PingFederate™ 4.2

WebLogic Integration Kit

Version 1.1

User Guide

PingIdentity™

Part Number 3007-129
Version 1.1
December, 2006
Ping Identity Corporation
1099 18th Street, Suite 2950
Denver, CO 80202
U.S.A.

Phone: 877.898.2905 (+1 303.468.2882 outside North America)
Fax: 303.468.2909
Web Site: http://www.pingidentity.com

Trademarks

Ping Identity and PingFederate are trademarks of Ping Identity Corporation.

All other trademarks or registered trademarks are the properties of their respective owners.

Disclaimer

This document is provided for informational purposes only, and the information herein is subject to change without notice. Ping Identity Corporation does not provide any warranties and specifically disclaims any liability in connection with this document.

Contents

Introduction **4**
System Requirements **6**
ZIP Manifest **6**
Known Issues **6**
Installing and Testing the Identity Asserter **7**
Setup PingFederate and QuickStart Sample Application 7
WebLogic Initial Setup 7
WebLogic Identity Asserter Installation 8
WebLogic Identity Asserter Configuration 8
WebLogic SP Application Setup 9
Testing 10

Introduction

The PingFederate 4 *WebLogic Integration Kit* adds a new Service Provider (SP) integration option to PingFederate.

The WebLogic Integration Kit consists of two parts:

- the Standard Adapter, which runs within the PingFederate server;
- the "PingFederate Identity Asserter for WebLogic", which resides with the application server.

The kit uses a proprietary, secure token format (`PFTOKEN`) to transfer the attributes between the PingFederate server and the WebLogic server.

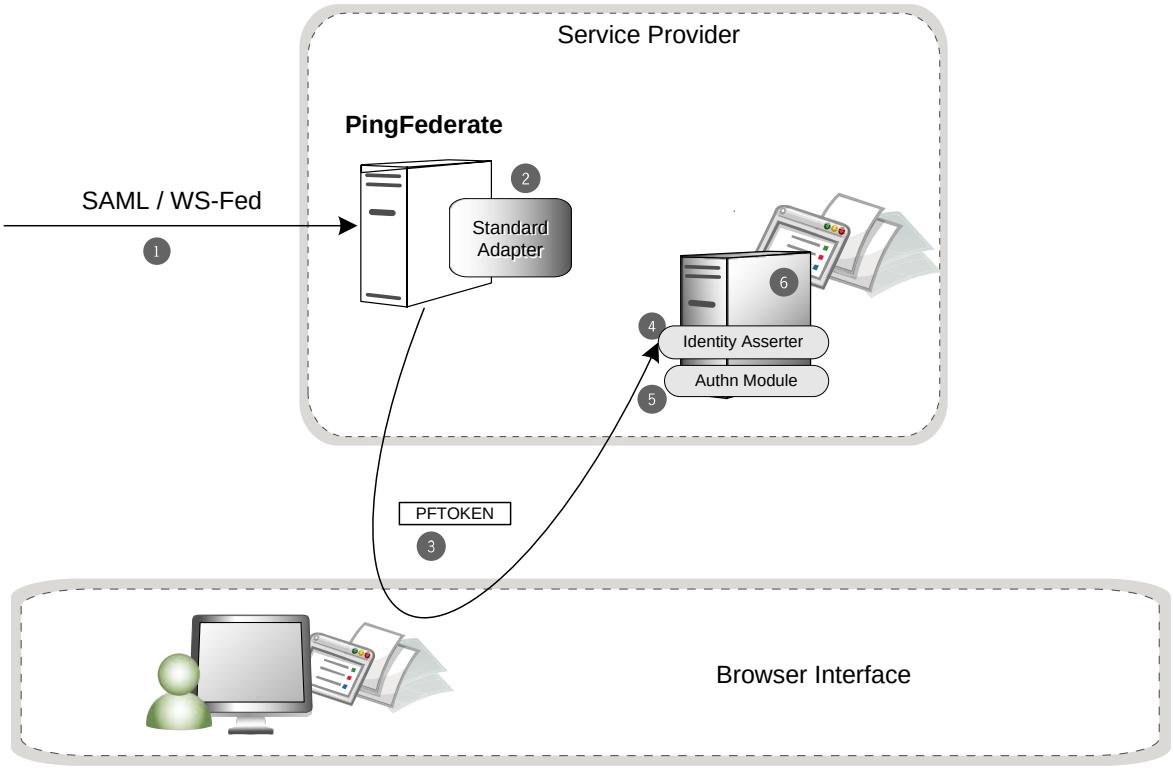
> **Note:** The Standard Adapter is bundled with the PingFederate installation. For details, see the "Standard Adapter Configuration" appendix in the PingFederate 4 *Administrator's Manual*.

The integration kit leverages the Standard Adapter, which is packaged with the PingFederate 4.x server. It uses BEA's Security Service Provider Interface (SSPI) to implement an identity asserter that is then used for perimeter authentication by WebLogic domain.

A WebLogic Identity Assertion provider is a specific form of authentication provider that allows users or system processes to assert their identity using tokens (that is, "perimeter authentication"). Identity Assertion providers enable perimeter authentication and support single sign-on. For more information, refer to the WebLogic server documentation (http://edocs.bea.com).

1. To integrate with WebLogic on the IdP side, use the Java Integration Kit. (http://www.pingidentity.com/products/integration)

The following figure shows the basic scenario in which the PingFederate server leverages the Standard Adapter and the Identity Asserter to allow SSO to a WebLogic domain:



1. PingFederate 4 server receives a SAML / WS-Federation assertion.
2. The PingFederate SP server parses the SAML assertion and passes the user attributes to the Standard SP Adapter. The Adapter encrypts the data internally and generates a `PFTOKEN`.

NOTE: Optionally, the PingFederate server can be configured to look up additional attributes from data stores and add them to the attributes received in the IdP's assertion. In many cases, the SP may want to persist an account and pass internal attributes for profiling or other reasons. (See the *Administrator's Manual* for more information.)

---

3. A request containing the `PFTOKEN` is redirected to the SP application. Additional attributes can be configured to transfer as part of the 'Request Header" or as "Cookies".
4. Ping Federate Identity Asserter for WebLogic is invoked. It retrieves the username from `PFTOKEN`.
5. The configured Authn Module with WebLogic is invoked and validates the username extracted in the previous step, thereby creating a valid `Principal`.
6. A local security context is created and the user has access to the protected resource.

System Requirements

The following software must be installed in order to implement the Java Integration Kit:

- PingFederate 4 server
- The J2SE Java Runtime Environment 1.4.2 or later for the agent side
- WebLogic Server 9.2

Note: WebLogic Server 9.1 was tested and several issues were discovered. Version 9.2 is recommended for use with PingFederate 4.

ZIP Manifest

The distribution ZIP file for the WebLogic Integration Kit contains the following:

- `/docs` – contains additional documentation:
  - `legal.pdf` – copyrights and license information
  - `Weblogic_Integration_Kit_Qualification_Statement.doc` – testing and platform information
  - `Weblogic_Integration_Kit_User_Guide.pdf` – this document
- `/dist` – contains libraries needed to run the adapter:
  - `pf4-pftoken-adapter-1.1.jar` – the Standard Adapter `JAR` file

Note: The Standard Adapter is bundled with the PingFederate 4 installation. Verify that you have the latest version of the jar file in `<pf_install_dir>\pingfederate\server\default\deploy`.

  - `commons-codec-1.3.jar` – from apache for common encoding/decoding operations
  - `pf4-pftoken-agent-1.1.jar` – the Agent Toolkit for Java (supports JDK 1.4.x and JDK 1.5.x)
  - pf4-identityasserter-1.1.jar – PingFederate Identity Asserter for WebLogic
- `/test` – contains the test setup
  - `webapps/wlsapp` – the sample web application

Known Issues

- The current version of the adapter doesn't support Single Logout (SLO) for WebLogic domain.
- A persistent cookie (`PFTOKEN_WLSIDENTITY`) is used to transfer user information between the PingFederate server and the Weblogic domain. The default `maxAge` for the cookie as specified through the PingFederate console is '300 seconds'. If the user closes the browser/application, opens a new browser within the `maxAge` interval and navigates to the WebLogic hosted

application, the browser will find the cookie again. It will then present the cookie to the WebLogic app server, and that server will then assert the identity and therefore log in the user again. To mitigate this behavior, you can set the `maxAge` value to a lower value through PingFederate Admin console.

See the *Qualification Statement* in the `/docs` directory for additional information.

Installing and Testing the Identity Asserter

Perform the following steps in sequence to install the PingFederate Identity Asserter for WebLogic 9.2:

Setup PingFederate and QuickStart Sample Application

In this section discusses how to setup PingFederate and the QuickStart sample application that is shipped with the product. Setting up QuickStart is not required for this integration, but it ensures that the basic issues regarding PingFederate configuration are resolved.

1. Download PingFederate 4.0 and follow the *Quick Start Guide* to set up the IdP/SP SSO scenario.
2. Verify that the SSO is working for the PingFederate IdP/SP Quick Start Sample Application.
3. Once the basic SSO is working with the Quick Start Sample application, make the following changes to update the Quick Start setup to test the WebLogic integration:
    a. From the main menu of the PingFederate Administration Console, click **Local Settings** and under **SP Events**, update **SSO Success URL** to http://hostname:7001/wlssample/secure.jsp, where `<hostname>` is the name of the machine where the WebLogic server is running.
    b. From the main menu of PingFederate Administration Console, click **SP Adapters**.
    c. Select the adapter configured while configuring the QuickStart.
    d. In the Adapter configuration screen, change the value of **PFTOKEN holder name** to `PFTOKEN_WLSIDENTITY`.
    e. Change the value of **Transfer Method** to **Cookie.**
    f. Change the Password. You will enter the same password in WebLogic console when configuring the identity asserter.

WebLogic Initial Setup

In this section, setup a WebLogic Server instance and configure a user /group in the embedded LDAP server. Once the basic setup (discussed above) is up and running, you can change the users/groups/roles as required by your environment. Instead of using the embedded LDAP Server, you can also use any external directory server that is supported by WebLogic. The user/group as specified in this section is used by the web application (`wlssample`) that is part of this distribution.

1. Download BEA WebLogic 9.2 Server (http://commerce.bea.com).
2. Create a WebLogic Domain. (You can use BEA's Domain Configuration Wizard to create one.) For the steps in this *Guide*, we will assume that the WebLogic Server is running at http://localhost:7001 and the domain name is `mydomain`.
3. Start the WebLogic Server and access the console at http://localhost:7001/console.

4. Use the navigation menu and select `mydomain -> Security Realms ->myrealm -> Users and Groups -> Groups`.
5. Click "`New`" and enter '`PingIdentity`' as the group name and click **OK**. Leave the **Provider** as **DefaultAuthenticator**.
6. Use the navigation menu and select **mydomain → Security Realms → myrealm → Users and Groups → Users**.
7. Click **New** and enter name=`joe`, password=`password` and click **OK**. Leave the **Provider** as **DefaultAuthenticator**.
8. Select the user `joe` again and add it to the group `PingIdentity` by clicking the tab **Group** and moving `PingIdentity` from `Available` to `Chosen`. (The group '`PingIdentity`' is specified in the bundled web application as the group that is allowed to access protected resources.)
9. Shutdown the WebLogic application server.

WebLogic Identity Asserter Installation

In this section, we discuss installing the Identity Asserter (part of this distribution) to the WebLogic Server. This can be done using either of the following approaches:

Option 1

Copy the following files to the `DOMAIN_DIR/lib`, where <`DOMAIN_DIR`> represents the root directory of your domain, for example `mydomain`.

- `pf4-identityasserter-1.1.jar`
- `commons-codec-1.3.jar`
- `pf4-pftoken-agent-1.1.jar`

This will allow for PingFederate Identity Asserter to be available for configuration though the WebLogic console.

Option 2

You can also install the identity asserter using the following steps:

1. Copy the file `pf4-identityasserter-1.1.jar` to the following directory: `<BEA_HOME>\weblogic91\server\lib\mbeantypes`, where `<BEA_HOME>` is the directory where BEA software is installed.
2. Copy the following jar files to your startup classpath of the WebLogic Server instance.
   - `pf4-pftoken-agent-1.1.jar`
   - `commons-codec-1.3.jar`

WebLogic Identity Asserter Configuration

This section discusses how to configure the Ping Federate Identity Asserter using the WebLogic Console. Once configured, the WebLogic server with invoke it for every new request. The order of various security providers can be configured through the WebLogic console. Sort the order and ensure that this identity asserter is the first one in the providers list.

1. Start the weblogic server using `startWebLogic.cmd`. Access the console at http://localhost:7001/console .

2. Through the navigation menu, Select **mydomain → Security-Realms → myrealm → Providers → Authentication**.
3. Click the **Lock & Edit** button in the Change Center to activate the buttons on this page.
4. Click **New.** The "Create a new Authentication Provider" screen is displayed.
5. Enter Name=`PingFed Identity Asserter`. Choose Type=`PingFedIdentityAsserter` and click **OK**.
6. Select `PingFedIdentityAsserter` again from the list and click on the tab **Provider Specific**. Enter the same password you entered while configuring the Standard Adapter in PingFederate console.
7. Click **Activate Changes** in the **Change Center**.
8. Shutdown the WebLogic Server.

Note: There is a known bug in the WebLogic Server 9.2. The '`Base64CodingRequired`' Flag as set in the WebLogic Console isn't effective. Make the following change manually in the `config.xml` to get around the issue.

Look for the following line (under element `authentication-provider`):
`<sec:name>PingFedIdentityAsserter</sec:name> and`

And add the following line under it:

```
<sec:base64-decoding-required>false</sec:base64-decoding-required>
```

`CR257702`

Contact BEA support with this CR number to get more details or patch.

WebLogic SP Application Setup

In this step, we deploy the web application that is shipped as part of this distribution. The security settings in the `web.xml` file have been set to use `CLIENT-CERT` and the page `secure.jsp` is secured to only allow access to the selected group/role.

The steps listed here are only recommendations. If you have preferred steps for deploying web applications to WebLogic Server, you can use them instead.

1. Copy the directory '`wlssample` from the attached zip to the following directory `<DOMAIN_DIR>/autodeploy`, where `<DOMAIN_DIR>` represents the root directory of the WebLogic domain; `mydomain`, for example.
2. Start the WebLogic server.
3. Access the web application by pointing the browser to the following location: http://locahost:7001/wlssample/secure.jsp. You should get a 401 (Unauthorized Access). This will validate the resource is protected.

Testing

This section discusses how to test the application. Use the Quickstart IdP Application for the IdP side and use the application shipped with this distribution for the SP side.

1. Access the IdP web application by pointing the browser to IdP URL as configured in Section A.
2. Login using `joe/test`.
3. Initiate SSO by clicking 'IdP-initiated SSO to Sample_SP'. The request will be redirected to the SP Application `secure.jsp` page and the principal/subject information will be displayed on the screen.