Server Audit Log Search Criteria
- Viewing the Server Audit Log Query Results
- Viewing the Server Audit Log Entry

Server Audit Log Search Criteria

You can view Server Audit Log by specifying initial substrings for any or all of the search criteria for viewing the audit logs.

All the following fields are optional:

- **By event type:** Federation event such as "Logged In," "Received Login Request", "Received Logout Request," or "Logged Out."

  ➤ The "Received Login Request" event has been set up so that the partner ID only displays when the partner issues a Login Request. In general, requests that are received by the Select Federation core from an application are logged with no partner ID.

- **By local user id**: ID of a local user.
- **By partner id**: Unique ID of the partner site.
- **By message id**: Messages exchanged with a particular site.
- **By origin IP**: IP address of the originating message/event.
- **From date**: Specified as YYYY-MM-DD.
- **To date:** Specified as YYYY-MM-DD.

Enter your search criteria in one or more fields, or leave all fields blank, and click the **Lookup** button. Clicking the **Clear All** button removes all entered criteria, leaving all fields blank.

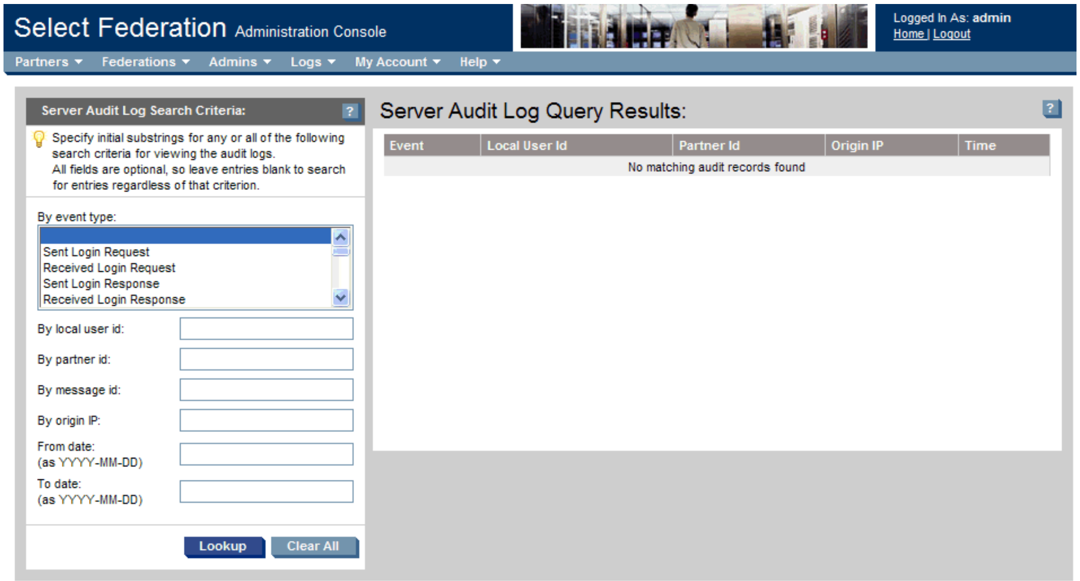
If you leave a field blank, the search is for all the entries in that category. For example, if you want to see a list of all enabled users, leave the **By local user id** field blank and click the **Lookup** button.

Viewing the Server Audit Log Query Results

Perform the following steps to view the Server Audit Log query results:

1 Select the **Logs** → **View Server Audit Log** menu option.

The Server Audit Log Query Results page opens. Initially, this page is blank.



Server Audit Log Search Criteria:

Specify initial substrings for any or all of the following search criteria for viewing the audit logs.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By event type:
Sent Login Request
Received Login Request
Sent Login Response
Received Login Response

By local user id:

By partner id:

By message id:

By origin IP:

From date: (as YYYY-MM-DD)

To date: (as YYYY-MM-DD)

Server Audit Log Query Results:

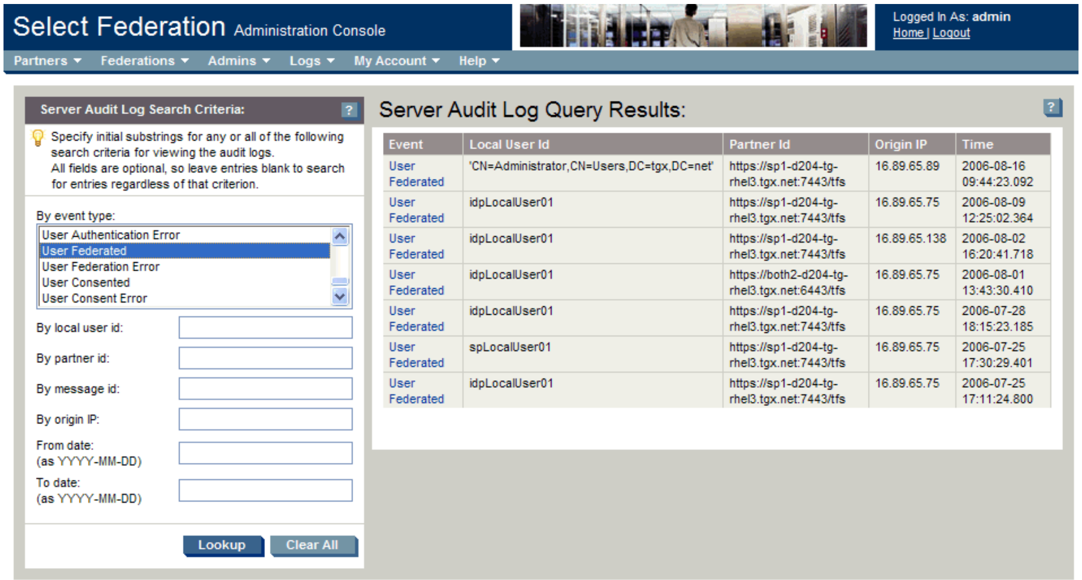
Event	Local User Id	Partner Id	Origin IP	Time
No matching audit records found				

2 Select one or more events in the left panel to view.

You can specify search criteria to show specific categories for the selected events. See Server Audit Log Search Criteria on page 64 for details.

3 Click the **Lookup** button.

The Server Audit Log Query Results page opens again with information about the selected event or events. Following is an example.



Server Audit Log Search Criteria:

Specify initial substrings for any or all of the following search criteria for viewing the audit logs.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By event type:
User Authentication Error
User Federated
User Federation Error
User Consented
User Consent Error

By local user id:

By partner id:

By message id:

By origin IP:

From date: (as YYYY-MM-DD)

To date: (as YYYY-MM-DD)

Server Audit Log Query Results:

Event	Local User Id	Partner Id	Origin IP	Time
User Federated	'CN=Administrator,CN=Users,DC=tgx,DC=net'	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.89	2006-08-16 09:44:23.092
User Federated	idpLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.75	2006-08-09 12:25:02.364
User Federated	idpLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.138	2006-08-02 16:20:41.718
User Federated	idpLocalUser01	https://both2-d204-tg-rhel3.tgx.net:6443/tfs	16.89.65.75	2006-08-01 13:43:30.410
User Federated	idpLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.75	2006-07-28 18:15:23.185
User Federated	spLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.75	2006-07-25 17:30:29.401
User Federated	idpLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	16.89.65.75	2006-07-25 17:11:24.800

This page displays information about all the selected events, based on the search criteria. For example, if you entered a specific **By partner id**, then this page would only display all the messages exchanged with a particular site.

➤ Local applications may issue login and other requests through the API. In these cases, in the Server Audit Log Query Results page, the Partner ID will be blank, as the request was not issued by a partner. In this case, when you view the logged entry details on the Server Audit Log Entry page (see the next section Viewing the Server Audit Log Entry), the Origin field will contain the "Local API," and the Message ID will be empty.

Other fields can sometimes be empty, as not all browsers and application servers provide the necessary information in all cases.

The Server Audit Log Query Results page displays the following information:

- **Event**: Federation event such as "Logged In", "Received Logout Request", "Logged Out", and so on.
- **Local User Id**: ID of the local user.
- **Partner Id**: Partner or Group site's unique ID in the federation. The Partner IDs appear as one of the following:
  - — Friendly name hyperlink of the Partner or Group whenever that Partner or Group ID is still in use. You can click on the Partner or Group hyperlink to quickly check the settings of the Partner or Group.

    If an admin deleted a Partner, then there is no link to a settings page for that Partner, and the friendly name of that Partner is deleted from the Select Federation records.
  - — providerId of the Partner or Group, which is not a hyperlink.
- **Origin IP**: IP address of the authentication request.
- **Time**: Date specified as YYYY-MM-DD and time the event was logged.

Click on an event to see more details on the Server Audit Log Entry page. See Viewing the Server Audit Log Entry on page 66 for details.

Viewing the Server Audit Log Entry

You can view more details about a particular event on the Server Audit Log Entry page.

Perform the following steps to view an event's details:

1 Select the **Logs** → **View Server Audit Log** menu option.

The Server Audit Log page opens.

2 Select one or more events in the left panel to view.

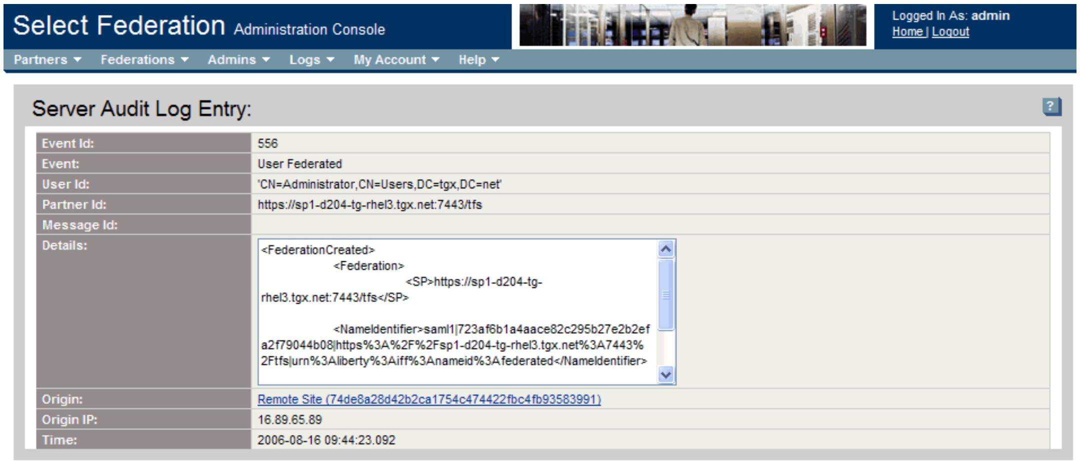
You can specify search criteria to show specific categories for the selected events. See Server Audit Log Search Criteria on page 64 for details.

3 Click the **Lookup** button.

The Server Audit Log Query Results page opens.

4 Click on an event.

The Server Audit Log Entry page opens with details about that event.



Server Audit Log Entry:

Event Id:	556
Event:	User Federated
User Id:	'CN=Administrator,CN=Users,DC=tgx,DC=net'
Partner Id:	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs
Message Id:	
Details:	<FederationCreated> <Federation> <SP>https://sp1-d204-tg-rhel3.tgx.net:7443/tfs</SP> <NameIdentifier>saml1|723af6b1a4aace82c295b27e2b2efa2f79044b08|https%3A%2F%2Fsp1-d204-tg-rhel3.tgx.net%3A7443%2Ftfs|urn%3Aliberty%3Aiff%3Anameid%3Afederated</NameIdentifier>
Origin:	Remote Site (74de8a28d42b2ca1754c474422fbc4fb93583991)
Origin IP:	16.89.65.89
Time:	2006-08-16 09:44:23.092

This page contains the following information:

- **Event Id**: Event number.
- **Event**: Federation event such as “Logged In”, “Received Logout Request”, “Logged Out”, and so on.
- **User Id**: ID of the local user related to the event.
- **Partner Id**: A Partner site's unique ID in the federation.
- **Message Id**: Message ID that was logged.
- **Details**: Event details.
- **Origin**: Site from which the event occurred.
- **Origin IP**: IP address of the authentication request.
- **Time**: Date specified as YYYY-MM-DD and time the event was logged.

Admin Audit Log

When Select Federation is running in Standalone mode, only the root administrator can view the administrator audit logs to see all the federated identity activities of each administrator. The **Logs** → **View Admin Audit Log** menu options will be hidden to all delegated administrators. When Select Federation is running in Select Access-Integrated mode, anyone who is logged in can view the administrator audit logs.

The following topics describe how to view the Admin Audit log:

- Admin Audit Log Search Criteria
- Viewing the Admin Audit Log Query Results
- Viewing the Admin Audit Log Entry

Admin Audit Log Search Criteria

You can view the Admin Audit Log by specifying initial substrings for any or all of the search criteria for viewing the audit logs.

All the following fields are optional:

- **By event type:** An administrator action such as “Viewed Audit Log” or “Logged In.” You can select more than one event.
- **By admin id:** User ID of the administrator.
- **By user id**: ID of the local user related to the event. If an admin deletes a federation, the user id is of the person whose federation was deleted.
- **By partner id:** Unique ID of the partner site.
- **By origin IP**: IP address of the authentication request.
- **From date**: Specified as YYYY-MM-DD.
- **To date:** Specified as YYYY-MM-DD.

Enter your search criteria in one or more fields, or leave all fields blank, and click the **Lookup** button. Clicking the **Clear All** button removes all entered criteria, leaving all fields blank.

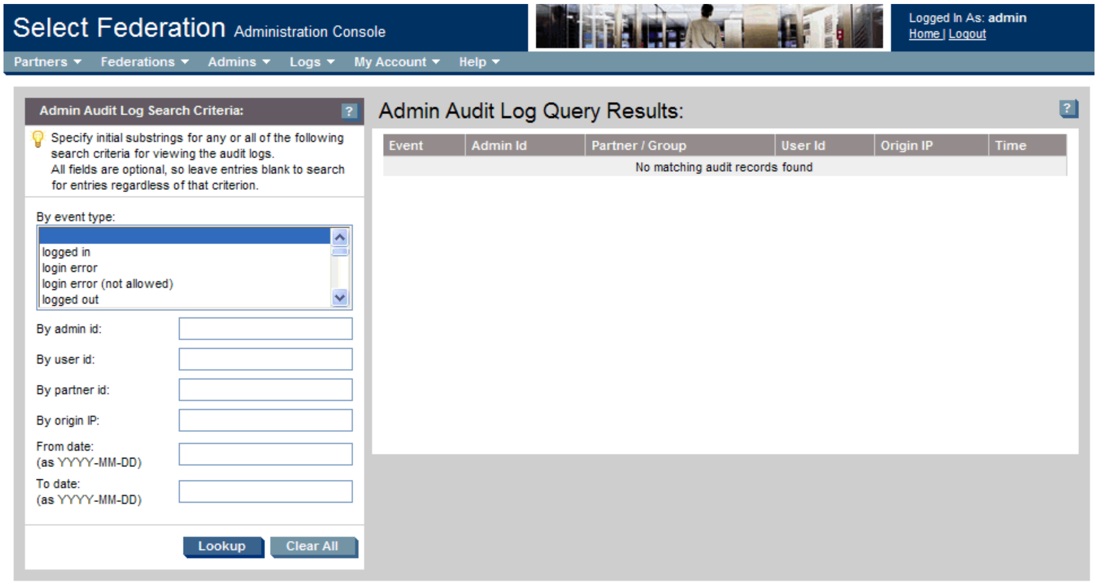
If you leave a field blank, the search is for all the entries in that category. For example, if you want to see a list of all enabled administrators, leave the **By admin id** field blank and click the **Lookup** button.

Viewing the Admin Audit Log Query Results

Perform the following steps to view the admin audit log query results:

1 Select the **Logs** → **View Admin Audit Log** menu option.

The Admin Audit Log Query Results page opens. Initially, this page is blank.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners
Federations
Admins
Logs
My Account
Help
Admin Audit Log Search Criteria:
Admin Audit Log Query Results:
Lookup
Clear All

Specify initial substrings for any or all of the following search criteria for viewing the audit logs.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By event type:
logged in
login error
login error (not allowed)
logged out

By admin id:

By user id:

By partner id:

By origin IP:

From date:
(as YYYY-MM-DD)

To date:
(as YYYY-MM-DD)

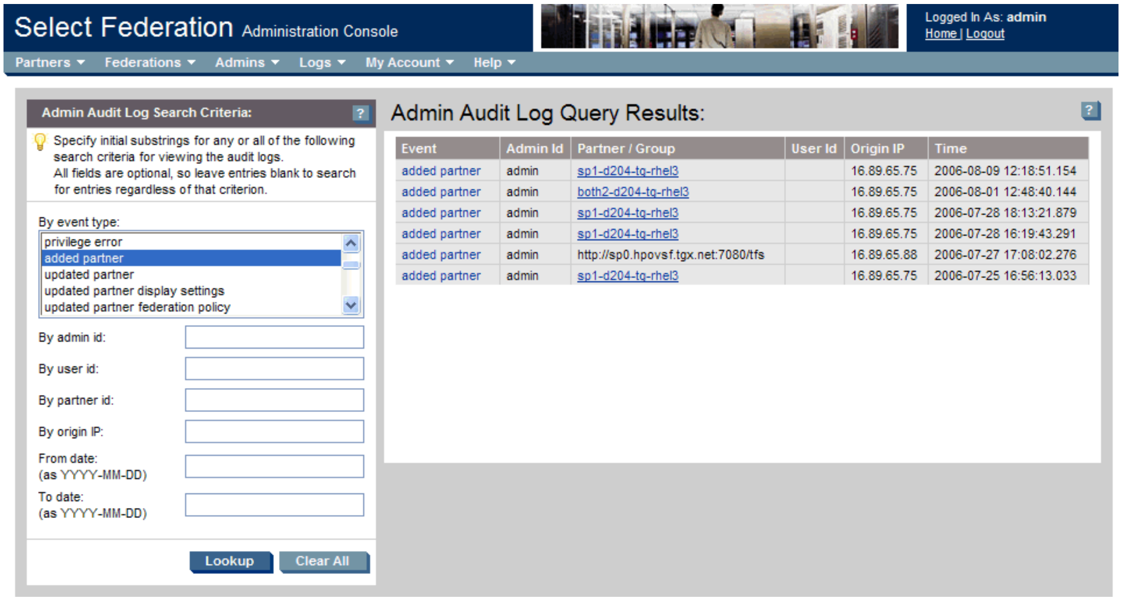
Event	Admin Id	Partner / Group	User Id	Origin IP	Time
No matching audit records found					

2 Select one or more events in the left panel to view.

You can specify search criteria to show specific categories for the selected events. See Admin Audit Log Search Criteria on page 67 for details.

3 Click the **Lookup** button.

The Admin Audit Log Query Results page opens again with information about the selected event or events. Following is an example.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Admin Audit Log Search Criteria:
?

Specify initial substrings for any or all of the following search criteria for viewing the audit logs.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By event type:
privilege error
added partner
updated partner
updated partner display settings
updated partner federation policy

By admin id:

By user id:

By partner id:

By origin IP:

From date: (as YYYY-MM-DD)

To date: (as YYYY-MM-DD)

Admin Audit Log Query Results:

Event	Admin Id	Partner / Group	User Id	Origin IP	Time
added partner	admin	sp1-d204-tq-rhel3		16.89.65.75	2006-08-09 12:18:51.154
added partner	admin	both2-d204-tq-rhel3		16.89.65.75	2006-08-01 12:48:40.144
added partner	admin	sp1-d204-tq-rhel3		16.89.65.75	2006-07-28 18:13:21.879
added partner	admin	sp1-d204-tq-rhel3		16.89.65.75	2006-07-28 16:19:43.291
added partner	admin	http://sp0.hpovsf.tgx.net:7080/tfs		16.89.65.88	2006-07-27 17:08:02.276
added partner	admin	sp1-d204-tq-rhel3		16.89.65.75	2006-07-25 16:56:13.033

This page displays information about all the selected events, based on the search criteria. For example, if you entered a specific **admin Id**, then this page would only display all the selected events performed by that admin.

This page contains the following information:

- **Event**: Admin event such as “Logged In”, “Added Partner Group”, “Deleted Federation”, “Logged Out”, and so on.
- **Admin Id**: Admin Username - email address for delegated administrators and “admin” for the root administrator.
- **Partner/Group**: Partner ID of the Partner or Group related to the event. The Partner IDs appear as one of the following:
  - — Friendly name hyperlink of the Partner or Group whenever that Partner or Group ID is still in use. You can click on the Partner or Group hyperlink to quickly check the settings of the Partner or Group.

    If an admin deleted a Partner, then there is no link to a settings page for that Partner, and the friendly name of that Partner is deleted from the Select Federation records.
  - — providerId of the Partner or Group, which is not a hyperlink.
- **User Id**: ID of the user related to the event.
- **Origin IP**: IP address of the authentication request.
- **Time**: Date specified as YYYY-MM-DD and time the event was logged.

Click on an event to see more details on the Admin Audit Log Entry page. See the next section Viewing the Admin Audit Log Entry for details.

Viewing the Admin Audit Log Entry

You can view more details about a particular event on the Admin Audit Log Entry page.

Perform the following steps to view an event's details:

1 Select the **Logs** → **View Admin Audit Log** menu option.

The Admin Audit Log page opens.

2 Select one or more events in the left panel to view.

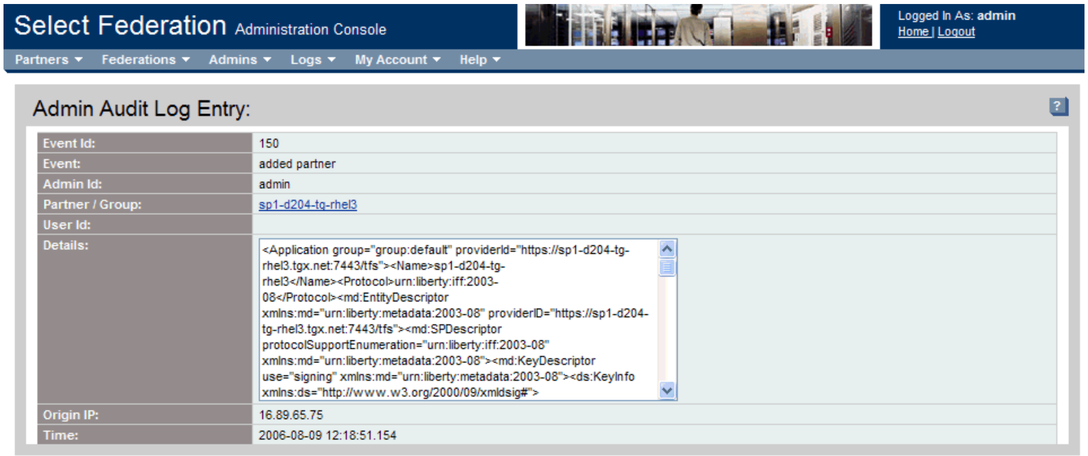
You can specify search criteria to show specific categories for the selected events. See Admin Audit Log Search Criteria on page 67 for details.

3 Click the **Lookup** button.

The Admin Audit Log Query Results page opens.

4 Click on an event.

The Admin Audit Log Entry page opens with details about that event.



Admin Audit Log Entry:

Event Id:	150
Event:	added partner
Admin Id:	admin
Partner / Group:	sp1-d204-tg-rhel3
User Id:	
Details:	<Application group="group:default" providerId="https://sp1-d204-tg-rhel3.tgx.net:7443/tfs"><Name>sp1-d204-tg-rhel3</Name><Protocol>urn:liberty:iff:2003-08</Protocol><md:EntityDescriptor xmlns:md="urn:liberty:metadata:2003-08" providerID="https://sp1-d204-tg-rhel3.tgx.net:7443/tfs"><md:SPDescriptor protocolSupportEnumeration="urn:liberty:iff:2003-08" xmlns:md="urn:liberty:metadata:2003-08"><md:KeyDescriptor use="signing" xmlns:md="urn:liberty:metadata:2003-08"><ds:KeyInfo xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
Origin IP:	16.89.65.75
Time:	2006-08-09 12:18:51.154

This page contains the following information:

- **Event Id**: Event number.
- **Event**: Admin event such as "Logged In", "Added Partner Group", "Deleted Federation", "Logged Out", and so on.
- **Admin Id**: Admin Username - email address for delegated administrators and "domineer the root administrator.
- **Partner/Group**: Name of the Partner or Group the event affected.
- **User Id**: ID of the user related to the event.
- **Details**: Details of the event.
- **Origin IP**: IP address of the authentication request.
- **Time**: Date specified as YYYY-MM-DD and time the event was logged.

Managing Admins

Select Federation has a "root" administrator account named "admin" when running in Standalone mode. It is recommended that the "admin" account only be used to create individual delegated administrative user accounts. This ensures that each administrative user's actions are logged with their own name instead of the generic "admin" account.

A delegated administrator logs in with their email address, not with their admin name.

The following topics describe how to manage administrators:

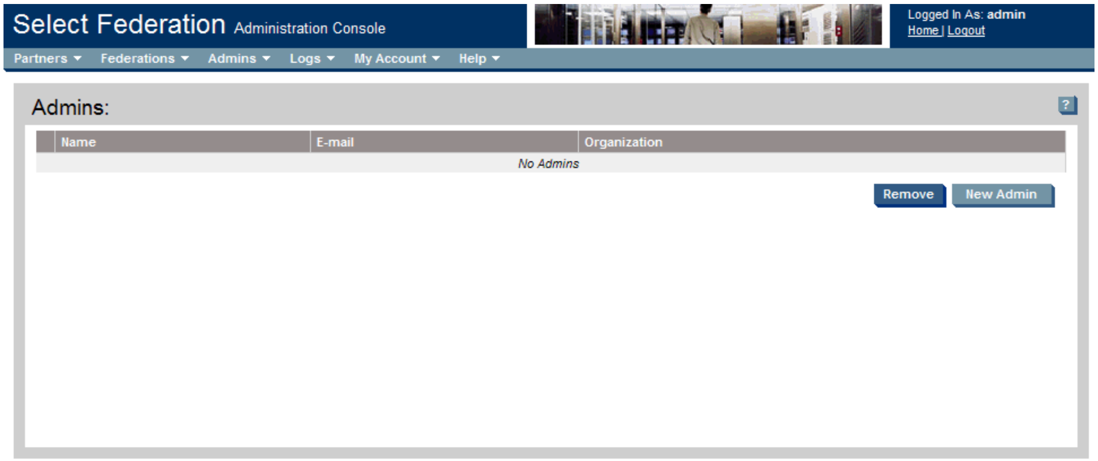
- Adding Admins (for the root administrator)
- Removing Admins (for the root administrator)
- Changing the Admin Password (for the root administrator)
- Viewing Delegated Admin Account Details (for delegated administrators)

Adding Admins

Perform the following steps to add a new admin:

1 Select the **Admins** → **Manage Admins** menu options.

The Admins page opens.

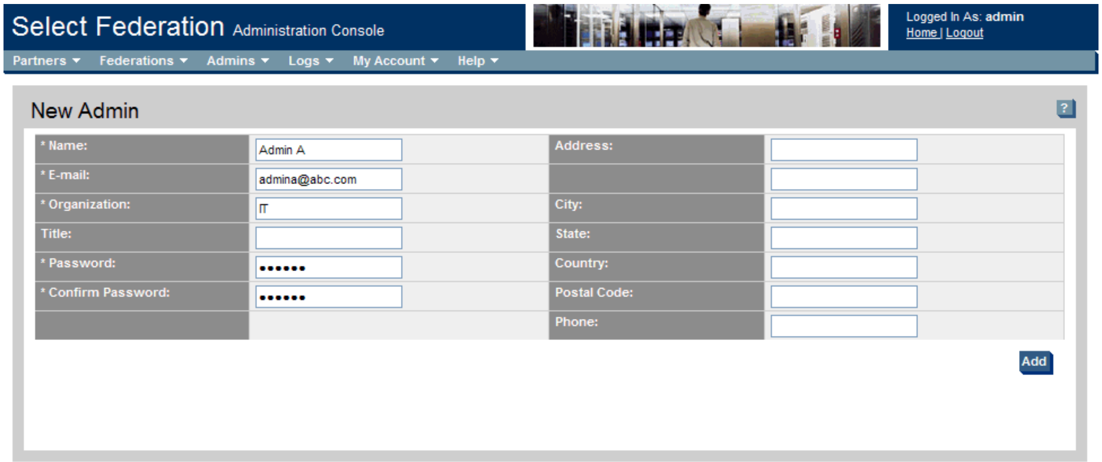


Name	E-mail	Organization
No Admins		

The first time you use Select Federation, this page is blank.

2 Click the **New Admin** button.

The New Admin page opens.



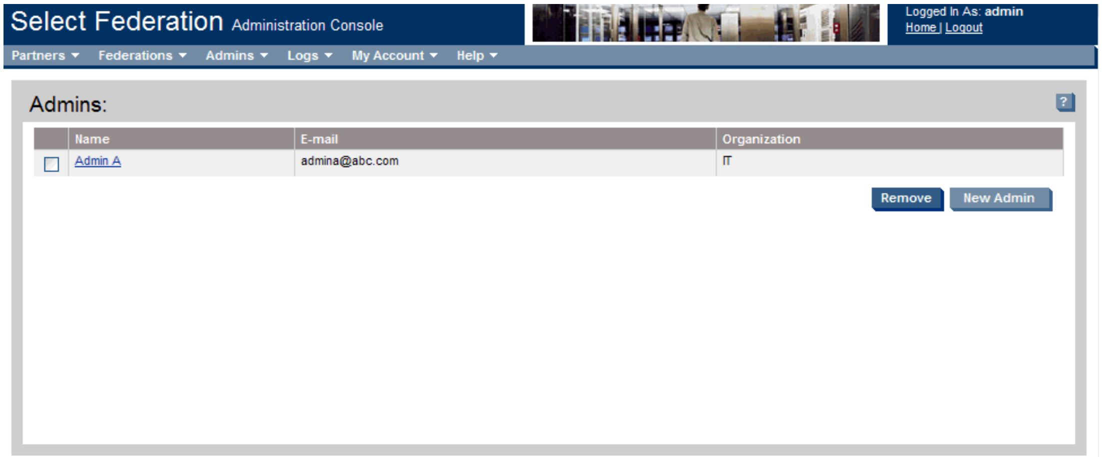
New Admin

* Name:	Admin A	Address:	
* E-mail:	admina@abc.com		
* Organization:	IT	City:	
Title:		State:	
* Password:	••••••	Country:	
* Confirm Password:	••••••	Postal Code:	
		Phone:	

3 Enter the required information indicated by the asterisk and any other information.

4 Click the **Add** button.

The Admins page opens again with the newly added admin.



Admins:

	Name	E-mail	Organization
□	Admin A	admina@abc.com	IT

An administrative user can then log into their own account using their email address as the login name.

Removing Admins

Perform the following steps to remove one or more admins.

1 Select the **Admins** → **Manage Admins** menu options to open the Admins page.

2 Click in one or more checkboxes of the admins you wish to remove.

3 Click the **Remove** button.

The admin is removed.

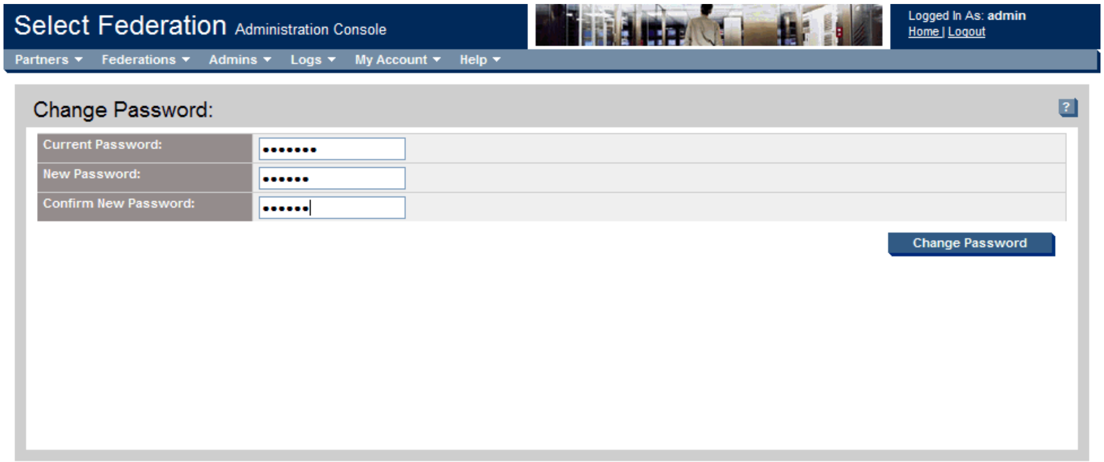
Changing the Admin Password

The root administrator can change the password of any other administrator. However, it is recommended delegated administrators change their own passwords.

Perform the following steps to change your password.

1 Click the **My Account** → **Change Password** menu options.

The Change Password page opens.



Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Change Password:
?
Change Password

Current Password:	•••••••
New Password:	••••••
Confirm New Password:	••••••

2 Enter the current password and then the new one and confirm it.

3 Click the **Change Password** button.

A highlighted message on this page confirms that the password has been changed.

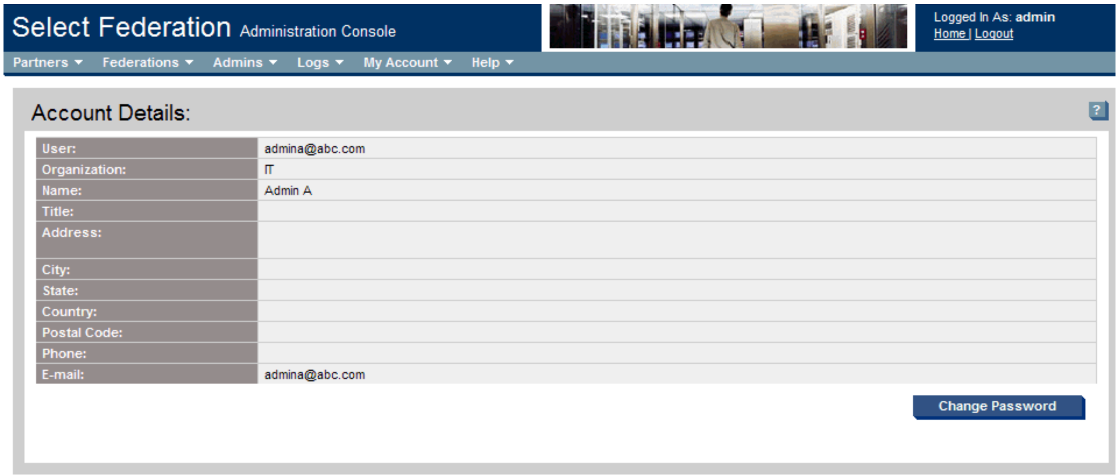
Viewing Delegated Admin Account Details

Delegated administrators can view their account details that were entered by the root administrator. The **View Details** option is only available to delegated administrators when Select Federation is running in Standalone mode.

To view your account details, perform the following step:

- Select the **My Account** → **View Details** menu options.

  The Account Details page opens.



Account Details:

User:	admina@abc.com
Organization:	IT
Name:	Admin A
Title:	
Address:	
City:	
State:	
Country:	
Postal Code:	
Phone:	
E-mail:	admina@abc.com

The Accounts Details page provides the following information about the delegated admin’s account:

- **User**: Delegated admin’s email address Username.
- **Organization**: Organization to which the delegated admin belongs.
- **Name**: Delegated admin’s name.
- **Title**: Delegated admin’s title within the organization.
- **Address**: Delegated admin’s street or P.O. Box address.
- **City**: City of the delegated admin’s address.
- **State**: State of the delegated admin’s address.
- **Country**: Country in which the delegated admin lives.
- **Postal Code**: Zip code of the delegated admin’s address.
- **Phone**: Delegated admin's phone number.
- **E-mail**: Delegated admin's email address.

Managing Federations

You can view all of the users that are federated with your Partners. You can also remove federated users. Use the Federation Search Criteria panel to quickly find all federated users.

The following topics describe how to search for, view and remove federated users:

- Viewing Federations
- Searching for Federated Users and Partners
- Removing Federated Partners

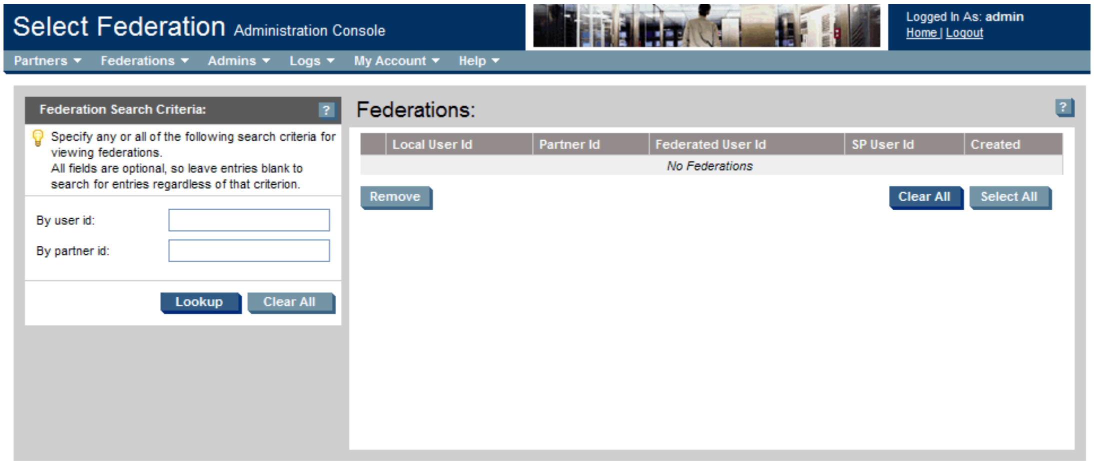
Viewing Federations

To view your federations, perform the following steps:

1 Click the **Federations** → **Manage Federations** menu options.

If you wish, you can skip this step and start with step 2 by searching for federations in the **Manage Federations** panel from the Welcome page.

The Federations page opens, which is initially blank.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Lookup
Clear All
Remove
Clear All
Select All

Federation Search Criteria:

Specify any or all of the following search criteria for viewing federations.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By user id:

By partner id:

Federations:

	Local User Id	Partner Id	Federated User Id	SP User Id	Created
			No Federations		

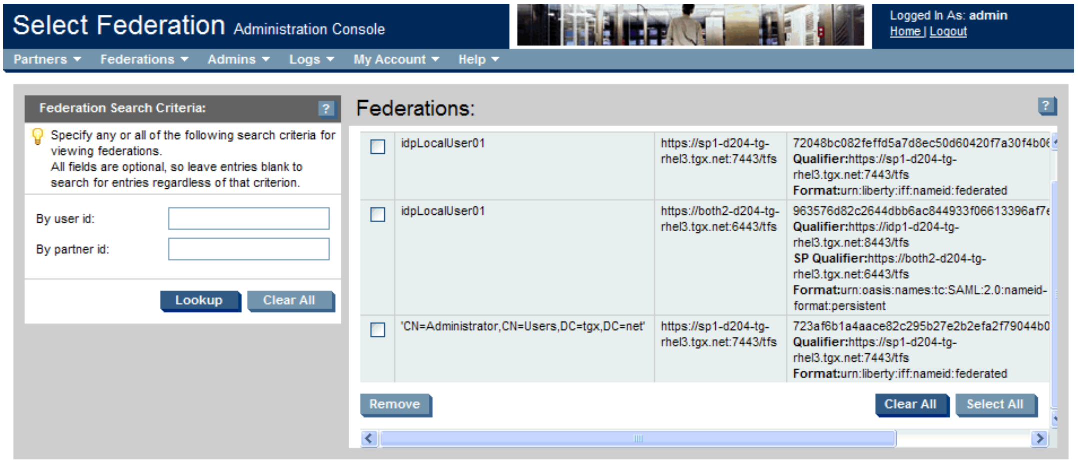
2 Use the Federation Search Criteria panel to search for and view federations.

See Searching for Federated Users and Partners on page 76 for instructions.

3 Click the **Lookup** button.

The Federations page opens displaying information based on the search criteria.

The following figure shows an example of the Federations page after a search has been performed.


Select Federation Administration Console
Logged In As: admin
Home | Logout
Partners ▾ Federations ▾ Admins ▾ Logs ▾ My Account ▾ Help ▾
Lookup
Clear All
Remove
Clear All
Select All

Federation Search Criteria:

Specify any or all of the following search criteria for viewing federations.
All fields are optional, so leave entries blank to search for entries regardless of that criterion.

By user id:

By partner id:

Federations:

□	idpLocalUser01	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	72048bc082feffd5a7d8ec50d60420f7a30f4b06 **Qualifier:**https://sp1-d204-tg-rhel3.tgx.net:7443/tfs **Format:**urn:liberty:iff:nameid:federated
□	idpLocalUser01	https://both2-d204-tg-rhel3.tgx.net:6443/tfs	963576d82c2644dbb6ac844933f06613396af7e **Qualifier:**https://idp1-d204-tg-rhel3.tgx.net:8443/tfs **SP Qualifier:**https://both2-d204-tg-rhel3.tgx.net:6443/tfs **Format:**urn:oasis:names:tc:SAML:2.0:nameid-format:persistent
□	'CN=Administrator,CN=Users,DC=tgx,DC=net'	https://sp1-d204-tg-rhel3.tgx.net:7443/tfs	723af6b1a4aace82c295b27e2b2efa2f79044b0 **Qualifier:**https://sp1-d204-tg-rhel3.tgx.net:7443/tfs **Format:**urn:liberty:iff:nameid:federated

Searching for Federated Users and Partners

Use the Federation Search Criteria panel to quickly find all federated users. You can search for federated users in different ways by entering a value in one or all fields, or by leaving the fields blank. For example, you can do any of the following:

- Enter the **User ID** and **Partner ID** and click **Lookup** to find information about that specific user's federation with a particular partner.
- Enter the **Partner ID** on the corresponding field and click **Lookup** to find information about that specific Partner's user federations.
- Click on the **Lookup** button without entering any information to search for all users and their federations.
- Click on the **Clear All** button to undo any previous selections.

Removing Federated Partners

Perform the following steps to remove federated partners:

1. Click the **Federations** → **Manage Federations** menu options.

   If you wish, you can skip this step and start with step 2 by searching for federations in the **Manage Federations** panel from the Welcome page.

2. Use the Federation Search Criteria panel to search for and view federations.

   See Searching for Federated Users and Partners on page 76 for instructions.

3. Click the **Lookup** button.

   The Federations page opens displaying information based on the search criteria.

4. You can do one of the following:

   - Remove a federation.

     Check the box to the left of the federation row you would like to delete and click the **Remove** button.

   - Remove all visible federations.

     Click on the **Select All** button and then click **Remove**.

   - Click on the **Clear All** button to undo any previous selections.

6 Enabling Applications

Overview

Ultimately, a key goal of any federated identity deployment is to provide seamless access to applications where previously one would have required a local credential to be presented. Select Federation provides a number of ways to “federation enable” applications. Providing protocol-independent application integration is a key feature of Select Federation.

Select Federation provides a number of ways to enable applications. The more advanced enablement options are provided in the SDK, which are described in the documentation on the Select Federation SDK CD. This chapter focuses on the following three enablement options:

- Application Helpers — Provide a simple means of generating URLs that initiate federation actions, which can be embedded in your portals, web sites or other applications.
- Demonstration Program — Provides a demonstration of federated identity in action, and also provides sample code for using the Select Federation software.
- Filters — Provide a convenient means of web access control for commonly used web-servers.

These enablement options are described in the following sections:

- Using the Application Helper
- Using the Demonstration Program
- Using Filters to Protect Web Applications

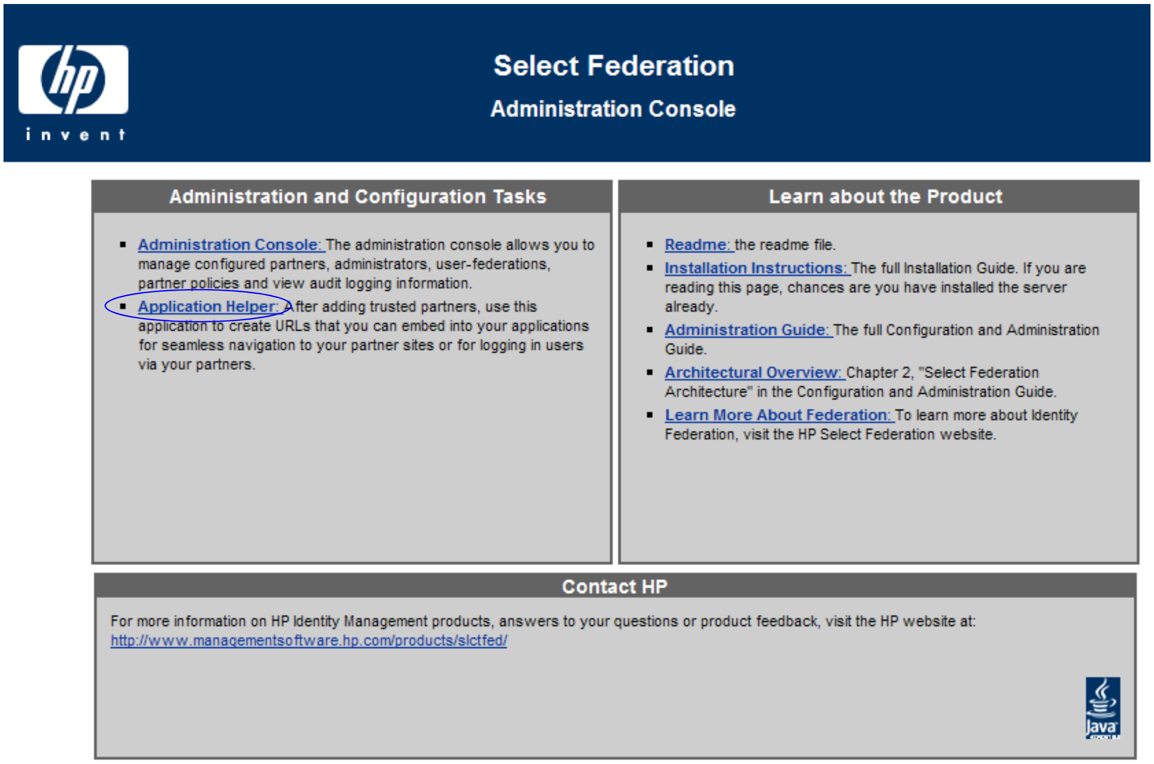
Using the Application Helper

For ease of integration into your existing environment, Select Federation provides a special Application Helper component.

The Application Helper is a unique feature of Select Federation that simplifies the way in which you initiate federation actions such as federated login and global logout. Using the Application Helper, you enter a “target URL” that you would like your users to go to after a federated login. The Application Helper will return a transformed URL that you can paste into your portal for your users to click on. When the users click on this transformed URL, they will arrive seamlessly at the target URL without being prompted for a credential.

The Administration Console startup page includes a link to the Application Helper as shown in the following figure.

Figure 8 Select Federation Administration Console



Select Federation

Administration Console

Administration and Configuration Tasks

- Administration Console: The administration console allows you to manage configured partners, administrators, user-federations, partner policies and view audit logging information.
- Application Helper: After adding trusted partners, use this application to create URLs that you can embed into your applications for seamless navigation to your partner sites or for logging in users via your partners.

Learn about the Product

- Readme: the readme file.
- Installation Instructions: The full Installation Guide. If you are reading this page, chances are you have installed the server already.
- Administration Guide: The full Configuration and Administration Guide.
- Architectural Overview: Chapter 2, "Select Federation Architecture" in the Configuration and Administration Guide.
- Learn More About Federation: To learn more about Identity Federation, visit the HP Select Federation website.

Contact HP

For more information on HP Identity Management products, answers to your questions or product feedback, visit the HP website at: http://www.managementsoftware.hp.com/products/slctfed/

You can also navigate to the Application Helper using the following address at the top-level URL:

```
<base-url>/tfs-internal/helperMain.html
```

The Application Helper can help you configure URLs in your application for seamless navigation to Service Provider (SAML Consumer) sites or for authentication via Identity Provider (SAML Producer) sites.

There are two useful pages in the Application Helper:

- `idphelper.jsp`: This helps you construct URLs to embed in your application that allow your users to seamlessly navigate to trusted third-party web sites. You may want users to go to a particular URL at that site, which you can enter on this page, or you can leave the target URL field blank, in which case the third-party site will navigate the user to an default URL after verifying the trust between the sites.
- `sphelper.jsp`: This shows how to construct login URLs that enable you to let users login, federate and de-federate via a trusted Identity Provider (IDP). It also provides a way of constructing “global logout” URLs that you can use to initiate a global logout for a user that has been authenticated at your site. Note that the global logout and federation / defederation features are not available when using SAML 1.0 or 1.1 protocols.

Using the Demonstration Program

Select Federation provides a Demonstration program that allows you to quickly see federated identity in action. The Demonstration program can also serve as sample code, which you can use to enable your own applications.

You can navigate to the Demonstration program using the following address at the top-level URL (just replace `tfs-internal` with `sf-demo`):

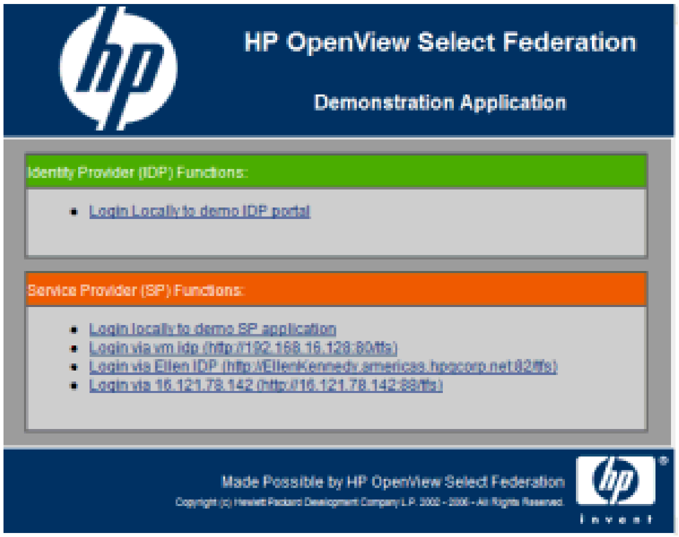
```
<base-url>/sf-demo
```

The Demonstration program consists of two parts:

- Identity Provider Demonstration program
- Service Provider Demonstration program

The following figure shows the Demonstration program landing screen.

Figure 9 Demonstration Application Landing Page



HP OpenView Select Federation

Demonstration Application

Identity Provider (IDP) Functions:

- Login Locally to demo IDP portal

Service Provider (SP) Functions:

- Login locally to demo SP application
- Login via vm idp (http://192.168.16.128:80/tfs)
- Login via Ellen IDP (http://EllenKennedy.americas.hpqcorp.net:82/tfs)
- Login via 16.121.78.142 (http://16.121.78.142:88/tfs)

Made Possible by HP OpenView Select Federation

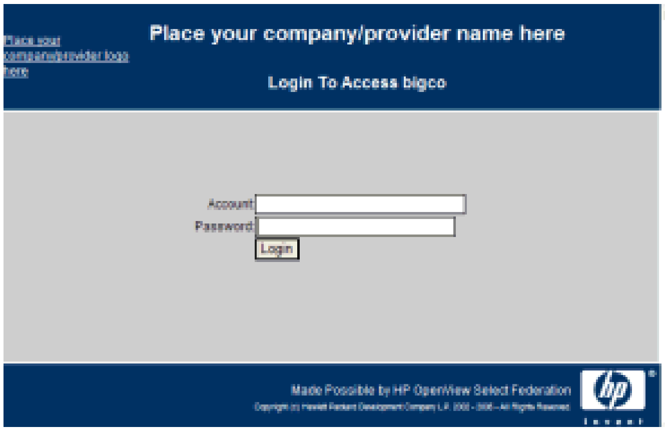
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved.

As seen in the above screen, the demo application shows all SP functionality with orange colored headers and all IDP functionality with green colored headers. Functionality that is shared by both IDP and SP is in neutral colors.

IDP Demonstration Program

Clicking on the **Login locally to demo IDP portal link** on the Demonstration program landing page opens an authentication screen similar to the following figure.

Figure 10 Authentication Screen



Place your company/provider logo here

Place your company/provider name here

Login To Access bigco

Account:

Password:

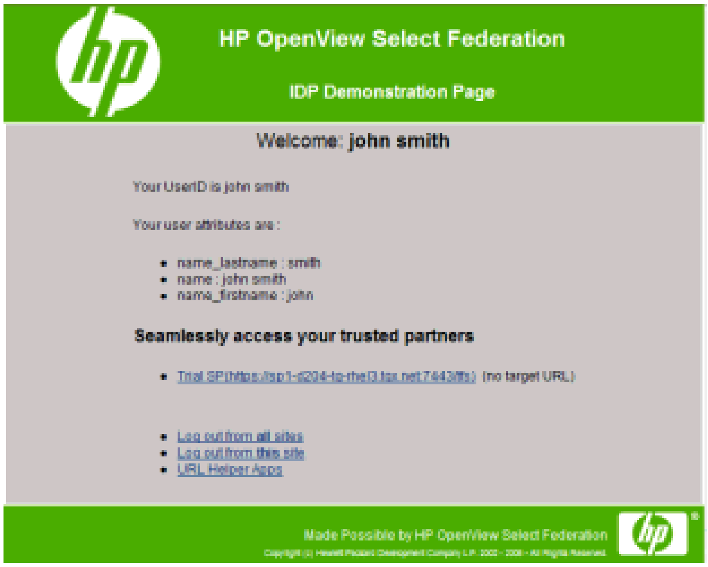
Made Possible by HP OpenView Select Federation

Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 – All Rights Reserved

On this screen, enter a username and password that can be validated against the directory server that was configured during the installation process.

Once you login to the IDP demo application, the IDP Demonstration page opens.

Figure 11 IDP Demonstration Page



HP OpenView Select Federation

IDP Demonstration Page

Welcome: john smith

Your UserID is john smith

Your user attributes are :

- name_lastname : smith
- name : john smith
- name_firstname : john

Seamlessly access your trusted partners

- Trial SP(https://sp1-d204-tp-rhel3.tpx.net:7443/ffs) (no target URL)

- Log out from all sites
- Log out from this site
- URL Helper Apps

Made Possible by HP OpenView Select Federation

Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved

This page allows you to do the following:

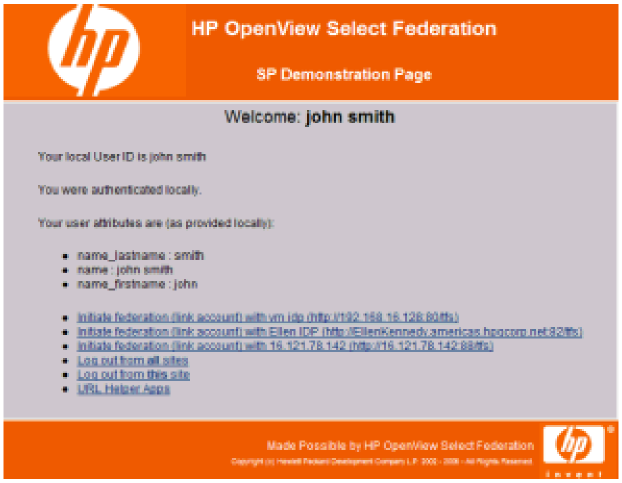
- Login to any configured SPs
- Logout from all sites at which the user is already logged on
- Logout locally from the IDP
- Access the helper applications

The SP Demonstration Program

You may access the SP demonstration program by either logging in via a configured IDP, or by logging in locally to the SP demonstration program. The local login is verified against the “internal IDP”. If an authentication mechanism is not configured locally (such as the SP-only mode of installation was chosen or the LDAP directory was not configured at install-time), the local login will not work, and users are forced to login through an IDP.

Logging in locally opens a page such as the one shown in the following figure.

Figure 12 SP Demonstration Page



HP OpenView Select Federation

SP Demonstration Page

Welcome: john smith

Your local User ID is john smith

You were authenticated locally.

Your user attributes are (as provided locally):

- name_lastname : smith
- name : john smith
- name_firstname : john

- Initiate federation (link account) with vm idp (http://192.168.16.128:80/tfs)
- Initiate federation (link account) with Ellen IDP (http://EllenKennedy.americas.hpqcorp.net:82/tfs)
- Initiate federation (link account) with 16.121.78.142 (http://16.121.78.142:88/tfs)
- Log out from all sites
- Log out from this site
- URL Helper Apps

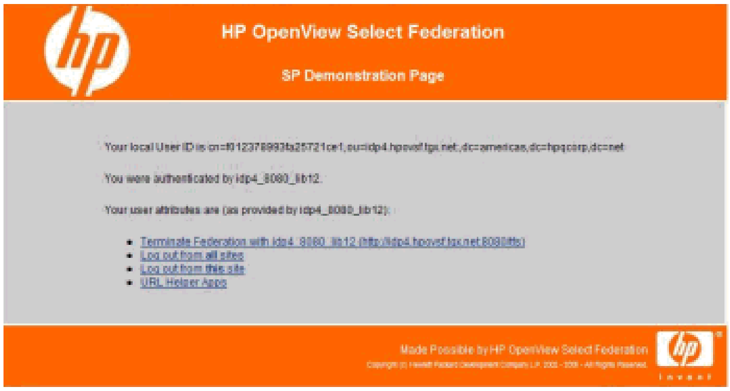
Made Possible by HP OpenView Select Federation

This page allows the user to do the following:

- Initiate federation with configured IDPs
- Terminate federations with IDPs that the user is already federated with
- Logout from all sites
- Logout locally
- Access the URL helper applications

If the user logs in via a configured IDP, the demonstration application opens the following screen:

Figure 13 SP Demonstration Page



HP OpenView Select Federation

SP Demonstration Page

Your local User ID is cn=f012378993fa25721ce1,ou=idp4.hpovsf.tgx.net,dc=americas,dc=hpqcorp,dc=net

You were authenticated by idp4_8080_lib12.

Your user attributes are (as provided by idp4_8080_lib12):

- Terminate Federation with idp4_8080_lib12 (http://idp4.hpovsf.tgx.net:8080/tfs)
- Log out from all sites
- Log out from this site
- URL Helper Apps

Made Possible by HP OpenView Select Federation

This screen shows the user's federated identity at the SP, any attributes that have been obtained from the IDP and allows the user to do the following actions:

- Terminate an existing federation
- Logout from all sites
- Logout locally
- Access the URL helper applications

Using Filters to Protect Web Applications

This section describes the Select Federation filters in the following topics:

- Overview of Filters
- Filter-Support
- IIS Filter
- Apache Filter
- Java Access Filter

Overview of Filters

A filter is a software component that changes how a web server handles the HTTP requests and responses for one or more web applications. A Select Federation filter validates user authentication state and if needed initiates. Single-Sign-On (SSO) before a web server serves an HTTP request for the following classes of URLs:

- Validates user authentication or Single-Sign-On (SSO) before a web server serves an HTTP request for the following classes of URLs:
  - **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). If, or once, the user is successfully authenticated the user's identity and attribute information is presented to the application. The application can then use this information to make further authorization decisions or to provide a personalized experience.
  - **Passive URLs** — Passive URLs are for resources where the user authentication state 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.
  - **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.

  The filter validates user authentication or SSO by looking for a special cookie (default cookie name `SFSession`) in the cookie header of the request. If the cookie header contains the special cookie, then the filter assumes that the user is authenticated. Otherwise, the filter redirects the user to the login page.

- Allows the user-specific information (such as profile and login information) to be available to the web applications by setting the request headers. See How Filters Work on page 83 for more details.

Select Federation provides the filters to protect and enhance web applications that are deployed on the following web servers:

- **IIS** — The Microsoft web server for the Windows platform.
- **Apache 2.0** — The Apache 2.0 web server for Linux, UNIX, or Windows platforms.

➤ The Select Federation Apache 2.0 module binaries are compatible with Apache 2.0.41+ versions that have a Module Magic Number that starts with 20020903. See Apache Filter on page 101 for more information.

However, the Select Federation Apache 2.0 module binaries are not compatible with Apache 1.3.

- **Java Servlet Containers** — Apache Tomcat or any other web server that is compliant with the Java Servlet Specification 2.3 or newer.

In some cases (such as during evaluation) it is not practical to install such a filter in the web server. Therefore, a PHP script is included on the Select Federation CD, which does the same work as the filters. This script may be helpful in better understanding the principles involved. See the `<`*`cd-base-directory`*`>\filters\custom\` directory for the PHP script.

How Filters Work

The filters are configured to protect (allow access only after authentication) certain directories, and possibly to allow access to certain files without authentication. It is also possible to configure what information (login event, IDP details from which a user is authenticated, user profile attributes and so on) should be made available to the applications.

When a filter receives the HTTP request, it first compares the request URL with the protected URLs (can be accessed only by authenticated users). If the request URL is not protected then it does not do any additional processing and passes the unmodified request along to the web server (or the next filter in the chain).

If the request URL is protected, then it retrieves the cookie header from the request headers and looks for a particular cookie in the cookie header. After the user is authenticated, this cookie is set for the domain shared between the web server and the Select Federation installation with an ID to the user federation session Event-Plugin running on an SP site (see “SP Event Plugin Interface” in the *HP OpenView Select Federation Web Application Developer's Guide* for more information). In the remainder of this chapter, the cookie is referred to as the “SF cookie.”

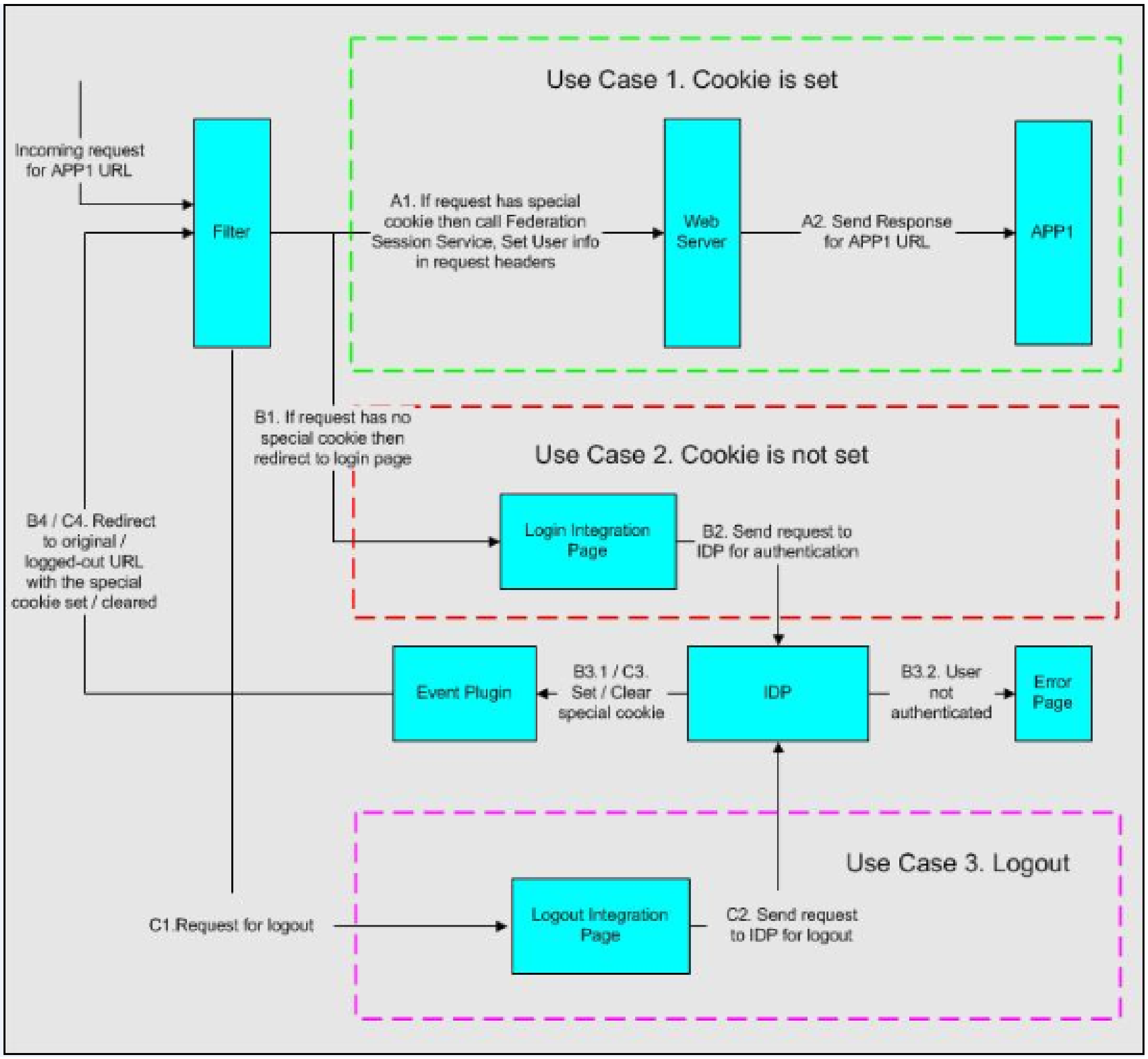
If the SF cookie is not present, the user is not authenticated and the filter redirects the user to the login page with the original request URL as parameter. Once the user is authenticated the system will redirect back to the application and hence the filter will be invoked again. This time however the SF cookie will be present.

If the filter finds a SF cookie in the cookie header, it gets the cookie value, which refers to a federated session id (sessionId). It then does a lookup for user information for that key in the filter cache. If the entry is found then it verifies that entry is valid. The filter maintains an in-memory cache of user information to ensure high performance.

In case the entry for sessionId is not found in the cache or the entry is expired, the filter calls the federation session service (`<BaseURL>/tfs-fs/FSS` installed on SP site) with the sessionId as the URL parameter. The federation session service returns the XML document with information about the principal.

If the filter receives the principal information successfully then it adds all the principal information in the cache. Next the filter adds extra headers (or in the case of the Java Access Filter extra attributes) to the request with the principal information (as per the directory configurations) and finally passes the HTTP request to the web server process.

Figure 14 Filter Flow Diagram



Use Case 1. Cookie is set
Incoming request for APP1 URL
Filter
A1. If request has special cookie then call Federation Session Service, Set User info in request headers
Web Server
A2. Send Response for APP1 URL
APP1
B1. If request has no special cookie then redirect to login page
Use Case 2. Cookie is not set
B4 / C4. Redirect to original / logged-out URL with the special cookie set / cleared
Login Integration Page
B2. Send request to IDP for authentication
Event Plugin
B3.1 / C3. Set / Clear special cookie
IDP
B3.2. User not authenticated
Error Page
Use Case 3. Logout
C1.Request for logout
Logout Integration Page
C2. Send request to IDP for logout

If it is not possible to use these native filters, applications such as Perl or PHP scripts can do the work of these filters. For this purpose, these applications need the following information:

- Cookie name (set by Event plugin - value of `filterSupport.cookieName` property from `tfsconfig.properties`)
- Location of login integration page to redirect if cookie not present
- Location of Federation Session Service (FSS)

How to Use Filters

To make use of headers set by the filter, you need to know the following header names:

- `SF-LocalUserId`

- SF-UserSession-Id
- SF-Login: When you configure the filters, you can either specify SF-Login to retrieve all the LoginInfo-related headers or specify each one of the following individually:

  — SF-Login-IdpFedUserId
  — SF-Login-AuthnContextClassRef
  — SF-Login-AuthnInstant
  — SF-Login-ReauthOnOrAfter

- SF-IDP: When you configure the filters, you can either specify SF-IDP to retrieve all the IDPInfo-related headers or specify each one of the following individually:

  — SF-IDP-IdpProviderId
  — SF-IDP-Home
  — SF-IDP-Name
  — SF-IDP-Description
  — SF-IDP-LogoRef
  — SF-IDP-LogoText

- SF-Profile: When you configure the filters, you can either specify SF-Profile to retrieve all the ProfileInfo-related headers or specify each one of the following individually (SF-Profile-<*AttributeName*> where <*AttributeName*> is the actual attribute name):

  — SF-Profile-name
  — SF-Profile-name-title
  — SF-Profile-name-firstname
  — SF-Profile-name-lastname
  — SF-Profile-home-street
  — SF-Profile-home-city
  — SF-Profile-home-state
  — SF-Profile-home-country
  — SF-Profile-home-postalCode
  — SF-Profile-personal-email
  — SF-Profile-personal-phone
  — SF-Profile-work-street
  — SF-Profile-work-city
  — SF-Profile-work-state
  — SF-Profile-work-country
  — SF-Profile-work-postalCode
  — SF-Profile-work-email
  — SF-Profile-work-phone

  This SF-Profile list assumes that the tfsconfig.properties file for the SP, includes the following attributes:

  ```
  userAttrs=name name_title name_firstname name_lastname
  home_street home_city home_state home_country home_postalCode
  personal_email personal_phone work_street work_city work_state
  work_country work_postalCode work_email work_phone
  ```

  The attribute names with underscores are replaced by hyphens. For example “name_title” becomes “SF-Profile-name- title.”

The information headers are prefixed as follows:

- Login information headers are prefixed with SF-Login-
- IDP information headers are prefixed with SF-IDP-

- Profile information headers are prefixed with `SF-Profile-`.

While configuring, if you specify the group header name (such as `SF-Login`), then the filter inserts all the subheaders from this group (`SF-Login-IdpFedUserId`, `SF-Login-AuthnContextClassRef`, `SF-Login-AuthnInstant`, `SF-Login-ReauthOnOrAfter`) in the request headers. In the case where you specify individual subheaders, then the filter inserts only those specified subheaders in the request headers.

A filter logs the high level or debug level information of the requests processed in the log file according to your configurations.

If it is not possible to use these native filters, applications such as Perl or PHP scripts can do the work of these filters. For this purpose, these applications need the following information:

- Cookie name (set by Event plugin - value of `filterSupport.cookieName` property from `tfsconfig.properties`)
- Location of login integration page to redirect if cookie not present
- Directories to be protected
- Unprotected or allowed URLs (like SSO, SLO pages, and so on)
- Passive URLs
- Location of Federation Session Service (FSS)
- `sessionId` as request parameter name for FSS with actual session ID as its value

Filter-Support

Select Federation 6.60 includes a dedicated Java web application that is used to integrate with the filters provided for the web servers mentioned above (or with web servers that cannot contact the Select Federation databases which are normally kept behind a firewall). This application is intended to be deployed on the same machine as the rest of the (SP) install of Select Federation.

How to Configure Filter-Support

To configure Filter-Support, you need to edit the configuration of the SP installation in the `tfsconfig.properties` file.

Perform the following steps to configure Filter-Support:

1. Open the `<`*`sp-installation-dir`*`>conf\tfsconfig.properties` file.
2. Uncomment `#spEventPlugin=` and provide the name for the filter's support plugin implementation:

   ```
   spEventPlugin=com.hp.ov.selectfederation.filters.support.FilterSupport
   Plugin
   ```

3. Add the following line underneath, replacing `.mycompany.com` with the domain that makes sense according to your installation:

   ```
   filterSupport.cookieDomain=.mycompany.com
   ```

   ➤ If the `#spEventPlugin=` line is not present because you have a previous version of the `tfsconfig.properties` file, then just add the two lines anywhere in the file.

4. Optionally, you may set the following parameters, but the defaults are assumed if you choose not to provide the values:

```
# If not specified, this value is assumed to be "SFSession"
filterSupport.cookieName=SFSession

# If not specified, this value is assumed to be "-1"
filterSupport.cookieMaxAge=-1

# A string that defaults to "targetIDP". A cookie with this name is set
# to remember the IDP that successfully authenticated the user. The
# cookie is cleared upon any authentication
filterSupport.idpSelection.cookieName=targetIDP

# An int that defaults to 7*24*60*60 seconds (one week). Setting this
# to 0 would disable the feature.
filterSupport.idpSelection.cookieAge=604800
```

5. Restart the Application server that hosts the SP to enable the EventPlugin.

   You are now ready to add the appropriate filter to your web server.

How Filter-Support Works

The following Filter-Support components are provided to facilitate the filter installation and configuration. In Select Federation 6.60, these components are a part of the Select Federation "Enterprise Application Archive" or EAR file:

- **FilterSupportPlugin** — You can configure and use this plugin at your SP site installation as explained in the How to Configure Filter-Support on page 86. This plugin sets the cookie (used by filters) in the domain you specify after a user returns from authenticating with an IDP. (The domain should be shared by both the web server and the Select Federation SP installation.)
- **FSService** — Provides user-specific information for a given federated session id.
- **tfs-fs.war** — Provides a login page (`login.jsp`), which can be configured as the login URL for the filters. The login page allows the user to choose any one of the SP's partner IDPs for being authenticated. This war also provides a logout page.

  The login page uses the following two URL parameters and the Select Federation APIs to do Single-Sign-On (SSO):

  - **targetURL** (required) -— The absolute URL of the page on the web server for which the user needs to authenticate. This URL is provided so that the user can be returned to the originating page on the web server as part of completing the SSO.
  - **targetIDP** (optional) — This URL is the providerId of the IDP to which the user should be directed.

Filter-Support enables the filter that is present at the web server to take the value of the cookie set by the FilterSupportPlugin to retrieve user information through the FSService, and to provide the headers to the filter's applications.

Configuring Server Authentication

This section describes how to enable the Filter-support for server authentication. Server authentication requires that you do the following:

- Enable server authentication.
- Specify the path to a PEM format server certificate of your SP installation.

- Specify the settings for the keystore that holds the client certificates used by the filters to connect to the Filter-Support service. When absent, they default to the top-level keystore. In that case, the client certificates used by the filters should be imported into the Select Federation keystore.

  ```
  fss.keystorePath=
  fss.keystoreType=
  fss.keystorePassword=
  ```

  If the `fss.keystorePath` exists but is empty, the default JVM truststore for SSL is used. Typically this is the `$JAVA_HOME/lib/security/cacerts` file.

➤ If the filter client certificates are self signed, they need to be added to the `cacerts` file. Otherwise, the SSL layer will not accept them.

IIS Filter

The IIS filter and configuration interface for the IIS filter are implemented as Windows `DLL` files.

➤ For the IIS filter, only Windows 2003 is supported.

These `DLL` files are installed and uninstalled by using the IIS console or by using scripts. Using the filter configuration interface, you can configure the filter with all required configurations (site level and virtual directory level) and store them in a metabase.

The following sections provide instructions for the IIS filter processes:

- How to Install and Uninstall the IIS Filter
- How to Configure the IIS Filter
- How to Use the IIS Filter
- IIS Filter Log File

How to Install and Uninstall the IIS Filter

The instructions in the following sections describe how to install and uninstall the IIS filter configuration interface and the IIS filter:

- How to Install the IIS Filter Configuration Interface
- How to Install the IIS Filter
- How to Uninstall the IIS Filter Configuration Interface
- How to Uninstall the IIS Filter

How to Install the IIS Filter Configuration Interface

When you install the IIS filter configuration interface, you are installing the DLL files required for virtual directory and site-level configurations.

Perform the following steps to install the IIS filter configuration interface:

1. Double-click on the **install.cmd** executable in `<`*`cd-base-directory`*`>\filters\iis\win32\conf\`.

   The Open File - Security Warning dialog opens.

2. Click on the **Run** button.

You are prompted to open a filter file.

3. Click on the **Open** button for each file until all are opened.
4. Close the IIS console and restart it again.

   The IIS Filter Configuration Interface is installed. Continue to install the IIS filter itself. See the next section, How to Install the IIS Filter, for instructions.

How to Install the IIS Filter

➤ Before installing the IIS filter, you must have installed the IIS filter configuration interface (see How to Install the IIS Filter Configuration Interface on page 88).

You can install the IIS filter in one of two ways:

- Using the IIS console
- Executing a script

Installing the IIS Filter Using the IIS Console

Perform the following steps to install the IIS filter using the IIS console:

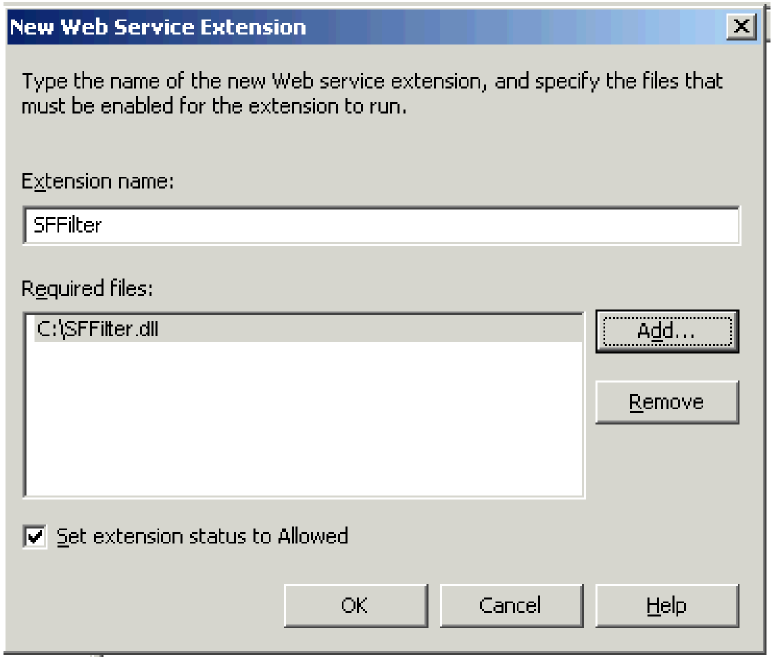
1. Save the `<cd-base-directory>\filters\iis\win32\SFFilter.dll` anywhere on your local hard drive.
2. Add the filter as follows:
   a. Select the Web Site you would like to protect in the left panel and right-click.
   b. Select the **Properties** menu option, then the **ISAPI Filters** tab in the Web Site Properties dialog.
   c. Click on the **Add** button.

      The Add/Edit Filter Properties dialog opens.
   d. Enter the **Filter** name. For example, **SFFilter.**
   e. Enter the path name or browse for the **Executable** `SFFilter.dll` file you saved on your hard drive.
   f. Click **OK**.
3. Add the Web Service extension (for IIS 6.x only) as follows:
   a. Right-click on the **Web Service Extension** in the left panel.
   b. Select the **Add a new Web service extension...** option.

      The New Web Service Extension dialog opens.
   c. Enter the Extension name. For example, **SFFilter.**
   d. Check the **Set extension status to Allowed** check box.
   e. Click on the **Add** button.

      The Add file dialog opens.
   f. Enter the path name or browse for the **Executable** `SFFilter.dll` file you saved on your hard drive.
   g. Click **OK**.

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



Type the name of the new Web service extension, and specify the files that must be enabled for the extension to run.

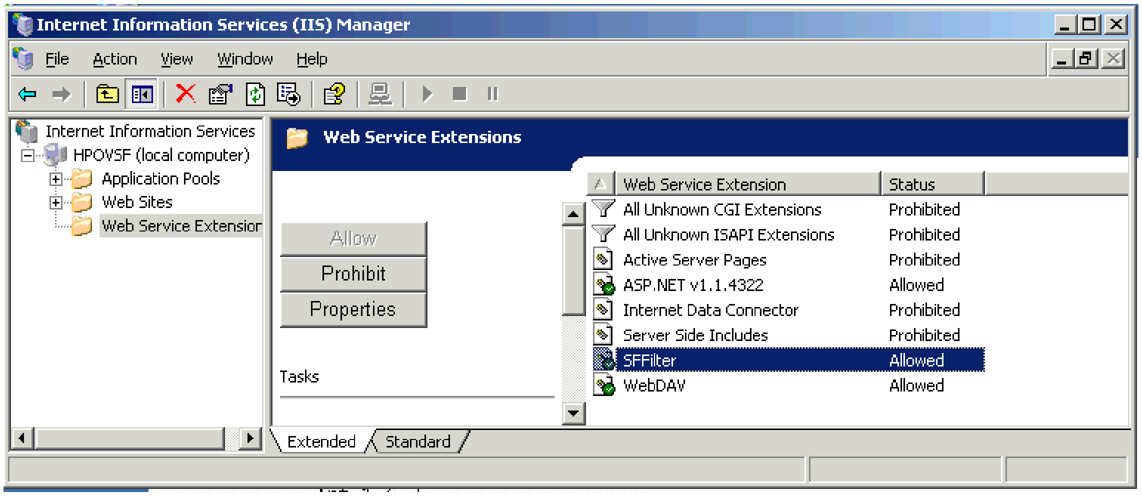
Extension name: SFFilter

Required files: C:\SFFilter.dll

☑ Set extension status to Allowed

h Click **OK** again.

The Web Service Extension displays in the right panel with the status set to Allowed.



Web Service Extension	Status
All Unknown CGI Extensions	Prohibited
All Unknown ISAPI Extensions	Prohibited
Active Server Pages	Prohibited
ASP.NET v1.1.4322	Allowed
Internet Data Connector	Prohibited
Server Side Includes	Prohibited
SFFilter	Allowed
WebDAV	Allowed

The IIS filter is installed.

4 After installing the IIS filter you must do the following steps or the filter may not load properly:

a Add *<cd-base-directory>*`\filters\iis\win32\redist` to the system PATH variable.

b Reboot the machine.

You are now ready to configure the IIS filter. See How to Configure the IIS Filter on page 92 for instructions.

Installing the IIS Filter Using a Script

Perform the following steps to install the IIS filter (both IIS 5.x and 6.x) using a script:

1. Save the `<cd-base-directory>\filters\iis\win32\SFFilter.dll` anywhere on your local hard drive.

2. Execute the Visual Basic script `Install_SF_isapi_filter.vbs`, such as:

   `Install_SF_isapi_filter.vbs` *`<SiteId>`* *`<FilterDLLFullPath>`*.

   For example:

   ```
   Install_SF_isapi_filter.vbs 1 <location-on-hard-disk>\SFFilter.dll
   ```

   The `SiteId: 1`, usually corresponds to the default web site, but the site you chose to protect may have a different `SiteId`. Be sure to determine which `SiteId` corresponds to your web site before running the script.

3. For IIS 6.0 users, add the web service extension by uncommenting the last two lines in the `Install_SF_isapi_filter.vbs` script.

   The IIS filter is installed.

4. After installing the IIS filter you must do the following steps or the filter may not load properly:

   a. Add `<`*`cd-base-directory`*`>\filters\iis\win32\redist` to the system PATH variable.

   b. Reboot the machine.

   You are now ready to configure the IIS filter. See How to Configure the IIS Filter on page 92 for instructions.

How to Uninstall the IIS Filter Configuration Interface

When you uninstall the IIS filter configuration interface, you uninstall the IIS filter configuration UI schema properties and the DLL files.

Perform the following steps to uninstall the IIS filter configuration interface:

1. Double-click on the un`install.cmd` executable in `<`*`cd-base-directory`*`>\filters\iis\win32\conf\`.

   This command uninstalls the IIS filter configuration UI schema properties and the DLL files.

2. Close the IIS console and restart it again.

How to Uninstall the IIS Filter

You can uninstall the IIS filter in one of two ways:

- Using the IIS console
- Executing a script

Uninstalling the IIS Filter Using the IIS Console

Perform the following steps to uninstall the IIS filter using the IIS console:

1. Select the Web Site you chose to protect in the left panel in which you want to uninstall the IIS filter, and right-click.

2. Select the **Properties** menu option, then the **ISAPI Filters** tab in the Web Site Properties dialog.

3 Select the filter you want to uninstall such as the **SFFilter** you added.

4 Click on the **Remove** button and click **OK**.

The filter is uninstalled.

Uninstalling the IIS Filter Using a Script

Perform the following steps to uninstall the IIS filter (both IIS 5.x and 6.x) using a script:

1 Execute the Visual Basic script `Remove_SF_isapi_filter.vbs`, such as:

```
Remove_SF_isapi_filter.vbs <SiteId> <FilterDLLFullPath>.
```

For example:

```
Remove_SF_isapi_filter.vbs 1 <cd-base-directory>\filters\iis\win32\
SFFilter.dll
```

The `SiteId: 1`, usually corresponds to the default web site, but the site you chose to protect may have a different `SiteId`. Be sure to determine which `SiteId` corresponds to your web site before running the script.

➤ For IIS 6.0, you need to uncomment (if commented) the last two lines from the `Install_SF_isapi_filter.vbs` and `Remove_SF_isapi_filter.vbs` scripts.

The filter is uninstalled.

How to Configure the IIS Filter

➤ Before configuring the IIS filter, you must have installed the IIS filter configuration interface (see How to Install the IIS Filter Configuration Interface on page 88) and the IIS filter (see How to Install the IIS Filter on page 89).

There are two ways in which you can configure the filter using the filter configuration interface:

- At the web site level — This configuration is useful when you would like to protect access to all the virtual directories under the site using the IIS filter.
- At the virtual directory level — This configuration gives you the ability to do more fine-grained access control, so that each virtual directory can have its own filter configuration.

The IIS filter is configured by using the IIS filter configuration interface integrated with the IIS console.

Perform the following steps to 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 Configure the IIS server or web site to be protected with server level configurations (cookie name, login page and so on):

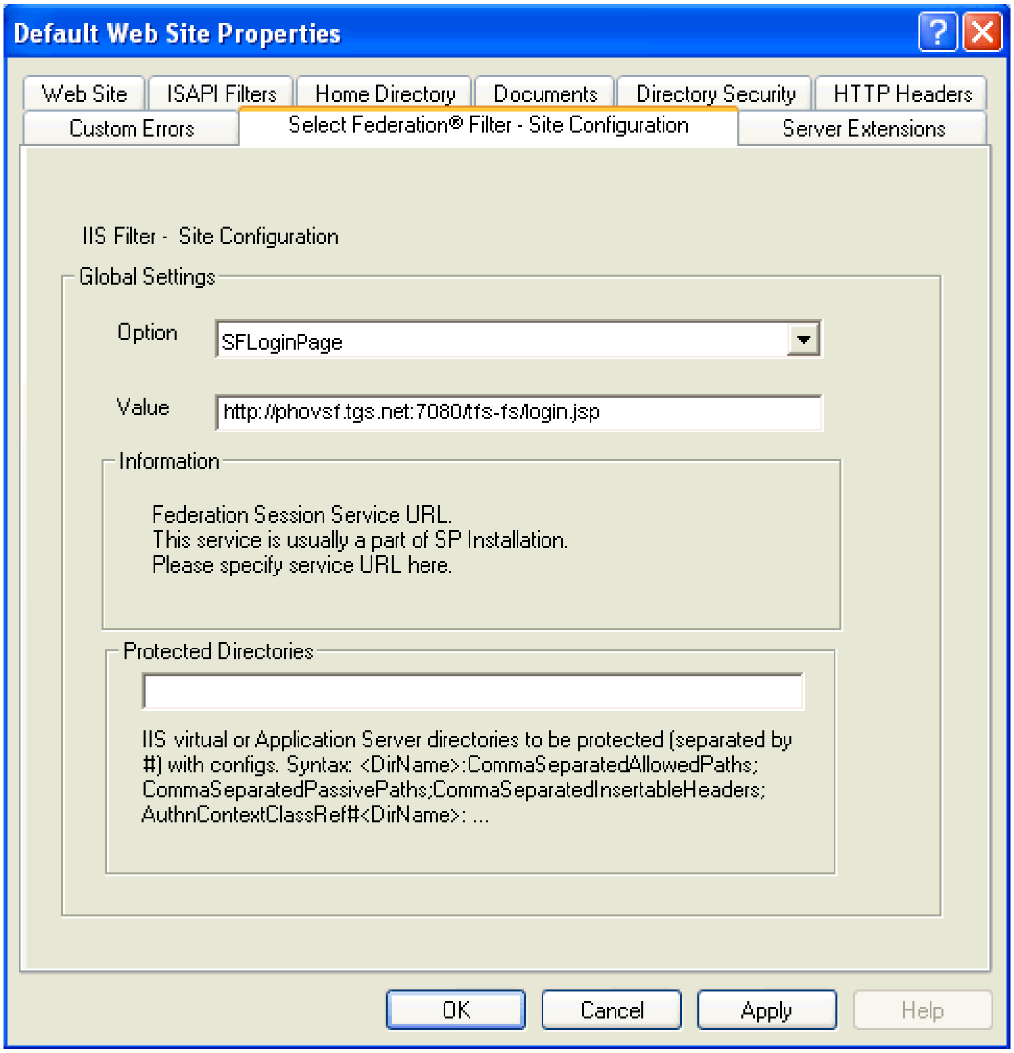
a Select the web site you want to protect → Right click → Properties
(On Windows 2000 and Windows 2003 servers, the same can be done as: Select IIS Server name (local computer) →Right click →Properties).

The Web Site Properties dialog opens.

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

c Configure the first property in the **Options** drop-down list by entering a value in the **Value** text box.

Following is an example of configuring a property.



IIS Filter - Site Configuration

Global Settings

Option: SFLoginPage

Value: http://phovsf.tgs.net:7080/tfs-fs/login.jsp

Information

Federation Session Service URL.
This service is usually a part of SP Installation.
Please specify service URL here.

Protected Directories

IIS virtual or Application Server directories to be protected (separated by #) with configs. Syntax: <DirName>:CommaSeparatedAllowedPaths;CommaSeparatedPassivePaths;CommaSeparatedInsertableHeaders;AuthnContextClassRef#<DirName>: ...

d Click on **Apply** to save your setting.

e Continue configuring each of the properties listed in the **Option** drop-down list.

Click on **Apply** after entering a value for each configuration parameter so that your entries are saved correctly.

Option	Value
SFLoginPage	Provide a URL to a page where a user without the proper cookie for the Filter (to use as credentials) is redirected. You can use the login.jsp page provided in the tfs-fs.war file. For example: **http://<sp-site>/tfs-fs/login.jsp** or **https://<sp-site>/tfs-fs/login.jsp**
SFFederationSessionService (FSService)	Provide a URL to the FSService servlet. If you have finished the steps in How to Configure Filter-Support on page 86, the URL can be formed as either **http://<sp-site>/tfs-fs/FSS** or **https://<sp-site>/tfs-fs/FSS** depending on your SP deployment.
SFLogFile	Provide a location on your system where you feel it is appropriate for the Filter to log its messages. For example: c:\Program Files\Select Federation\logs\SFFilter.log. It is essential to specify the log file location since the log file is important for re-confirming the configuration used by the filter, understanding the filter functionality and gathering information to debug any issues. **Note**: 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 will need to figure out to which virtual directory the filter writes the file.
SFDebug	Provide the value as **TRUE** for the filter to log debug statements or **FALSE** not to log debug statements.
SFCACerts - (Certificate Authority Certificates)	Provide the file path name of the CA certificates for server validation.
SFSkipServerAuth - (Skip Server Authentication)	Provide the value as **True** to skip server authentication or **False** to authenticate. See “Appendix C, Configuring Server and Client Authentication” in the *HP OpenView Select Federation Web Application Developer’s Guide* for more information.
SFCookieName	Provide the name of the cookie. The filter checks this in the request headers. The cookie name must be the same as the cookie set by the Event Plugin.

Option	Value
`SFSkipClientAuth` - (Skip Client Authentication)	Provide the value as **True** to skip client authentication or **False** to authenticate. See "Appendix C, Configuring Server and Client Authentication" in the *HP OpenView Select Federation Web Application Developer's Guide* for more information.
`SFKeyFile`	Provide the path name of the key file containing the private key to use along with the certificate.
`SFCertFile` (Certification File)	Provide the path name of the certification file that contains the client certificate to be used for client authentication.
`SFPassPhrase`	If the Key file is encrypted, provide the passphrase to use the key file.

f Click on **OK** to save the changes and close the window, or click on **Apply** to save the changes and keep the window open.

3 Configure virtual directories with directory level configurations:

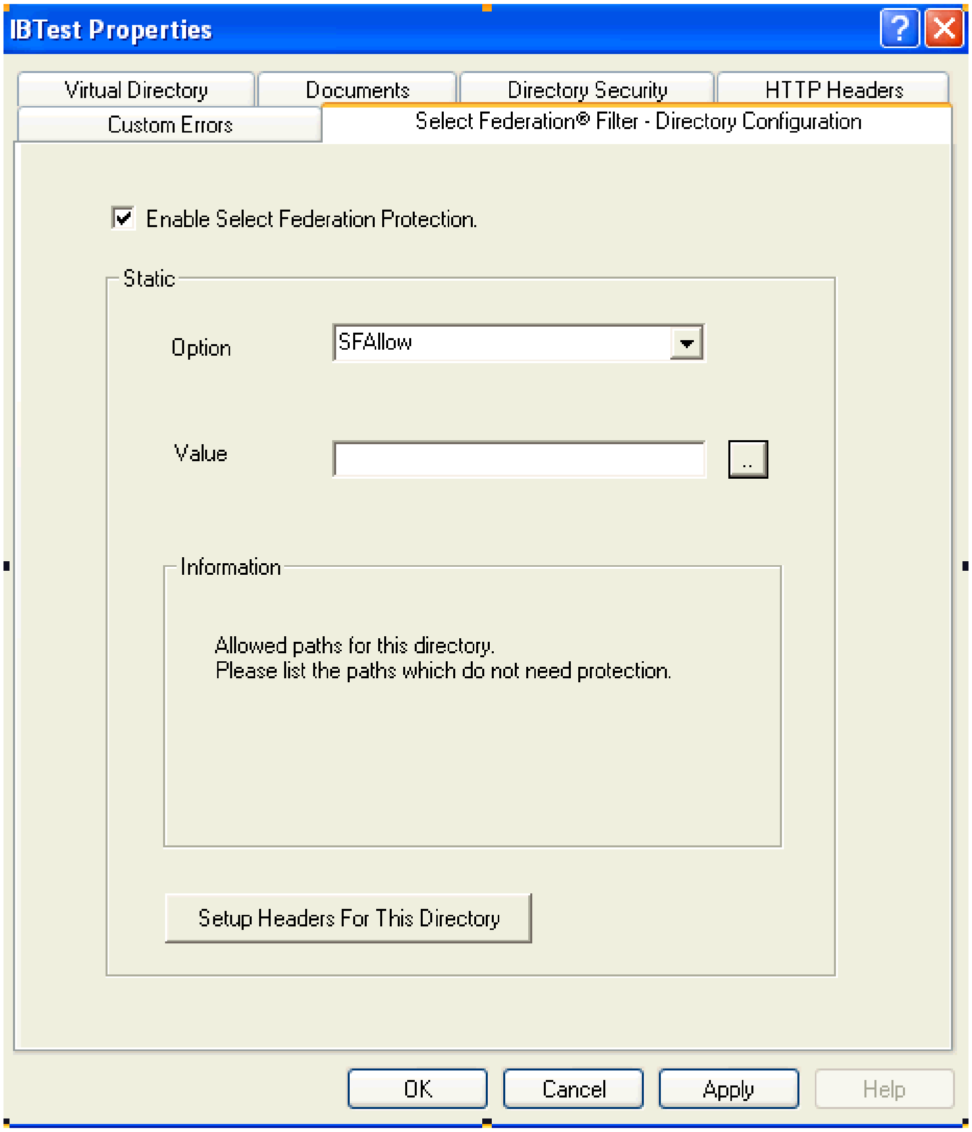
a Select a virtual directory in a web site.

b Right-click on the virtual directory and select the **Properties** menu option.

The web site Properties dialog opens.

c Select the **Select Federation Filter - Directory Configuration** tab.

d Check the **Enable Select Federation Protection** check box if you want to protect the virtual directory as shown in the following figure:



☑ Enable Select Federation Protection.

Option: SFAllow

Value:

Allowed paths for this directory.
Please list the paths which do not need protection.

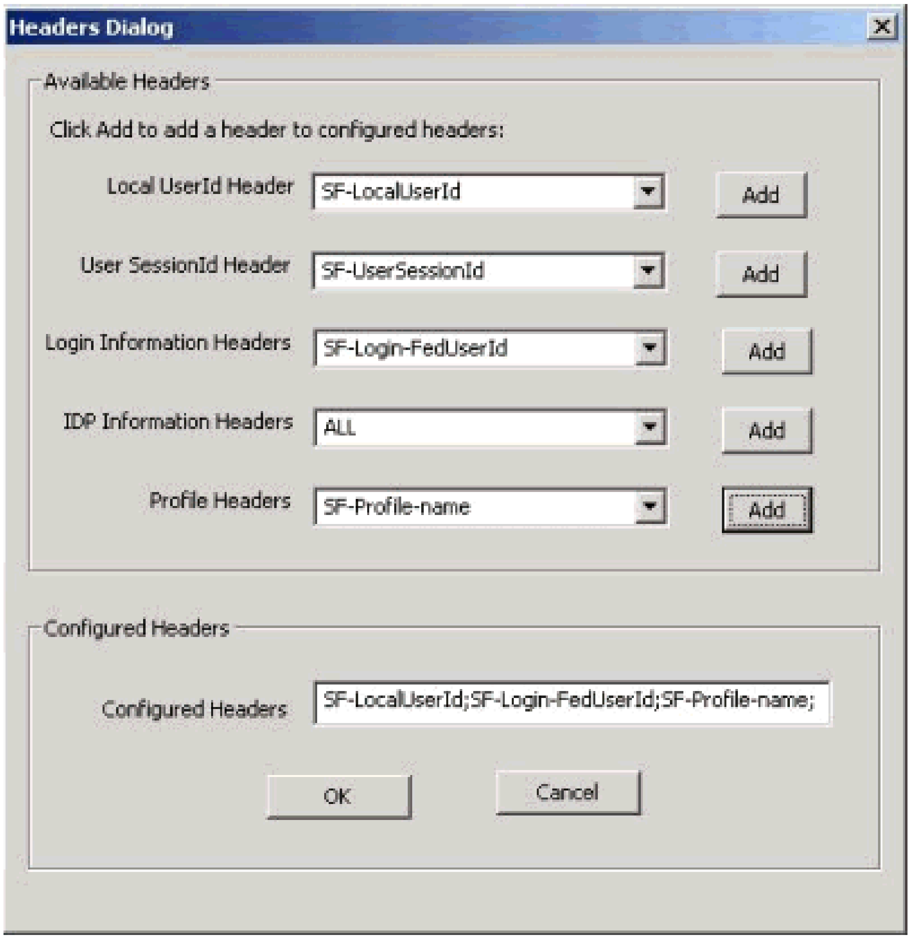
e Optionally, configure the following available properties in the **Option** drop-down list:

Option	Value
`SFAllow`	Provide a list of semicolon-separated relative URLs that should not be protected by the filter, in one of two ways: • Browse for the relative URLs by clicking the browse button (**..**) next to the Value input field. • Enter the relative URLs manually. For example, with URLs such as: **http://superSecure.it.com/secureNow.asp** and **http://superSecure.it.com/secureLater.asp**, the relative URLs would be entered as: **secureNow.asp;secureLater.asp;**
`SFPassive`	Provide a list of semicolon-separated relative URLs that should be treated as passive URLs (see Overview of Filters on page 82). The format is similar to the example given above for `SFAllow`.
`SFAuthnContext`	Choose an Authentication Context from the drop-down list, which serves as the minimum level of authentication needed to access the resources in the virtual directory. Note that the strength of the Authentication Context you configure should be less than or equal to the one specified in your SP's `tfsconfig.properties` configuration file. For example, if you select `SmartCard` in the filter configuration, and the default AuthnContext configured at the SP is `PasswordProtectedTransport`, you will not be granted access to the protected resource. However, if you specify `Password` in the filter configuration, you will be granted access. You can choose from the following values (listed in increasing levels of strength): • `Password` • `PasswordProtectedTransport` • `MobileContract` • `MobileDigitalID` • `PreviousSession` • `Smartcard` • `SmartcardPKI` • `SoftwarePKI` • `TimeSyncToken` If you do not specify a value for the authentication context, the default authentication context is used. This means that any level of authentication will be allowed.

f Optionally, click on the **Setup headers for this directory** button to customize which headers are set.

By default, all the information pertaining to the session, IDP, and user profile is made available as HTTP header variables, which you can customize.

The Headers Dialog opens.



Headers Dialog

Available Headers

Click Add to add a header to configured headers:

Local UserId Header: SF-LocalUserId

User SessionId Header: SF-UserSessionId

Login Information Headers: SF-Login-FedUserId

IDP Information Headers: ALL

Profile Headers: SF-Profile-name

Configured Headers

Configured Headers: SF-LocalUserId;SF-Login-FedUserId;SF-Profile-name;

g Select the header variables you wish to add and click the **Add** button after every selection.

The consolidated list displays in the **Configured Headers** section of the dialog box.

h Click on **OK** to save the changes and close the window, or click on **Apply** to save the changes and keep the window open.

4 Optionally, verify whether the properties are successfully added by doing one of the following:

a Select the properties of the virtual directory or server and open the Select Federation tab.

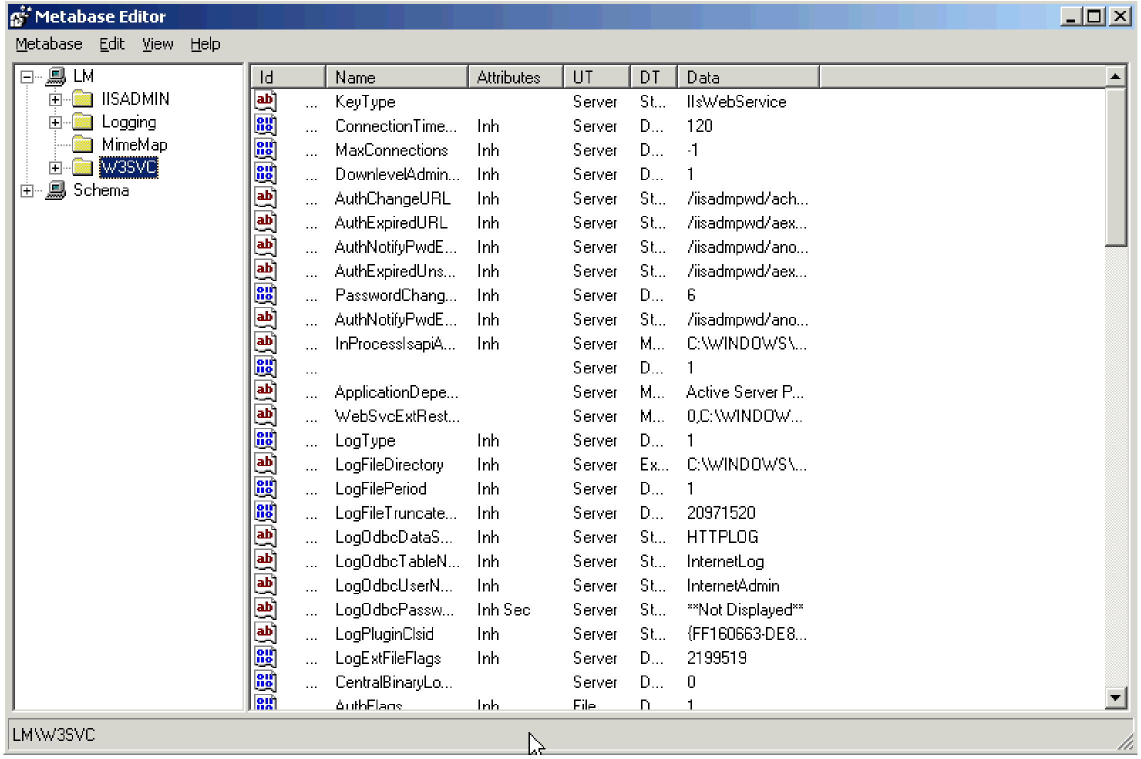
You should see the properties with the configured values.

b You can use the MetaEdit tool to view the configured properties and their respective values. Download this tool from the following page and click the **Mtaedt22.exe** link:

http://support.microsoft.com/default.aspx?scid=kb;en-us;301386&sd=tech

- To view the server level properties: select **LM → W3SVC**
- To view the web site level properties: select **LM → W3SVC→ 1 → ROOT**
- To view the virtual directory level properties: select **LM → W3SVC→ 1→ROOT →** *<DirName>*.

On the right-hand side you will see the configured properties under the Name column and their values under the Data column as shown in the following figure.



Id	Name	Attributes	UT	DT	Data
...	KeyType		Server	St...	IIsWebService
...	ConnectionTime...	Inh	Server	D...	120
...	MaxConnections	Inh	Server	D...	-1
...	DownlevelAdmin...	Inh	Server	D...	1
...	AuthChangeURL	Inh	Server	St...	/iisadmpwd/ach...
...	AuthExpiredURL	Inh	Server	St...	/iisadmpwd/aex...
...	AuthNotifyPwdE...	Inh	Server	St...	/iisadmpwd/ano...
...	AuthExpiredUns...	Inh	Server	St...	/iisadmpwd/aex...
...	PasswordChang...	Inh	Server	D...	6
...	AuthNotifyPwdE...	Inh	Server	St...	/iisadmpwd/ano...
...	InProcessIsapiA...	Inh	Server	M...	C:\WINDOWS\...
...			Server	D...	1
...	ApplicationDepe...		Server	M...	Active Server P...
...	WebSvcExtRest...		Server	M...	0,C:\WINDOW...
...	LogType	Inh	Server	D...	1
...	LogFileDirectory	Inh	Server	Ex...	C:\WINDOWS\...
...	LogFilePeriod	Inh	Server	D...	1
...	LogFileTruncate...	Inh	Server	D...	20971520
...	LogOdbcDataS...	Inh	Server	St...	HTTPLOG
...	LogOdbcTableN...	Inh	Server	St...	InternetLog
...	LogOdbcUserN...	Inh	Server	St...	InternetAdmin
...	LogOdbcPassw...	Inh Sec	Server	St...	**Not Displayed**
...	LogPluginClsid	Inh	Server	St...	{FF160663-DE8...
...	LogExtFileFlags	Inh	Server	D...	2199519
...	CentralBinaryLo...		Server	D...	0
	AuthFlags	Inh	File	D	1

With IIS 6.0 (Win2K3) and IIS 5.0 (Win2K), you can configure the server either by right clicking on the server name (local computer), or specific web site you chose to protect and then selecting the properties menu. However, for IIS 5.1 (Win XP), the server configurations can ONLY be done by right clicking on the specific web site and selecting the properties menu.

After making any changes to the filter settings using the configuration interface, do one of the following:

- It is recommended that you browse for the following dummy URL in order for the filter settings to take effect:

  http://<iis-server-machine>/SFFilterConfigure.html

  The filter generates a `200 OK` response and sends a page with the following text to the client:

  ```
  SF filter configured!
  ```

- You can verify that the configuration changes have taken effect by looking at the filter log file.

How to Use the IIS Filter

This section provides an IIS filter sample to help you know how to use the IIS filter. This filter sample is provided as an example of how the filter may be used to retrieve information about the federated session, user profile, and partner IDP. This sample is very basic in that it prints all the headers that have been configured for that particular site or virtual directory.

➤ Be sure that you have installed a partner IDP, which is configured to use an Access Management system or a directory server. The IDP is required to authenticate a user that initiates a federation from the SP.

Filter Sample

To access the sample, perform the following steps:

1 Copy the `getheaders.asp` sample from the `<`*`cd-base-directory`*`>\filters\iis samples\` directory to your protected site or virtual directory.

2 Open the `getheaders.asp` file and search for `SFRoot` to set `it` to the root URL of your Select Federation Application server SP instance.

3 Save your changes.

4 Now try to access the sample. For example:

http://<iis_servername>/<protectedvirtual_directory>/getheaders.asp

The browser should now be redirected to a login page, as configured in the Select Federation filter configuration interface (see Step 2 in How to Configure the IIS Filter on page 92).

5 Select the IDP that you would like to authenticate with from the drop-down list and click the **Login** button.

The IDP login page opens where you are asked to enter your credentials.

6 Login with any user account that exists in the access management system or directory server that has been configured at the IDP.

Once you as the user is authenticated by the IDP, the IIS filter then calls the FSService servlet to get the user information. The FSService should have been configured in the Select Federation filter configuration interface as a URL value for `SFFederationSessionService` (see Step 2 in How to Configure the IIS Filter on page 92).

The request is then forwarded to the desired page: `getheaders.asp` in this sample, which prints out all the headers it receives.

IIS Filter Log File

The IIS filter log file path can be configured using the configuration interface. If the log file is not configured from the interface, then the filter looks for the `SFHome` registry entry under `HKEY_LOCAL_MACHINE` → `SOFTWARE` → `\Hewlett-Packard\OpenView` and creates the `Server\Logs\SFFilter.log` file relative to `SFHome`.

If the `SFHome` registry entry is not found, the IIS filter logs details into the `C:\SFFilter.log` file. This generated log file can be used to see the request processing details, headers set and cookie information.

➤ If you set `SFDebug` to `True` in the Select Federation configuration interface, then the filter will output debug statements to the log file. This will help gain a better understanding of the inner working of the filter and help with troubleshooting.

Following is an example of a log file with the filter configuration information and request processing details.

```
Filter Configuration Details:
-----------------------------
Server configs:
ookieName=SFSession
LoginPage=http://localhost:10432/fs/login.jsp
FederationSessionService=http://localhost:10432/fs/fss.jsp
Debug=TRUE
SkipClientAuth=TRUE
CertFile=
KeyFile=
PassPhrase=
SkipServerAuth=TRUE
CACerts=
Virtual directory configs:
   /SFTest/:index.html;;SF-IDP-Home,SF-Profile-name-firstname,;

Protected URL: /SFTest/getheaders.asp
Cookie-Header:ASPSESSIONIDAARSDBSD=HNOJGPMCAPGJIBFAOKCONDDF;
SFSession=2dc4d67ae068b6979bf9f8097239e8b14eab1f2c
Expected cookie found
Successfully received info from Federation Session Service...
Cached info for sessionId=2dc4d67ae068b6979bf9f8097239e8b14eab1f2c
User information is set in the headers
SF-IDP-Home:null SF-Profile-name-firstname:testname
```

Apache Filter

The Apache filter is implemented as a shared library (`.DLL` for Windows or `.SO` for Linux).

For the Apache filter, the following platforms are supported:

- Windows 2000 and 2003
- Red Hat Linux AS 3.0 and 4.0

You install the Apache filter by copying the filter shared library into the `<apache_home>/modules>` directory. You then configure the Apache filter by modifying the `httpd.conf` file. If you need to change default settings, you can modify the `filter.conf` file.

Before you install and configure the Apache filter, be sure to carefully look at its prerequisites.

In Select Federation, the Apache 2.0 module binaries are compatible with specific versions of Apache on Linux, but not with Apache 1.3. For details on supported Apache versions, see Supported Apache Versions on Linux on page 108.

The following sections provide instructions for the Apache filter processes and information on supported versions for the Apache filter on Linux platforms:

- How to Install the Apache Filter
- How to Configure the Apache Filter
- How to Use the Apache Filter
- Apache Filter Log File
- Supported Apache Versions on Linux

How to Install the Apache Filter

Complete the following steps to install the Apache filter on Windows and Linux platforms.

Installing the Apache Filter On Windows Platforms

Perform the following steps to install the Apache filter on Windows platforms:

1. Copy the Apache filter binary from `<cd-base-directory>/filters/apache/win32/SFModule.dll` to the `<apache_home>/modules/` directory.
2. Copy the files present in the `<cd-base-directory>/filters/apache/win32/redist/` folder into a local folder such that they will be in the system PATH. Or copy them into a new local folder and edit the system PATH to point to it as well.
3. Add the following line to the `<apache_home>/conf/httpd.conf` file:

   ```
   LoadModule SFModule_module modules/SFModule.dll
   ```
4. Copy the sample configuration file `<cd-base-directory>/filters/apache/conf/filter.conf` to the `<apache_home>/conf/` directory.
5. Restart your Apache web server.

Installing the Apache Filter On Linux Platforms

Perform the following steps to install the Apache filter on Linux platforms:

1. Configure Filter-Support as described in How to Configure Filter-Support on page 86.
2. Copy the Apache filter binary from `<cd-base-directory>/filters/apache/linux/SFModule.so` to the `<apache_home>/modules/` directory.

   The Apache filter binary includes the following run-time dependencies:

   - Apache APR
   - `OpenSSL`
   - `libcurl` (built with `OpenSSL`)
   - `libxml2`
3. Check the run-time dependencies of the Apache filter module to be sure all required components are present.

   a. Enter the following command at the prompt (#):

   ```
   # ldd <apache_home>/modules/SFModule.so
   ```

   The output may look something like the following:

   ```
   libxml2.so.2 => /usr/lib/libxml2.so.2 (0xb74bb000)
   libz.so.1 => /usr/lib/libz.so.1 (0xb74ad000)
   libcurl.so.3 => not found
   libdl.so.2 => /lib/libdl.so.2 (0xb74aa000)
   libstdc++-libc6.2-2.so.3 => /usr/lib/libstdc++-libc6.2-2.so.3 (0xb7468000)
   libm.so.6 => /lib/tls/libm.so.6 (0xb7446000)
   libc.so.6 => /lib/tls/libc.so.6 (0xb730e000)
   libpthread.so.0 => /lib/tls/libpthread.so.0 (0xb72fe000)
   /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x80000000)
   libgcc_s.so.1 => /lib/libgcc_s.so.1 (0xb72f5000)
   ```

   Any `not found` results indicate that either the dependencies are not installed, or do not exist in the linker path.

b Enter the following command at the prompt (#) to copy the missing components provided in the <*cd-base-directory*>/filters/apache/linux/redist/ directory to a new local directory:

```
# cp -r <cd-base-directory>/filters/apache/linux/redist/ <local-dir>/filters/
```

c Add the path of the new local directory (<*local-dir*>/filters/redist) to the /etc/ld.so.conf file.

d Execute the following command at the prompt (#):

```
# ldconfig
```

You might see some warnings such as:

```
ldconfig: /var/tmp/filters/redist/libxml2.so.2 is not a symbolic link
ldconfig: /var/tmp/filters/redist/libcurl.so.3 is not a symbolic link
ldconfig: /var/tmp/filters/redist/libapr-1.so.0 is not a symbolic link
```

Ignore these warnings and proceed to the next step.

e Enter the following command at the prompt again:

```
#ldd <apache_home>/modules/SFModule.so
```

This time, the results should show that all dependencies are satisfied.

```
libxml2.so.2 => /var/tmp/filters/redist/libxml2.so.2 (0xb74a4000)
libz.so.1 => /usr/lib/libz.so.1 (0xb7496000)
libcurl.so.3 => /var/tmp/filters/redist/libcurl.so.3 (0xb7464000)
libdl.so.2 => /lib/libdl.so.2 (0xb7461000)
libstdc++-libc6.2-2.so.3 => /usr/lib/libstdc++-libc6.2-2.so.3 (0xb741f000)
libm.so.6 => /lib/tls/libm.so.6 (0xb73fd000)
libc.so.6 => /lib/tls/libc.so.6 (0xb72c5000)
libpthread.so.0 => /lib/tls/libpthread.so.0 (0xb72b5000)
libssl.so.0.9.8 => /var/tmp/filters/redist/libssl.so.0.9.8 (0xb727a000)
libcrypto.so.0.9.8 => /var/tmp/filters/redist/libcrypto.so.0.9.8 (0xb7155000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x80000000)
libgcc_s.so.1 => /lib/libgcc_s.so.1 (0xb714c000)
```

➤ If libstdc++-libc6.2-2.so.3 is not found, you need to install the rpm appropriate to your system to satisfy this dependency.

4 Add the following line to the <apache_home>/conf/httpd.conf file:

```
LoadModule SFModule_module modules/SFModule.so
```

5 Copy the sample configuration file <cd-base-directory>/filters/apache/conf/filter.conf to the <apache_home>/conf/ directory.

6 Restart your Apache web server.

How to Configure the Apache Filter

➤ Before configuring the Apache filter, you must have installed the Apache filter (see How to Install the Apache Filter on page 102).

Configure the Apache filter by completing the following tasks:

- Task 1: Edit the <apache_home>/conf/filter.conf file to configure the filter-specific options to match your SP installation:

  The following filter-specific options are shared across all directories:

  — CookieName

- `LoginPage`
- `FederationSessionService`
- `Debug`
- `SkipClientAuth`
- `CertFile`
- `KeyFile`
- `PassPhrase`
- `SkipServerAuth`
- `CACerts`

- Task 2: Edit the <apache_home>/conf/httpd.conf file to configure the directories and locations that you wish to protect:

Task 1: Edit the <apache_home>/conf/filter.conf file to configure the filter-specific options to match your SP installation:

Provide values for the following filter-specific options to match your SP installation:

Uncomment the following line:	**Provide the following value:**
`#CookieName=..`	The same cookie name as the one you configured for the `filterSupport.cookieName` parameter in the `tfsconfig.properties` file. (See How to Configure Filter-Support on page 86.)
`#LoginPage=...`	A URL to a page where a user without the proper cookie for the filter (to use as credentials) is redirected. You can use the `login.jsp` page provided in the `tfs-fs.war` file. For example: **http://<sp-site>/tfs-fs/login.jsp** or **https://<sp-site>/tfs-fs/login.jsp**
`#FederationSessionService=...`	A URL to the FSService servlet which is called by the filter to get LoginInfo, IDPInfo, User ProfileInfo and so on. If you have finished the steps in How to Configure Filter-Support on page 86, the URL can be formed as either **http://<sp-site>/tfs-fs/FSS** or **https://<sp-site>/tfs-fs/FSS** depending on your SP deployment.
`#Debug=...`	**TRUE** for the filter to log debug statements or **FALSE** not to log debug statements.
`#SkipServerAuth=...`	**TRUE** to skip server authentication or **FALSE** to authenticate. See Configuring Filter Authentication on page 115 for more information.
`#CACerts=...`	The path to the file that contains the CA certificates for server validation.
`#SkipClientAuth=...`	**TRUE** to skip client authentication or **FALSE** to perform client authentication for https communication. See Configuring Filter Authentication on page 115 for more information.

Uncomment the following line:	Provide the following value:
`#CertFile=...`	The path to the file that contains the client certificate, to be used for client authentication during the SSL Handshake. The certificate should be in PEM format.
`#KeyFile=...`	The path to the file that contains the private key, to be used with the client certificate for client authentication during the SSL Handshake.
`#PassPhrase=...`	If the key file is encrypted, provide the passphrase to use the key file.

Task 2: Edit the <apache_home>/conf/httpd.conf file to configure the directories and locations that you wish to protect:

1 Add the `SFProtect` option to the specified directories to register them with the filter.

```
<Directory "/path/to/directory">
   ... Other Standard Options ...
   SFProtect
   ...
</Directory>
```

2 Specify the headers for the values to be retrieved:

```
<Directory "/path/to/directory">
   ... Other Standard Options ...
   SFProtect
   SFHeaders SF-LocalUserId,SF-UserSessionId,SF-Login,SF-IDP,SF-Profile
   ...
</Directory>
```

➤ The above configuration for `SFHeaders` is equivalent to specifying all the attributes individually, for example:

```
SFHeaders SF-Login-IdpFedUserId,SF-Login-AuthnContextClassRef,SF-
Login-AuthnInstant,SF-Login-ReauthOnOrAfter,SF-IDP-IdpProviderId,SF
-IDP-Home,SF-IDP-Name,SF-IDP-Description,SF-IDP-LogoRef,SF-IDP-Logo
Text,SF-Profile-name,SF-Profile-name-title,SF-Profile-name-firstnam
e,SF-Profile-name-lastname,SF-Profile-home-street,SF-Profile-home-c
ity,SF-Profile-home-state,SF-Profile-home-country,SF-Profile-home-p
ostalCode,SF-Profile-personal-email,SF-Profile-personal-phone,SF-Pr
ofile-work-street,SF-Profile-work-city,SF-Profile-workstate,SF-Prof
ile-work-country,SF-Profile-work-postalCode,SF-Profile-work-email,
SF-Profile-work-phone
```

See How Filters Work on page 83 for more details.

3 Specify the resources that should not be protected by the filter:

```
<Directory "/path/to/directory">
   ... Other Standard Options ...
   SFProtect
   SFHeaders...
   SFAllow allow.html,allow2.html
   ...
```

```
</Directory>
```

4. Specify the resources that should be passive:

```
<Directory "/path/to/directory">
   ... Other Standard Options ...
   SFProtect
   SFHeaders...
   SFAllow...
   SFPassive passiveurl1,passiveurl2
   ...
</Directory>
```

5. Optionally, specify a value for the Authentication Context to be the minimum level of authentication needed to access the resource.

   The strength of the Authentication Context you configure should be less than or equal to the one specified in your SP's `tfsconfig.properties` configuration file.

   For example, if you select `SmartCard` in the filter configuration, and the default AuthnContext configured at the SP is `PasswordProtectedTransport`, you will not be granted access to the protected resource. However, if you specify `Password` in the filter configuration, you will be granted access.

   You can choose from the following values (listed in increasing levels of strength):

   - `Password`
   - `PasswordProtectedTransport`
   - `MobileContract`
   - `MobileDigitalID`
   - `PreviousSession`
   - `Smartcard`
   - `SmartcardPKI`
   - `SoftwarePKI`
   - `TimeSyncToken`

   If you do not specify a value for the authentication context, the default authentication context is used. This means that any level of authentication will be allowed.

6. Specify an alternate Authentication context if you are not satisfied by the default one (usually you would not have a need to specify this):

```
<Directory "/path/to/directory">
   ...Other Standard Options ...
   SFProtect
   SFHeaders...
   SFAllow...
   SFPassive...
   SFAuthContext minContext
   ...
</Directory>
```

7. Specify a line such as `SFPassive passive.*, which` marks all the paths that begin with passive as passive paths.

How to Use the Apache Filter

This section provides an Apache filter sample to help you learn how to use the Apache filter. This filter sample shows an example of how the filter may be used to retrieve information about the federated session, user profile, and partner IDP. This sample is very basic in that it displays all the headers that have been configured for that particular site or virtual directory.

➤ Be sure that you have installed a partner IDP, which is configured to use an Access Management system or a directory server. The IDP is required to authenticate a user that initiates a federation from the SP.

Apache Filter Sample

To access and use the sample, perform the following steps:

1 Copy the `getheaders.php` sample from the `<`*`cd-base-directory`*`>/filters/samples/` directory to your protected site or protected directory.

2 Open the `getheaders.php` file and search for `SFRoot` to set it to the root URL of your Select Federation Application server SP instance.

3 Save your changes.

4 Now try to access the sample. For example:

http:// <servername:port>/<protectedDirectory>/getheaders.php

The browser should now be redirected to a login page, as configured in the Select Federation filter configuration file (see How to Configure the Apache Filter on page 103).

5 Select the IDP that you would like to authenticate with from the drop-down list and click the **Login** button.

The IDP login page opens where you are asked to enter your credentials.

6 Login with any user account that exists in the access management system or directory server that has been configured at the IDP.

Once you as the user is authenticated by the IDP, the Apache filter then calls the FSService servlet to get the user information. The FSService should have been configured in the Select Federation filter configuration file as a URL value for `SFFederationSessionService` (see How to Configure the Apache Filter on page 103).

The request is then forwarded to the desired page: `getheaders.php` in this sample, which displays all the headers it receives.

7 If `php` is not enabled, check the `<apache-root>/logs/error.log` log file. The filter logs debug (`if DEBUG=TRUE`) and normal log messages into a separate file `filter.log` in the `logs/` directory.

This makes it easier to see the filter- specific logs. Fatal errors go into the Apache `error_log` file.

Apache Filter Log File

The Apache filter module logs its messages into the `<`*`apache_home`*`>/logs/filter.log` file. If there are any problems in generating the log file, the messages are logged to the Apache error log file. For example: `<`*`apache_home`*`>/logs/error_log file`.

Following is an example of a log file with the filter configuration information and request processing details:

```
[SFFilter.init] Total directories Configured = 2.
```

```
[SFFilter.init] Cookie Name : SFSession.
[SFFilter.init] Login Page : https://sp1.HPOVSF.tgx.net:7443/tfs-fs/
login.jsp.
[SFFilter.init] Federation Session Service : https://sp1.HPOVSF.tgx.net:7443/
tfs-fs/FSS.
[SFFilter.init] Client Certificate File Path : newcert.pem.
[SFFilter.init] Client Key File Path : newreq.pem.
[SFFilter.init] Key File Passprhase : NO_PWD.
[SFFilter.init] CA Certificates File : /usr/local/test-scripts/trustedSP.pem.
[SFFilter.init] Skip Client Authentication : TRUE.
[SFFilter.init] Skip Server Authentication : FALSE.
[SFFilter.init] Debug : TRUE
[DEBUG: 07/Apr/2006:12:46:23 AM PDT] Match found at /sffilter/.*
[LOG: 07/Apr/2006:12:46:23 AM PDT]  Found Protected Url /sffilter/
getheaders1.php
[DEBUG: 07/Apr/2006:12:46:23 AM PDT] Did not match any Allowed Path : (/
sffilter/getheaders1.php)
[LOG: 07/Apr/2006:12:46:23 AM PDT]  Session ID Not Found.
[DEBUG: 07/Apr/2006:12:46:23 AM PDT] Did not match any Passive Path : (/
sffilter/getheaders1.php)
[LOG: 07/Apr/2006:12:46:23 AM PDT]  Redirected to sign-in page.
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Match found at /sffilter/.*
[LOG: 07/Apr/2006:12:46:37 AM PDT]  Found Protected Url /sffilter/
getheaders1.php
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Did not match any Allowed Path : (/
sffilter/getheaders1.php)
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Cookie Value
:SFSession=76ce83b37d20934e0d7d90e2760338e31510f62c
[LOG: 07/Apr/2006:12:46:37 AM PDT]  Found SessionId
(SFSession=76ce83b37d20934e0d7d90e2760338e31510f62c)
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Fetching Resource: (https://
sp1.HPOVSF.tgx.net:7443/tfs-fs/
FSS?sessionId=76ce83b37d20934e0d7d90e2760338e31510f62c)
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Status:Success
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Following headers are available.
-------
[DEBUG: 07/Apr/2006:12:46:37 AM PDT]
LocalUserId:d3d6a8fd470c3b1e7f0083cb5179b202f507a31c
[DEBUG: 07/Apr/2006:12:46:37 AM PDT]
UserSessionId:76ce83b37d20934e0d7d90e2760338e31510f62c
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Name:idp1
[DEBUG: 07/Apr/2006:12:46:37 AM PDT] Description:
....
```

Supported Apache Versions on Linux

This section provides information on supported versions for the Apache filter run-time dependencies on Linux platforms.

The Apache filter on the Linux platforms includes the following run-time dependencies:

- Apache httpd Web Server
- Apache APR
- cURL/libcURL
- Libxml2
- OpenSSL

The following sections describe the supported versions for each of the above dependencies.

Apache httpd Web Server

The Apache filter module binary is compatible with Apache 2.0.x versions that have a Module Magic Number that starts with 20020903 (has a major version component of). This means that this version of Select Federation is compatible with Apache 2.0.41+ to Apache 2.0.55.

Following is an example of how to determine the server version number and Module Magic Number for your installed version of apache:

```
# /usr/sbin/httpd -V
Server version: Apache/2.0.53
Server built: Sep 5 2005 09:28:47
Server's Module Magic Number: 20020903:9
```

➤ The Apache filter module binary is **not** compatible with Apache 1.3.x.

The Apache filter module was tested successfully with the following versions of Apache httpd web server:

- 2.0.55 on Red Hat Linux AS 3.0
- 2.0.52 on Red Hat Linux AS 4.0 ES Basic Edition

Apache APR

The Apache filter module binary is compatible with Apache APR 0.9.4, 1.2.2 and 1.2.6.

The Apache filter module was tested successfully with the following versions of Apache APR:

- 1.2.2 on Red Hat Linux AS 3.0
- 0.9.4 on Red Hat Linux AS 4.0

cURL/libcURL

The Apache filter module binary is compatible with `cURL` 7.12.1 and 7.15.3.

The Apache filter module was tested successfully with the following versions:

- `cURL` 7.15.3 on Red Hat Linux AS 3.0
- `libcURL` 7.12.1 on Red Hat Linux AS 4.0

Libxml2

The Apache Filter module binary is compatible with `Libxml2 2.6.16` and `2.6.23`.

The Apache Filter module was tested successfully with the following versions of `Libxml2`

- `2.6.23` on Red Hat Linux AS 3.0
- `2.6.16` on Red Hat Linux AS 4.0

OpenSSL

The Apache Filter module binary is compatible with `OpenSSL 0.9.7a` and `0.9.8a`.

The Apache Filter module was tested successfully with the following versions of `OpenSSL`:

- 0.9.8a on Red Hat Linux AS 3.0
- 0.9.7a on Red Hat Linux AS 4.0

Java Access Filter

The Java Access filter (JAF) can be used on any web server that acts as Java Servlet container. It is a simple, but powerful way to enable applications for federated access and personalization.

The following sections provide instructions for the JAF processes:

- How to Install the Java Access Filter
- How to Configure the Java Access Filter
- How to Use the Java Access Filter
- Java Access Filter Log

How to Install the Java Access Filter

The JAF is implemented as a servlet filter. To install the JAF, copy the `sffilter.jar` file from the `<cd-base-directory>/filters/servlet` directory into the `WEB-INF/lib` directory of the web application to be enabled. The next step is to configure your web application's deployment descriptor (`WEB-INF/web.xml`) file.

How to Configure the Java Access Filter

To add the JAF functionality to your application, you need to modify your applications deployment descriptor (`WEB-INF/web.xml`). The JAF configuration parameters are set in the deployment descriptor.

Deployment Descriptor Structure

The deployment descriptor contains an element called <filter>, which describes all the required parameters for access filters. The following parameters are specified in the <filter> element as child nodes:

- filter-name: Used for filter mapping.
- filter-class: Class that should be present in the jar that you copied during installation.

You need to specify various initialization parameters in the deployment descriptors. Initialization parameters are specified as child nodes of the <filter> element.

Filter global level parameters, which are configured in the <init-param> node, are as follows:

- SFCookieName: Name of the cookie. The filter checks this in the request headers. This cookie name should be the same as the cookie set by Event Plug-in. Default value is `SFSession`.
- SFLoginPage: Login page to which the filter redirects if the expected cookie is not found. For example: **http://SPMachine:port/tfs-fs/login.jsp**
- SFFederationSessionService: Location of the Federation Session Service which is called by the filter to get authentication-related information if the user is already authenticated and its principal is not available in cache. For example: **http://SPMachine:port/tfs-fs/FSS**.
- SFDebug: Enable or disable debug logs: TRUE (enable) and FALSE (disable). Default value is FALSE. If the debug flag is enabled, then the user receives detailed logging information. If the debug flag is disable, only high-level messages are logged in the configured log file.

- SFLogFile: Location of the file in which log messages are logged. The default value is AccessFilter.log, which is created in the default directory of the web server being used by the web application.

Directory Configuration Parameters

By default all resources in the application are protected. To unprotect a resource, it must be added to the allow list

The following parameters apply to directory-level configurations:

- SFAllow: Comma separated list of allowed files, or file paths.
- SFPassive: Comma separated list of passive files.
- SFAttributes: Comma separated list of user authentication information elements to add to requests.
- SFAuthnContext: Authentication context ('level') for the specified directory path.

Administrators can specify the above parameters at various levels. Directory-level configurations are specified in same format as the <init-param> nodes, but the param-name values are prefixed with the directory path.

To specify directory-specific values for (SFAllow, SFPassive, SFAttributes, SFAuthnContext) you need to configure a parameter called SFProtect. The parameter value for SFProtect is a comma separated list of resource names. For example, to specify specific values for “test” as a protected directory, add:

```
<init-param>
   <param-name>SFProtect</param-name>
   <param-value>/test, /protectme</param-value>
</init-param>
```

Then if needed, add specifics as follows:

```
<init-param>
   <param-name>/test.SFPassive</param-name>
   <param-value>passive.html</param-value>
</init-param>
```

Sample Configuration File

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE web-app
   PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
   "http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>
   <display-name>Servlet Filter Demo</display-name>
   filter>
      <filter-name>AccessFilter</filter-name>
      <filter-class>
      com.hp.ov.selectfederation.filters.AccessFilter
      </filter-class>


      <init-param>
         <param-name>SFProtect</param-name>
         <param-value>/test, /protected</param-value>
```

```
</init-param>

<init-param>
   <param-name>SFAttributes</param-name>
   <param-value>
         SF-UserSessionId,SF-Login,SF-IDP,SF-Profile
   </param-value>
</init-param>

<init-param>
   <param-name>SFAllow</param-name>
   <param-value>gifs</param-value>
</init-param>

<init-param>
   <param-name>/test.SFAllow</param-name>
   <param-value>
         allow.html, nestedtest
   </param-value>
</init-param>

<init-param>
   <param-name>/test.SFPassive</param-name>
   <param-value>passive.html</param-value>
</init-param>
<init-param>
   <param-name>/test.SFAttributes</param-name>
   <param-value>SF-Login, SF-IDP, SF-LocalUserId</param-value>
</init-param>
<init-param>
   <param-name>/test.SFAuthnContext</param-name>
   <param-value>minContext</param-value>
</init-param>


<init-param>
   <param-name>SFSkipClientAuth</param-name>
   <param-value>true</param-value>
</init-param>
<init-param>
   <param-name>SFSkipServerAuth</param-name>
   <param-value>true</param-value>
</init-param>
<init-param>
   <param-name>SFKeyStore</param-name>
   <param-value>conf/keystore.jks</param-value>
</init-param>
<init-param>
   <param-name>SFKeyStorePass</param-name>
   <param-value>password</param-value>
</init-param>
<init-param>
   <param-name>SFKeyPass</param-name>
   <param-value>password</param-value>
</init-param>
```

```
      <init-param>
         <param-name>SFCACertsStore</param-name>
         <param-value>conf/keystore.jks</param-value>
      </init-param>
      <init-param>
         <param-name>SFCACertsPass</param-name>
         <param-value>password</param-value>
      </init-param>
         <param-name>SFCookieName</param-name>
         <param-value>SFSession</param-value>
      </init-param>
      <init-param>
         <param-name>SFLoginPage</param-name>
         <param-value>
               http://127.0.0.1:4001/tfs-fs/login.jsp
         </param-value>
      </init-param>
      <init-param>
         <param-name>SFLogFile</param-name>
         <param-value>AccessFilter.log</param-value>
      </init-param>
      <init-param>
         <param-name>SFDebug</param-name>
         <param-value>false</param-value>
      </init-param>
</filter>
<filter-mapping>
      <filter-name>AccessFilter</filter-name>
      <url-pattern>/*</url-pattern>
</filter-mapping>
<taglib>
      <taglib-uri>
         http://jakarta.apache.org/taglibs/i18n-1.0
      </taglib-uri>
         <taglib-location>/WEB-INF/taglibs-i18n.tld</taglib-location>
      </taglib>
</web-app>
```

Authentication Configuration Parameters

The following parameters apply to authentication configurations:

- SFSkipClientAuth: Client authentication for https communication. Skip client authentication: TRUE (skip) and FALSE (authenticate). Default is TRUE.
- SFSkipServerAuth: Server authentication for https communication. Skip server authentication: TRUE (skip) and FALSE (authenticate). Default is TRUE.
- SFKeyStore: Holds the path to a keystore with the client certificate.
- SFKeyStorePass: Password for the keystore.
- SFKeyPass: Password for the private key that belongs to the client certificate.

The client certificate in this keystore should be trusted by the server.

It is possible to use a non-default trust store for the TLS client authentication connection to the Select Federation install. The trust store is not needed if the TLS certificate that is presented by the Select Federation installation web server, is trusted by the application server that runs the filter (that is in the cacerts file).

- SFCACertsStore: Keystore with CA certificates.
- SFCACertsPass: Password for that store.

Other Configuration Parameters

- SFCacheSize: Number of objects to hold in cache (defaults to 1000).
- SFCacheTime: Number of seconds to determine objects as “valid” (defaults to 15 minutes).
- SFSloResource: Special URL for application pages to use to initiate SLO.

Defaults to /SingleLogOut. When present in web-xml but empty the SLO handling by the filter is disabled.

How to Use the Java Access Filter

After installation and configuration, try to access any URL from protected directory under web-application where JAF is installed. It should redirect the browser to login page. Once authentication is done, FSS will be called to get valid principal information. This principal information will be put in cache. Using the principal, filters will set the attributes in the request according to header configuration and request will be forwarded to desired protected page.

If you try to access the same page again, it will display the contents, since the user is authenticated and the principal information is also available within the filter in cache.

For allowed pages, no authentication will be done and request will be forwarded directly to the desired location. For passive pages, no user authentication will be done. Once you select your IDP, your request will be directly forwarded to required location without any authentication.

If any problem occurs, the message will be logged to log file as per logging configuration done in the deployment descriptor.

Java Access Filter Sample

See the `Readme.txt` file in `<cd-base-directory>/filters/servlet/samples/` for instructions on how to access the Java Access Filter sample.

Java Access Filter Log

The Java Access filter logs can be configured in the `web.xml` file. The Java Access Filter always logs to the web application server logging utility. This configuration is server dependent.

The following two parameters are used in the `web.xml` file for logging:

```
SFDebug
SFLogFile
```

If `SFLogFile` is present but empty no log file is created.

Configuring Filter Authentication

Enabling a JAF to Authenticate Itself to the Select Federation Installation

To use the TLS client authentication to authenticate to the Select Federation installation, perform the following steps:

1. Set SFSkipClientAuth to false.
2. Provide the location and password of the JKS keystore that holds the TLS client authentication certificate in the SFKeyStore and SFStorePass parameters
3. Provide the password to protect the key in SFKeyPass, unless that password is the same password used to protect the keystore.

Scenarios for Setting Up Filters for Server and Client Authentication

The following sections provide scenarios that closely reflect actual deployments.

- Setting Up the Apache Filter for Server and Client Authentication
- Setting Up the Java Filter for Server and Client Authentication

Setting Up the Apache Filter for Server and Client Authentication

Set up the Apache filter for server and client authentication by completing the following tasks:

- Task 1: Enable Server Authentication
- Task 2: Enable Client Authentication

Task 1: Enable Server Authentication

1. Edit the `<apache_home>/conf/filter.conf` file to enable debug logging for the filter for the duration of the setup:

   ```
   Debug=TRUE
   ```

2. Edit the `<apache_home>/conf/filter.conf` file to enable Server Authentication:

   ```
   SkipServerAuth=FALSE
   ```

3. If your SP certificate is in DER format, convert it to PEM format as follows:

   > ➤ If you are using a java-based keystore or you generated your certificate using a java-based utility (for example keytool), your certificate may be in DER format.

   - Optionally, if your certificate is stored in a java keystore, export it out of the keystore as follows:

     ```
     keytool -export -alias <mycert> -keystore </path/to/keystore>
     -storepass <password> -file </path/to/spcert>
     ```

   - Convert the SP certificate from DER to PEM format by entering the following command prompt:

     ```
     openssl x509 -inform DER -in <path/to/spCert> -outform PEM -out <path/
     to/trustedSP.pem>
     ```

4. Provide a path to the newly created `trustedSP.pem` file in the `filter.conf` file:

   ```
   CACerts=/path/to/trustedSP.pem
   ```

How you provide the path depends on one of the following:

- If the `trustedSP.pem` file is on the same system as the Apache web server, just provide the path.
- If the SP and Apache web server are on different machines, first transfer the `trustedSP.pem` file to the system with the Apache web server, and then provide the path.

5 Restart the Apache server and browse to a protected URL to test and check if the server authentication succeeds.

If you fail to see the protected URL, you can look at the `<apache_home>/logs/filter.conf` file for more details.

When you have enabled and configured server authentication successfully, enable client authentication as described in the next section.

Task 2: Enable Client Authentication

➤ Before you begin, be sure that you have a port configured and listening on your Application Server for performing Client Authentication. See step 1 in Configuring TLS Client Authentication on page 165 for more information.

Perform the following steps to enable client authentication:

1 For demonstration purposes, create your own Certificate Authority (CA) as follows:

➤ Most certificates are signed by a CA.

a Generate a Certificate Service Request (CSR) for a Certificate Authority (CA) using the `openssl` utility:

```
openssl req -new -newkey -rsa:<key_length> -nodes -out <path/to/ca.csr>
-keyout <path/to/ca.key>
```

b Since you are your own CA, generate a self-signed CA certificate in PEM format:

```
openssl x509 -trustout -signkey <path/to/ca.key> -days
<number_of_days_that_the_cert_will_be_valid_for> -outform PEM -req -in
<path/to/ca.csr> -out ./ca.pem
```

c Optionally, if your deployment needs to make use of a fingerprint for the CA certificate, generate the CA certificate as follows:

```
openssl x509 -fingerprint -in <path/to/ca.pem> -out ./ca.fingerprint
```

2 Generate a CSR and private key for the filter component:

```
openssl req -new -newkey -rsa:<filter_cert_key_length> -nodes -out <path/
to/filter.csr> -keyout <path/to/filter.key>
```

3 Create the certificate for the filter by signing the filter's CSR with the CA:

```
openssl x509 -req -days <number_of_days_that_the_cert_will_be_valid_for>
-in <path/to/filter.csr> -CA <path/to/ca.pem> -CAkey <path/to/ca.key>
-CAcreateserial -outform PEM -out ./filter.pem
```

4 Import the CA certificate to a java-based keystore.

a The CA certificate must be in DER format to be imported to a java-based keystore. Therefore, for this scenario, convert the filter certificate from PEM to DER format:

```
openssl x509 -inform PEM -in <path/to/ca.pem> -outform DER -out <path/
to/ca.der>
```

b Import the CA certificate to the file used as the truststore for your SP installation.

You can often use the keytool utility to do this:

```
keytool -import -alias <CA_Alias> -file <path/to/ca.der> -keystore
<path/to/cacerts_file> -storepass <password>
```

5 Edit the filter.conf file to provide the necessary paths and enable client authentication.

You do not need to configure the login URL to use a port that performs client authentication. However, be sure to alter the URL used to query the SP for information, to use the port configured for client authentication.

For example, if you used port 7443 (which does not use client authentication) for the login URL, and port 7444 does use client authentication, you would change the port to 7444 for the URL used to query the SP, as follows:

```
LoginPage=https://sp0.hpovsf.tgx.net:7443/tfs-fs/login.jsp
FederationSessionService=https://sp0.hpovsf.tgx.net:7444/tfs-fs/FSS
SkipClientAuth=FALSE
CertFile=/path/to/filter.pem
KeyFile=/path/to/filter.key
```

➤ You do not need to provide a passphrase because the -nodes option was enabled to avoid encrypting the key file. Be sure the passphrase is commented:

```
#PassPhrase=...
```

6 Restart your SP and the Apache server, and browse to a protected URL to test and check if the client authentication succeeds.

If you fail to see the protected URL, you can look at the <*apache_home*>/logs/ filter.conf file for more details.

7 Optionally, if the server and client authentication succeeded, you can turn off debug logging in the filter.conf file:

```
Debug=FALSE
```

Be sure to restart the Apache server for the changes to take effect.

Setting Up the Java Filter for Server and Client Authentication

Set up the Java filter for server and client authentication by completing the following tasks:

- Task 1: Authenticate the Select Federation installation
- Task 2: Use the TLS client authentication endpoint to authenticate to the Select Federation installation

Task 1: Authenticate the Select Federation installation

When the tfs-fs war file is deployed on a https (TLS) endpoint, the Java HTTP layer verifies that the server certificate is trusted. By default this requires that either the server certificate or the CA that issued the server certificate be in the cacerts file.

However, it is possible to use a non-default trust store for the TLS connection to the Select Federation install. Set the following configuration parameters in the web.xml file:

```
SFCACertsStore: keystore with CA certs
SFCACertsPass: password for that store
```

`SFSkipServerAuthn` controls whether the server certificate is checked for hostname. If set to `false` the server certificate subject name (CN) should match the hostname in the URL to the `tfs-fs war` file. This will not be the case for the typical certificate that is generated for a Select Federation install.

Task 2: Use the TLS client authentication endpoint to authenticate to the Select Federation installation

To use the TLS client authentication endpoint to authenticate to the Select Federation installation, perform the following steps:

1. Set `SFSkipClientAuth` to `false`.
2. Provide the location and password of the JKS keystore that holds the TLS client authentication certificate.
3. Provide the password to protect the key, unless that password is the same password used to protect the keystore.

   `SFKeyStore`: Holds the path to a keystore with the client certificate.

   `SFKeyStorePass`: Password for the keystore.

   `SFKeyPass`: Password for the private key that belongs to the client certificate.

   The client certificate in this keystore should be trusted by the server.

7 Configuring Attributes

Introduction

Attributes are often important for a useful federation setup. Installations that act as authorities provide attributes to application partners. Attributes are conveyed using the various federation protocols; these protocols require that attributes have designated names. Authorities that provide attributes need to fetch these attributes from a data source. Select Federation includes built-in support for LDAP directories, and relational databases as attribute sources. In addition, a plugin interface is available that enables development plugins that retrieve or compute attributes from alternative sources (see Directory Plugin Interface on page 122).

This chapter describes how the attributes used in Select Federation are defined in the system configuration `tfsconfig.properties` file.

Configuration of an Attribute

The system configuration `tfsconfig.properties` file has an entry `userAttrs`, a space-separated list of keys for the attributes that are available to the system. For each listed key a corresponding attribute needs to be configured. For example, if the `tfsconfig.properties` file contains the following entry, then the system configuration needs to have entries for the attributes `name_title` and `name_firstname`:

```
userAttrs=name_title name_firstname
```

The entries for `name_firstname` could be as follows:

```
    name_firstname.dstSvc=pp name_firstname.dstSelect=/pp:PP/pp:CommonName/
    pp:AnalyzedName/pp:FN

    name_firstname.samlAttr=name_firstname

    name_firstname.samlAttrNS=http://schemas.trustgenix.com/samlattr

    name_firstname.saml2Attr=name_firstname

    name_firstname.saml2AttrFormat=urn:oasis:names:tc:SAML:2.0:attrname-

    format:basic

    name_firstname.wsfedAttr=name_firstname

    name_firstname.wsfedAttrNS=http://schemas.xmlsoap.org/claims

    name_firstname.ldapAttr=givenName
```

Each entry starts with the “key” and is followed by a dot. For each attribute the recognized subentries are:

- `dstSvc`: the key to a DST (Liberty ID-WSF Data Services Template) service. This means that the attribute will be available through the named DST service. The service name should have a corresponding entry with its namespace, such as `pp.dstNS=urn:liberty:id-sis-pp:2003-08`.
- `dstSelect`: the Select statement for the given attribute as specified by the DST service in question.

The presence of `dstSvcC` and `dstSelect` indicate that the attribute is available through an ID-WSF DST type of service.

- `samlAttr`: the name of the attribute in SAML 1 attribute statements. This name has to be agreed upon with partners.
- `samlAttrNS`: the namespace of name of the SAML attribute. This namespace has to be agreed upon with partners.
- `saml2Attr`: the name of the attribute in SAML 2 attribute statements. This name has to be agreed upon with partners.
- `saml2AttrFormat`: the identifier of the format of the attribute name in SAML 2.0 attribute statements. SAML 2.0 profiles specify a number of these format identifiers. Partners may agree upon other identifiers too.
- `wsfedAttr`: the name of the attribute in ADFS (WS-Federation 1.0) attribute claims This name has to be agreed upon with partners.
- `wsfedAttrNS`: the namespace of the name of the ADFS (WS-Federation 1.0) attribute. This namespace has to be agreed upon with partners and defaults to **http://schemas.xmlsoap.org/claims.**

The presence of the `samlAttr` or `saml2Attr` means that the attribute can be pushed in assertions, and is available for SAML attribute queries; subject to per partner settings for attributes (see Chapter 6, Enabling Applications) and possibly subject to end user policy (see Chapter 9, Configuring Privacy Manager).

The presence of the `wsfedAttr` means that the attribute can be pushed in ADFS (WS-Federation 1.0) assertions; subject to per partner settings for attributes (see Editing Partner Parameter Settings on page 44) and possibly subject to end user policy (see Chapter 9, Configuring Privacy Manager).

➤ The `wsFedAttrName` allows you to give another name to the attribute. However, it is required to map the incoming attributes in ADFS, so that a second mapping on the Select Federation side is rarely necessary. If the `tfsconfig.properties` file includes the `name` entry such as `name_firstname`, then attributes that do not have `wsFed` in the `tfsconfig.properties` file are sent (pushed) to an ADFS SP.

- `dispName`: the name shown to the end user by the Privacy Manager. If absent, the attribute `key` is used. Note that the dictionaries for internationalization may have entries based upon attribute keys, display names or both. So the final text presented to the user may not be the display name that is in the system configuration.
- `ldapAttr`: At an authority site (IDP site), the name of the LDAP attribute that is used to find the value of the configured attribute. This is only used when an LDAP is used to resolve attribute queries. If Select Federation is configured to work with Select Access, then at an application site (SP site), the ldapAttr value is used to populate the destination LDAP directory.

  ➤ For a federation to work successfully, you need to make sure that every attribute that is used to populate the LDAP directory can be created in the destination LDAP directory.

- `allowedValues`: An optional entry with no default that takes a String with a list of values separated by semicolons. Applies to Authority installations. Is used to filter the values of this attribute when sent to an Application partner. Only values that are in the list are sent. For example, if the following entry is in the `tfsconfig.properties` file, then only `sales` and `engineering` will make it to the Application partner:

  ```
  group.allowedValues= sales;engineering
  ```

  This means that it is all right to inform partners that a user belongs to “sales” or “engineering”, but if the same person belongs to “admins” or “documentation”, those values remain hidden.

  ➤ If semicolons are not suitable as separators, you can define an alternative with the entry `valueSeparator`. This takes a one-character String and defaults to a semicolon.

- `filter`: An optional entry that provides an alternative way to filter sent attribute values. Filter takes a single string that should be a (Java-compliant) regular expression. For example, the following string would let group names that end in this string to be sent. There is no default value for this.:

  ```
  group.filter=OU=Sales,O=Acme Inc,C=US\\z
  ```

Configuring Group Claims for ADFS (WS-Federation 1.0)

Outgoing Group Claims

An IDP install can issue Group claims over the ADFS (WS-Federation 1.0) protocol. This requires proper system configuration (in the `tfsconfig.properities` file) and corresponding partner configuration through the Select Federation Administration Console.

First the directory used by the IDP must contain an attribute (for each user) that holds zero or more values. Each value is the name of a group to which the authenticated user belongs. This directory attribute must be added in the same way that other attributes are added, even if it is the same as given in the `ldapGroupMembershipAttr` configuration parameter. For example:

```
groups.ldapAttr=businessCategory

groups.dispName=Groups
```

➤ This `groups` attribute should also be added to the `userAttr` list.

You need to indicate that this local directory attribute needs to be used for ADFS Group claims by setting the correct `wsfedAttr` name:

```
groups.wsfedAttr=Group
```

You can filter outgoing attribute values for any attribute but this may be especially useful for group membership values. For example:

```
groups.allowedValues=sales;engineering
```

Finally, ensure that this `groups` attribute is pushed into each assertion to the ADFS (WS-Federation 1.0) partners through the Select Federation Administration Console. See Configuring the Attribute Policy on page 51 for instructions.

Incoming Group Claims

An SP installation can receive group claims from an ADFS (WS-Federation 1.0) Authority site partner. To support incoming ADFS (WS-Federation 1.0) group claims it is sufficient to ensure that an attribute is defined with the correct wsfedAttr name in the tfsconfig.properties file. For example:

```
groups.wsfedAttr=Group
```

```
groups.dispName=Groups
```

The attribute needs to be in the userAttrs list. Then, in the Administration Console, add the Groups attribute as one of the attributes to obtain on each logon. To do this, select **Manage Partners** → **Authority Attribute Policy**.

Directory Plugin Interface

Select Federation specifies a Java interface called DirPlugin that it calls to verify user credentials or to fetch attribute information about users. Customers can implement this interface to integrate with their directory infrastructure. Out-of-the-box, Select Federation provides the following implementations of this directory plugin interface:

- `DirPlugin_ADS`: For use with Active Directory
- `DirPlugin_LDAP`: For use with other LDAPv3 directory servers
- `DirPlugin_JDBC`: For use with a supported JDBC database
- `DirPlugin_File`: For use with a text file.

The GUI installer has configuration screens for connecting with LDAP directories / Active Directory unless you choose the **I will not be configuring a profile service at this time** option. If you choose this option, you can edit the tfsconfig.properties configuration file to use the JDBC or File directory plugins or a custom directory plugin that you may have developed.

In general, an authority installation obtains attributes from a DirPlugin implementation. HP OpenView Select Federation ships with implementations for relational databases and LDAP directories as well as with an implementation for a simple file-based user database. The DirPlugin implementation is set using the `dirPlugin` configuration entry, which is set during installation. See the `tfsconfig.properties` file and Appendix A, Configuration Parameters for alternative values. Also note that it is possible to develop custom DirPlugin implementations, see the *Web Application Developer's Guide* on the Select Federation SDK CD.

Using Multiple Directory Plugins

It is possible to set up multiple directory plugin implementations, such that a particular plugin is responsible for one or more designated attributes. Multiple plugins can be set using these steps:

1. Add the `directory` entry to one or more attributes. For example:

   `someAttribute.directory=2ndPlugin`.

2 Add at least a class entry for the additional plugin to the system configuration. For example:

2ndPlugin.class=myPlugin.

In addition it is then possible to add a jarFile entry and plugin specific entries. For example:

2ndPlugin.jarFile=/home/ib/newDirPlugin.jar

2ndPlugin.someEntry=foo.

When the system instantiates the plugin it will be provided with the configuration parameters, such as with someEntry=foo.

For each attribute the responsible plugin can be given in the directory subentry for that attribute. For example:

newAttr.directory=2ndPlugin.

When the directory subentry is absent the DirPlugin that is configured with dirPlugin is used. It is possible to configure all attributes with an explicitly responsible directory plugin but it is strongly recommended to have a default plugin configured with dirPlugin=. This as some authentication plugins authenticate against the configured default plugin.

Configuring the Built-In LDAP Directory Plugins

As mentioned in the Directory Plugin Interface on page 122, Select Federation provides two built-in directory plugins to authenticate users against an LDAP directory, and to store user attributes:

- DirPlugin_ADS — Use with Active Directory
- DirPlugin_LDAP — Use with all other LDAPv3 directory servers

The information in the following sections describe different configuration options available when using these plugins.

DirPlugin_ADS

If you selected “Active Directory” during installation, the initial configuration parameters are set in the tfsconfig.properties file as follows:

```
## Required attributes
idpAuthnPlugin=com.trustgenix.tfsIDP.util.IDPAuthnPlugin_ADS
dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_ADS
ldapURL=ldap://<server>:<port>
# userPrincipalName of the administrative user
ldapPrincipal=Administrator@domain.com
ldapPassword=password
ldapUserBaseDN=CN=FederatedUsers, DC=Domain, DC=Com
DirPlugin_ADS.useSAMAccNameOnly=0

## Optional attributes
#ldapUserAttr=
#ldapUserObjectClass=
```

During installation, the rootDSE (defaultNamingContext) of the specified Active Directory server is queried. The query result (DC=domain,DC=com) is stored as the value for the ldapUserBaseDN parameter in the tfsconfig.properties file. All search operations are performed using this value as the base DN.

➤ Setting the ldapUserBaseDN parameter to the exact value of the baseDN that users are authenticated against, improves the performance of your Select Federation installation. For example: ldapUserBaseDN=cn=Users,dc=domain,dc=com

By default, the Active Directory plugin assumes that the entered userId is either a userPrincipalName, or a sAMAccountName. It is possible to change this RDN user attribute to some other value such as. cn or telephoneNumber as follows:

```
ldapUserAttr=cn
```

The default objectClass used for performing searches is person. You can change this to be some other objectClass such as. user as follows:

```
ldapUserObjectClass=user
```

Changing the Bind Mechanism to GSSAPI

Select Federation supports the GSSAPI bind mechanism for Kerberos v4 authentication to Active Directory. The following changes are required in the tfsconfig.properties file:

- ldapPrincipal needs to be set to the sAMAccountName of the administrative user instead of the userPrincipalName.
- Uncomment the other GSSAPI attributes and set values according to your setup as shown in the following example:

```
ldapPrincipal=Administrator
ldapAuthentication=GSSAPI
GSSAPI.defaultRealm=TEST-SERVER.HP.COM
GSSAPI.defaultRealmKDC=test-server.hp.com:88
```

DirPlugin_LDAP

If you selected “LDAPv3” during installation, the initial configuration parameters are set in the tfsconfig.properties file as follows:

```
## Required attributes
idpAuthnPlugin=com.trustgenix.tfsIDP.util.IDPAuthnPlugin_Dir
dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_LDAP
ldapURL=ldap://16.89.65.92:389
ldapPrincipal=cn=Directory Manager
ldapPassword=password
ldapUserBaseDN=dc=cup,dc=hp,dc=com
## Optional attributes
ldapUserAttr=cn
ldapSearchSubtree=1
#ldapUserObjectClass=
```

➤ Setting the ldapUserBaseDN parameter to the exact value of the baseDN that users are authenticated against improves the performance of your Select Federation installation. For example: ldapUserBaseDN=ou=People,dc=cup,dc=hp,dc=com

During installation, there is an advanced configuration screen which allows you to configure the RDN user attribute and enable sub-tree search. It is possible to change this RDN user attribute to some other value such as. cn or telephoneNumber as follows:

```
ldapUserAttr=cn
```

The default objectClass used for performing searches is person. You can change this to be some other objectClass such as. user as follows:

```
ldapUserObjectClass=user
```

JDBC and File Directory Plugins

This section describes how to use the built-in non-LDAP directory plugins.

DirPlugin_JDBC: The JDBC Directory Plugin

The DirPlugin_JDBC allows you to configure Select Federation to leverage a supported JDBC database for authenticating users and for storing user-attributes.

➤ Select Federation can separately use a JDBC database or an LDAP directory as a federation repository. That capability is independent of this Directory Plugin feature.

Configuring Select Federation to use the JDBC Directory Plugin

To enable Select Federation to use the JDBC directory plugin, add a line to the tfsconfig.properties configuration file that reads as follows:

```
dirPlugin=myDirPlugin

myDirPlugin.class=com.trustgenix.tfsIDP.util.DirPlugin_JDBC
```

In the above lines, myDirPlugin may be replaced by an alias of your choice. In this document, myDirPlugin is used as a placeholder for that alias.

Make sure that this is the only uncommented line in the tfsconfig.properties that begins with dirPlugin=. This allows Select Federation to load the JDBC directory plugin for user-authentication and attribute fetching purposes. The next step is to configure the JDBC Directory Plugin to point to the database. You can do this by either directly connecting to the database from the Directory Plugin or by using a JDBC Data Source configured in the application server.

Direct Connection to Database

To connect directly to a database, add the following lines to the tfsconfig.properties file:

```
myDirPlugin.jdbcProvider=
myDirPlugin.jdbcAddr=
myDirPlugin.jdbcUser=
myDirPlugin.jdbcPassword=
```

The values of each of these variables are as follows:

- If you are connecting directly to the database, the value of jdbcProvider can be either JDBCProvider_Oracle or JDBCProvider_MSSQL or JDBCProvider_Derby.

- `jdbcAddr` is set to the name of the server on which the database listener is listening and port number (if not default) separated by a colon, such as `localhost:1592`
- `jdbcUser` is set to the userid to be used to connect to the database. For the JDBC Directory Plugin to function correctly, this user should have read access to the database.
- `jdbcPassword` is set to the password for the `userid`.

Connecting Using a JDBC Data Source

If you would like to connect via a JDBC Data Source configured in the application server, you should add the following lines to the tfsconfig.properties file instead of the ones mentioned in the previous paragraph.

```
myDirPlugin.jdbcProvider=
```

```
myDirPlugin.jdbcDataSource=
```

When using Data Sources, the value of the `jdbcProvider` variable depends upon the database and application server you are using.

- If you are using Microsoft SQL Server or Derby, the value must be one of the following based on the database, regardless of the application server:

  ```
  JDBCProvider_MSSQL
  ```

  ```
  JDBCProvider_Derby
  ```

- However, if you are using Oracle, the value must be one of the following if you are using the built-in application server, WebSphere or WebLogic respectively:

  ```
  JDBCProvider_Oracle_TomcatDS
  ```

  ```
  JDBCProvider_Oracle_WebSphereDS
  ```

  ```
  JDBCProvider_Oracle
  ```

The value of the `jdbcDataSource` variable must be the name of the Data Source configured in the application server. For example if you configured a Data Source called `SFDataSource`, the value of this variable must be `jdbc/SFDataSource`.

Configuring the JDBC Directory Plugin

Configuring the JDBC Directory Plugin involves specifying SQL queries and related parameters in the `tfsconfig.properties` file for various actions that the Directory Plugin is expected to take. Note that these variables are mandatory and the plugin will fail to load if any of these are not specified, even if you do not intend to use that feature of the directory plugin.

The variables to be configured are:

- `myDirPlugin.jdbcUserPassQuery=`
- `myDirPlugin.jdbcUserPassFormat=`
- `myDirPlugin.jdbcUserAttrQuery=`
- `myDirPlugin.jdbcUserGroupQuery=`

jdbcUserPassQuery

You would execute this SQL statement to retrieve a user's password for verification. Assuming you had a user table such as where the password is stored in the clear:

```
CREATE TABLE users (
```

```
                 user VARCHAR(100) NOT NULL,
                 pass VARCHAR(100),
                 name_firstname VARCHAR(100)
                 name_lastname VARCHAR(100)
                 PRIMARY KEY (userId)
         );
```

You would add the following line in the tfsconfig.properties file:

```
myDirPlugin.jdbcUserPassQuery=select password from users where
user=’’{0}’’
```

The placeholder ’’{0}’’ refers to the user id that is passed to the Directory Plugin by Select Federation.

jdbcUserPassFormat

This SQL statement specifies the format of the user's password. The format is either hash or plain, but cannot be null. The hash method indicates an LDAP style SHA or SSHA hash.

Therefore you would add a line to the tfsconfig.properties file of the format:

```
myDirPlugin.jdbcUserPassFormat=hash
```

jdbcUserAttrQuery

This is the SQL statement used to retrieve user attributes. The column names of the attributes must match the value of the attribute's jdbcCol parameter in the tfsconfig.properties file. Make sure that there are no unmatched attributes in the list of userAttrs.

For example, if you had the following:

```
userAttrs=firstname, lastname
firstname.jdbcCol=name_firstname
lastname.jdbcCol=name_lastname
```

Then the value of the jdbcUserAttrQuery can be:

```
myDirPlugin.jdbcUserAttrQuery=select name_firstname, name_lastname from
users where user=’’{0}’’
```

The placeholder ’’{0}’’ is the value of the local user id passed to the directory plugin by Select Federation.

jdbcUserGroupQuery

This is the SQL statement used to verify membership of a user in a particular group. The Select Federation administration console allows you to specify a group name per-partner. The value that you specify there is the value that will be passed to this query. Therfore, if you also had a group table such as the following group table:

```
CREATE TABLE groups (
                 group VARCHAR(100) NOT NULL,
                 user VARCHAR(100) NOT NULL,
         );
```

Then you could add:

```
myDirPlugin.jdbcUserGroupQuery=select group from groups where user=''{0}''
and group=''{1}''
```

The placeholder `''{0}''` is the value of the local user id whose membership is being verified. The placeholder `''{1}''` is the value of the group name as specified in the administration console.

DirPlugin_File: The File-based Directory Plugin

DirPlugin_File is a simple file based directory plugin that uses a flat-file to fetch user attributes, verify group membership and verify user passwords. If you choose not to configure a profile service at the time of installation, a file directory plugin will get configured automatically. To configure the file directory plugin, add the following lines to the tfsconfig.properties:

```
dirPlugin=com.trustgenix.tfsIDP.util.DirPlugin_File
```

The DirPlugin_File takes two configuration parameters, the location of the file containing all the inputs and the character that separates values in that file. The file to be used is specified as:

```
DirPlugin_File.filePath=<path to input file>
```

The value separator is specified as:

```
DirPlugin_File.valueSep=<sepChar>
```

If this is not specified, the default value for the separator character is semi-colon.

A sample input file is given below:

```
# directory entries for user 'john'

john.name_firstname=John

john.name_lastname=Doe

john.personal_email=john@doe.com

# password (for verifyPassword)

john.password=password

# group membership (for isMember)

john.memberOf=foo bar


jane.name_firstname=Jane

jane.name_lastname=Doe

jane.personal_email=jane@doe.com

# password (for verifyPassword)

jane.password=password

# group membership (for isMember)

jane.memberOf=foo baz
```

In this example, the separator character for multiple values is space.

8 Configuring Integrated Windows Authentication

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. IWA consists of the following:

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

How IWA Works

To provide Integrated Windows Authentication it is a given that the users have to be logged on to their windows machines and that their browsers (such as Internet Explorer (IE)) are enabled to use their windows credentials to perform automatic logons.

The following scenario outlines a typical workflow when IWA is in use:

1. Given the federation model, any user attempting to login to a protected resource/ application ends up at an SF-IDP (Select Federation IDP).
2. For IWA, the Authentication Plugin at the SF-IDP redirects the user to the configured loginURL, which happens to be a NTLM-protected page hosted on your IIS server.
3. If the credentials are valid then IIS will let the user through to the `login.asp` page. But if it determines that the credentials are not valid then the user gets prompted for authentication.
4. The ASP login page captures the user's default windows credentials from the user’s browser session with IIS.
5. Once the ASP page has the user credentials, it invokes FSS to add the user and create a session for the user with the SF-IDP.
6. The FSS checks that the ASP presents a valid client certificate as the actual web service call (ASP page invoking FSS) must be placed over a secured connection (mutually authenticated SSL connection).

   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).

7 If the connection is secure and the certificates valid, then the FSS servlet reads the request and logs the user in to the SF-IDP and responds back to the ASP page.

8 The ASP page redirects the user back to the resource/application that the user had initially attempted to access.

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: Enable Server and Client Authentication

Task 2: Set up the 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: Enable Server and Client Authentication

A small part of setting up the Server and Client Authentication for the SSL/TLS channel is dependent on the application server you are using for your SF-IDP install. The following instructions are for the built-in application server, which are similar for the WebSphere and WebLogic application servers. However, for complete configuration instructions for WebSphere or WebLogic, refer to their respective application server guides.

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>"
```

Task 2: Set up the FSS component

The FSS component should already be deployed on any application server where Select Federation 6.60 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.

   ```
   # This defaults to the providerBaseURL + securePortNumber "/tfs-fs/FSS"
   fssURL=https://<providerBaseURL>:<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:

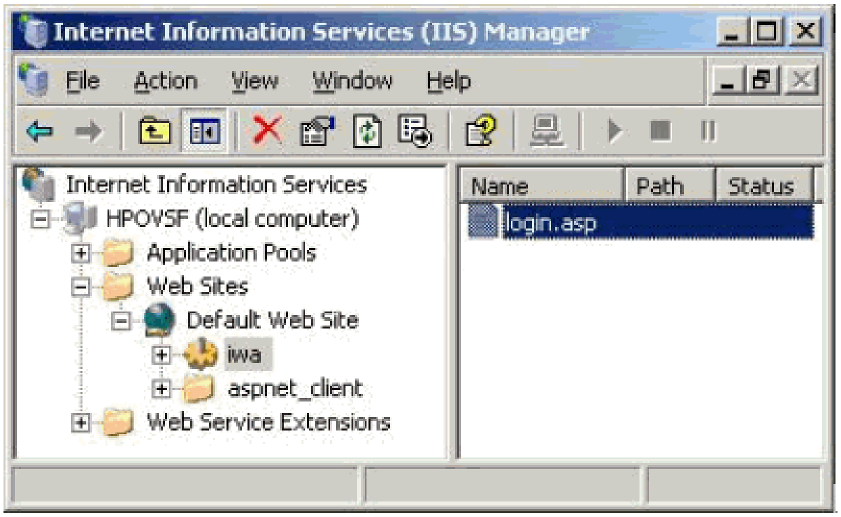
1. Copy the `login.asp` file (and other ASP files) from `<`*`cd-base-directory`*`>/adapters/iwa/` to the machine where your IIS server is located.

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



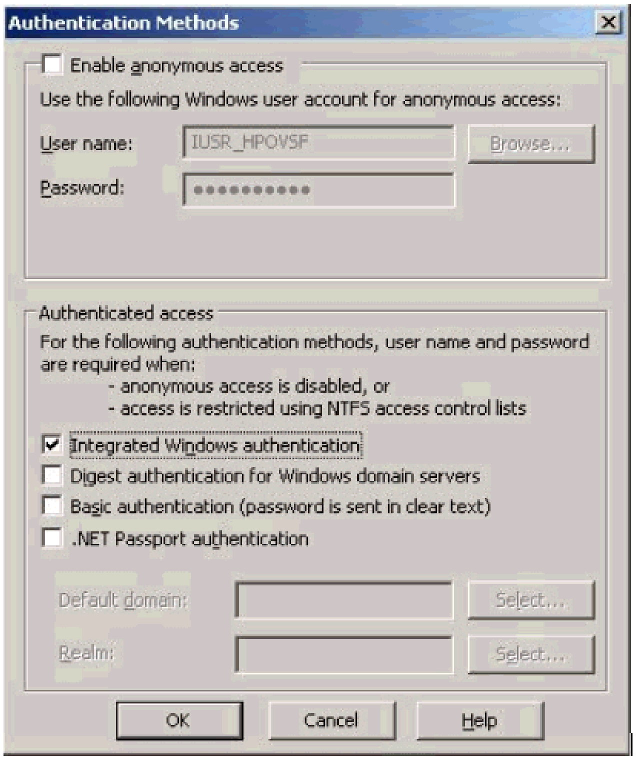
4 Be sure that the **Status** of the **Active Server Pages** (under **Web Service Extensions**) is set to **Allowed**.

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 16 Authentication Methods Dialog


Authentication Methods

☐ Enable anonymous access

Use the following Windows user account for anonymous access:

User name: IUSR_HPOVSF

Password: ••••••••••

Authenticated access

For the following authentication methods, user name and password are required when:
- anonymous access is disabled, or
- access is restricted using NTFS access control lists

☑ Integrated Windows authentication

☐ Digest authentication for Windows domain servers

☐ Basic authentication (password is sent in clear text)

☐ .NET Passport authentication

Default domain:

Realm:

7. Select **Integrated Windows authentication** so that the location where you placed `login.asp` is NTLM-authentication enabled.

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 IE, 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.

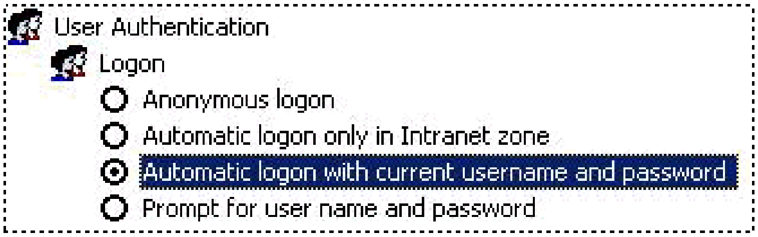
The following steps provide an example of how to enable the user’s browser manually for IE:

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 17 User Authentication Setting in the Security Settings Window



User Authentication
Logon
○ Anonymous logon
○ Automatic logon only in Intranet zone
● Automatic logon with current username and password
○ Prompt for user name and password

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

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 the 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 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 in one of the following ways:
  - — You may use any tool or utility of your preference to 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.

- — You may use any tool or utility of your preference to create a self-signed certificate that matches the above requirements and your security constraints.
- — 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 OpenView Select Federation Certificate Management User's Guide* for more information.)

- 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 137.

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.
- The input box for the **Your Name** field is the same as providing a value for the `cn`.

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.

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 textbox 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 textfield 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 get the certificate for that CA 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.

   g Copy the exported certificate file to your SF-IDP 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>/_jvm/lib/security/cacerts` 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, a password and the location for the client certificate that you copied from the IIS machine to your SF-IDP machine.

   f Click the **Import** button.

   The client 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 you IWA client certificates then:

   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 Or if you prefer to simply use the default keystore used by your SF-IDP then:

   a Import the client certificate to the default keystore. The keystore is located at `<sf-home>/conf/sslkeystore.jks`.

   b You can find the location of the default keystore by taking a look at your tfsconfig.properties and checking what the values for the following properties are:

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

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

9 Configuring Privacy Manager

Overview

Select Federation Privacy Manager is a unique feature that empowers end users to control the exchange of their personal attributes and their preferences about exchanging such information between trusted sites. The Privacy Manager is used by Select Federation to interact with end users to ask for permission to link accounts (federate), to ask for permission to release attributes to partners, and so on.

In Select Federation, the Privacy Manager allows for end users to have the system remember privacy decisions as privacy policy rules. In this case, the Privacy Manager also offers a facility to end users to review such privacy policy rules. In all cases, the Privacy Manager uses a presentation engine to render pages to users. This presentation engine allows for localization on a per user basis (see Localizing Select Federation on page 181). The presentation engine also can adapt to various browsers that end users may have, and enables branding of the end user Privacy Manager pages (see Branding the End User Pages on page 174).

Privacy Manager Configuration

Privacy Manager is disabled by default. To enable and configure Privacy Manager, change the settings in the `tfsconfig.properties` file, as described in the following sections:

- Enabling the Use of Privacy Manager for ID-WSF Services
- Configuring the Use of End User Privacy Policies

Enabling the Use of Privacy Manager for ID-WSF Services

If ID-WSF services such as the Personal Profile are used, you should also enable the ID-WSF interaction redirect service, one of the components of the Privacy Manager. This is performed by uncommenting the following line in the `tfsconfig.properties` file, where *<BASE_URL>* is the URL of your installation:

```
## Location of interaction redirect service used by
## ID-WSF WSPs deployed here
userInteractionURL=https://<BASE_URL>/pm/irs
```

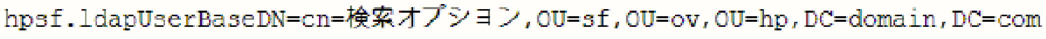
In the Privacy Manager web pages, Personal Profile can be localized from the `tfsconfig.properties` file, by changing the `profile.name` setting as follows:

Change: `profile.name=Personal Profile`

To: `profile.name=<`*`specific_language_using_encoding_UTF-8`*`>`

To avoid mistakes, any characters that are not covered under the ISO-8859-1 encoding by default, should be entered in their Unicode Escaped representation.

For example the following entry that includes unsupported characters:



```
hpsf.ldapUserBaseDN=cn=検索オプション,OU=sf,OU=ov,OU=hp,DC=domain,DC=com
```

must be entered in the `tfsconfig.properties` file as follows:

```
hpsf.ldapUserBaseDN=cn=\u691c\u7d22\u30aa\u30d7\u30b7\u30e7\u30f3,OU=sf,O
U=ov,OU=hp,DC=domain,DC=com
```

Configuring the Use of End User Privacy Policies

By default the use of end user privacy policies is disabled. Enabling such policies will allow end users to ask the system to remember privacy decisions, and to review the thus constructed policies. To enable privacy policy rules changes the following line in `tfsconfig.properties`:

```
# Uncomment/edit following line to allow use of user
# specific policies for the listed services
userPolicy.services=profile
```

This line accepts a list of space separated service names; `profile` is the built-in service name for the set of SAML and ID-WSF attribute services (see Configuring the Attribute Policy on page 51).

For the profile service it is now possible to set some parameters related to pages that users may encounter. The name of the service as displayed to user (before localization) is set by:

`profile.name=Personal Profile`.

The possible outcomes of user privacy policy rules are governed by this line:

`profile.possibleDecisions=DENY PROMPT GRANT`.

For example to allow users only to create rules that deny access to certain profile attributes and, hence ensure positive explicit consent before attribute release, the line could read:

`profile.possibleDecisions=DENY`.

`DENY`, `PROMPT` and `GRANT` are all keywords that should be used as is. The actual text shown to end users is subject to localization and branding.

End users can review their policies by visiting this URL:

`https:{BASE_URL}/pm/privacy`.

Branding the End User Privacy Manager Pages

Privacy Manager uses the Select Federation presentation engine to render pages to end users — both the pages that ask for consent as well as the pages that enable viewing and modification of the user privacy rules. You can brand these pages (changing title, logos, colors, and to some degree layout) in the following ways:

- Branding by configuration settings — changing configuration entries in the `tfsconfig.properties` file
- Branding by CSS stylesheets — changing or replacing the CSS stylesheets
- Branding by XSLT stylesheets — changing or replacing the XSLT stylesheets

See Branding the End User Pages on page 174 for detailed instructions on how to brand the end user Privacy Manager pages.

10 Authentication Contexts

The Liberty ID-FF and the SAML 2.0 standard suites have the notion of an “Authentication Context”. An authentication context is a description of a particular method for authenticating users, such as verifying a password. An application may require that a user be authenticated in a manner that is consistent with a particular authentication context and likewise an authority informs an application about the actual authentication context that was used. Authentication contexts need to be agreed upon between partners. This chapter describes how to configure authentication contexts in HP OpenView Select Federation.

➤ In many cases no changes are required. Making changes is not recommended unless the administrator has a thorough understanding of both the standards as well as the policy that the partnership wants to deploy.

Statements and Classes

Authentication context descriptions are typically fairly large (XML) documents that contain details about password length, verification of photo IDs, and so on. It would be impractical to always move these large documents around over the network. Also, there are a number of widely-used authentication methods that are well understood even if details are slightly different between different implementations. Hence, there are Authentication Context Classes, which represent such well-known methods such as “password over a protected transport”. Authentication Context Statements however, are actual description documents that define the details of the authentication context, such as the number of bits in the password. As there rarely is a need to move actual authentication context documents around, references are used.

Relative Order of Authentication Context

Without mutual agreement between partners a particular authentication context classes cannot be interpreted as referring to “stronger” authentication as another. When partners agree about relative ordering of authentication contexts, it becomes possible for an application partner to request an authority for authentication of a user at “at least” a given authentication context. The ID-FF and SAML 2.0 protocols allow for this notion, but as explained it requires agreement between partners.

Configuring Authentication Context Statements and References

The system configuration contains information about the authentication contexts. The authentication context classes and statements that are supported by the currently configured authentication plugin (or page) can be configured. These are set by the parameters:

- `authnContextClassRef`: the space separated references to the supported authentication context classes; used for processing Liberty ID-FF authentication requests.
- `authnContextStatementRef`: the space separated references to the corresponding authentication context statements that are supported.
- `saml2AuthnContextClassRef`: the space separated references to the supported authentication context classes; used for processing SAML 2.0 authentication requests.
- `saml2AuthnContextStatementRef`: the space separated references to the corresponding authentication context statements that are supported.
- `samlAuthMethod`: the space separated identifiers of the supported authentication methods according to the SAML 1 specifications.

The relative ordering of authentication contexts is set by the parameters:

- `rankedAuthnContextClassRefs`: space separated list of references to authentication context classes; from weak to strong, weakest first.
- `rankedAuthnContextStatementRefs`: space separated list of references to authentication context statements; from weak to strong, weakest first.

➤ These are the Liberty ID-FF classes / statements. Normally the set of supported authentication contexts is a subset of the ranked contexts.

11 Configuring Liberty Introduction Service

A common problem in a federated system is that when a user is about to login to a service provider (SP) site, the SP site needs to be able to help the user identify which site the user can login from. This is the IDP Selection problem.

The Liberty Alliance Standard includes a specialized mechanism for addressing this issue called the Introduction Protocol. This is done by all the IDPs and the SP setting up a common DNS Domain that hosts a server from each one of the sites. Thus if an “Extranet” acting as an SP site and several “Portals” acting as IDPs form a circle-of-trust to use the introductions protocol, they need to define a common DNS domain, such as `commondomain.com`.

Each site gets a sub DNS address such as `extranet.commondomain.com` and `portal1.commondomain.com`, for each participant in the circle of trust. An IDP is expected to write a cookie in this common domain by redirecting its users to its server in the common domain. The cookie domain is set to the entire common domain `commondomain.com` so that it may be read by any web server in this domain.

When the user then navigates to the SP, the SP redirects the user to the common domain and finds out if any IDP has stored a cookie in the common domain. If it has, then the SP redirects the user to that IDP in order to sign-on the user.

Configuration and Usage

HP OpenView Select Federation provides certified interoperable support for the introduction protocol. The WAR file to be hosted on the introductions server is `tfs-intro.war`.

In order to use the Introduction service, you need to setup DNS appropriately and then run the `tfs-intro.war` on the server in the common DNS domain. This WAR requires the configuration file `tfsconfig.properties`. In addition, the Select Federation operational WAR, `tfs.war`. needs to know where the introduction service is located.

At an IDP Site, the `tfs.war` requires the following variables:

```
cookieReaderServiceURL=https://sp.commondomain.com/tfs-intro/
CookieReaderService
```

The value of the `cookieWriterServiceURL` variable is the URL of the common domain cookie writer. The `useSSOCookieWriter` variable turns the introduction functionality on or off. The value of this variable should be “1” in order to use the introduction service.

➤ Currently, HP OpenView Select Federation only supports the use of the Cookie Writer Service when using the artifact profile.

At an SP Site, the `tfs.war` file requires the following variables:

```
cookieWriterServiceURL=https://idp.commondomain.com/tfs-intro/
CookieWriterService

useSSOCookieWriter=1
```

12 Configuring the ADFS (WS-Federation 1.0) Protocol

Select Federation 6.60 supports a new protocol specification, WS-Federation 1.0 to be interoperable with Active Directory Federation Services (ADFS). Using this protocol requires that you add the following lines to the `tfsconfig.properties` configuration file of Select Federation:

- `wsfed10.logoutConfirmURL`: Points to a page that will provide content to iframe elements in ADFS Single Log Out pages. If this parameter is empty or commented out Select Federation will create content through the `slo-response.xml` stylesheet. Not required, when absent, defaults to null.
- `wsfed10.idpDomain`: A string representing the domain suffix that a SF Authority represents from the point of view of ADFS (WS-Federation 1.0) partners. Defaults to the host part of the providerId variable. Required on an IDP installation.
- `wsfed10.defaultLogoutReturnURL`: Points to a page where users end up after an ADFS SP initiates Single Log Out and does not provide a return URL. Is used to override the more generic `defaultLogoutReturnURL` that will be set during installation. Not required; defaults to the value of `defaultLogoutReturnURL`.
- `wsfed10.localUserIdFormat`: A string that indicates to an ADFS (WS-Federation 1.0) Application partner how to map a local name identifier sent by the SF Authority to an Identity Claim. Defaults to http://schemas.xmlsoap.org/claims/UPN.

  For example to make the partner treat such a local identifier as a `CommonName` claim set:

  ```
  wsfed10.localUserIdFormat=http://schemas.xmlsoap.org/claims/CommonName
  ```
- `wsfed10.localUserIdPrefix`: An optional string that defaults to an empty string, used to add a constant string before a local name when the SF Authority is configured to send local names to a WS-Federation 1.0 partner.
- `wsfed10.localUserIdSuffix`: An optional string that defaults to: @[wsfed10.idpDomain]. It is used to add a constant string after a local name when the Select Federation Authority is configured to send local names to a WS-Federation 1.0 partner. For example, if the users' email addresses are used as local authority user ids, and if local names are to be sent to an ADFS (WS-Federation 1.0) partner, you would set the `localUserIdFormat` to match the ADFS Email claim and deliberately set the suffix to an empty string, as email addresses already contain a “suffix”. For example:

  ```
  wsfed10.localUserIdFormat=http://schemas.xmlsoap.org/claims/EMail

  wsfed10.localUserIdSuffix=
  ```

13 Configuring LDAP Directories or an Active Directory for Federation and Partner Data

Select Federation can use an LDAP directory or an Active Directory to authenticate users and to create and populate newly activated user entries at an SP. In addition, Select Federation can store some of the data it normally stores in an RDBMS (the federation repository), in an LDAP directory or an Active Directory.

This chapter describes how to use an LDAP directory or an Active Directory to store such data, including federation mappings (federation data) or partner configuration information.

➤ If you use an LDAP directory or Active Directory to store federation data for Select Federation, then Select Federation must be installed as either an SP-only site or an IDP-only site, but not both simultaneously.

Overview

➤ The way that Select Federation uses an LDAP directory or an Active Directory to store federation and trust data is very similar. Therefore, throughout this chapter, the term “LDAP” applies to both an LDAP directory and an Active Directory, except where specified.

Select Federation can store its user federation data as well as partner data in an LDAP directory. This may be beneficial if existing CRM and similar applications exist, which use the same LDAP directory. Setting up LDAP as a federation data store requires some effort, so unless there are good reasons to choose this option the use of a relational database is recommended.

➤ Select Federation needs to use a relational database (RDBMS) to store federation session data. The Select Federation installer can set up a built-in database if no RDBMS is available.

A further consideration is that partner data can be stored in the LDAP directory only if user federation data is also stored in the LDAP directory.

The setup for LDAP consists of two steps:

1. Prepare the LDAP directory by extending the schema and setting up branches.
2. Modify the Select Federation configuration to point to the LDAP directory for federation and partner storage.

LDAP Repository Usage

Select Federation can be configured to store user-federation data in the LDAP directory (that is, any associations of pseudonyms with local user IDs) and can also store partner data in the LDAP directory. Partner data is always stored in a separate partner branch or sub-tree. User federation data is stored as additional attribute values for existing user entries in the case of an IDP and as new user entries in a separate federation branch or sub-tree in the case of an SP.

Sample Schema Files

Three sample schema files have been included in the `docs` subdirectory of the Select Federation CD: `sf.schema, sf.schema.ad` and `hpsf.ldif`. These files may be used as sample references for the instructions below. These files are not intended to be applied directly to your LDAP directory.

Preparing the LDAP Directory or Active Directory Installation for Federation and Partner Data

Before Select Federation can store federation and partner data in an LDAP directory or Active Directory, the directory must be set up with the right placeholders, similar to creating the right table in an RDBMS database.

The following tables list and describe the recommended names of attributes and object classes for both the LDAP directory and Active Directory. However, you may choose a different name for a certain attribute or object class, as long as the same name is specified in the `tfsconfig.properties` file. See Changing Select Federation Configuration for LDAP Federation and Partner Data on page 155 for instructions on changing the attribute or object class names from their standard names.

➤ For Active Directory, the attributes must use “Unicode String” instead of “text” and “Generalized Time” instead of “general time.”

Table 1 Recommended Names of Attributes

Recommended name	Type	Single valued	Comments
created	general time	X	
modified	general time	X	
providerid	text	X	Also used to identify affiliations
uid	text	X	Local user ID. This attribute name might already exist in the standard schema. You may use a different name to avoid conflict with a differing definition of the same attribute in the existing schema.
spid	text	X	Used in case provider identifies an affiliation
spuid	text	X	
feduid	text	X	

Recommended name	Type	Single valued	Comments
entryid	text	X	This attribute name might already exist in the standard schema. You may use a different name to avoid conflict with a differing definition of the same attribute in the existing schema.
resourceid	text	X	
servicetype	text	X	
options	text		
abstract	text	X	This attribute name might already exist in the standard schema. You may use a different name to avoid conflict with a differing definition of the same attribute in the existing schema.
flags	text	X	
secmechs	text		
wsdlref	text	X	
servicenameref	text	X	
endpoint	text	X	
soapaction	Text	X	
displayName	text	X	This attribute might already exist and likely is set up as you need it.
description	text	X	This attribute name might already exist in the standard schema. You may use a different name to avoid conflict with a differing definition of the same attribute in the existing schema.
homepage	text	X	
logourl	text	X	
logotext	text	X	
succinctid	text	X	
protocol	text	X	
roles	text		
properties	text	X	Should allow for storage of an encoded object, such as a “blob.”
metadata	text	X	Should allow for storage of a very large chunk of text, such as a “blob.”

Table 2 Recommended Names of Object Classes.

ObjectClass recommended name	Attributes
federation	cn created feduid modified providerid spid spuid uid
service	abstract entryid flags options providerid resourceid servicetype
servicedescription	endpoint entryid flags secmechs servicenameref soapaction wsdlref
idp (SP only)	providerid
provider	created description displayname homepage logotext logourl metadata modified properties protocol providerid roles succinctid

Preparing the LDAP Directory

Perform the following steps to prepare the LDAP directory:

1. Be sure that distinct attributes with the properties in Table 1 are in the schema of the LDAP directory. The text attributes should allow for 1024 characters.
2. Add the object classes to the schema of the LDAP directory (see Table 2 for the object classes).

   Attributes are not mandatory, which means they can be empty.
3. If you are using Select Federation as an IDP, be sure that the directory has a branch for federations (`ou=federations,dc=example,dc=com`).

   This branch must have either your real user records within it or records in which the RDN attribute is specified as the `ldapUserAttr` in the `tfsconfig.properties` file. This enables the federation data to be stored as attributes of the corresponding user entry.

   Also be sure that the LDAP directory has a branch for partners: ou=partners, dc=example, dc=com.

   ➤ At an IDP, the attribute used for describing the user ID is always the RDN of the user entry (such as `uid, cn`). The value of `ldap_fed.uidName` in the `tfsconfig.properties` file must be the same as the value of `ldapUserAttr` (the RDN attribute) in the `tfsconfig.properties` file.

Preparing the Active Directory

For the Active Directory, use the sample schema file, `sf.schema.ad`, provided in the `docs` subdirectory.

Perform the following steps to prepare the Active Directory using the `sf.schema.ad` file:

1. Edit the `sf.schema.ad` file, keeping in mind the following requirements:
   - You may change the attribute names (shown in Table 1) and object class names (shown in Table 2) in the sample AD schema file based on your deployment. However, you must keep the attribute syntax the same. Be sure your changes are reflected where necessary.
   - If you are using your Select Federation install as an SP, you must use `hpovsffederationspoc` as the example federation objectclass. The `rDNAttID` for this objectclass must be `hpovsfspfeduid` (or the attribute name you have for `hpovsfspfeduid`).
   - If you are using your Select Federation install as an IDP, you must use `hpovsffederationidpoc` as the example federation objectclass. The `rDNAttID` for this objectclass must be `hpovsfidpspid` (or the attribute name you have for `hpovsfidpspid`).
   - Change the `mayContain` and `rDNAttID` entries as appropriate.
   - You may change the `governsID` to use your deployment-specific OIDS.
   - Change the `possSuperiors` of `hpprovideroc` based on your deployment to reflect the appropriate container.
2. Load the edited schema sample file into your deployment Active Directory, keeping in mind the following requirement:

Be sure to load the schema for the Active Directory from an external tool such as LDIFDE and not from the Active Directory Schema snap-in. The LDIFDE tool is a utility program that can import and export the Active Directory schema objects using LDIF-formatted files.

Following is a sample command line using LDIFDE to import the schema file:

```
ldifde -i -k -f c:\sf.schema.ad
```

3 If you are using Select Federation as an IDP, be sure that the directory has a branch for federations (`ou=federations,dc=example,dc=com`).

This branch must have either your real user records within it or records in which the RDN attribute is specified as the `ldapUserAttr` in the `tfsconfig.properties` file. This enables the federation data to be stored as attributes of the corresponding user entry.

Also be sure that the LDAP directory has a branch for partners: ou=partners, dc=example, dc=com.

➤ At an IDP, the attribute used for describing the user ID is always the RDN of the user entry (such as uid, cn). The value of `ldap_fed.uidName` in the `tfsconfig.properties` file must be the same as the value of `ldapUserAttr` (the RDN attribute) in the `tfsconfig.properties` file.

Changing Select Federation Configuration for LDAP Federation and Partner Data

The Select Federation configuration should be changed to point to the LDAP repository for federation and partner data.

Perform the following steps to change the Select Federation configuration to point to the LDAP repository:

1 Check that the following entries in the `tfsconfig.properties` file are correct:

- `ldapURL`
- `ldapPrincipal`
- `ldapPassword`

2 Add the following lines to the `tfsconfig.properties` file, based on either an IDP or SP installation:

a For an IDP install:

```
idpFedDataProvider=com.trustgenix.tfsIDP.util.IDPFedDataProvider_LDAP

ldap_fed.baseDN=the DN of the federations branch, e.g. ou=federations,
dc=example, dc=com

cotDataProvider=com.trustgenix.tfs.util.COTDataProvider_LDAP

ldap_partner.baseDN=the DN of the partner branch, e.g. ou=partners,
dc=example, dc=com
```

➤ If at an IDP you are not creating a separate branch for federations, the `ldap_fed.baseDN` needs to point to the branch where your actual user entries are:

b For an SP install:

```
spFedDataProvider=com.trustgenix.tfsSP.util.SPFedDataProvider_LDAP
```

```
ldap_fed.baseDN=the DN of the federations branch, e.g. ou=federations,
dc=example, dc=com
```

```
cotDataProvider=com.trustgenix.tfs.util.COTDataProvider_LDAP
```

```
ldap_partner.baseDN=the DN of the partner branch, e.g. ou=partners,
dc=example, dc=com
```

➤ For an SP install, `ldap_partner.baseDN` can be equal to the `ldap_fed.baseDN`.

3 If you are using LDAP and the attribute or object class names are not the recommended names listed in Preparing the LDAP Directory or Active Directory Installation for Federation and Partner Data on page 152, do the following:

a Copy the relevant line from the following list into the `tfsconfig.properties` file.

b Uncomment the line and change the value to the name used in the LDAP schema.

```
#ldap_common.createdName=created
#ldap_common.modifiedName=modified
#ldap_common.providerIdName=providerid
#ldap_fed.objectClass=federation
#ldap_fed.uidName=uid
#ldap_fed.spidName=spid
#ldap_fed.feduidName=feduid
#ldap_fed.createdName=created
#ldap_fed.spuidName=spuid
#ldap_ro.objectClass=service
#ldap_ro.entryIdName=entryid
#ldap_ro.resourceIdName=resourceid
#ldap_ro.serviceTypeName=servicetype
#ldap_ro.optionsName=options
#ldap_ro.abstractName=abstract
#ldap_ro.flagsName=flags
#ldap_ro_desc.wsdlRefName=wsdlref
#ldap_ro_desc.serviceNameRefName=servicenameref
#ldap_ro_desc.endpointName=endpoint
#ldap_ro_desc.soapActionName=soapaction
#ldap_ro_desc.secMechsName=secmecs
#ldap_ro_decs.soapActionName=soapaction
#ldap_partner.objectClass=provider
#ldap_partner.providerIdName=providerid
#ldap_partner.providerNameName=displayname
#ldap_partner.providerDescName=description
#ldap_partner.providerURLName=homepage
#ldap_partner.logoURLName=logourl
#ldap_partner.logoTextName=logotext
#ldap_partner.succinctIdName=succinctid
#ldap_partner.protocolName=protocol
#ldap_partner.rolesName=roles
#ldap_partner.propertiesName=properties
#ldap_partner.metadataName=metadata
```

4 If you are using Active Directory as your federation and partner repository, do the following:

a Copy the following list into the tfsconfig.properties file.

➤ If you have modified the schema from the provided sample schema for Active Directory, change the value to the name used in the LDAP schema.

```
ldap_common.createdName=hpovsfcreated
ldap_common.modifiedName=hpovsfmodified
ldap_common.providerIdName=hpovsfproviderid
ldap_fed.uidName=hpovsfuid
ldap_fed.createdName=hpovsfcreated
ldap_fed.spuidName=hpovsfspuid
ldap_ro.objectClass=hpovsfserviceoc
ldap_ro.entryIdName=hpovsfentryid
ldap_ro.resourceIdName=hpovsfresourceid
ldap_ro.serviceTypeName=hpovsfservicetype
ldap_ro.optionsName=hpovsfoptions
ldap_ro.abstractName=hpovsfabstract
ldap_ro.flagsName=hpovsfflags
ldap_ro_desc.objectClass=hpovsfservicedescriptionoc
ldap_ro_desc.secMechsName=hpovsfsecmechs
ldap_ro_desc.wsdlRefName=hpovsfwsdlref
ldap_ro_desc.serviceNameRefName=hpovsfservicenameref
ldap_ro_desc.endpointName=hpovsfendpoint
ldap_ro_desc.soapActionName=hpovsfsoapaction
ldap_partner.objectClass=hpovsfprovideroc
ldap_partner.providerIdName=hpovsfproviderid
ldap_partner.providerNameName=displayName
ldap_partner.providerDescName=description
ldap_partner.providerURLName=hpovsfhomepage
ldap_partner.logoURLName=hpovsflogourl
ldap_partner.logoTextName=hpovsflogotext
ldap_partner.succinctIdName=hpovsfsuccinctid
ldap_partner.protocolName=hpovsfprotocol
ldap_partner.rolesName=hpovsfroles
ldap_partner.propertiesName=hpovsfproperties
ldap_partner.metadataName=hpovsfmetadata
```

b If you are using your Select Federation as an IDP, then copy the following lines to the tfsconfig.properties file.

➤ You must modify the values appropriately if you have not used the provided sample Active Directory schema.

```
# When instance is used as an IDP
ldap_fed.idp.objectClass=hpovsffederationidpoc
ldap_fed.idp.spidName=hpovsfidpspid
ldap_fed.idp.feduidName=hpovsfidpfeduid
```

c If you are using your Select Federation as an SP, then copy the following lines to the tfsconfig.properties file:

➤ You must modify the values appropriately if you have not used the provided sample Active Directory schema.

```
# When instance is used as an SP
ldap_fed.sp.objectClass=hpovsffederationspoc
ldap_fed.sp.spidName=hpovsfspspid
ldap_fed.sp.feduidName=hpovsfspfeduid
```

API Configuration

If you are storing Partner data in LDAP, to ensure that the API works properly, you must configure the `tfsconfig.properties` file and the `idpapiconfig.properties` file (for an IDP site) and the `spapiconfig.properties` file (for an SP site) as follows:

- For an IDP site, edit the `tfsconfig.properties` and the `idpapiconfig.properties` files to include the following entries and any entries for non-default attribute or object class names:

  ```
  idpApiPartnerData=LDAP

  idpApiFedData=LDAP

  ldapURL

  ldapPrincipal

  ldapPassword

  ldap_partner.baseDN

  ldap_fed.baseDN
  ```

- For an SP site, edit the `tfsconfig.properties` and the `spapiconfig.properties` files to include the following entries and any entries for non-default attribute or object class names:

  ```
  spApiPartnerData=LDAP

  spApiFedData=LDAP

  ldapURL

  ldapPrincipal

  ldapPassword

  ldap_partner.baseDN

  ldap_fed.baseDN
  ```

For a list of all the configuration parameters in each configuration file, see Appendix A, Configuration Parameters.

14 Certificate Management

Overview

Select Federation uses digital certificates primarily for ensuring the security of communication with partners. Digital certificates are cryptographically sound untamperable identities issued to an entity such as a person or organization. In the case of Select Federation, the certificates in question are organizational certificates issued to the federation end-point that communicates with its trusted partner federation end-points. It is possible to use individual certificates with Select Federation as a mode of authenticating individual users, however, such usage is not covered in this chapter. This chapter only describes the use of organizational certificates

Digital Certificates may be self-signed or issued by third-parties. Self-signed certificates are inherently untrustworthy and may be used in real production environment with independent, manual verification of their authenticity with the identity asserted in that certificate. Third-party certificates may be issued by “Trusted Third Party CA” organizations which may be already trusted by the underlying application server or Java Virtual Machine. Typically, in such cases extra configuration is minimal or not needed at all. However, if you are using a self-signed certificate or a third-party certificate that has been issued by a CA that is not recognized by the application server or the Java Virtual Machine, you will need to perform additional configuration to ensure that the certificates are appropriately trusted.

Select Federation uses certificates for security of communication with partners. Security has several aspects to it, including authenticity, integrity and privacy. In the context of a federation, authenticity means the confidence of knowing that information being claimed as coming from one party is indeed from that party and not from anyone else. Integrity ensures that any information sent by one party to another has not been tampered with en-route and privacy is the guarantee that no other party other than the intended recipients of a communication could have knowledge of the contents of that communication. When used for digital signatures, certificates ensure the authenticity and integrity of messages. Server certificates for SSL / TLS ensure the authenticity of the server and the integrity and privacy of communication between the client and the server. Client certificates for SSL / TLS ensure the authenticity of the client to the server and are used in setting up the private channel between the client and the server.

Various federated identity protocols provide various options for using certificates. Some, like Liberty ID-FF require digital signatures whereas others like SAML provide options such as digital signatures or SSL / TLS client-authentication. Select Federation supports almost all security profiles of all protocols and therefore has broad interoperability. However, this creates some amount of work in configuring some of the advanced options.

This chapter provides information on how to configure the various keystores and certificate stores used by Select Federation, the application server and the Java Virtual Machine.

Using a Signing Certificate Issued by a Third-Party CA

The Select Federation installer generates a self-signed certificate for use in signing and Transport Layer Security (TLS) client authentication (see Configuring TLS Client Authentication on page 165).

If you have a certificate issued by a certificate authority (CA), you may use the issued certificate instead. You can manage third-party CAs using one of the following options:

- Keytool — see Using the Java Keytool to Manage Third-Party CAs for instructions.
- Certificate Management Tool (CMT) — provided with Select Federation in `$SF-HOME/tools`. See the *HP OpenView Certificate Management User's Guide* for instructions.

Using the Java Keytool to Manage Third-Party CAs

This section provides instructions for installing a third-party certificate.

Prerequisites

- Java keystore with your private key and the matching certificate issued by a CA
- File with CA certificate in PEM format:

  ```
  -----BEGIN CERTIFICATE-----
  ```

  and

  ```
  -----END CERTIFICATE-----
  ```

Installing a Third-Party Certificate

Perform the following steps to install a third-party certificate:

1. Edit the `tfsconfig.properties` file generated by the Select Federation installer and change the following lines:

   ```
   keystoreType=JKS
   keystorePath=<path to your keystore.jks>
   keystorePassword=<password for your keystore>
   certAlias=<alias for your certificate in the keystore>
   keyAlias=<alias for your key in the keystore>
   keyPassword=<password for your key in the keystore>
   ```

2. If you have a certificate issued by a private CA that is not included in the default Java trust list, you need to install the CA's certificate in your Java `cacerts` file using the Java keytool. Following are the default paths:

   - **WebLogic:** `BEA_HOME/jdk142_05/jre/lib/security/cacerts`
   - **WebSphere:** `IBMWS_HOME/java/jre/lib/security/cacerts`
   - **Built-in application server:** `SF_HOME/_jvm/lib/security/cacerts`

Certificates in Select Federation

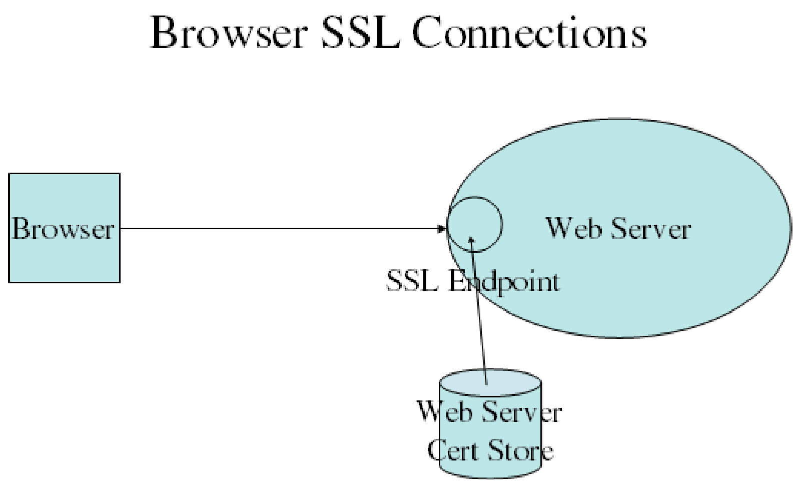
Certificates in Select Federation may be used from different locations depending upon which particular security option is being used, as follows. Note that sometimes the application server that runs Select Federation does not terminate the SSL connection as it is behind a web-server proxy. For simplicity, the word “Web Server” is meant to indicate either an application server directly available to partners (terminating the SSL connection) or a web-server proxy that terminates the SSL connection from the partner.

- CA Certificates installed in JVM cacerts trust store (a Java keystore)
  - Used for Secure Sockets Layer (SSL) Client/Server Authentication
- SSL Server Certificates installed in Web Server cert store (such as the built-in application server Java keystore)
  - Used for SSL Server Authentication
- SSL Client / XML Signing Certificates installed in Select Federation cert store (a java keystore)
  - Used for SSL Client Authentication, XML Signing
- Encryption Certificates installed in Select Federation cert store (a java keystore)
  - Used for XML Encryption

Browser SSL Connections

The browser must trust the CA that issued the web server’s SSL Certificate.

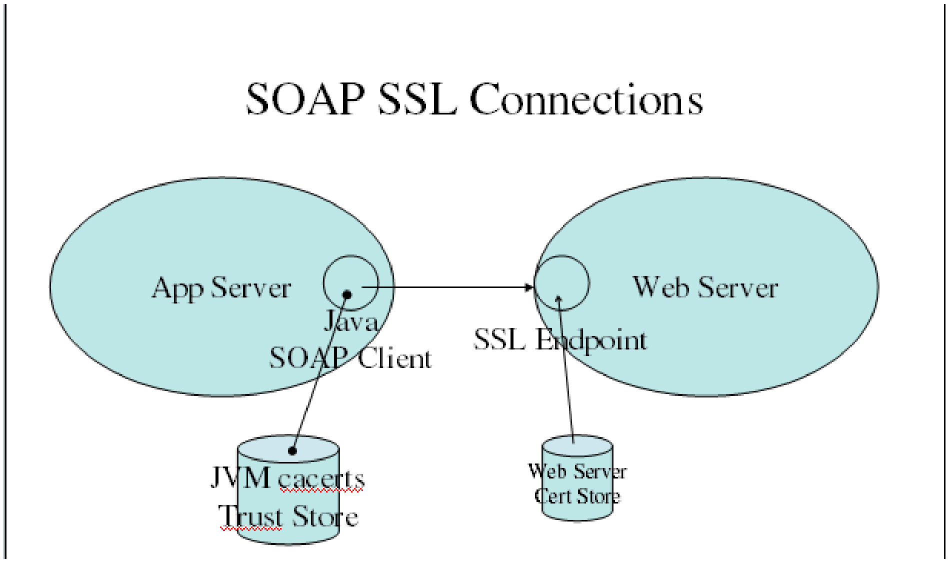
Figure 18 Browser SSL Connections



SOAP SSL Connections

The Client JVM must trust the CA that issued the web server’s SSL Server Certificate (JVM cacerts Trust Store lists trusted CA Certificates).

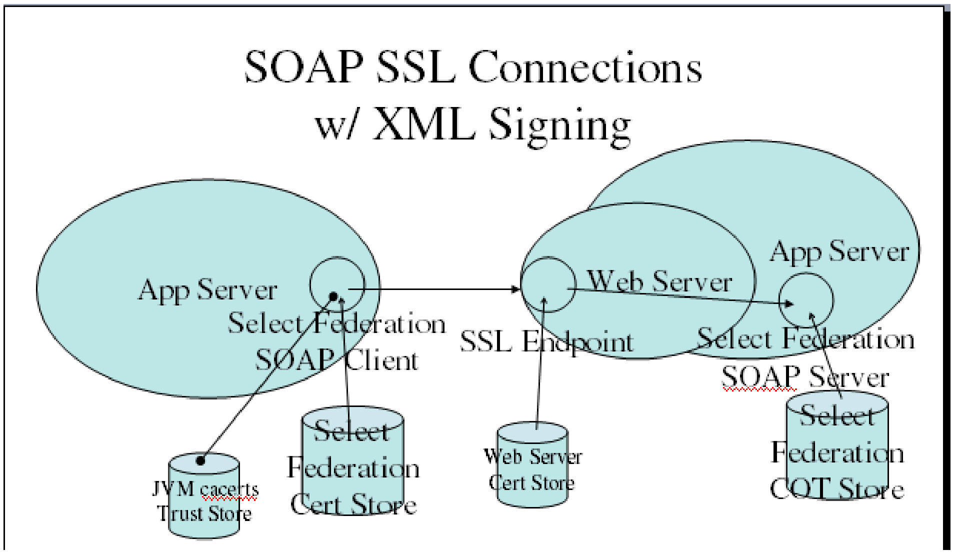
Figure 19 SOAP SSL Connections



SOAP SSL Connections with XML Signing

- The client JVM must trust the CA that issued the web server's SSL Server Certificate.
- The Select Federation SOAP server must trust the client's Signing Certificate (from metadata in the COT Store).

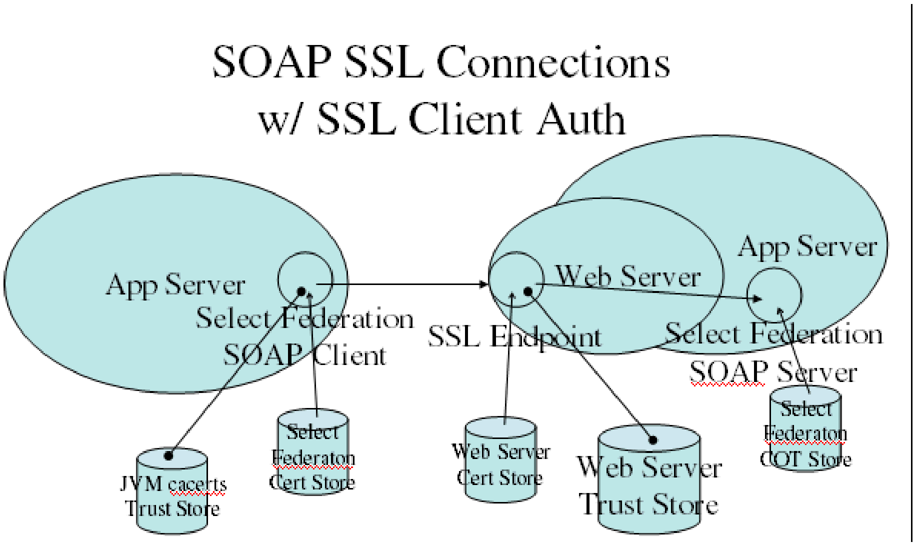
Figure 20 SOAP SSL Connections with XML Signing



SOAP SSL Connections with SSL Client Authentication

- Client JVM must trust the CA that issued the web server's SSL Server Certificate.
- Web server must trust the CA that issued the client's SSL Client Certificate (for Java the web server this is also JVM cacerts Trust Store).
- Select Federation SOAP Server must trust the client's SSL Client Certificate (from metadata in the COT Store).

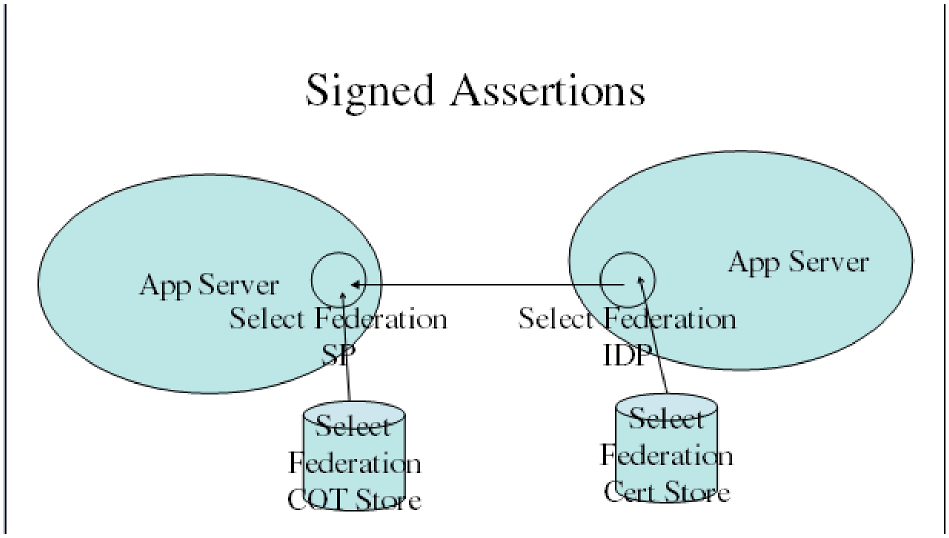
Figure 21 SOAP SSL Connections with SSL Client Authentication



Signed Assertions

Select Federation SP must trust the IDP's Signing Certificate (from metadata in the COT Store). The signing certificate is used from the Select Federation Certificate Store (typically a file-based store). The trusted certificate(s) used to verify the authenticity of the certificate used by the other party to sign its assertion is used from the COT store (where Select Federation stores the metadata, such as in the database or LDAP).

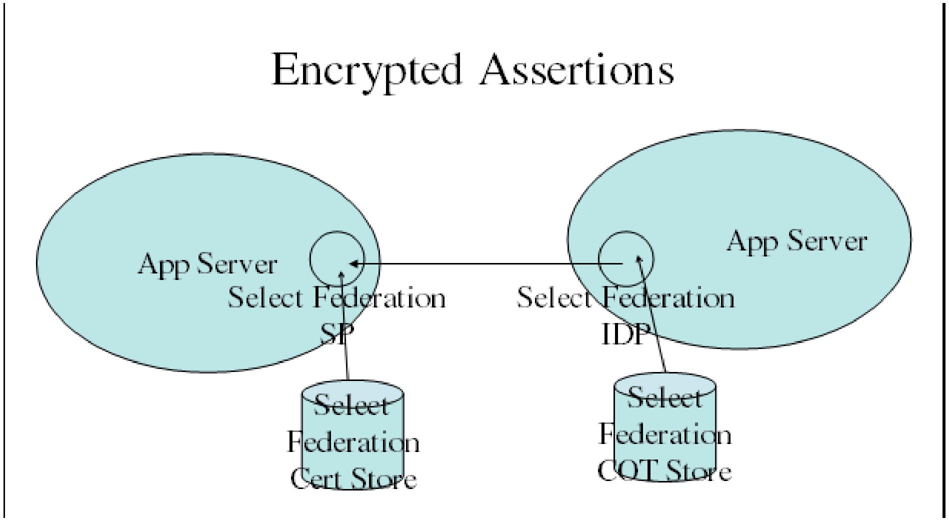
Figure 22 Signed Assertions



Encrypted Assertions

- Select Federation IDP must know the SP's Encryption Certificate (from metadata in the COT Store).

Figure 23 Encrypted Assertions



Select Federation SSL Client/XML Signing Certificate

Add the following parameters to the `tfsconfig.properties` file for signing of assertions by an IDP (IDP's signing certificate verified by an SP):

```
# Keystore configuration

keystoreType=JKS

keystorePath=/Users/grw/trustgenix/iop/certs/jks/h-idp-sign.jks

keystorePassword=changeit

certAlias=tomcat

keyAlias=tomcat

keyPassword=changeit
```

Select Federation Encryption Certificate

Add the following parameters to the `tfsconfig.properties` file for encryption of assertions for an SP (SP's encryption certificate verified by an IDP):

```
# Encryption keystore configuration

encKeystoreType=JKS

encKeystorePath=/Users/grw/trustgenix/iop/certs/jks/h-wsp-enc.jks

encKeystorePassword=changeit

encCertAlias=tomcat

encKeyAlias=tomcat

encKeyPassword=changeit
```

Other Certificates

- Use the Select Federation Administration Console to manage the COT Store

  - Certificates are in partner metadata
- Use Java keytool to manage the JVM cacerts Trust Store
  - $JAVA_HOME/lib/security/cacerts
  - Default password `changeit`
  - `keytool -import -keystore cacerts -file ca.pem`
- Web Server certificates / trust stores are product specific
  - Built-in application server
    - Java keystores for SSL Server Certificates, configured in `server.xml`
    - JVM cacerts Trust Store

Configuring TLS Client Authentication

You need to configure a TLS client authentication endpoint (host/port) on your application server (or on the front-end web server, if you have one). This should be a separate endpoint from the one used for normal non-TLS client authentication requests.

Perform the following steps to configure a TLS client authentication:

1 Configure a TLS client authentication endpoint on the application server.

Configuring a TLS client authentication is slightly different on different application servers. The following example shows how to configure a TLS client authentication on the built-in application server that is included with Select Federation.

Example of configuring a TLS client authentication on the built-in application server:

There is a regular TLS on port 8443 and TLS with client authentication on port 9443. Make the following changes to the `server.xml` file located in the `SF_HOME/conf/` directory of the Select Federation installation:

```
<Connector port="8443" maxHttpHeaderSize="8192"
         maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
         enableLookups="false" disableUploadTimeout="true"
         acceptCount="100" scheme="https" secure="true"
         clientAuth="false" sslProtocol="TLS" />

<Connector port="8443" maxHttpHeaderSize="8192"
         maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
         enableLookups="false" disableUploadTimeout="true"
         acceptCount="100" scheme="https" secure="true"
         clientAuth="true" sslProtocol="TLS"
                    keystoreFile="<SF Install dir>conf\sslkeystore.jks"
                    keystorePass="password"
                    truststoreFile="<SF Install dir>conf\sslkeystore.jks"
                    truststorePass="password"/>
```

2 Edit the Select Federation server configuration.

Set the server configuration setting in the `tfsconfig.properties` file located in the `SF_HOME/conf/` subdirectory. Following is an example:

```
providerBaseTLSClientAuthSOAPURL=https://sf.example.com:9443/tfs-soap
```

3 Import partner certificates.

Be sure that the certificates that are to be presented by your partners will be trusted by the application server (or web server, whichever is terminating the TLS connection).

On the built-in application server, for example, ensure that the issue CA certificates (or the end certificates, if self-signed) are imported into the java `cacerts` file (`$JAVA_HOME/lib/security/cacerts`) and are marked as trusted. You can use the Certificate Management Tool that is provided with Select Federation for managing your certificates. See the *HP OpenView Certificate Management User's Guide* for details.

4 Download and exchange metadata.

When downloading metadata on the server side, be sure to select the download link (in the Administration console) that specifies TLS Client Authentication. Then exchange this metadata, which specifies the server TLS endpoint with your partner.

5 Configure the Application Protocol Policy.

See Configuring the Protocol Policy and Metadata on page 53 for instructions for each protocol.

Select Federation SSL Deployment on WebSphere 6.02 with TLS Client Authentication Enabled

The following instructions describe how to configure Select Federation deployed on WebSphere to be TLS client authenticated in a Select Federation SSL deployment. These instructions apply whether Select Federation is deployed as an SP or IDP on WebSphere.

➤ FixPack 15 needs to be installed for Websphere 6.0.2 for SSL Client Authentication to work correctly.

Deploy Select Federation SSL on WebSphere 6.x by completing the following tasks:

- Task 1: Add a new SSL Repertoire
- Task 2: Enable the Application Server to use the new SSL Repertoire

Task 1: Add a new SSL Repertoire

1 Log into the WebSphere Administrative console.

2 Click on **Security** → **SSL** on the left navigation bar.

3 Click on the **New JSSE repertoire** button.

4 Add and verify the following entries:

- **Alias** = SF Alias
- **Client Authentication** is unchecked <default value>
- **Security Level** = HIGH <default value>
- **Cipher Suites** = add nothing <default value>
- **Crytographic token** is unchecked <default value>
- **Predefined JSSE provider** is selected <default value>
- **SelectProvider** = IBMJSSE2 <default value>

- **Custom JSSE Provider** is NOT selected <default value>
- **Custom Provider** has no entry in it <default value>
- **Protocol** = SSLv3
- **Key File name** = path/to/your/<webkeystore.jks>
- **Key file password** = <*password for keystore*>
- **Key file format** = JKS
- **Trust File name** = `path/to/your/<`*`webkeystore`*`.jks>` — this is the location where you need to import your partner certificate

  `webkeystore` is the keystore that contains the key exchanged between the IDP `jvm` keystore (`cacerts`) and Select Federation deployed on WebSphere as an SP.
- **Trust file password** = <*password for keystore*>
- **Key file format** = JKS

5 Click **OK** to save your settings.

Task 2: Enable the Application Server to use the new SSL Repertoire

1 Click on **Servers** → **Application servers** → (your server) →**Web Container** settings.

2 Select **Web container transport chains** → **WCInboundDefaultSecure** and click on **SSL Inbound Channel (SSL_2)**.

3 In the SSL Repertoire list select the repertoire created in Task 1: Add a new SSL Repertoire.

4 Click **OK** to save the configuration.

5 Set the value of `keyAlgorithmName` to **`IBMX509`** in the `conf/tfsconfig.properties` file in the WebSphere SP.

6 Stop and restart the WebSphere server after making all the above changes.

- For Windows:

  `stopWebSphere.cmd`

  `startWebSphere.cmd`
- For UNIX:

  `./stopWebSphere.sh`

  `./startWebSphere.sh`

Make sure that you have exchanged all necessary certificates needed for Server Side SSL configuration.

Make sure that the Select Federation IDP side of WebSphere is installed and set up correctly for client authentication. See Configuring TLS Client Authentication on page 165.

Configuring Certificate Revocation Checking

Select Federation uses certificates for ensuring security of communication via established methods such as SSL / TLS, XML Digital Signatures, and so on. In many deployments, it is important to verify the validity of certificates against a certification authority's published list of revoked certificates. Select Federation 6.60 provides a simple means of enabling certificate revocation checking via Certificate Revocation Lists or CRLs.

Select Federation checks certificate revocation when properly enabled, and when the certificate contains a CRL Distribution Point or CDP. Select Federation uses the PKIX certificate validation and revocation checking mechanisms provided by the Java runtime to achieve this. Note that this requires that all partner certificates are issued by Certificate Authorities listed in the Java cacerts file, or that the certificates themselves are added to the Java cacerts file.

It is recommended that when using revocation checking, certificates from trusted third-party CAs that include the CRL Distribution Point extension should be used. It is possible to use other certificates, but the CRLs need to be pre-loaded manually in the trust store for that to work.

Enabling Revocation Checking

To enable certificate revocation checking, add the following lines to the `tfsconfig.properties` file.

```
pkixCertValidation=1

pkixCertValidation.checkRevocationStatus=1
```

Each of the following parameters have default settings. If your deployment configuration does not match the defaults, then add these parameters to the `tfsconfig.properties` file:

`pkixCertValidation.caKeystoreType=<`*`Key_store_type`*`>`
(defaults to `JKS`)

`pkixCertValidation.caKeystorePath=<`*CA_Key_store_path*`>`
(defaults to `JAVA_HOME/lib/security/cacerts`)

`pkixCertValidation.caKeystorePassword=<`*password_for_the_CA_Key_Store*`>`
(defaults to `changeit`)

Make sure the following Java option is set when you start the application server:

```
-Dcom.sun.security.enableCRLDP=true
```

The steps to enable certificate revocation checking vary depending on which application server Select Federation is deployed on.

Enabling Revocation Checking for the Built-In Application Server on Windows

Select Federation is installed as a Windows service when deployed on the built-in application server. To set the java option for enabling revocation checking, the service must be uninstalled first, and then reinstalled after adding the java option to the service installation script.

Perform the following steps to set the java option for enabling revocation checking:

1 Stop the Select Federation server from the Windows services panel.

2. Go to the Select Federation install directory and run the **service.bat** script (located in the bin subdirectory) to remove the Select Federation service:

   ```
   C:\Program Files\HP OpenView\Select Federation>bin\service.bat remove
   SelectFederation
   ```

   ```
   The service SelectFederation has been removed.
   ```

3. Open service.bat and go to the end of the file to see the following statement:

   ```
   "%EXECUTABLE%" //US//%SERVICE_NAME% ++JvmOptions "-Dfile.encoding=UTF-8"
   ```

4. Add a line for the java option corresponding to revocation checking directly below this line:

   ```
   "%EXECUTABLE%" //US//%SERVICE_NAME% ++JvmOptions
   "-Dcom.sun.security.enableCRLDP=true"
   ```

5. Run service.bat to install Select Federation Windows service:

   ```
   C:\Program Files\HP OpenView\Select Federation>bin\service.bat install

   Installing the service 'SelectFederation' ...

   Using CATALINA_HOME:    C:\PROGRA~1\HPOPEN~1\SELECT~1

   Using JAVA_HOME:        C:\PROGRA~1\HPOPEN~1\SELECT~1\_jvm

   Using JVM:
   C:\PROGRA~1\HPOPEN~1\SELECT~1\_jvm\bin\client\jvm.dll

   Using jvm opts ..

   The service 'SelectFederation' has been installed.
   ```

6. Start the Select Federation server from the Windows Services panel.

Enabling Revocation Checking for the Built-In Application Server on UNIX

On UNIX (Linux, Solaris or HP-UX), set the java option for enabling revocation checking in catalina.sh, which is located in the /bin/ subdirectory.

Perform the following steps:

1. Stop the Select Federation server by executing the command **bin/shutdown.sh** from the installation directory.

2. Open catalina.sh and look for the following line:

   ```
   JAVA_OPTS="$JAVA_OPTS"-Dfile.encoding=UTF-8
   ```

3. Add the following revocation checking java option below this line:

   ```
   JAVA_OPTS="$JAVA_OPTS "-Dcom.sun.security.enableCRLDP=true
   ```

4. Start the server by executing the following command from the installation directory:

   ```
   bin/startup.sh
   ```

   The new java option has taken effect.

Enabling Revocation Checking for WebLogic 8.1 and 9.1

The way that you set the java option for enabling revocation checking for WebLogic 8.1 and WebLogic 9.1 depends on whether you are using a managed server or an unmanaged server. The following subsections describe how to enable revocation checking for WebLogic 8.1 and 9.1 on managed and unmanaged servers.

Enabling Revocation Checking for WebLogic 8.1 Using a Managed Server

Perform the following steps to set the java option for enabling revocation checking for WebLogic 8.1 if you are using a managed server:

1 Launch the WebLogic Admin Console.

2 Expand Servers in the left pane of the Console and select the desired <*server* >.

3 Select **Configuration** → **Remote Start** on the Main panel.

4 Specify the java options in the **Arguments** field.

Enabling Revocation Checking for WebLogic 9.1 Using a Managed Server

Perform the following steps to set the java option for enabling revocation checking for WebLogic 9.1 if you are using a managed server:

1 Launch WebLogic Admin Console.

2 Click on **Environment** → **Servers** in the left pane.

3 Click on the server where Select Federation is deployed in the main pane.

4 Select the **Configuration** tab and then the **Server Start** tab.

5 Specify the java options in the **Arguments** field.

Enabling Revocation Checking for WebLogic 8.1 and 9.1 Using an Unmanaged Server

The way that you enable revocation checking on unmanaged servers is the same for WebLogic 8.1 and 9.1.

Perform the following steps to set the java option for enabling revocation checking for WebLogic 8.1 or 9.1 if you are using an unmanaged server:

1 Create a backup copy of the WebLogic Server start scripts in domain-name.

`domain-name` is the directory in which you located the domain. By default, this directory is `BEA_HOME\user_projects\domains\domain-name`.

- For scripts that start an Administration Server, back up the following command:

  On Windows: ***domain-name*\startWebLogic.cmd**

  On UNIX: ***domain-name*/startWebLogic.sh**

- For scripts that start a Managed Server, back up the following command:

  ON Windows: ***domain-name*\startManagedWebLogic.cmd**

  On UNIX: ***domain-name*/startManagedWebLogic.sh**

2 Open the start scripts in a text editor.

3 Edit the `set JAVA_OPTIONS` command to specify the Java options.

If you specify multiple options, separate each option by a space, and place quotes around the entire set of options. For example:

On Windows: **`set JAVA_OPTIONS="-Dcom.sun.security.enableCRLDP=true"`**

On UNIX: **`JAVA_OPTIONS="-Dcom.sun.security.enableCRLDP=true" ; export JAVA_OPTIONS`**

4 Save the start script.

5 Restart the server.

Enabling Revocation Checking for WebSphere 6.0.2

Perform the following steps to set the java option for enabling revocation checking for WebSphere 6.0.2:

1 Start the WebSphere Admin Console.

2 Select **Servers** → **Application Servers** on the Left pane.

3 Click on the server instance on the main menu that needs to be configured.

4 Select **Java and Process Management** → **Process Definition** under the Server Infrastructure.

5 Select **Java Virtual Machine** on the Process Definition page under Additional properties.

6 Add the following JVM parameters:

```
-Dcom.ibm.security.enableCRLDP=true
```

```
-Dcom.ibm.jsse2.checkRevocation=true
```

7 Optionally, specify additional JVM options in the Generic JVM Arguments on the Java Virtual machine page.

Additional Requirements for SSL Deployments

If you configured `https` as the preferred protocol during the Select Federation installation, be sure that the Certificate Authority (CA) that issued the server SSL certificate is included in the Java trust list **at the partner site**. For example, if your installation is a Service Provider (SP) that was issued a self-signed SSL certificate by Select Federation during installation, that certificate must be imported in the java `cacerts` file of the Identity Provider (IDP). Look for a certificate called **tomcat.cer** under `SF_HOME/conf/`, and import it into the IDP's trust store. You could use a simple java utility like `keytool` to install the CA's certificate in your Java `cacerts` file (see step 2 in Installing a Third-Party Certificate on page 160 for the default paths).

A sample `import` command looks somewhat like the following:

```
% keytool -import -trustcacerts -alias MyCA -file <CA certificate file>
-keystore <path to your Java installation>/lib/security/cacerts -storepass
<default value is "changeit">
```

The same is true if the SP side SSL certificate is issued by a third-party CA. In that case, the only change would be to import the third-party CA certificate instead of the SP self-signed certificate. Note that the above process needs to be repeated at the SP site if the IDP site is also deployed over SSL.

The location of the Java `cacerts` file varies depending on the application server you chose during installation. See Installing a Third-Party Certificate on page 160 for the default paths for the application servers.

15 Customizing Select Federation

This chapter describes how to customize and configure the end user visible pages of Select Federation. These pages are mainly the Privacy Manager and the federation consent pages. Customizing these pages refers to changing the look and feel – the styles, logos, colors, and so on.

Customization

Simple Customization

Most common requirements for customization, such as changing the styles used for various visual elements, logos, and colors can be met by simply adding a few variables to the Select Federation configuration file `tfsconfig.properties`. The variables are listed in the following table:

Table 3 Customization Variables in tfsconfig.properties

Variable	Type	Description
`presentation.css-url`	String	The URL of the style sheet to be used for rendering Privacy Manager pages.
`presentation.logo-src`	String	The URL of the logo to be displayed on the Privacy Manager pages.
`presentation.logo-text`	String	Alternative text to be displayed when the mouse is hovered over the logo.
`presentation.logo-href`	String	The URL to which the user is navigated upon clicking the logo.

Advanced Customization

If the above customization parameters do not meet your requirements for customization of the Privacy Manager, then further customization can be done by modifying the XSLT files that are used to render the pages. The original XSLT files are typically in the `conf/stylesheets` subdirectory relative to the current working directory of the JVM (the directory from where the application server is launched).

It is also possible to create XSLT files that are specific to a particular user agent. This requires you to define possible user-agent aliases in the `tfsconfig.properties` file as follows:

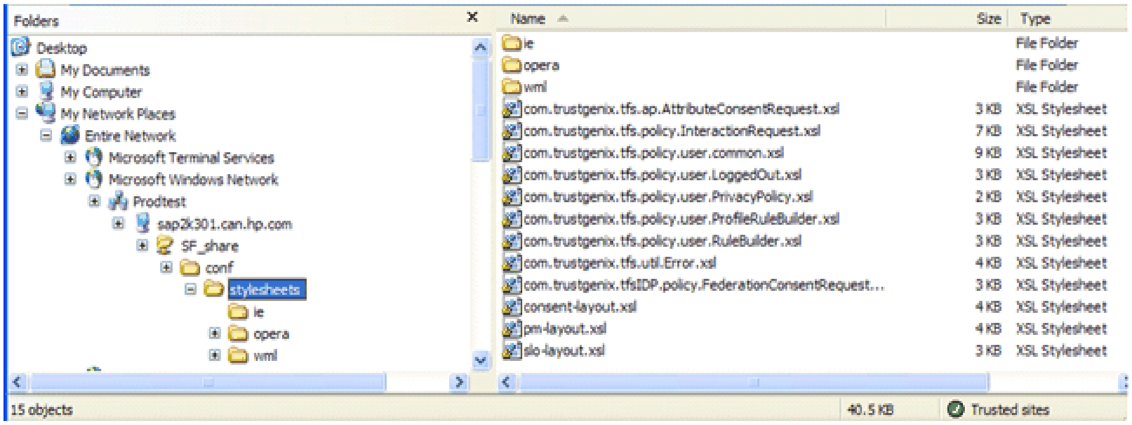
```
presentation.browsers=wml ie mogw
presentation.mogw=MOGW

presentation.ie=MSIE Internet%20Explorer

presentation.wml=wml
```

As you can see above, the `presentation.browsers` variable indicates the complete list of defined browser types. Subsequently, for each browser type defined, a user-agent search string is defined. Select Federation will determine if the user-agent string presented by the browser contains the string defined above. If so, it will use the first matching browser type. The resulting directory structure is shown below:

Figure 24 Directory for XSLT template files



```
Desktop
  My Documents
  My Computer
  My Network Places
    Entire Network
      Microsoft Terminal Services
      Microsoft Windows Network
        Prodtest
          sap2k301.can.hp.com
            SF_share
              conf
                stylesheets
                  ie
                  opera
                  wml
```

Name	Size	Type
ie		File Folder
opera		File Folder
wml		File Folder
com.trustgenix.tfs.ap.AttributeConsentRequest.xsl	3 KB	XSL Stylesheet
com.trustgenix.tfs.policy.InteractionRequest.xsl	7 KB	XSL Stylesheet
com.trustgenix.tfs.policy.user.common.xsl	9 KB	XSL Stylesheet
com.trustgenix.tfs.policy.user.LoggedOut.xsl	3 KB	XSL Stylesheet
com.trustgenix.tfs.policy.user.PrivacyPolicy.xsl	2 KB	XSL Stylesheet
com.trustgenix.tfs.policy.user.ProfileRuleBuilder.xsl	3 KB	XSL Stylesheet
com.trustgenix.tfs.policy.user.RuleBuilder.xsl	3 KB	XSL Stylesheet
com.trustgenix.tfs.util.Error.xsl	4 KB	XSL Stylesheet
com.trustgenix.tfsIDP.policy.FederationConsentRequest...	3 KB	XSL Stylesheet
consent-layout.xsl	4 KB	XSL Stylesheet
pm-layout.xsl	4 KB	XSL Stylesheet
slo-layout.xsl	3 KB	XSL Stylesheet

Branding the End User Pages

The Select Federation presentation engine allows branding (changing title, logos, colors, and to some degree layout) of the following end user pages:

- Privacy Manager pages
- Consent pages
- Login page
- Single logout page
- Error page

The presentation engine allows you to change the branding of these pages in the following ways:

- Branding by Configuration Settings — changing configuration entries in the `tfsconfig.properties` file
- Branding by Using CSS Stylesheets — changing or replacing the CSS stylesheets
- Branding by Using XSLT Stylesheets — changing or replacing the XSLT stylesheets

Branding by Configuration Settings

You can change the branding of end user pages using configuration settings in the `tfsconfig.properties` file. This section shows how to do this with the Privacy Manager pages.

In the Privacy Manager web pages, “Personal Profile” can be localized from the `tfsconfig.properties` file, by changing the `profile.name` setting as follows:

Change: `profile.name=Personal Profile`

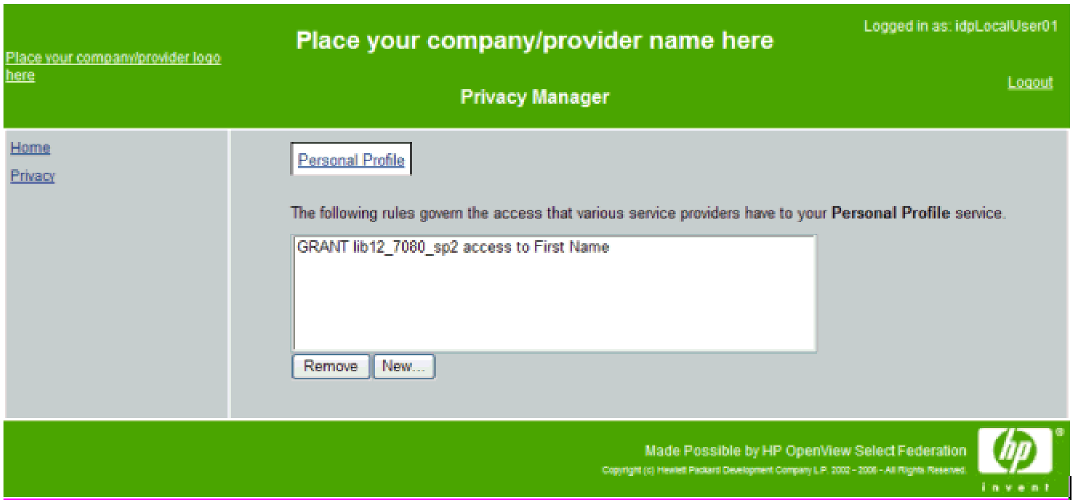
To: `profile.name=<`*`specific_language_using_encoding_UTF-8`*`>`

To avoid mistakes, any characters that are not covered under the ISO-8859-1 encoding by default, should be entered in their Unicode Escaped representation. For example:

```
hpsf.ldapUserBaseDN=cn=検索オプション,OU=sf,OU=ov,OU=hp,DC=domain
```

A minimum amount of branding is critical towards a complete and successful deployment of the Privacy Manager. Without any branding, the page visited by the end users to review their policies (`https://<BASE_URL>/pm/privacy`) appears as follows:

Figure 25 Privacy Manager Without Branding



Place your company/provider logo here

Place your company/provider name here

Privacy Manager

Logged in as: idpLocalUser01

Logout

Home

Privacy

Personal Profile

The following rules govern the access that various service providers have to your **Personal Profile** service.

GRANT lib12_7080_sp2 access to First Name

Remove New...

Made Possible by HP OpenView Select Federation

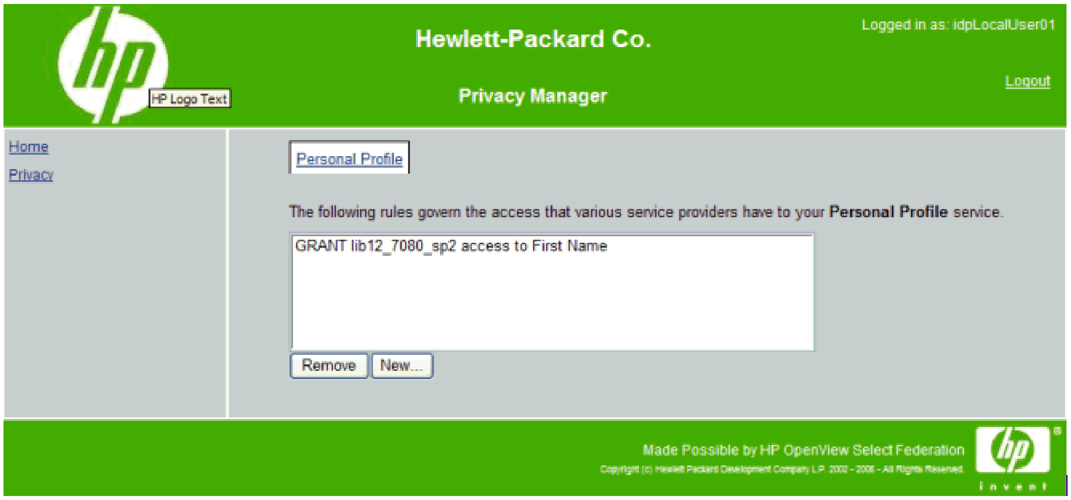
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved.

Perform the following steps to provide bare-minimum rebranding of the end user pages using configuration settings in the `tfsconfig.properties` file:

1 Uncomment and specify meaningful values for the following parameters in the `tfsconfig.properties` file:

```
## CSS stylesheet reference, that controls colors and layout
#presentation.css-url=/styles/users.css
## Logo source
presentation.logo-src=/styles/greenStretch.jpg
## Logo alternative text
presentation.logo-text=HP Logo Text
## Logo hyperlink, this is the page users will go when they click the logo
presentation.logo-href=http://www.hp.com
## Header text, this is the text that will be placed at the very top of the page
presentation.header-text=Hewlett-Packard Co.
```

The following figure shows the bare-minimum rebranding, which provides a more complete experience for end users when they review their policies.



The following rules govern the access that various service providers have to your **Personal Profile** service.

GRANT lib12_7080_sp2 access to First Name

Made Possible by HP OpenView Select Federation

Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved.

2 Restart your application server for these changes to take effect.

You can also restyle the end user pages by changing some of the parameters in the `tfsconfig.properties` file. For example, for the Footer Logo source, you may change the HP logo for another one of the following to meet the needs of your color scheme:

```
blueInvent.jpg
greenInvent.jpg
orangeInvent.jpg
```

You can set all of the following parameters to be different or the same:

```
#presentation.generic-footer-logo-src=/styles/
blueInvent.jpg#presentation.idp-footer-logo-src=/styles/
greenInvent.jpg#presentation.sp-footer-logo-src=/styles/orangeInvent.jpg
```

➤ The changes made to these values require that you edit and re-evaluate the following values in the default CSS stylesheet (`users.css`) as well: `genericHeaderTable`, `genericFooterTable`, `idpHeaderTable`, `idpFooterTable`, `spHeaderTable`, `spFooterTable`. See the next section, Branding by Using CSS Stylesheets, for details.

Branding by Using CSS Stylesheets

The CSS stylesheet can be used to make drastic changes. Color changes are easy and "safe" whereas layout changes require careful testing. The CSS file that you may edit is in the same location where you deployed the `styles.war` file during your installation (`styles/users.css`). This section describes how to use the CSS stylesheet to change the branding of the Privacy Manager pages.

By default, the Privacy Manager pages are styled to reflect the color scheme of the type of site from which users navigate. Following are the different color schemes:

- Green for IDP-type pages (coming from an IDP site role)
- Orange for SP-type pages (coming from an SP site role)
- Blue for generic pages

As a small exercise to become familiar with the stylesheet, go through the following scenario to rebrand the Privacy Manager pages to reflect a color scheme of a portal from which end users navigate. The steps outline the simple changes you can make to apply one single color scheme for all the Privacy Manager pages. These changes override the default colors.

Perform the following steps to change the color scheme of the Privacy Manager pages:

1 Change the background color for the footer of IDP-type pages from green to blue in the `styles/users.css` file:

```
table.idpFooterTable, table.idpFooterTable tr td table {
   ...
   background-color: #003366;
   ...
}
```

2 Change the background color for the footer of SP-type pages from orange to blue in the `styles/users.css` file:

```
table.spFooterTable, table.spFooterTable tr td table {
background-color: #003366;
}
```

3 Change the background color for the header of IDP-type pages from green to blue in the `styles/users.css` file:

```
table.idpHeaderTable, table.idpHeaderTable tr td table {
   ...
   background-color: #003366;
   ...
}
```

4 Change the background color for the header of the SP-type pages from orange to blue in the `styles/users.css` file:

```
table.spHeaderTable, table.spHeaderTable tr td table {
background-color: #003366;
}
```

5 Refresh the page to see the changes.

The background colors have all been changed to blue.

Now you need to change the logo color to blue to apply one single color scheme for all the Privacy Manager pages. You do this by editing the `tfsconfig.properties` file.

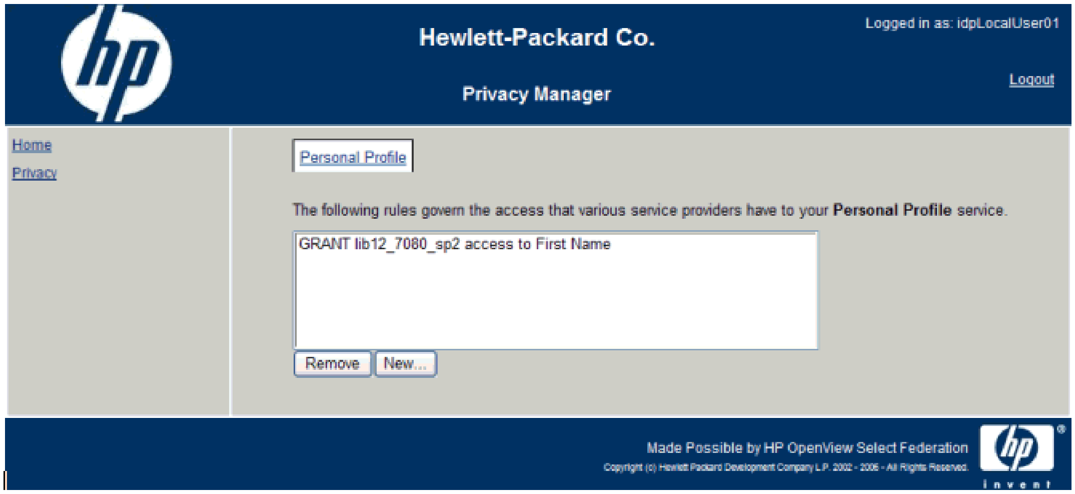
6 Edit the `tfsconfig.properties` file to provide a meaningful logo of your company (`logo-src`) and use the provided HP Logo for the footer (`footer-logo-src`).

Use a color that is the closest match to your required color scheme. In this scenario, the color is `blueInvent.jpg`:

```
## Logo source
presentation.logo-src=/styles/blueStretch.jpg
## Footer Logo Source:
#presentation.generic-footer-logo-src=/styles/blueInvent.jpg
#presentation.idp-footer-logo-src=/styles/blueInvent.jpg
#presentation.sp-footer-logo-src=/styles/blueInvent.j
```

7 Restart your application server for the changes made to the `tfsconfig.properties` file to take effect.

You now have a cohesive look and feel for your end users. The following example shows the IDP-type pages with the color scheme changes:


Hewlett-Packard Co.
Privacy Manager
Logged in as: idpLocalUser01
Logout
Home
Privacy
Personal Profile
Remove
New...
Made Possible by HP OpenView Select Federation
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved.
invent

The following rules govern the access that various service providers have to your **Personal Profile** service.

GRANT lib12_7080_sp2 access to First Name

The CSS stylesheet includes many HTML elements with specific class names. It is possible to accomplish many of the styling and branding goals without touching anything other than the stylesheet and the `tfsconfig.properties` file.

Branding by Using XSLT Stylesheets

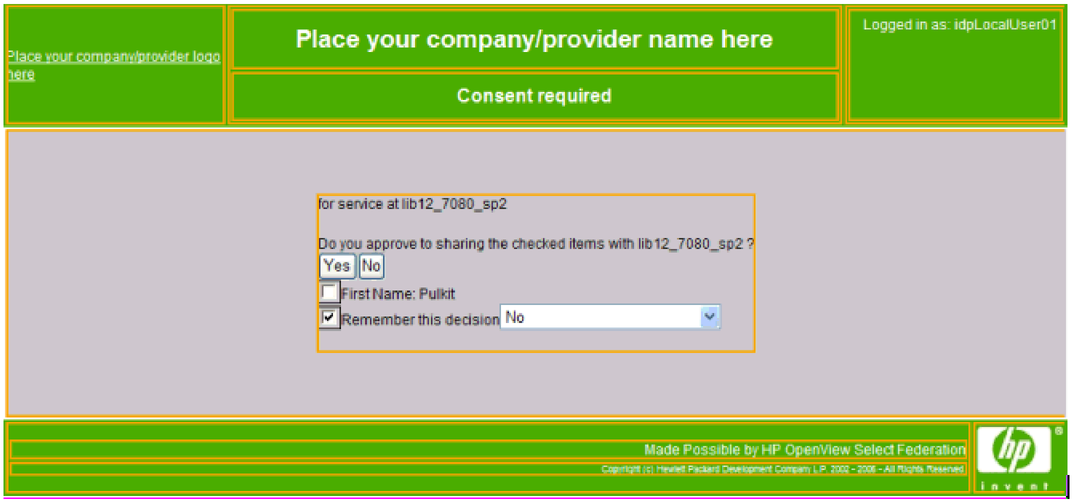
The Select Federation presentation engine uses XSLT stylesheets to transform semantic XML documents into browser pages. These stylesheets provide ultimate control over the end user look and feel, but modifying these is recommended only to experts. A good reason to modify these files is to allow for better localization.

The stylesheets are located in a subdirectory of the `/conf/` directory of the installation. The files are structured in a hierarchy according to locale and browser type. The names of the XSLT stylesheets and the directory structure must **not** be changed. Most layout changes can be achieved by changing those stylesheets that have names ending in "-`layout.xsl`".

Following are the stylesheets that can be changed.

- `consent-layout.xsl`— This stylesheet affects all user-facing consent pages. Edit this stylesheet only if the structure displayed in the following figure cannot be adapted to meet your rebranding and restyling needs by only working with the `css` and `tfsconfig.properties` files.

Figure 26 consent-layout.xsl Stylesheet Layout



Place your company/provider logo here

Place your company/provider name here

Consent required

Logged in as: idpLocalUser01

for service at lib12_7080_sp2

Do you approve to sharing the checked items with lib12_7080_sp2 ?

Yes No

□ First Name: Pulkit

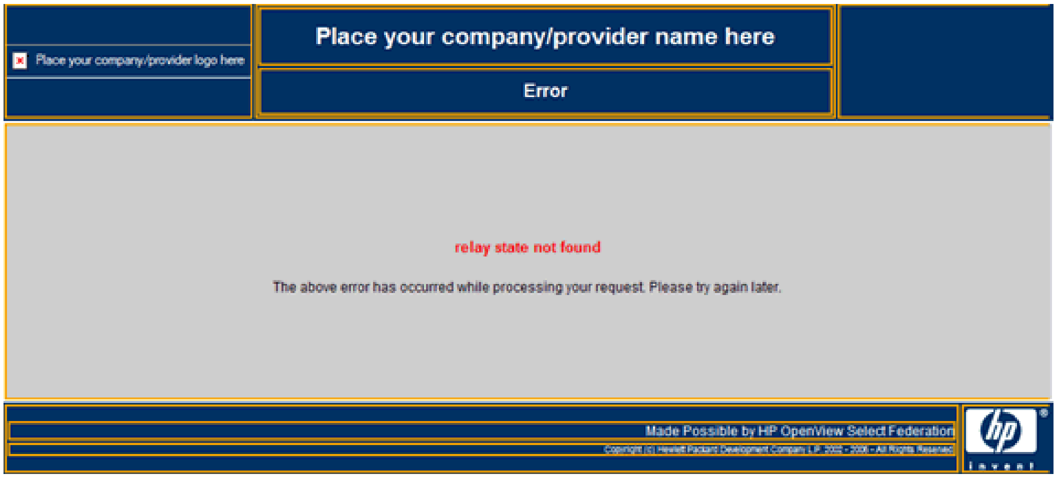
☑ Remember this decision No

Made Possible by HP OpenView Select Federation

Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved

- `error-layout.xsl` - This stylesheet affects all user-facing error pages. Edit this stylesheet only if the structure displayed in the following figure cannot be adapted to meet your rebranding and restyling needs by only working with the `css` and `tfsconfig.properties` files.

Figure 27 error-layout.xsl Stylesheet Layout



Place your company/provider logo here

Place your company/provider name here

Error

relay state not found

The above error has occurred while processing your request. Please try again later.

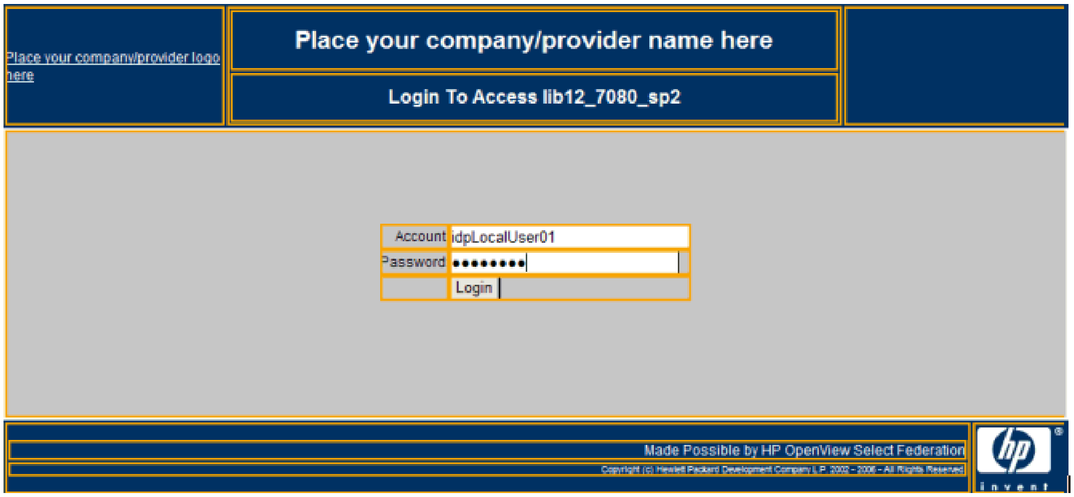
Made Possible by HP OpenView Select Federation

Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved

- `login-layout.xsl` - This stylesheet affects all user-facing login pages. Edit this stylesheet only if the structure displayed in the following figure cannot be adapted to meet your rebranding and restyling needs by only working with the `css` and `tfsconfig.properties` files.

  This stylesheet is used to render the login page only when the loginURL is not explicitly specified in the `tfsconfig.properties` file.

Figure 28 login-layout.xsl Stylesheet Layout

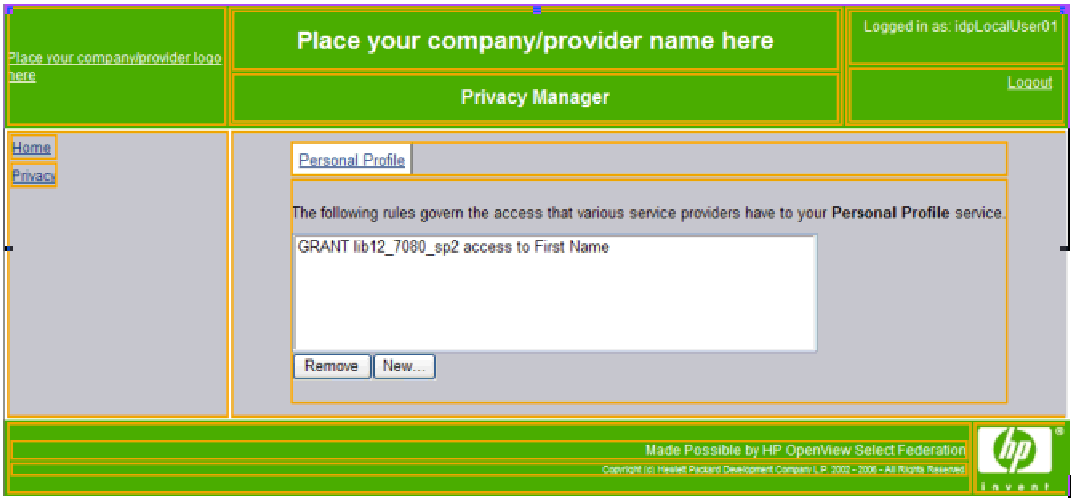

Place your company/provider logo here
Place your company/provider name here
Login To Access lib12_7080_sp2
Login
Made Possible by HP OpenView Select Federation
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved
invent

Account idpLocalUser01

Password ••••••••

- pm-layout.xsl - This stylesheet affects all user-facing Privacy Manager pages. Edit this stylesheet only if the structure displayed in the following figure cannot be adapted to meet your rebranding and restyling needs by only working with the css and tfsconfig.properties files.

Figure 29 pm-layout.xsl Stylesheet Layout

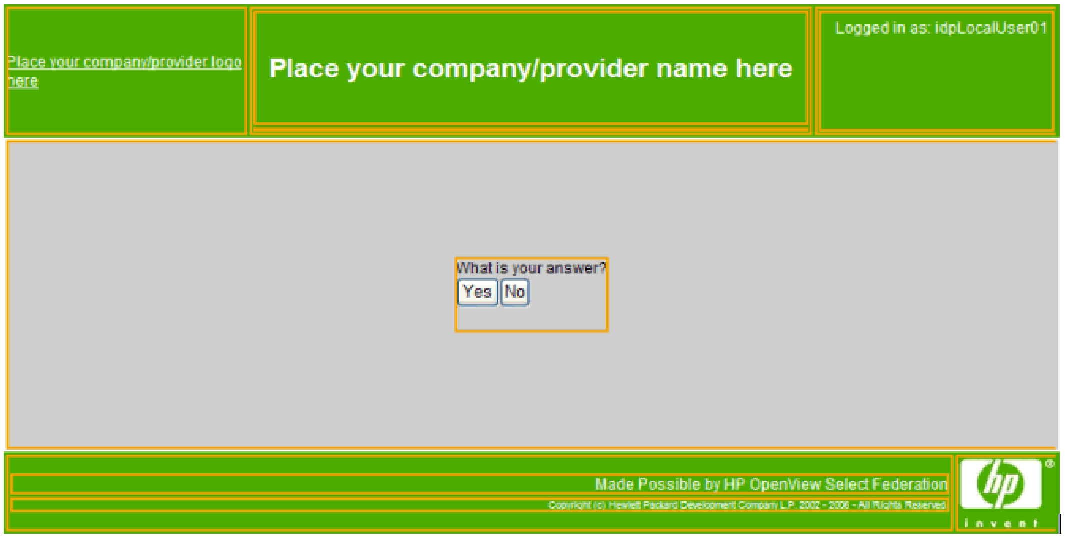

Place your company/provider logo here
Place your company/provider name here
Logged in as: idpLocalUser01
Privacy Manager
Logout
Home
Privacy
Personal Profile
Remove
New...
Made Possible by HP OpenView Select Federation
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved
invent

The following rules govern the access that various service providers have to your **Personal Profile** service.

GRANT lib12_7080_sp2 access to First Name

- slo-layout.xsl. - This stylesheet affects all user-facing Single Log-out (SLO) pages. Edit this stylesheet only if the structure displayed in the following figure cannot be adapted to meet your rebranding and restyling needs by only working with the css and tfsconfig.properties files.

Figure 30 slo-layout.xsl Stylesheet Layout


Place your company/provider logo here
Place your company/provider name here
Logged in as: idpLocalUser01
What is your answer?
Yes
No
Made Possible by HP OpenView Select Federation
Copyright (c) Hewlett Packard Development Company L.P. 2002 - 2006 - All Rights Reserved
invent

Localizing Select Federation

This section describes localization of the end user visible pages of Select Federation. These pages are mainly for the Privacy Manager and the federation consent page. Localization refers to changing these pages so that they are rendered effectively in a particular locale (language and country).

International Character Support

Setting Unicode Escaped Character Representations in the tfsconfig.properties File

Select Federation uses the `tfsconfig.properties` system configuration file in the `conf/` subdirectory, which is a Java properties file. If the `tfsconfig.properties` parameter values contain characters that are not supported in the `ISO-8859-1` encoding option, they must be entered in their Unicode Escaped representation.

For example the following entry that includes unsupported characters:

```
hpsf.ldapUserBaseDN=cn=検索オプシヨン,OU=sf,OU=ov,OU=hp,DC=domain,DC=com
```

must be entered in the `tfsconfig.properties` file as follows:

```
hpsf.ldapUserBaseDN=cn=\u691c\u7d22\u30aa\u30d7\u30b7\u30e7\u30f3,OU=sf,O
U=ov,OU=hp,DC=domain,DC=com
```

Optionally you can use tools such as native2ascii (available in the Sun JDK) to convert a file with native-encoded characters (characters which are non-Latin 1 and non-Unicode) to one with Unicode-encoded characters.

Setting JVM Character Encoding Options

If you want your installation to support international characters for user attributes exchanged between federated sites, it is strongly recommended that you set the default locale of the systems on which Select Federation is installed, so that the character encoding is `UTF-8`.

In addition, you need to set the `-Dfile.encoding=UTF-8` option for your application server. If you are using WebSphere or WebLogic as your application server for Select Federation, follow the steps in the following sections to set the JVM options. This option is already set in the built-in application server for Select Federation.

Setting Character Encoding for WebSphere

Perform the following steps to set the `-Dfile.encoding=UTF-8` option for WebSphere:

1. Start the WebSphere Administrative Console.
2. Click **Servers**.
3. Click **Application Servers**.
4. Click your server.
5. Click the **Configuration** tab.
6. Expand the **Java and Process Management** link.
7. Click **Process Definition**.
8. Click **Java Virtual Machine**.
9. Scroll to the Generic JVM Arguments text box and add **`-Dfile.encoding=UTF-8`**.
10. Click **Apply**.
11. Click **Save link to the master configuration** at the top.
12. Restart the WebSphere application server.

Setting Character Encoding for WebLogic

Perform the following steps to set the `-Dfile.encoding=UTF-8` option for WebLogic:

1. Edit `<BEA DOMAIN HOME DIRECTORY>/bin/setDomainEnv.sh`.
2. In the `JAVA_PROPERTIES` variable, add **-Dfile.encoding=UTF-8**.

   For example, the `JAVA_PROPERTIES` variable after it has been modified is as follows:

   ```
   JAVA_PROPERTIES="-Dplatform.home=${WL_HOME} -Dwls.home=${WLS_HOME}
   -Dwli.home=${WLI_HOME} -Dfile.encoding=UTF-8"
   ```

3. Save the file.

4. Restart the WebLogic application server.

Simple Localization

As with customization, most of the needs for localization to a particular language and country combination can be met by simply creating a resource bundle specific to that language, and if required to a country specific version of it. This resource bundle then needs to be added to the system classpath before starting the application server. The base name of the Java class for which you can create locale specific resource bundles is com.trustgenix.tfs.i18n.User. Sample locale specific resource bundles are provided on the SDK CD under the localization top-level directory.

Advanced Localization

In case you need to localize the end user pages (modify formatting or look and feel elements or alter layouts), you can create locale (language and country) specific subdirectories to the conf/stylesheets subdirectory. The name of the language specific subdirectory is the two letter **ISO Language Code** optionally followed by an underscore followed by the two letter **ISO Country Code**. For example, you may create a directory for the Swedish language as:

```
conf/stylesheets/sv.
```

Or create a directory for the Finland country specific variant of Swedish as:

```
conf/stylesheets/sv_FI.
```

Select Federation will attempt to find a locale specific directory for the XSLT for any end user facing page by matching the locale specification supplied by the browser. If the user agent specifies a country specific language variant, Select Federation will attempt to find the country specific and language specific subdirectory, but if it does not find that, Select Federation will attempt to find the generic language specific language subdirectory. If neither is not found, it will load the generic XSLTs at the top-level conf/stylesheets subdirectory.

A Configuration Parameters

This appendix lists and describes the configuration parameters for the following files:

- tfsconfig.properties — Contains general settings for Select Federation to customize your installation.
- spapiconfig.properties — Contains API application configuration file parameters for an SP install.
- idpapiconfig.properties — Contains API authority configuration file parameters for an IDP install.

Types of Configuration Parameters

There are five types of configuration parameters that are possible when you manually enter data.

Table 4 Types of Configuration File Parameters

Type	Example	Format
String	param=value	A String value. See Java Properties documentation for the list of special characters that require escaping.
StringList	param=value1 value2	A StringList is a space-separated list of String values (spaces appearing in values, if allowed, must be escaped).

Boolean	param=0
	param=1
		A Boolean has a value of 0 (false) or 1 (true).

Integer	param=123
	param=-1
		An Integer value.

TimeDuration	param=1s
	param=1h30m
	param=500
		A TimeDuration is a measure of time in days, hours, minutes, seconds (and milliseconds). Use the unit suffix ‘d’ or ‘D’ for days, ‘h’ or ‘H’ for hours, ‘m’ or ‘M’ for minutes, and ‘s’ or ‘S’ for seconds. A number without a unit suffix is treated as milliseconds.

tfsconfig.properties

If you edit the `tfsconfig.properties`, be sure to do the following:

1. Make a backup copy of the `tfsconfig.properties` 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 `tfsconfig.properties` file includes configuration parameters for specific categories. The following tables list and describe all the configuration parameters by category in the `tfsconfig.properties` file:

- Table 5, Core Configuration File Parameters
- Table 6, Application Configuration File Parameters
- Table 7, Authority Configuration File Parameters
- Table 8, DirPlugin_LDAP Plugin Configuration File Parameters
- Table 9, DirPlugin_ADS Plugin Configuration File Parameters
- Table 10, HPSA Adapter Configuration File Parameters
- Table 11, Privacy Manager Configuration File Parameters

Table 5 Core Configuration File Parameters

Name	Type	Default (if not required)	Description
jdbcDataSource	String	null	JNDI name of J2EE Data Source to use, by default, for connections to the database. If this option is provided then `jdbcAddr/Driver/User/Password` is ignored.
jdbcDataSource	String	null	JDBC provider class. Depends upon database being used. Supported classes are: • `com.trustgenix.tfs.JDBCProvider_Derby` • `com.trustgenix.tfs.JDBCProvider_MSSQL` • `com.trustgenix.tfs.JDBCProvider_MySQL` • `com.trustgenix.tfs.JDBCProvider_Oracle` • `com.trustgenix.tfs.JDBCProvider_Oracle_TomcatDS` • `com.trustgenix.tfs.JDBCProvider_Oracle_WebsphereDS`
jdbcAddr	String	null	JDBC address to use, by default, for connections to the database.
jdbcDriver	String	null	JDBC driver to use, by default, for connections to the database.

Name	Type	Default (if not required)	Description
jdbcUser	String	null	JDBC user to use, by default, for connections to the database.
jdbcPassword	String	null	JDBC password to use, by default, for connections to the database.
tblPrefix	String	"tfs"	Prefix to use, by default, for database tables.
userAttrs	StringList	null	List of user profile attributes to support.
<attralias>.dstSelect	String	null	If non-null, the DST select expression that maps to this profile attribute.
<attralias>.dstNS	String	null	The DST service namespace for this profile attribute.
<attralias>.samlAttr	String	null	If non-null, the SAML attribute that maps to the user's profile attribute.
<attralias>.samlAttrNS	String	null	The SAML attribute namespace for this profile attribute.
relayTimeout	Time Duration	20m	The time to allow for messages to be relayed through user agent (browser) connections. Determines various cache lifetimes and notOnOrAfter values in SAML assertions. Default is 20 minutes.
sessTimeout	Time Duration	1 day	Determines which session entries are eliminated. Those entries that have not been accessed in the last `sessTimeout` interval are eliminated from the session table.
purgeInterval	Time Duration	1h	Determines how often the runtime tables are purged of entries for abandoned sessions. Default is 1 hour.
auditPrune	Time Duration	0	Audit logs are pruned of all entries older than auditPrune. If auditPrune is 0, logs are not pruned.
adminEventPlugin	String	null	Specifies the `com.trustgenix.tfsAdmin.plugin.AdminEventPlugin` to be used.
defaultMetadata	String	"saml20"	Specifies the format of the metadata available at the providerId of the installation. It can be one of: liberty12, liberty11idp, liberty11sp, saml10, saml11, saml20, wsfed10.

Name	Type	Default (if not required)	Description
preferFrontChannel	Boolean	false	Indicates that protocols should prefer front-channel browser profiles (such as SAML POST) instead of back-channel profiles (such as SAML Artifact).
presentation.default Language	String	"en"	Indicates the language of the top-level XSLT files in the `conf/stylesheet` directory.
presentation.stylesheetDir	String	"conf/stylesheets"	Sets the directory with the XSLT files that control the presentation of end user pages. The default directory is where the installer copies these files.
presentation.css-url	String	"/styles/users.css"	The URL to the CSS stylesheet that will be used by the XSLT files of the presentation service.
presentation.logo-src	String	"/styles/logo.gif"	The logo is used for the top-left logo in pages served by the presentation service.
presentation.logo-text	String	"HP"	Used as an alternative text for the logo used by the presentation service. The same string is also used in some titles and headers.
presentation.logo-href	String	"http://www.hp.com"	Link to the page users will go when they click the logo rendered by the presentation service.
useSelectAccess	Boolean	true	Indicates if HP OpenView Select Access is used together with this HP OpenView Select Federation installation.
useSLOGetProfile	Boolean	false	This is set to true by the installer and indicates that the GET based method for Single Logout is to be used. If true, checkmarkURL should be set too.
checkmarkURL	String		An optional String that typically points to a picture that should be shown upon successful logout of a partner. The installer sets this to `<baseURL>/tfs/checkmark.gif`. This entry is required if useSLOGetProfile is true.

Name	Type	Default (if not required)	Description
auditDataProvider	String	`com. trustgenix. tfs.util. AuditData Provider_ JDBC`	If set, this channels audit logging to the specified class, at an IDP and SP. It sends administrative and operational audit log events to this class. If Select Audit is used for auditing system events, you can uncomment this parameter and set it to: `com.trustgenix.hpsf.selectaudit.AuditDataProvider_SelectAudit`. If Select Audit is not used, you can uncomment this parameter and set it to: `com.trustgenix.tfs.util.AuditDataProvider_JDBC`.
AuditDataProvider_ SelectAudit.auditTo HPSF	Boolean	False	If Select Audit is used for auditing and this entry is true, system events are logged to both Select Audit and Select Federation databases.
AuditDataProvider_SelectAudit.port	String	<Select Access default connector port>	The port number of the Select Access connector can be found in the *HP OpenView Select Access Installation Guide*.

auditProtocolMessages	String	True	Logs protocol events. You can prefix this parameter with the name of a particular protocol. For example, `saml20.auditProtocolMessages=0`. For each protocol the prefixed setting takes precedence over the generic setting. The recognized protocol prefixes are:
			`saml10`
			`saml11`
			`saml20`
			`idff11`
			`idff12`
			`wsfed10`
			Note that system state changes, such as when a user logged in, are logged in the Server Audit log. This setting does not affect the system log (Java VM output).

Name	Type	Default (if not required)	Description
logXML	String	INFO	Defines the `log4j.properties` level of XML message details logged in the system log. Therefore, if the logging level specified in the `log4j.properties` file is the same as or more verbose than the level specified as the value of this parameter, the diagnostic logs will contain the XML message details.
debugHttpParams	Boolean	False	Indicates if HTTP parameters need to be output when logging level is at debug.
debugHttpParams.maskAtts	StringList	Ecom_User_Password	List of HTTP headers, parameters, cookies, session variables to suppress from being output when logging level is at DEBUG.

Table 6 Application Configuration File Parameters

Name	Type	Default (if not reqd)	Description
defaultIDP	String	null	Default IDP to use when authenticating users.
spAutoGenerate LocalUserId	Boolean	false	If true, new users are assigned an automatically generated unique local identifier, bypassing the activation process. This is equivalent to specifying the F_AUTOGENERATELOCALID flag in the SPAPI.loginUser call.
includeSAML AssertionInProfile	Boolean	false	If true, the SAML Assertion will be included in the SPAPI profile as an XML string under the key "_samlAssertion".
includeSAML SubjectNameIn Profile	Boolean	false	If true, the SAML Assertion Subject Name will be included in the SPAPI profile as three strings under the keys: "_samlSubjectName", "_samlSubjectNameQualifier", and "_samlSubjectNameFormat".

Name	Type	Default (if not reqd)	Description
spEventPlugin	String	null	SPEventPlugin implementation class name. 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 OpenView Select Access, this should be set to "`com.hp.ov.selectfederation.HPSA_SPEventPlugin`".
spDefaultURL	String	Required	The default application URL to send users to following receipt of an unsolicited authentication assertion with no accompanying target URL.
spProxyReturn	Boolean	false	If true, the server will act as a proxy to load the return URL for authenticated users during the login process. This eliminates a user agent redirect, but requires that the return URL is re-written to include any needed session IDs (since cookies will not be available).
signAuthnRequests	Boolean	Required	If true, AuthnRequest messages will be signed.
spFederation Termination NotificationProtocolProfiles	StringList	Required	List of Liberty protocol profile URIs to support for FT at the SP.
spRegisterName IdentifierProtocol Profiles	StringList	Required	List of Liberty protocol profile URIs to support for RNI at the SP.
spSingleLogout ProtocolProfiles	StringList	Required	List of Liberty protocol profile URIs to support for SLO at the SP.
supportLECProfile	Boolean	true	If true, AuthnRequests will be sent using the LECP profile whenever a compatible Liberty-Enabled header is detected.
lecpIDPs	StringList	null	If non-null, the list of IDPs (identified by ProviderID) to include in the AuthnRequestEnvelope IDPList.

Name	Type	Default (if not reqd)	Description
useSSOCookie Writer	Boolean	false	If true, the SSO service will use the configured cookie writer to update the Liberty common domain cookie.
cookieWriter ServiceURL	String	null	The URL of the CookieWriterService to use. Required if useSSOCookieWriter is true.
spAuthTimeout	String	Value of authTimeout	If the IDP sends a **reauthOnOrAfter** which is later than what the SP configuration dictates or for absent values, the SP will use its own, stricter, value. If **spAuthTimeout** is set to 0, the SP uses the IDP provided timestamps as is, even if null (never expire). Hence when Select Federation is being used with Select Access, **spAuthTimeout** should never be set to 0.

Table 7 Authority Configuration File Parameters

Name	Type	Default (if not reqd)	Description
idpAuthnPlugin	String	null	IDPAuthnPlugin implementation class name. If non-null, the class will be instantiated and called to authenticate users and invalidate login sessions. If null, loginURL will be used.
idpAuthnPlugin. characterEncoding	String	“UTF-8”	idpAuthnPlugin.characterEncoding can be set to indicate the character encoding used by a login page. This may be needed if an IDPAuthnPlugin is used (by setting idpAuthnPlugin=..). The default value is "UTF-8" and it is strongly recommended that all login pages use "UTF-8" for forms (such as: <form action="..." accept-charset="utf-8" method="post">.
loginURL	String	null	If idpAuthnPlugin is null, users are redirected to this URL for authentication. The page at this URL must use the IDPAPI to record the user authentication.
logoutURL	String	null	If idpAuthnPlugin is null, and logoutURL is non-null, users are redirected to this URL during logout processing.
consentURL	String	null	If non-null, users are redirected to this URL to provide consent for new federations.

Name	Type	Default (if not reqd)	Description
authTimeout	Time Duration	0	Default timeout for user authentications at the IDP, within a browser session. After this time, the user is required to re-authenticate. A value of 0 disables expiration of user authentications.
reauthMaxAge	TimeDuration	1s	Maximum age of an authentication that can satisfy a forced re-authentication request. Default is 1 second.
authnContext ClassRef	StringList	Required	List of Liberty AuthnContextClassRef URIs supported by the configured idpAuthnPlugin or loginURL.
authnContext StatementRef	StringList	Required	List of Liberty AuthnContextStatementRefs corresponding to the AuthnContextClassRefs listed in authnContextClassRef (in order).
saml2Authn ContextClassRef	StringList	Required	List of SAML 2.0 AuthnContextClassRefURIs supported by the configured idpAuthnPlugin or loginURL.
saml2Authn ContextStatementRef	StringList	Required	List of SAML 2.0 AuthnContextStatementRefs corresponding to the AuthnContextClassRefs listed in saml2AuthnContextClassRef (in order).
samlAuthMethod	StringList	Required	List of SAML 1.1 AuthenticationMethod URIs corresponding to the Liberty AuthnContextClassRefs listed in authnContextClassRef (in order).
rankedAuthn ContextClassRefs	StringList	Required	List of AuthnContextClassRefs in comparison order according to local policy, from weakest to strongest.
rankedAuthn ContextStatement Refs	StringList	Required	List of AuthnContextStatementRefs in comparision order according to local policy, from weakest to strongest.
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.

Name	Type	Default (if not reqd)	Description
dirPlugin_File.filePath	String	Required if dirplugin=com.trustgenix.tfsIDP.util.DirPlugin_FIle	The file path for the properties file used by the file based DirPlugin.
dirPlugin_File.valueSep	String	";"	A configurable value separator that is used to split attribute properties into multiple values. For example, "`user.foo=one;two;three`". The default is semicolon. If valueSep is set to an empty value, then attribute properties will NOT be split into multiple values (each attribute will have only one value).
<attralias>.ldapAttr	String	null	If non-null, the LDAP user attribute that maps to the user's profile attribute.
idpFederationTerminationNotificationProtocolProfiles	StringList	Required	List of Liberty protocol profile URIs to support for FT at the IDP.
idpRegisterNameIdentifierProtocolProfiles	StringList	Required	List of Liberty protocol profile URIs to support for RNI at the IDP.
idpSingleLogoutProtocolProfiles	StringList	Required	List of Liberty protocol profile URIs to support for SLO at the IDP.
idpSingleSignOnProtocolProfiles	StringList	Required	List of protocol profile URIs to support for SSO at the IDP.
idwsfSupportAttributeQuery	Boolean	false	If true, the built-in ID-WSF profile service front-end to the dirPlugin is enabled and advertised via the DiscoveryResourceOffering attribute in ID-FF assertions. The DS is also enabled in read-only mode to advertise the profile service(s).
useSSOCookieWriter	Boolean	false	If true, the SSO service will use the configured cookie writer to update the Liberty common domain cookie.

Name	Type	Default (if not reqd)	Description
cookieWriter ServiceURL	String	null	The URL of the CookieWriterService to use. Required if useSSOCookieWriter is true.
cookieDomain	String	null	If non-null, the CookieWriterService is enabled to write Liberty introduction cookies in the specified domain (for example, ".cot.com").
cookieMaxAge	Integer	0	If zero, the default, introduction cookies are created as session cookies. If greater than zero, introduction cookies are created as persistent cookies with the specified lifetime.
cookieSecure	Boolean	true	If true, introduction cookies are flagged as secure.
samlRequest Auth	StringList	"sign ssl http"	The list of SOAP authentication mechanisms configured for the SAML SOAP service.
samlInclude Audience Restriction Condition	Boolean	false	If true, SAML SSO assertions include an audience restriction condition identifying the intended consumer.
samlIncludeSubjectI P	Boolean	false	If true, SAML SSO assertions include the authenticated user's IP address (as determined by examining the network connection over which the user is authenticated).
samlSupport AttributeQuery	Boolean	false	If true, attribute query requests will be accepted on the SAML SOAP endpoint.
wantAuthn RequestsSigned	Boolean	false	If set to 1, authentication requests from all partners are expected to be signed, irrespective of the partner specific setting for the Application Protocol Policy.
workaroundCookieQ uotes	Boolean	false	Enables a workaround for a bug in Tomcat cookie handling, explicitly adding quotes around the cookie value.

Table 8 DirPlugin_LDAP Plugin Configuration File 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.
ldapAuthentication	String	"simple"	LDAP authentication mode to use, by default, for connections to the directory (see JNDI documentation for possible values.
ldapUserAttr	String	Required	LDAP RDN user attribute to use for username in constructing user DN.
ldapUserBaseDN	String	Required	Base DN to use in constructing user DN from username and ldapUserAttr. User DN looks like `<ldapUserAttr>=<username>,<ldapUserBaseDN>`.
<attralias>.ldap Attr	String	null	If non-null, the LDAP user attribute that maps to the user's profile attribute.
ldapUserObjectClass	String	person	ObjectClass used for performing lookups on the directory. You could change this to some other value, for example, `user`.
ldapSearchSubtree	Boolean	false	If set to "true", the plugin will perform a sub-tree search for users. Hence, you do not need to specify the exact base DN as long as you have this flag enabled

Table 9 DirPlugin ADS Plugin Configuration File Parameters

Name	Type	Default (if not reqd)	Description
ldapURL	String	Required	LDAP URL to use, by default, for connections to the directory.
ldapPrincipal	String	null	LDAP user to use, by default, for connections to the directory. This can be either the userPrincipalName or sAMAccountName of the administrative user (such as Administrator@domain.com, or Administrator).
ldapPassword	String	null	LDAP password to use, by default, for connections to the directory.

Name	Type	Default (if not reqd)	Description
ldapUserAttr	String	null	LDAP RDN user attribute to use for username in constructing user DN. By default, DirPlugin_ADS supports authentication using userPrincipalName and sAMAccountName. This can be changed to some other value such as. cn or telephoneNumber.
ldapUserBaseDN	String	Value of the "defaultNamingContext" of your ADS installation	Base DN to use in constructing user DN from username and ldapUserAttr. User DN looks like <ldapUserAttr>=<username>,<ldapUserBaseDN>. It is recommended to set this value for performance reasons if you have a large/complex Active Directory forest. By default, after installation, it will be set to the rootDSE (defaultNamingContext) of your Active Directory installation.
ldapUserObjectClass	String	person	ObjectClass used for performing lookups on the directory. You could change this to some other value, for example, "user".
<attralias>.ldapAttr	String	null	If non-null, the LDAP user attribute that maps to the user profile attribute.
ldapAuthentication	String	“simple”	LDAP authentication mode to use, by default, for connections to the directory (see JNDI documentation for possible values; "GSSAPI" is supported for Kerberos v4 authentication to AD).
GSSAPI.defaultRealm	String	null	Default realm for GSSAPI Kerberos authentication, for example: TEST-SERVER.HP.COM
GSSAPI.defaultRealmKDC	String	null	The KDC address for this realm. For example: test-server.hp.com:88
DirPlugin_ADS.useSAMAccNameOnly	Boolean	true	This flag is used for backward compatibility. In a new installation of Select Federation, it is set to "0", which allows users to authenticate with `sAMAccountName` or `userPrincipalName` by default. Other formats (such as `cn`) can be supported by setting the appropriate value for `ldapUserAttr`. To enable backward compatible behavior, that is, login with `sAMAccountName` only, you can either set it to "1", or comment it out.

Table 10 HPSA Adapter Configuration File Parameters

Name	Type	Default (if not reqd)	Description
hpsf.debugLevel	String	null	Debugging level for enforcer.
hpsf.enforcerName	String	null	Name of the enforcer used by Select Federation.
hpsf.enforcerPath	String	/selectFederation	The path at which the enforcer is used.
hpsf.serviceURL	String	Automatically computed	The base URL of the server at which the enforcer is running. This is automatically set by select Federation, but can be overridden by this variable.
hpsf.spLogoutURL	String		The URL at an SP to which Select Federation redirects when it receives a logout request from an IDP.
hpsf.ldapServer Type	String		This can be either “ads” for active directory or “sun” for all other.
hpsf.ldapUserAttr	String		The attribute for creating the full path to the user object in the LDAP directory.
hpsf.ldapUserBaseDN	String		The base DN within which the LDAP path will be created.
hpsf.enforcerConf	String	null	Adds a path to a HP OpenView Select Access enforcer config file (`enforcer-servlet.xml`). This is useful to overcome the SA bug in Linux with the default SA enforcer config path.

Table 11 Privacy Manager Configuration File Parameters

Name	Type	Default (if not reqd)	Description
profileDispAttrs	StringList	null	List of profile attributes to display in consent dialogs, if present in request.
<attralias>.disp Name	String		Display name for profile attribute.
userPolicy.services	String	“profile”	Allows use of user-specific policies for the listed services. For listed services, user consent will be required.
userInteraction URL	String	null	Location of interaction redirect service used by ID-WSF WSPs. Should be set to the base URL followed by `/pm/irs`.

Name	Type	Default (if not reqd)	Description
userPolicyData Provider	String	"com. trustgenix.tfs.policy.user. Policy Data Provider_JDBC"	User privacy store type. It can be either "com.trustgenix.tfs.policy.user.PolicyData Provider_JDBC" (for relational databases), "com.trustgenix.tfs.policy.user.PolicyData Provider_Dir" (when storing policies in the configured directory) or "com.trustgenix.tfs.policy.user.PolicyData Provider_File" (for storing policies as XML documents in a file system directory).
PolicyDataProvider_ File.cache	Integer	10	File-based PolicyDataProvider file cache size.
PolicyDataProvider_ File.directoryPath	String	"properties"	Directory where user privacy files will be stored.
PolicyDataProvider_ Dir.policy AttributeName	String	"privacypolicy"	The name of the single valued LDAP attribute that will hold a user privacy policy as a blob.
profile.name	String	Required	The label for the tabbed page that shows user privacy rules for the profile service. The installer sets it to "Personal Profile".
profile.default ConsentRequired	Boolean	true	Reserved for future usage. If set to false, end user consent will not be required for attribute release. Has the same effect as removing "profile" from the value of userPolicy.services.
profile.possible Decisions	StringList	"DENY, GRANT"	The possible decisions that rules about attribute release can state. The installer sets it to "DENY PROMPT GRANT".
profile.showValues InConsentRequest	Boolean	false	If true values of attributes that are about to be released are presented to the end user.
federation.name	String	Required	The label for the tabbed page that shows user privacy rules for the federation consent. The installer sets it to "Single Sign-on".
federation.possibleDecisions	StringList	"DENY, GRANT"	The possible decisions that rules about federation consent can state.
privacy.invalid RuleDeleteDelay	Time Duration	24h	Controls how long rules in user privacy policies that are no longer valid (because of reference to partners that no longer exist) remain in the policy before being removed. Note that rules are only marked as invalid, and only removed when users log in to the Privacy Manager.

spapiconfig.properties

The following table lists and describes the application API configuration file parameters included in the `spapiconfig.properties` file for an SP site:

Table 12 Application API Configuration File Parameters

Name	Type	Default (if not required)	Description
providerId	String	Required	Server's Liberty ProviderID.
providerBaseURL	String	Required	URL for the server's front-channel WAR.
jdbcProvider	String	Required	Database-specific JDBC provider to use, by default, for connections to the database. For example, `com.trustgenix.tfs.JDBCProvider_Oracle`.
jdbcDataSource	String	null	JNDI name of J2EE Data Source to use, by default, for connections to the database. If this option is provided then `jdbcAddr/Driver/User/Password` are ignored.
jdbcAddr	String	null	JDBC address to use, by default, for connections to the database.
jdbcDriver	String	null	JDBC driver to use, by default, for connections to the database.
jdbcUser	String	null	JDBC user to use, by default, for connections to the database.
jdbcPassword	String	null	JDBC password to use, by default, for connections to the database.
tblPrefix	String	"tfs"	Prefix to use, by default, for database tables.
cookieReader ServiceURL	String	null	If non-null, the URL of the CookieReaderService to use.

idpapiconfig.properties

The following table lists and describes the authority API configuration file parameters included in the `idpapiconfig.properties` file for an IDP site:

Table 13 Authority API Configuration File Parameters

Name	Type	Default (if not required)	Description
providerId	String	Required	Server's Liberty ProviderID.
providerBaseURL	String	Required	URL for the server's front-channel WAR.

Name	Type	Default (if not required)	Description
jdbcProvider	String	Required	Database-specific JDBC provider to use, by default, for connections to the database (for example, “com.trustgenix.tfs. JDBCProvider_Oracle”).
jdbcDataSource	String	null	JNDI name of J2EE Data Source to use, by default, for connections to the database. If this option is provided then jdbcAddr/ Driver/User/Password are ignored.
jdbcAddr	String	null	JDBC address to use, by default, for connections to the database.
jdbcDriver	String	null	JDBC driver to use, by default, for connections to the database.
jdbcUser	String	null	JDBC user to use, by default, for connections to the database.
jdbcPassword	String	null	JDBC password to use, by default, for connections to the database.
tblPrefix	String	"tfs"	Prefix to use, by default, for database tables.
timeOutSeconds	Integer	0	If greater than zero, determines the reauthenticateOnOrAfter time in assertions (overriding the value established by authTimeout in server). It is preferable to use the authTimeout parameter in the server. This parameter should only be used to override the authTimeout setting in the server with a shorter time, if needed.
tfsSIDCookie Domain	String	null	If non-null, the cookie domain used for the tfsSID cookie that records the user’s IDP session. Can be used to share the tfsSID cookie between IDP web applications.
cookieReader ServiceURL	String	null	If non-null, the URL of the CookieReaderService to use.
cookieWriter ServiceURL	String	null	If non-null, the URL of the CookieWriterService to use.
cookieReader ReturnPrefixes	String	null	If non-null, the list of prefixes for allowed cookie reader return URLs.

Table 14 ID-WSF DS Configuration File Parameters

Name	Type	Default (if not reqd)	Description
idwsfSupportDS	Boolean	false	If true, the DS is enabled.
idwsfDSSecMechId	StringList	null	List of ID-WSF security mechanism URNs to support on DS endpoint. If null, defaults to urn:liberty:security:2003-08:TLS:X509, if providerBaseSOAPURL starts with https and urn:liberty:security:2003-08:null:X509 otherwise. If the AS is supported, the default security mechanisms will also include urn:liberty:security:2004-04:TLS:Bearer or urn:liberty:security:2004-04:null:Bearer, as appropriate based on providerBaseSOAPURL.
idwsfDSToken Timeout	Time Duration	0	Timeout for credential tokens issued by the DS. A value of 0 causes the value of authTimeout to be used (if authTimeout is also 0, credential tokens issued by the DS do not expire).
idwsfDSAllow UpdatesFrom	StringList	null	If non-null, discovery service updates will only be allowed from the listed SPs (identified by ProviderID).

Table 15 LECP Service Configuration File Parameters

Name	Type	Default (if not reqd)	Description
lecpAllowIDPLoc Prefixes	StringList	null	If non-null, the LECP service will only consider IDPList entries (in a received AuthnRequestEnvelope) with locations that have a match in this list of URL prefixes.
lecpDenyIDPLoc Prefixes	StringList	null	If non-null, the LECP service will ignore IDPList entries (in a received AuthnRequestEnvelope) with locations that have a match in this list of URL prefixes.
lecpDefaultIDPLoc	String	null	The location of the IDP to use by default, when no IDPList is provided.
lecpDefaultIDPLoc_ Liberty11	String	null	The location of the IDP to use by default for Liberty 1.1 requests, when no IDPList is provided. This overrides lecpDefaultIDPLoc for Liberty 1.1 requests.

Name	Type	Default (if not reqd)	Description
lecpDefaultIDPLoc_ Liberty12	String	null	The location of the IDP to use by default for Liberty 1.2 requests, when no IDPList is provided. This overrides lecpDefaultIDPLoc for Liberty 1.2 requests.
lecpStripHeaders IDP	StringList	null	List of headers (received by the LECP service) that should not be forwarded in requests sent to IDPs.
lecpStripHeaders SP	StringList	null	List of headers (received by the LECP service) that should not be forwarded in requests sent to SPs.
lecpSessionHdr	String	"X-LECP Session"	If non-null, identifies the header (received by the LECP service) that should be used to track the user's session. See LECP Service manual for more information.

B Running Apache Derby as a Network Service

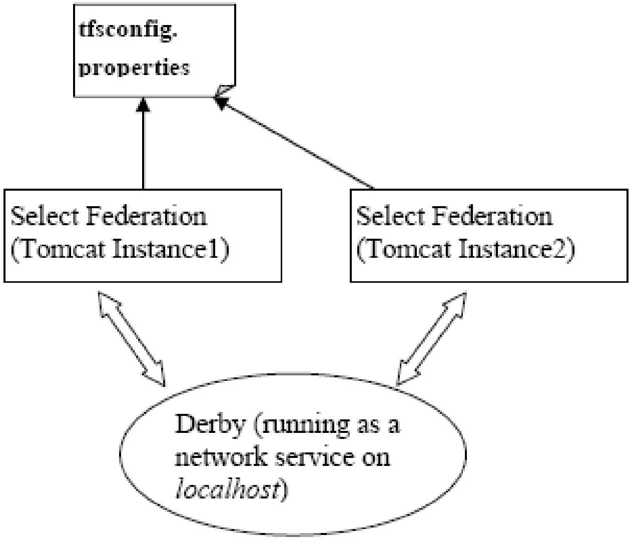
Summary

Select Federation uses Apache Derby as an embedded database. The database used by the system can be selected during the installation (see the “Installing Select Federation” chapter in the *HP OpenView Select Federation Installation Guide)*.

This appendix describes the steps for running Apache Derby as a network service, rather than using it in embedded mode. This is especially useful in production environments, when multiple (possibly replicated) instances of Select Federation need to share a database over the network.

Test Scenario

The following experimental set up shows two instances of Select Federation deployed on Tomcat 4.1. The host platform is WinXP. Apache Derby will run as a network service on the host. The aim is to have both Select Federation instances share the database available on the network.


tfsconfig.
properties
Select Federation (Tomcat Instance1)
Select Federation (Tomcat Instance2)
Derby (running as a network service on *localhost*)

Perform the following tasks to run Apache Derby as a network service:

Task 1: Download the Apache Derby binary

Go to the “Distributions” section on the download page:

http://db.apache.org/derby/releases/release-10.0.2.1.html

Task 2: Get the "jar" files required for the Derby server and client

On the server side, the jar files that are part of the Derby installation package are in /lib:

- `derby.jar`
- `derbynet.jar`

On the client side, the following jar files are available at:

`http://www-128.ibm.com/developerworks/db2/downloads/jcc/`

- `db2jcc.jar`
- `db2jcc_license_c.jar`

Task 3: Set the CLASSPATH

Ensure that the classpath includes these jars. For example, on the server side, you would have to edit your classpath to include the following:

```
C:\<derbyhome>\lib\derby.jar;C:\<derbyhome>\lib\derbytools.jar;C:\<derbyh
ome>\lib\derbynet.jar;%CLASSPATH%
```

The client jars should be copied to the "`/shared/lib`" subdirectory for all Tomcat instances that will be connecting to Derby.

Task 4: Start and shut down Derby through the command line

Startup command:

```
java org.apache.derby.drda.NetworkServerControl start -h localhost
-p 1527
```

Shutdown command:

```
java org.apache.derby.drda.NetworkServerControl shutdown -h localhost -p
1527
```

It is convenient to put these commands in batch files under the Derby home directory (`startup.bat` and `shutdown.bat`). On successfully starting up the Derby server, you should see the following message:

```
Server is ready to accept connections on port 1527.
```

Task 5: Edit the tfsconfig.properties file

Use the following syntax for the JDBC connection parameters when using a Derby network driver.

```
jdbcProvider=com.trustgenix.tfs.JDBCProvider_Derby

jdbcDriver=com.ibm.db2.jcc.DB2Driver

jdbcAddr=jdbc:derby:net://localhost:1527/"c:/ib-cd/
TFSDB":user=APP;password=APP;retrieveMessagesFromServerOnGetMessage=true;
```

Comment out the properties `jdbcUser` and `jdbcPassword`. Note that the address string above is with the database authentication turned off. Refer to the troubleshooting and references section for additional information on the syntax.

Task 6: Modify the Deployment Descriptor of the Second Select Federation Instance

Edit the web.xml of the “tfs-internal” web-app of the second Select Federation instance so that it points to the tfsconfig.properties file of the first instance. This file can be found under webapps\tfs-internal\WEB-INF in the Tomcat installation directory.

```
<web-app>

…………..

<env-entry>

 <env-entry-name>com.trustgenix.tfs.propFile</env-entry-name>

 <env-entry-value>../<Select Federation 1st

 instance>/conf/tfsconfig.properties</env-entry-value>

 <env-entry-type>java.lang.String</env-entry-type>

</env-entry>

…………….

</web-app>
```

Task 7: Start the Derby server and the Select Federation Instances

Use the startup scripts to start Derby and both the Tomcat instances. Navigate to the admin console (tfs-internal) and login. This completes the setup.

Additional References

1. The Derby documentation is fairly good. Several manuals are available (look under doc/manuals/index.html in your Derby installation). These manuals are also available online.
2. *Accessing the Network Server using the DB2 Universal Driver* (contains syntactical notations and examples for accessing the network server).
3. Note on using *authentication and encryption* with Derby.
4. Setting Derby *system-wide properties*.
5. IBM *Cloudscape documentation*.

C Troubleshooting

This chapter provides troubleshooting information in the following sections:

- Troubleshooting Select Federation
- Troubleshooting Filters
- Troubleshooting Integrated Windows Authentication
- Using HP OpenView Self-Healing Services

Troubleshooting Select Federation

Use the Select Federation application server log file to view logged messages. There could be some exceptions caused due to incorrect syntax or configuration. Here are some common problems:

Why do I get the error "schema does not exist"?

The current schema for any connection defaults to a schema corresponding to the user name. If no user name is supplied then the user name (and hence current schema) defaults to APP. However, even though the current schema is set to the user name, that schema may not exist. A schema is only created by CREATE SCHEMA or creating an object (such as a table) in that schema (this is implicit schema creation). **The one exception to this is the APP schema, which is always created**, though applications should not depend on that. This is the reason why the userid and password have been set to "APP".

"Userid length, 0, is not allowed"

The DB2 driver *requires* that a user ID be specified when establishing a connection. Not specifying a user ID or leaving it blank will result in this error.

The Select Federation installer reported an issue with the directory server SSL certificate. How do I fix this?

There are several possibilities that might have generated this warning, such as, a name mismatch or an expired certificate. However, the most common scenario is that of an "untrusted certificate", when the CA that issued this certificate is unknown to the Select Federation Java trust store. You could use a simple utility like **keytool** to install the CA's certificate in your Java cacerts file. In the case of the built-in application server, the cacerts file would be located in the `_jvm\lib\security` sub-directory of the install location. If you installed Select Federation on an existing application server in your environment, locate the cacerts file for the JDK that is being used by the application server. You can then issue a command (example given below) for importing your CA cert:

```
% keytool -import -trustcacerts -alias MyCA -file <CA certificate file>
-keystore <path to your Java installation>/lib/security/cacerts -storepass
<default value is "changeit">
```

In Select Access integrated mode, browser is stuck after federated login at IDP

This can happen when you are using Internet Explorer, and Select Access and Select Federation are running on different ports of the same computer. This happens due to a known issue with the Microsoft Internet Explorer browser which causes wrong URLs to be generated during the redirect from Select Federation to Select Access.

Troubleshooting Filters

To troubleshoot effectively, be sure to configure a log file location, and enable debug logging. For configuration guidelines, see How to Configure the IIS Filter on page 92 and How to Configure the Apache Filter on page 103.

If your issue is not covered in this chapter, be sure your log file is available when reporting the issue to Support (see Support on page 4 for information).

Problems

Problem

The IIS filter configuration interface is not seen after running the installation script.

Solution

First, check that you closed and re-opened the IIS administrative console. Interface changes do not take effect until you do so.

Then, if that did not work, uninstall and re-run the install script.

Error Messages

Error Message

```
Got unexpected Exception...please try again.
“Exception: com.trustgenix.tfs.TFSException: SSO Failed, probably because of
a missing cookie"
```

Problem

The cookie names specified in the Select Federation SP instance configuration file and the filter configuration should be the same.

Solution

Do the following:

1. Open the `tfsconfig.properties` file of the SP instance and search for `filterSupport.cookieName`.
2. Confirm that the value of this attribute is the same as the cookie name being set as part of the Select Federation filter configuration.

➤ A good way to debug issues related to cookies is to change the browser setting so as to **prompt** before cookies are set.

Error Message

```
Missing IDP
```

Solution

To test the Select Federation filter, you need to have configured an IDP that has been set up to validate users against an Access Management system (such as HP OpenView Select Access) or a Directory Server (such as Active Directory).

Error Message

```
Java Index OutofBounds exception when accessing protected resource
```

Problem

This can happen if you tried to access a URL without specifying the protected resource. For the filter to function correctly, the resource that is being accessed should have three components:

- protocol
- server name
- resource name

Solution

When accessing the protected resource, be sure to specify the full URL containing all three components. For example:

http://myserver.com/myprotectedresource

Error Message

```
http://localhost/SFFilterConfigure.html is "not found" [IIS filter only]
```

Problem

This could happen if the Select Federation filter is not loaded.

Solution

- Make sure that the filter has been added to the Web service extension list, and that the status should be set to "Allowed".
- The machine should have been rebooted after modifying the system path.

Troubleshooting Integrated Windows Authentication

This section lists the most common errors that would show up due to a broken workflow. It provides scenarios and suggest solutions to help you find the root of the problem.

Error Messages

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.

Troubleshooting GSSAPI

Following are some potential issues you might run into when using the built-in Active Directory plugin "DirPlugin_ADS" with the Kerberos GSSAPI bind mechanism. These are exceptions you will see in the Select Federation application server log file.

Error Message

```
2006-10-17 00:33:34,030|http-7080-Processor23|ERROR] -
com.trustgenix.tfs.ldap.LDAPv3Provider:
javax.security.auth.login.LoginException: KrbException:: Pre-authentication
information was invalid (24) - Preauthentication failed
```

Possible Causes and Solutions

Cause 1: The password entered is incorrect.

Solution 1: Verify the password.

Cause 2: If you are using the keytab to get the key (by setting the useKeyTab option to true in the Krb5LoginModule entry in the JAAS login configuration file), then the key might have changed since you updated the keytab.

Solution 2: Consult your Kerberos documentation to generate a new keytab and use that keytab.

Cause 3: Clock skew - If the time on the KDC and on the client differ significantly (typically 5 minutes), this error can be returned.

Solution 3: Synchronize the clocks (or have a system administrator do so).

Error Message

```
[Krb5LoginModule] authentication failed
Could not load configuration file c:\winnt\krb5.ini (The system cannot find
the file specified)
com.trustgenix.tfs.DataProviderException: Could not load configuration file
c:\winnt\krb5.ini
```

Solution

Create the "kr5.ini" under "c:\winnt" (on Windows) with contents as given in the GSSAPI section. Note that this is an alternative to specifying the following GSSAPI values in tfsconfig.properties:

```
#GSSAPI.defaultRealm=
```

```
#GSSAPI.defaultRealmKDC=
```

Error Message

```
com.trustgenix.tfs.DataProviderException: Cannot get kdc for realm
hpsfov-espoo.fin.hp.com
```

Solution

Make sure that the supplied user credentials are correct, and that the administrative username corresponds to the value of the sAMAccountName attribute of the administrative user.

Problem

I don't see detailed DEBUG information in the logs pertaining to GSSAPI.

Solution

It is possible to enable detailed logging for GSSAPI by setting the debug flag to “true” in the gss.cfg file, which is located in the “conf” subdirectory.

Using HP OpenView Self-Healing Services

The Self-Healing Services enables more accurate, timely problem diagnosis and correction within OpenView Support and Services. The process of investigating a software fault consists of three distinct phases: data collection, problem analysis, and reporting.

Select Federation provides a Data Collector utility, which collects log files, configuration data, and any other information that would be useful in debugging a crash. You can launch the Data Collector in command line mode after you experience a problem. The information is transferred to HP support, who diagnoses the problem and contacts you with a solution.

Data Collector

The Data Collector is a standalone utility that you run from the command line. It is shipped with the Select Federation install. The Data Collector collects data when prompted, but does not provide any kind of evaluation or self-diagnosis.

The data being collected for Select Federation includes the following files:

- `installshield_log.txt`
- `tfsconfig.properties`
- `idapiconfig.properties`
- `spapiconfig.properties`

- `version.txt`
- `sf-URLs.txt`
- Application server log file (this is collected only when the collector is run from the command line and the log file name is passed as a parameter to the collector)

Running Data Collector From the Command Line

Scripts are available for Windows and UNIX platforms to run the Select Federation Data Collector from the command line. The output of the collector is stored in `<SF-Install-Dir>/tools/shs/out`. This directory contains the data collected as well as a `summary.xml` and a `collector.log` file.

When running the data collector from the command line you need to manually submit all these files to HP Support. The data collector takes a command line parameter which can be used to specify the application server log file that needs to be created.

Be sure there is no sensitive information in this file before asking the data collector to gather its contents.

To specify the application server log file name you need to pass `-app_log` followed by the file name as a parameter to the collector. When no parameter is specified to the collector the file list mentioned above is collected from your installation.

For Windows

At a DOS command prompt enter:

```
cd <SF-Install-Dir>\tools\shs
run-data-collector.bat
```

Following is an example of running collector using a Windows command prompt without specifying the Application Server Log file or the Application server log file name.

```
C:\test-area\idp\tools\shs>run-data-collector.bat

C:\test-area\idp\tools\shs>cd out

C:\test-area\idp\tools\shes\out>ls
SLCTFED_000.txt  SLCTFED_002.txt  SLCTFED_004.txt  collector.log
SLCTFED_001.txt   SLCTFED_003.txt  SLCTFED_005.txt   summary.xml

C:\test-area\idp\tools\shes\out>cd ..

C:\test-area\idp\tools\shes\out>run-data-collector.bat -app_log c:/test-area/idp/
logs/catalina.2006-07-21.log

C:\test-area\idp\tools\shs\out>cd out

C:\test-area\idp\tools\shes\out>ls
SLCTFED_000.txt  SLCTFED_002.txt  SLCTFED_004.txt  SLCTFED_006.txt  summary.xml
SLCTFED_001.txt  SLCTFED_003.txt   SLCTFED_005.txt   collector.log

C:\test-area\idp\tools\shes\out>
```

For Unix

At the UNIX prompt, enter:

```
cd <SF-Install-Dir>/tools/shs
./run-data-collector.sh
```

Following is an example of running collector on UNIX without specifying the Application Server Log file or the Application server log file name.

```
bash-3.00# ./run-data-collector.sh
bash-3.00# cd out
bash-3.00# ls
SLCTFED_000.txt  SLCTFED_001.txt  SLCTFED_002.txt  SLCTFED_003.txt  SLCTFED_004.txt
SLCTFED_005.txt  collector.log  summary.xml
bash-3.00# cd ..
bash-3.00# ./run-data-collector.sh -app_log /opt/OV/SelectFederation/logs/
catalina.out
bash-3.00# cd out
bash-3.00# ls
SLCTFED_000.txt  SLCTFED_001.txt  SLCTFED_002.txt  SLCTFED_003.txt  SLCTFED_004.txt
SLCTFED_005.txt  SLCTFED_006.txt  collector.log  summary.xm
bash-3.00#
```

Contacting Support

If the troubleshooting information in this appendix does not resolve your issue, do the following:

1. Run the “Self Healing Services” tool (see Using HP OpenView Self-Healing Services on page 214).
2. Provide the following additional information when you contact the Select Federation support team:
   - Exact description of the use case and problem you are facing.
   - Select Federation application server log file (with DEBUG logging enabled).
   - Product version.
   - Platform, database, and LDAP server.
   - Federation protocol being used (Liberty 1.X, / SAML 1.X / SAML 2.0).
   - Name and version of Application server on which Select Federation is running (Built-in, WebLogic, WebSphere).
   - JDK major and minor version.
   - Federation software that your partner is using (if it is from a different vendor).
   - If you are an IDP, how are you authenticating your users? Is it through one of the built-in directory or authentication plugins offered by Select Federation, or are you using a custom plugin?
   - Have you integrated Select Federation with any access management system (such as Select Access). If yes, provide the product name and version.

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.

ADFS (WS-Federation 1.0)

Active Directory Federation Services (ADFS) is a feature of Microsoft Windows 2003 Server R2. ADFS allows a federation with Active Directory-based users, by using the WS-Federation 1.0 protocol.

Administrator

An identity with full permission to manage Select Federation.

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 Site Role

An application site (also called a SAML Consumer or 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.

ASP

Microsoft Active Server Pages log users in by invoking the IDP-FSS over a secure channel.

Attribute

One or more characteristics that are part of an identity profile. Attributes are name/value pairs with a type that is assigned a 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 a SAML Producer or 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. Authentication is a prerequisite for authorization. See Access Control and Authentication.

CA

Certificate Authority

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).

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.

ID-WSF

Liberty Identity Web Services Framework security mechanism.

IDP

An Identity Provider or IDP is an 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.

IDP-FSS

IDP filter-support service, which is 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.

IIS

The Internet Information Server (IIS) is the web server that is bundled with Windows 2003 Server.

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.

Keystore

A keystore is 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 (Lightweight Directory Access Protocol)

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.

LECP

Liberty Enabled Client/Proxy Service.

MMC

Microsoft Management Console, used to set up server authentication and to import the `pkcs` / `pfx` format file into your local store on the IIS machine.

NTLM

NT LAN Manager [web definition: is a challenge/response form of authentication that was the default network authentication protocol in Windows NT 4.0.]

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).

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.

Presence Service

A service that informs the WSC if a user is online, available, and so on.

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. See Delegated Administrator.

SOAP

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

SP

A Service Provider (SP) is an application that allows authenticated access based on an authentication performed by an IDP using a federated identity protocol such as Liberty or SAML.

SSL

Secure Sockets Layer handshake protocol, which supports server and client authentication.

SSO

Single Sign-On session/authentication process that permits a user to enter one set of credentials (name and password) to access multiple applications. A Web SSO is a specialized SSO system for web applications.

SAML

Security Assertion Markup Language protocol.

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.

WSC

A Web Service Consumer (WSC) is 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.

WSP

A Web Service Provider (WSP) is a web service application that services requests it receives based on XML and typically SOAP-based communication.

Index