

Identity Bridge Service Provider Edition User's Guide

**Lana Frost
Nicolas Philippe**

December 20, 2013

Copyright © 2013 ForgeRock AS

Abstract

Guide to installing and configuring Identity Bridge Service Provider Edition.



This work is licensed under the [Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License](http://creativecommons.org/licenses/by-nc-nd/3.0/).

To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA. Trademarks are the property of their respective owners.

UNLESS OTHERWISE MUTUALLY AGREED BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

DejaVu Fonts

Bitstream Vera Fonts Copyright

Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Bitstream" or the word "Vera".

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Bitstream Vera" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of Gnome, the Gnome Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the Gnome Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

Arev Fonts Copyright

Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the modifications to the Bitstream Vera Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Tavmjong Bah" or the word "Arev".

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Tavmjong Bah Arev" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL TAVMJONG BAH BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the name of Tavmjong Bah shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from Tavmjong Bah. For further information, contact: tavmjong @ free . fr.

Table of Contents

Preface	v
1. Overview of an Identity Bridge Deployment	1
2. Before You Install	5
3. Getting Identity Bridge Up and Running	7
4. Configuring Connections Between Identity Bridge, Active Directory, and OpenDJ	17
5. Mapping Data Between Active Directory and OpenDJ	25
6. Data Synchronization and User Association Management	29
7. Configuring Single Sign-On	37
8. Configuring Identity Bridge for Integrated Windows Authentication	49
9. Customizing the Identity Bridge Interface	61
10. Securing an Identity Bridge Deployment	63
11. Installing an Alternative Repository	67
12. Deploying Identity Bridge for High Availability	71
13. Advanced Configuration	77
14. Troubleshooting an Identity Bridge Installation	83
Identity Bridge Glossary	89
Index	91

DRAFT

Preface

This guide shows you how to install, configure, and manage Identity Bridge.

1. Who Should Use this Guide

This guide is written for administrators of Identity Bridge and covers the install, configuration, and removal procedures that you theoretically perform only once per version. This guide also covers the configuration and management of the synchronization mechanism that ensures consistency across two disparate data stores.

The Identity Bridge software is based on the OpenIDM and OpenAM products. For a deeper understanding of how the product works, you can have a look at the OpenIDM and OpenAM documentation, although such information is not required for basic installation, configuration, and management of Identity Bridge.

2. Formatting Conventions

Most examples in the documentation are created on GNU/Linux or Mac OS X. Where it is helpful to make a distinction between operating environments, examples for UNIX, GNU/Linux, Mac OS X, and so forth are labeled (UNIX). Mac OS X specific examples can be labeled (Mac OS X). Examples for Microsoft Windows can be labeled (Windows). To avoid repetition, however, file system directory names are often given only in UNIX format as in `/path/to/server`, even if the text applies to `C:\path\to\server` as well.

Absolute path names usually begin with the placeholder `/path/to/`. This path might translate to `/opt/`, `C:\Program Files\`, or somewhere else on your system.

Command line, terminal sessions are formatted as follows.

```
$ echo $JAVA_HOME
/path/to/jdk
```

Program listings are formatted as follows.

```
class Test {
    public static void main(String [] args) {
        System.out.println("This is a program listing.");
    }
}
```

DRAFT

Chapter 1. Overview of an Identity Bridge Deployment

Identity Bridge enables you to upload user data from your enterprise data store (Active Directory) to your application data store, and automatically to synchronize this data when user entries are added, changed, or removed. In addition, Identity Bridge enables single sign-on (SSO) to resources, using the Security Assertion Markup Language (SAML).

This sample Identity Bridge deployment uses a scripted REST connector to provision and synchronize data from Active Directory to an OpenDJ LDAP data store. You can replace the connector, and the data store, depending on the requirements of your organization. The sample deployment uses OpenAM as the Service Provider to demonstrate the SSO capability.

1.1. Overview of the Identity Bridge Architecture

Identity Bridge includes a browser-based user interface, and is installed “on premises”, inside your DMZ. A customizable UI wizard enables you to configure data synchronization from your Active Directory server to another data store, in this case OpenDJ.

When Identity Bridge has been installed and configured, access to your applications (in this case OpenAM) can be configured to go through Identity Bridge.

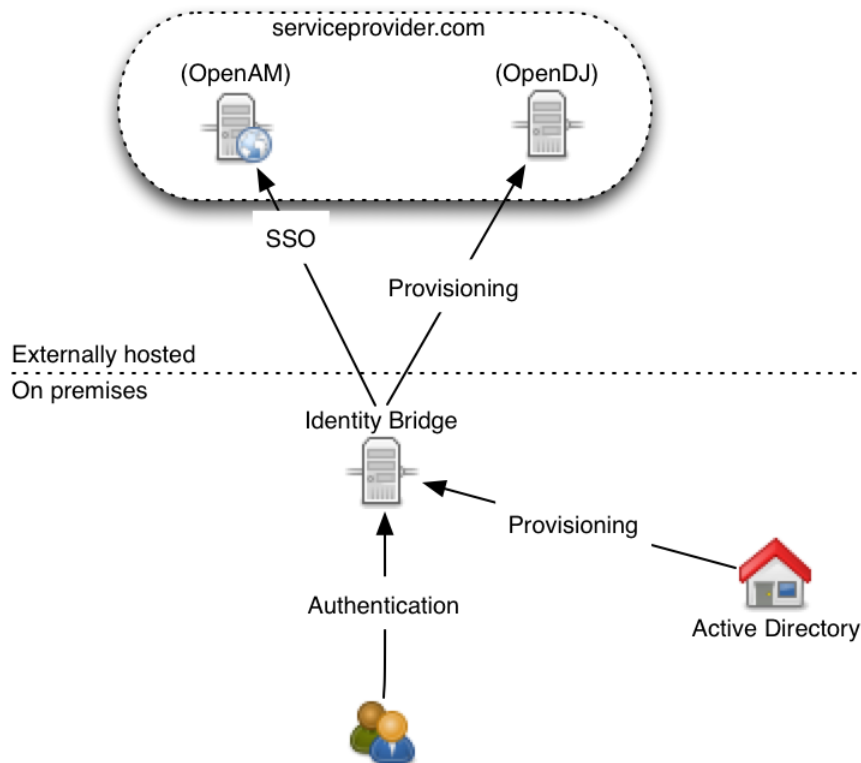
Although Identity Bridge manages primarily user data across disparate data stores, users and passwords are generally not stored in Identity Bridge itself. Administrative access to Identity Bridge relies on the credentials of administration users in Active Directory.

When an administrative user logs into Identity Bridge (at the URL `https://hostname.domain:8443/admin/`), he is able to configure, manage and monitor data synchronization between Active Directory and the application data store. If single sign-on has been configured, a regular user can log into Identity Bridge (at the URL `https://hostname.domain:8443/connect/`), and be routed directly to the required resources, via SAML.

The session for the administration UI is shared with the user UI. Therefore, when an administrator is logged into Identity Bridge, and logs into a resource that is protected by Identity Bridge, he does so without entering additional authentication details.

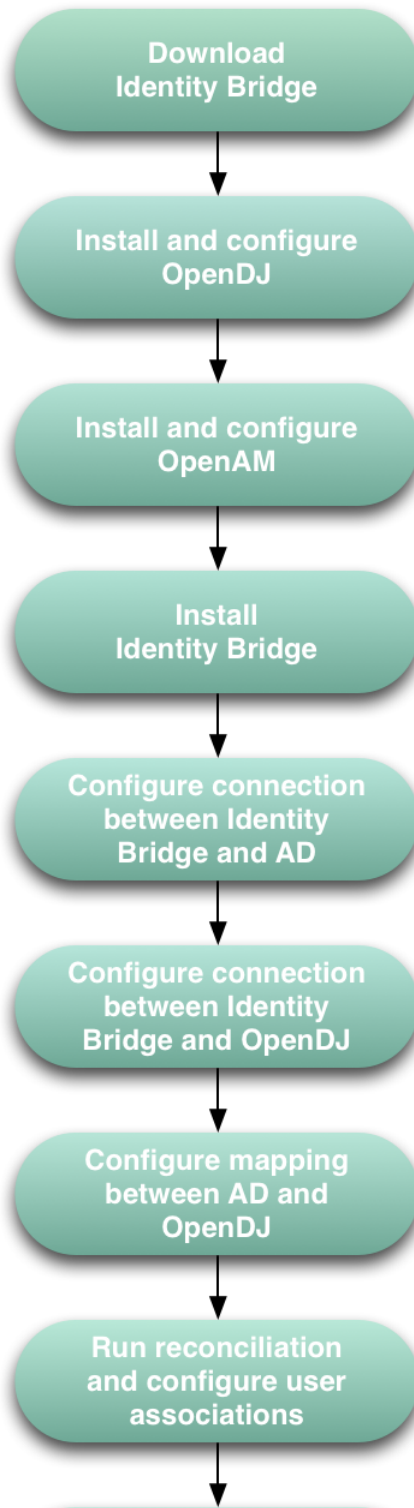
By default, access to Identity Bridge is controlled with forms-based authentication. Users of Identity Bridge provide the login credentials of their Active Directory account when they log in. You can configure Identity Bridge for Integrated Windows Authentication (IWA) in addition to forms-based authentication. For more information, see [Configuring Identity Bridge for Integrated Windows Authentication](#). For more information about configuring single sign-on (SSO), see [Configuring Single Sign-On](#).

The following figure provides a high-level overview of the Identity Bridge components.



1.2. Outline of the Setup Process

Setting up Identity Bridge involves the configuration of multiple systems. The following flowchart provides a high-level overview of what happens between these systems during the setup process. Each step is discussed in more detail in the rest of this guide.



Chapter 2. Before You Install

This chapter covers software and hardware prerequisites for installing and running the Identity Bridge software.

2.1. Java Requirements

Make sure you have an appropriate version of Java installed. Identity Bridge requires Java SE JDK 6 update 24 or later.

To check the Java version on UNIX systems:

```
$ java -version
java version "1.7.0_25"
Java(TM) SE Runtime Environment (build 1.7.0_25-b17)
Java HotSpot(TM) 64-Bit Server VM (build 23.25-b01, mixed mode)
```

To check the Java version on Windows systems:

- Open the Control Panel and type Java in the search field.
- Click the Java icon to display the Java Control Panel.
- Click About to display the Java version information.



In addition, on Windows systems, you must set the JAVA_HOME environment variable to point to the root of a valid Java installation.

2.2. Supported Platforms

Identity Bridge 1.0.1 has been tested primarily on CentOS 6.4 64-bit with Oracle Java 1.7.0 update 25 and 1.6.0 update 45, and on Microsoft Windows 2008 R2 server with Java 1.7.0 update 25. The testing was performed with an Active Directory running on Microsoft Windows 2008 R2 server.

2.3. Supported Repositories

OrientDB is provided with Identity Bridge as an internal (embedded) repository. The following JDBC repository is also supported but requires a separate download:

- MySQL 5.1 or 5.5 with Connector/J 5.1.18 or later

Note

MySQL support is not available in this 1.0.1 release and will be provided in the next Identity Bridge release.

2.4. Supported Browsers

Identity Bridge has been tested with Google Chrome, Mozilla Firefox, and Microsoft Internet Explorer version 10.

2.5. Hardware Requirements

You need at least 200 MB disk space and 1 GB memory for a minimal evaluation installation. For a production installation, disk space and memory requirements will depend on the number of Active Directory users, and on the size of the log files that Identity Bridge writes.

Chapter 3. Getting Identity Bridge Up and Running

This chapter describes how to complete the initial configuration of an Identity Bridge instance.

3.1. Downloading, Installing, and Starting Identity Bridge

Download the Identity Bridge .zip file, then use one of the following procedures to install Identity Bridge, depending on your operating system.

Procedure 3.1. To Install Identity Bridge on UNIX-Like Systems

1. Unpack the contents of the .zip file into the install location.

```
$ cd /path/to  
$ unzip ~/Downloads/forgerockBridge-1.0.0-SNAPSHOT-development.zip
```

2. Run the setup script.

```
$ cd /path/to/forgerockBridge  
$ ./setup.sh
```

3. Enter the SSL port to listen on for the Identity Bridge user interface. The default is 8443.
4. Enter y to have the Identity Bridge server start immediately after setup, and run in the background.

When Identity Bridge runs in the background, any log messages are output to the file `/path/to/forgerockBridge/logs/console.out`.

If you select not to have the server start immediately, you must start Identity Bridge manually, using the `startup.sh` script. In this case, log messages are output to the terminal in which you started Identity Bridge. To redirect log messages to `console.out`, follow the instructions in [Section 3.2, “Stopping and Restarting Identity Bridge”](#).

5. Point your browser to `https://hostname.domain:8443/admin/`, (specifying an alternate port if you entered an alternate port during the setup).

Procedure 3.2. To Install Identity Bridge on Windows Systems

1. Double-click the .zip file and select Extract all files.

2. In a command window, change to the *install-location*\forgerockBridge directory.

```
C:\>cd \path\to\forgerockBridge
```

3. Before you start the setup, consider the HTTPS port on which Identity Bridge should listen. By default, Identity Bridge listens on port 8443. To specify a different port, edit the `openidm.port.https` property in the `conf/boot/boot.properties` file before you start Identity Bridge.
4. Run the `startup.bat` script.

```
C:\install-location\forgerockBridge>startup.bat
```

There will be some initial errors output to the log as the connectors between the data stores have not yet been configured. You can safely ignore these errors in the initial setup.

Messages are output to the Felix shell in the command window in which you launched Identity Bridge.

5. Point your browser to `https://hostname.domain:8443/admin/`, (specifying an alternate port if you changed the default port).

You will receive a warning about the website's security certificate if you have not replaced the default certificate with a trusted certificate. For more information, see [Managing SSL Certificates](#).

Note

The remainder of this Guide assumes that Identity Bridge is accessed at the URL `hostname.domain`. Replace `hostname.domain` in all URLs pertaining to the UI with the FQDN of the host on which Identity Bridge is installed.

If you have the "ADBlock" extension enabled for your browser, disable it. The "ADBlock" extension filters all pages that include "AD" which interferes with several Active Directory pages.

3.2. Stopping and Restarting Identity Bridge

You can check whether an instance of Identity Bridge is running, stop, and restart the server as outlined in the following sections.

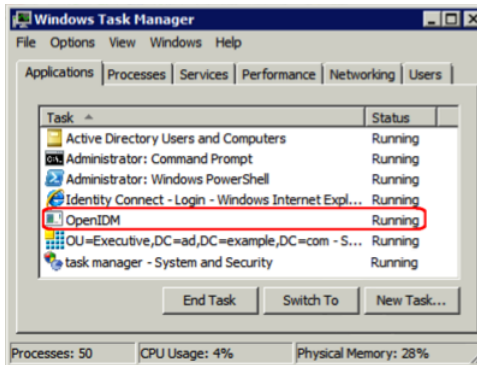
- To check whether Identity Bridge is running on UNIX-like systems, run the following command on the system on which you started Identity Bridge:

```
$ ps -ef | grep openidm
```

If an instance of Identity Bridge is running, you should see output similar to the following:

```
501 91957      1  0  4:47PM ttys001  12:03.23 /usr/bin/java
-Djava.util.logging.config.file=/path/to/forgerockBridge/conf/logging.properties
-Xmx1024m -Xms1024m
-Djava.endorsed.dirs=
-classpath /path/to/forgerockBridge/bin/*:/path/to/forgerockBridge/framework/*
-Dopenidm.system.server.root=/path/to/forgerockBridge
-Djava.awt.headless=true org.forgerock.commons.launcher.Main
-c bin/launcher.json
```

- To check whether Identity Bridge is running on Windows systems, check the running applications in the Windows Task Manager. Identity Bridge runs under the application "OpenIDM".



- To stop Identity Bridge on UNIX-like systems, run the shutdown script, located in the install directory.

```
$ cd /path/to/forgerockBridge
$ ./shutdown.sh
./shutdown.sh
Stopping OpenIDM (91957)
```

- To stop Identity Bridge on Windows systems, stop the OpenIDM application in the Windows Task Manager, or type shutdown in the Felix console that opened when you started Identity Bridge.
- To restart Identity Bridge on UNIX-like systems, run the startup script, located in the install directory. Use the nohup command to keep Identity Bridge running after you log out, and redirect the console output to console.out, as follows.

```
$ cd /path/to/forgerockBridge
$ nohup ./startup.sh > logs/console.out 2>&1&
[1] 32548
```

- To restart Identity Bridge on Windows systems, run the startup.bat script in the install directory.

3.3. Installing Identity Bridge as a Windows Service

You can install Identity Bridge to run as a Windows service, so that the server starts and stops automatically when Windows starts and stops. You must be logged in as an administrator to install Identity Bridge as a Windows service.

Note

On a 64-bit Windows server, you must have a 64-bit Java version installed to start the service. If a 32-bit Java version is installed, you will be able to install Identity Bridge as a service, but starting the service will fail.

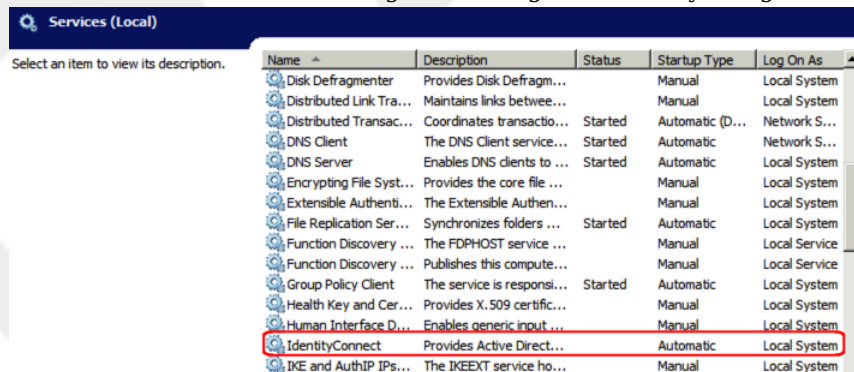
1. Unpack the Identity Bridge .zip file, as described previously, and change to the bin directory:

```
C:\>cd install-location\forgerockBridge\bin
```

2. Launch the Identity Bridge service, with the following command:

```
C:\install-location\forgerockBridge\bin>install-service.bat
Identity Bridge Service successfully installed as "IdentityBridge" service
```

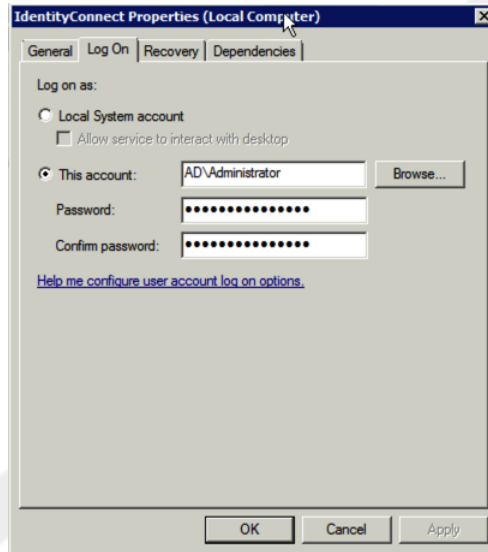
3. Use the Windows Service manager to manage the Identity Bridge service.



4. Change the user account for this service from the default (local system) account to an account with administrative privileges. The local system account has limited permissions and an Identity Bridge service that runs with this account will encounter problems during synchronization.

To change the user account:

- Double click the "IdentityBridge" service in the Windows Service manager.
- Select the Log On tab.
- Select This Account and browse for an Active Directory administrative account.
- Enter the password for the administrative account.



- Click Apply to save the changes.

5. Use the Windows Service Manager to start, stop, or restart the service.

To uninstall the Identity Bridge service, run the following command:

```
C:\install-location\forgerockBridge\bin>launcher.bat /uninstall  
Service "IdentityBridge" removed successfully
```

3.4. Configuring OpenDJ to Connect to Identity Bridge

This sample Identity Bridge deployment connects to an OpenDJ data store. Specific configuration changes are required to the OpenDJ instance, to enable it to connect to Identity Bridge. Install OpenDJ, as described in the [OpenDJ Installation Guide](#). After you have installed OpenDJ, follow these steps.

1. Import the `Example.ldif` file that is provided with Identity Bridge, either during the setup, or manually, after the setup.

If you import the file after the setup, make sure that the `userRoot` backend exists before the import, then import the file as follows:

```
$ cd /path/to/OpenDJ/bin
$ import-ldif
--port 4444
--hostname opendj.example.com
--bindDN "cn=Directory Manager"
--bindPassword password
--includeBranch dc=example,dc=com
--backendID userRoot
--ldifFile /path/to/forgerockBridge/script/OpenDJ/Example.ldif
--trustAll
```

The sample LDIF file contains one user entry (`uid=jdoe`), one administrator entry (`uid=idm`), and two group entries.

2. Set up REST access to OpenDJ, as follows:
 - a. Enable the HTTP Connection Handler.

```
$ cd /path/to/opendj/bin
$ ./dsconfig set-connection-handler-prop
--hostname opendj.example.com
--port 4444
--bindDN "cn=Directory Manager"
--bindPassword password
--handler-name "HTTP Connection Handler"
--set enabled:true
--no-prompt
--trustAll
```

- b. Enable the HTTP access log.

```
$ ./dsconfig set-log-publisher-prop
--hostname opendj.example.com
--port 4444
--bindDN "cn=Directory Manager"
--bindPassword password
--publisher-name "File-Based HTTP Access Logger"
--set enabled:true
--no-prompt
--trustAll
```

3. Edit the configuration file for the HTTP connection handler (`/path/to/opendj/config/http-config.json`) to match the specifics of your OpenDJ directory. This file defines the mapping between JSON resources and LDAP entries.

The configuration file must enable basic authentication ("supportHTTPBasicAuthentication" : true, and must specify a search DN that includes the OpenIDM administrative user that will be used to create the connector to OpenDJ ("searchBaseDN" : "dc=example,dc=com").

The following example shows a configuration file that works for this sample deployment.

```
{
  // The Rest2LDAP authentication filter configuration. The filter will be
  // disabled if the configuration is not present. Upon successful
  // authentication the filter will create a security context containing the
  // following principals:
  //
  // "dn" - the DN of the user if known (may not be the case for sasl-plain)
  // "id" - the username used for authentication.
  "authenticationFilter" : {
    // Indicates whether the filter should allow HTTP BASIC authentication.
    "supportHTTPBasicAuthentication" : true,

    // Indicates whether the filter should allow alternative authentication
    // and, if so, which HTTP headers it should obtain the username and
    // password from.
    "supportAltAuthentication" : true,
    "altAuthenticationUsernameHeader" : "X-OpenIDM-Username",
    "altAuthenticationPasswordHeader" : "X-OpenIDM-Password",

    // The search parameters to use for "search-simple" authentication. The
    // %s filter format parameters will be substituted with the
    // client-provided username, using LDAP filter string character escaping.
    "searchBaseDN" : "dc=example,dc=com",
    "searchScope" : "sub", // Or "one".
    "searchFilterTemplate" : "(&(objectClass=inetOrgPerson)(uid=%s))"
  },

  // The Rest2LDAP Servlet configuration.
  "servlet" : {
    // The REST APIs and their LDAP attribute mappings.
    "mappings" : {
      "/users" : {
        "baseDN" : "ou=people,dc=example,dc=com",
        "readOnUpdatePolicy" : "controls",
        "useSubtreeDelete" : false,
        "usePermissiveModify" : true,
        "etagAttribute" : "etag",
        "namingStrategy" : {
          "strategy" : "clientDNNaming",
          "dnAttribute" : "uid"
        },
        "additionalLDAPAttributes" : [
          {
            "type" : "objectClass",
            "values" : [
              "top",
              "person",
              "organizationalPerson",

```

```

    "inetOrgPerson"
  ]
}
},
"attributes" : {
  "schemas" : { "constant" : [ "urn:scim:schemas:core:1.0" ] },
  "_id" : { "simple" : {
    "ldapAttribute" : "uid",
    "isSingleValued" : true,
    "isRequired" : true,
    "writability" : "createOnly" } },
  "_rev" : { "simple" : {
    "ldapAttribute" : "etag",
    "isSingleValued" : true,
    "writability" : "readOnly" } },
  "userName" : { "simple" : {
    "ldapAttribute" : "mail",
    "isSingleValued" : true,
    "writability" : "readOnly" } },
  "displayName" : { "simple" : {
    "ldapAttribute" : "cn",
    "isSingleValued" : true,
    "isRequired" : true } },
  "name" : { "object" : {
    "givenName" : { "simple" : {
      "ldapAttribute" : "givenName",
      "isSingleValued" : true } },
    "familyName" : { "simple" : {
      "ldapAttribute" : "sn",
      "isSingleValued" : true,
      "isRequired" : true } }
  } },
  "manager" : { "reference" : {
    "ldapAttribute" : "manager",
    "baseDN" : "ou=people,dc=example,dc=com",
    "primaryKey" : "uid",
    "mapper" : { "object" : {
      "id" : { "simple" : {
        "ldapAttribute" : "uid",
        "isSingleValued" : true,
        "isRequired" : true } },
      "displayName" : { "simple" : {
        "ldapAttribute" : "cn",
        "isSingleValued" : true,
        "writability" : "readOnlyDiscardWrites" } }
      } }
  } },
  "groups" : { "reference" : {
    "ldapAttribute" : "memberOf",
    "baseDN" : "ou=groups,dc=example,dc=com",
    "writability" : "readOnly",
    "primaryKey" : "cn",
    "mapper" : { "object" : {
      "id" : { "simple" : {
        "ldapAttribute" : "cn",
        "isSingleValued" : true } }
      } }
  } },
  "contactInformation" : { "object" : {
    "telephoneNumber" : { "simple" : {
      "ldapAttribute" : "telephoneNumber",
      "isSingleValued" : true } },

```

```

    "emailAddress" : { "simple" : {
      "ldapAttribute" : "mail",
      "isSingleValued" : true } }
  },
  "meta" : { "object" : {
    "created" : { "simple" : {
      "ldapAttribute" : "createTimestamp",
      "isSingleValued" : true, "writability" : "readOnly" } },
    "lastModified" : { "simple" : {
      "ldapAttribute" : "modifyTimestamp",
      "isSingleValued" : true,
      "writability" : "readOnly" } }
  } }
},
"/groups" : {
  "baseDN" : "ou=groups,dc=example,dc=com",
  "readOnUpdatePolicy" : "controls",
  "useSubtreeDelete" : false,
  "usePermissiveModify" : true,
  "etagAttribute" : "etag",
  "namingStrategy" : {
    "strategy" : "clientDNNaming",
    "dnAttribute" : "cn"
  },
  "additionalLDAPAttributes" : [
    {
      "type" : "objectClass",
      "values" : [
        "top",
        "groupOfUniqueNames"
      ]
    }
  ],
  "attributes" : {
    "schemas" : { "constant" : [ "urn:scim:schemas:core:1.0" ] },
    "_id" : { "simple" : {
      "ldapAttribute" : "cn",
      "isSingleValued" : true,
      "isRequired" : true,
      "writability" : "createOnly" } },
    "_rev" : { "simple" : {
      "ldapAttribute" : "etag",
      "isSingleValued" : true,
      "writability" : "readOnly" } },
    "displayName" : { "simple" : {
      "ldapAttribute" : "cn",
      "isSingleValued" : true,
      "isRequired" : true,
      "writability" : "readOnly" } },
    "members" : { "reference" : {
      "ldapAttribute" : "uniqueMember",
      "baseDN" : "dc=example,dc=com",
      "primaryKey" : "uid",
      "mapper" : { "object" : {
        "_id" : { "simple" : {
          "ldapAttribute" : "uid",
          "isSingleValued" : true,
          "isRequired" : true } },
        "displayName" : { "simple" : {
          "ldapAttribute" : "cn",
          "isSingleValued" : true,

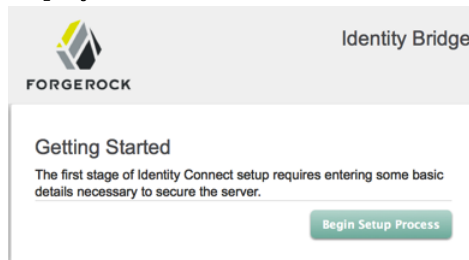
```

```
        "writability" : "readOnlyDiscardWrites" } }  
    } }  
  } },  
  "meta" : { "object" : {  
    "created" : { "simple" : {  
      "ldapAttribute" : "createTimestamp",  
      "isSingleValued" : true,  
      "writability" : "readOnly" } },  
    "lastModified" : { "simple" : {  
      "ldapAttribute" : "modifyTimestamp",  
      "isSingleValued" : true,  
      "writability" : "readOnly" } }  
    } }  
  } }  
}
```

Chapter 4. Configuring Connections Between Identity Bridge, Active Directory, and OpenDJ

Part of the Identity Bridge setup involves defining connections between Identity Bridge and Active Directory, and between Identity Bridge and OpenDJ. This chapter describes how to configure these connections.

After you have set up Identity Bridge and pointed your browser to `https://hostname.domain:8443/admin/`, the Identity Bridge Getting Started page is displayed.



Click Begin Setup Process to start the setup.

A message requesting you to confirm the Identity Bridge URL is displayed. The URL displayed here is the one that you are using to access the Identity Bridge setup. This *must* be the same URL as the URL with which your users will access Identity Bridge. If it is not the same URL, your SAML configuration will ultimately fail. For example, if you are configuring Identity Bridge using the URL `https://localhost.com:8443`, but your users will ultimately be accessing Identity Bridge at `https://connect.example.com:8443`, the URL that is configured with SAML will not match the URL your users are using, and they will therefore be unable to log in with SAML.

If you realize at this point that this is not the URL with which your users will be accessing Identity Bridge, cancel the setup, then access Identity Bridge using the correct URL.

Confirm Your URL

It is **critically important** that the URL used to configure Identity Connect is the same the URL that users will access in production.

It is **highly recommended** that you use a valid SSL certificate in production. If that is your plan, ensure that the host name you will use for that certificate also matches this URL.

You will have an opportunity to provide an SSL certificate later on in the setup process. For now continue with the self-signed certificate and ignore the SSL warning in your browser.

The current URL you are using is :

https://localhost:8443

Confirm that this will be your production URL: ☒

[Configure Data Source](#)

Select "Confirm that this will be your production URL" and click Configure Data Source to continue.

4.1. Configuring the Active Directory Connector

The first step in setting up Identity Bridge is configuring the Active Directory connector, or data source.

1. Click *Configure Data Source* to configure the Active Directory connector.
Data Source: Active Directory

Be sure to add base contexts which include both your users and your groups.

[Add Base Context](#)

Enter the path to your base dn
CN=Users,DC=example,DC=com ✓

Enter the path to your base dn
OU=groups,dc=example,dc=com ✓

Host name or IP
host.example.com ✓

Port
389 ✓ ☐ Use SSL?

Account Distinguished Name (DN)
CN=Administrator,CN=Users,DC=example,DC=com ✓

Password
..... ✓

User Filter
(&(!((userAccountControl:1.2.840.113556.1.4.803:=2)))

[Add Objectclasses For Users](#)

objectclasses for users
user ✓

Group Filter
(&(!((cn=Domain Users)))

[Add Objectclasses For Groups](#)

objectclasses for groups
group ✓

[Validate settings](#)

2. On the *Data Source: Active Directory* page, enter the following information:

- *Host name or IP.* Enter the fully qualified host name, or IP address, of the machine that hosts the Active Directory instance.
- *Port.* Enter the port number on which the Active Directory server listens for LDAP connections. The default LDAP port is 389. The default LDAPS port is 636. If you are connecting to a Global Catalog, the default port is 3268, or 3269 if you are using SSL.

Check *Use SSL* to connect to the LDAPS port.

Make sure that remote LDAP or LDAPS access to the Active Directory server is allowed through the Firewall.

If you select to use SSL, and you are using a self-signed certificate (or the root CA for your Active Directory certificate is not in the trust store), you need to provide the public SSL certificate for your Active Directory server as follows:

- On your Active Directory server, type the following command into a Command Prompt window:

```
c:\>certutil -ca.cert client.crt
```

This command will output the certificate (from -----BEGIN CERTIFICATE----- to -----END CERTIFICATE-----) to the command line.

- Copy the contents of the certificate to the clipboard.
- On the Identity Bridge Active Directory configuration screen, click *show certificate*.
- Paste the certificate contents into the SSL Certificate window and click the close icon.
- *Account Distinguished Name (DN).* Enter an administrator bind DN for the Active Directory server. This user must have domain administration rights to all of the base contexts that will be managed by Identity Bridge.
- *Password.* Enter the bind password for the user specified previously.
- *Base Contexts.* Enter the path to one or more base DN's that will be synchronized during the data synchronization phase.

Note

Make sure that the user and group entries that will be managed through Identity Bridge are included in the base contexts that you specify here.

- *User Filter.* Specify one or more LDAP filters that will be applied to the Active Directory users, to determine which users will be mapped to their corresponding accounts in OpenDJ.

By default, Identity Bridge filters out Active Directory user entries that are disabled, with the filter "`(!(userAccountControl:1.2.840.113556.1.4.803:=2))`". Click the User Filter field and use the Update User Filter dialog to create additional filters.

For information about LDAP filter syntax, see <http://social.technet.microsoft.com/wiki/contents/articles/5392.active-directory-ldap-syntax-filters.aspx>.

Provide one or more Active Directory object classes to search for Identity Bridge user entries. The default object class for user entries is user.

- *Group Filter.* Specify one or more LDAP filters that will be applied to Active Directory group entries, to determine which groups will be mapped the corresponding OpenDJ groups.

By default, Identity Bridge filters out Active Directory entries under the organizational unit `cn=Domain Users`, with the filter `(!(cn=Domain Users))`. `Domain Users` is a special group that typically includes *all* user entries in the directory, but is not displayed under a user's `memberOf` attribute (so the group displays no members when it is searched). Do not remove this filter from the configuration. Click the Group Filter field and use the Update Group Filter dialog to create additional filters.

For information about LDAP filter syntax, see <http://social.technet.microsoft.com/wiki/contents/articles/5392.active-directory-ldap-syntax-filters.aspx>.

Provide one or more Active Directory object classes to search for Identity Bridge group entries. The default object class for group entries is group.

3. When you have completed all of the preceding fields, click *Validate Settings* to validate the data source configuration. If the configuration is valid, a validation message is displayed at the top of the page. If the configuration is not valid, a validation error is displayed, with additional details provided at the bottom of the page.

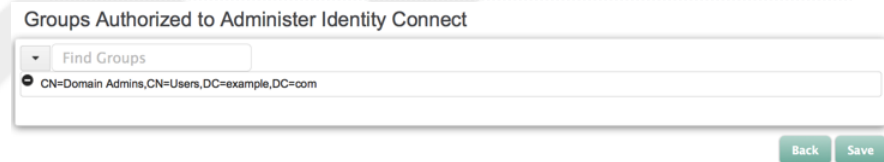
When the data source configuration is valid an Active Directory connector is created on the Identity Bridge host to facilitate access between Identity Bridge and Active Directory.

4. The following page lists all the defined groups on the Active Directory server, for the base contexts and object classes that were specified in the previous step. It might take a couple of seconds for this list to be populated, depending on your network latency, as Identity Bridge accesses Active Directory here and reads the list of defined groups.

Select the group or groups whose members will be granted administration privileges for Identity Bridge and click *Save*.

Caution

If you select a group here to which your own administrative account does not belong, you will be locked out of the Identity Bridge administrative interface immediately and will be required to reinstall Identity Bridge from scratch. You should therefore take care when selecting these groups.



Groups Authorized to Administer Identity Connect

Find Groups

☒ CN=Domain Admins,CN=Users,DC=example,DC=com

Back Save

The Active Directory connector has now been configured. At this point, the user interface exits and you are forced to authenticate (using the credentials established previously) before you can proceed with the configuration.

4.1.1. Working with Multiple Active Directory Domains

If your directory service has only one domain controller, or if all your users are in the same domain, Identity Bridge can connect to a single domain controller. If your directory service spans multiple domains, Identity Bridge must connect to the Global Catalog (GC) to have a comprehensive view of all the domains. Multiple connections to multiple Domain Controllers from a single Identity Bridge instance are not possible.

Using a GC as the authoritative data source has the following limitations:

-
- Only a subset of attributes is replicated from other domains to the GC.

Certain required attributes might be missing for the purposes of Identity Bridge. To avoid this problem, you must modify the Active Directory schema to ensure that the required attributes are replicated to the GC. For more information, see [Section 4.1.2, “Updating the Active Directory Schema for a Global Catalog”](#).

- Delete operations are not detected immediately.

A liveSync operation will therefore not update the OpenDJ resource with the result of a delete operation. Delete operations are detected by a reconciliation operation, so data stores are only temporarily "out of sync" with regard to deletes.

- Not all group types are supported.

Group membership information is replicated to the GC for *universal* groups only. You must therefore use universal groups if your directory service has more than one domain.

4.1.2. Updating the Active Directory Schema for a Global Catalog

To ensure that the attributes required by Identity Bridge are replicated to the GC, you must update the Active Directory schema to include the required attributes. Before you update the schema, note the following:

- Only a member of the Schema Admins group can modify the Active Directory schema.
- Modifying the Active Directory schema requires a change to the registry on the Schema Master. For information about how to change the registry, see the Microsoft Knowledge Base article on [Registry Modification Required to Allow Writing to Schema](#).

Modifying the registry incorrectly can severely compromise your system so exercise caution.

If you attempt to change the schema before you change the registry key, Active Directory will reject the change.

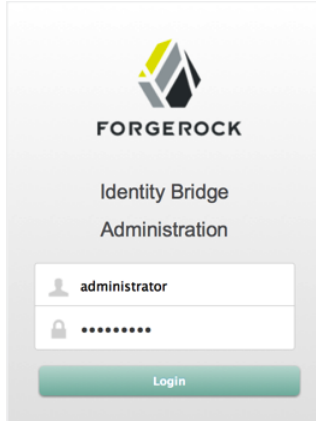
- Increasing the number of attributes that are replicated to the GC will invariably have an impact on network replication traffic.

Use the Active Directory Schema Microsoft Management Console (MMC) to modify the schema. For more information, see the Microsoft Knowledge Base article on [Modifying Attributes That Replicate to the Global Catalog](#).

4.2. Configuring the ForgeRock Connector

Identity Bridge connects to an OpenDJ data store over a scripted SQL connector called the ForgeRock connector.

After you have completed the Active Directory connector configuration, the administration login page is displayed.

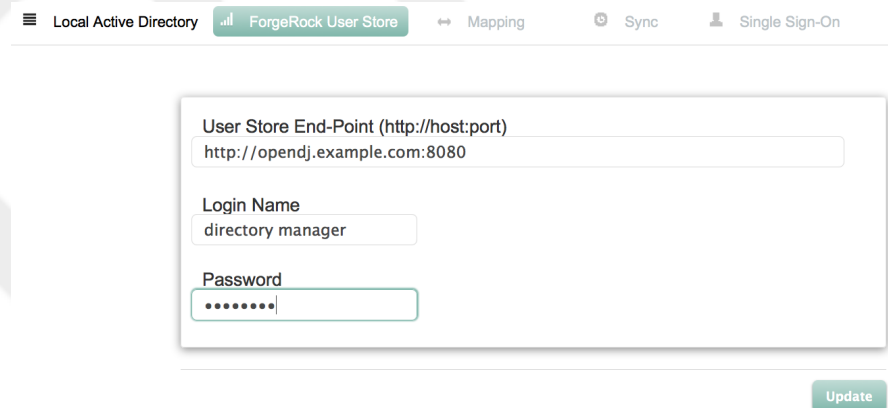


The image shows the ForgeRock Identity Bridge Administration login page. At the top is the ForgeRock logo, followed by the text "Identity Bridge Administration". Below this is a login form with two input fields: the first is labeled "administrator" and the second is masked with dots. A green "Login" button is positioned below the password field.

Configure the ForgeRock connector as follows:

1. Log in with the credentials of a user who belongs to one of the groups that you specified in the previous step.

The ForgeRock User store configuration page is displayed.



The image shows the ForgeRock User Store configuration page. At the top, there is a navigation bar with tabs: "Local Active Directory", "ForgeRock User Store" (which is selected), "Mapping", "Sync", and "Single Sign-On". Below the navigation bar is a configuration form with three fields: "User Store End-Point (http://host:port)" with the value "http://opendj.example.com:8080", "Login Name" with the value "directory manager", and "Password" which is masked with dots. An "Update" button is located at the bottom right of the form.

2. *User Store End-Point (http://host:port)*. Enter the URL, including the port, on which the OpenDJ http connection handler listens, for example, `http://opendj.example.com:8080`.

3. *Login Name*. Enter the name of an administrative user in OpenDJ, for example, `idm`, if you imported the sample `Example.ldif`.
4. *Password*. Enter the password of the administrative user specified in the previous step.

You are now ready to move on to the mapping configuration.

Chapter 5. Mapping Data Between Active Directory and OpenDJ

Identity Bridge enables you to specify how attributes and other data are mapped from the Active Directory data source to the OpenDJ data store.

After you have configured the ForgeRock connector, select the Mapping tab.

The Mapping page enables you to map Active Directory users and groups to OpenDJ users and groups.

- *Attribute Mapping* maps all the attributes of an Active Directory user entry to a comparable attribute in OpenDJ. A default set of mapped attributes is presented, with sample values for each attribute.
- *Group Mapping* maps Active Directory groups to OpenDJ groups. At least one group mapping required.

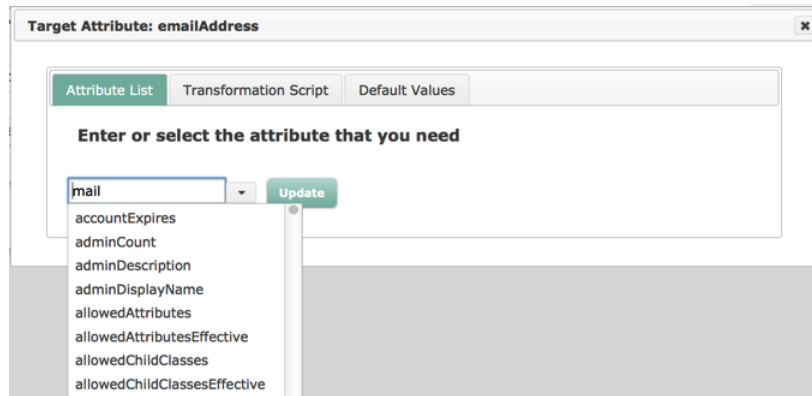
The first time you open the mapping page, only the Attribute Mapping tab is displayed. After you have saved the initial attribute mapping, the Group Mapping tab is displayed.

5.1. Mapping Attributes

Attribute mapping enables you to specify how the value of an OpenDJ attribute is defined, based on a corresponding Active Directory attribute. A default set of mapped attributes is presented, with sample values for each attribute.

Configure attribute mapping by following these steps:

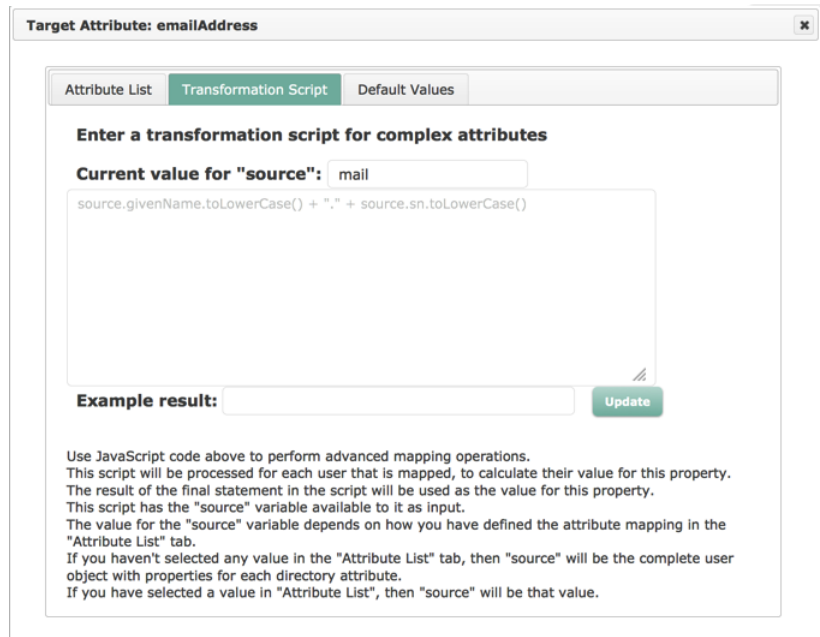
1. On the Mapping page, select the Attribute Mapping tab.
2. In the DJ Property column, select the OpenDJ attribute whose value you want to define. For example, click on `emailAddress` to specify the value that will be used for the OpenDJ email attribute.
3. The Attribute List tab enables you to specify an Active Directory attribute to be mapped directly. Enter the name of the attribute or type a few characters of the attribute name to select it from the list.



4. The Transformation Script tab enables you to specify how an Active Directory attribute is transformed to provide a value for the OpenDJ attribute. The transformation script is a JavaScript that takes a source (Active Directory) attribute, and does something with its value to provide the corresponding OpenDJ attribute value.

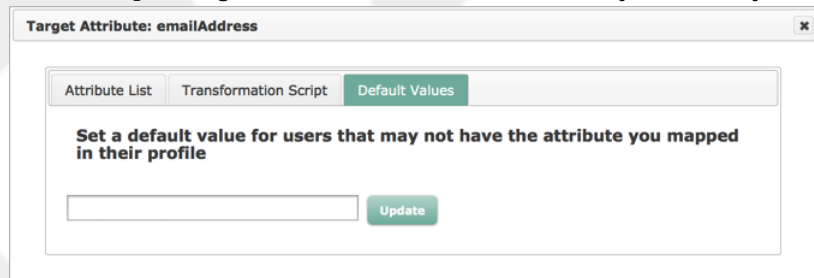
For example, the sample transformation script `source.mail ? source.mail.toLowerCase() : null` takes the value of the mail attribute from Active Directory and converts it to lower case to provide the value of the emailAddress attribute in OpenDJ. If no value exists for the mail attribute, a null value is inserted as the value in OpenDJ.

The format of the transformation script depends on whether you have selected an attribute on the Attribute List tab. If you do not specify an attribute on the Attribute List tab, the entire object is regarded as the source, and you must include the attribute name in the script (for example, `source.mail.toLowerCase()`). If you specify an attribute on the Attribute List tab, that attribute is regarded as the source, so the transformation script would be `source.toLowerCase()`.



The screenshot shows a configuration window titled "Target Attribute: emailAddress". It has three tabs: "Attribute List", "Transformation Script" (which is selected), and "Default Values". Under the "Transformation Script" tab, there is a heading "Enter a transformation script for complex attributes". Below this, it says "Current value for 'source': mail" next to a text input field. A code editor area contains the script: `source.givenName.toLowerCase() + "." + source.sn.toLowerCase()`. Below the code editor is an "Example result:" label followed by an empty text box and an "Update" button. At the bottom, there is explanatory text: "Use JavaScript code above to perform advanced mapping operations. This script will be processed for each user that is mapped, to calculate their value for this property. The result of the final statement in the script will be used as the value for this property. This script has the 'source' variable available to it as input. The value for the 'source' variable depends on how you have defined the attribute mapping in the 'Attribute List' tab. If you haven't selected any value in the 'Attribute List' tab, then 'source' will be the complete user object with properties for each directory attribute. If you have selected a value in 'Attribute List', then 'source' will be that value."

5. The Default Values tab enables you to specify a default value that should apply for the OpenDJ attribute in the event that the user does not have the corresponding attribute in her Active Directory user entry.



The screenshot shows the same configuration window, but now the "Default Values" tab is selected. It has a heading "Set a default value for users that may not have the attribute you mapped in their profile". Below this heading is a text input field and an "Update" button.

6. The Sample User field indicates the user whose entry is used to show what the attribute transformations will look like. By default, the entry of the user who is logged into Identity Bridge is used. To display potential attribute values for a different user, enter the uid of that user in the Sample User field.
7. When the attribute mapping configuration is complete, click *Save* to save the mapping.

5.2. Mapping Groups

The group to group mapping feature enables you to specify the Active Directory groups that correspond to configured OpenDJ groups. You *must* configure at least one group mapping in order for synchronization to work.

Configure group mapping by following these steps:

1. On the Mapping page, select the Group Mapping tab.
2. The left hand column lists all the defined OpenDJ groups. The right hand column indicates the Active Directory groups to which these OpenDJ groups are mapped. No groups are mapped by default.

The screenshot shows the 'Group Mapping' tab in the configuration interface. On the left, under 'ForgeRock', the 'openidm2' group is listed. Below it, the 'openidm' group is shown with an edit icon and the text 'None'. A 'Find Groups' dropdown menu is open, displaying a list of Active Directory groups: Domain Admins, Domain Computers, Domain Controllers, Domain Guests, Enterprise Admins, Enterprise Read-only Domain Controllers, and Group Policy Creator Owners. The 'Update' button is visible below the dropdown. At the bottom, it says 'Cached Data Last Updated December 19, 2013 17:33'.

3. Click the edit icon adjacent to an OpenDJ group to map the corresponding Active Directory group.

You can select more than one Active Directory group to map to an OpenDJ group. The following selection maps all members of the Active Directory group "forgerockGroup" to the "openidm" group in OpenDJ.

The screenshot shows the 'Group Mapping' tab. The 'openidm' group is now mapped to 'forgerockGroup' in the Active Directory column. The 'Update' and 'Cancel' buttons are visible. At the bottom, it says 'Cached Data Last Updated December 19, 2013 17:33' with a link to 'Update Now'.

Chapter 6. Data Synchronization and User Association Management

The main purpose of Identity Bridge is to maintain data consistency between your Active Directory and any other data store. This consistency is achieved by a process called *synchronization*, which modifies user data on the target system (OpenDJ, in this case) to match the data in Active Directory.

This section provides an overview of the synchronization process and walks you through the synchronization configuration to establish associations between user entries.

6.1. Overview of the Synchronization Process

Synchronization changes user data on a target system so that it matches the data on a source system. Before synchronization can occur, a *reconciliation* report is run. Reconciliation is the process by which two data sources are assessed and the consistency of the data across the two systems is analyzed. Part of the reconciliation process involves identifying the user accounts that exist in the two data stores, and assessing their potential for matching.

After a reconciliation run, the Reconciliation Report identifies all user and group accounts and categorizes them, based on the extent to which a match is found between the source and the target. User accounts are divided into two main categories:

- *Valid Active Directory Users* are user accounts that exist in Active Directory and are candidates for synchronization. A valid AD user account can be one of the following:
 - *1-1 Match*, meaning that a clear and unique match exists in the OpenDJ data store, with no ambiguity.
 - *No Match Found*, meaning that there is no corresponding entry in OpenDJ, although the Active Directory user is a valid user for synchronization.

In an initial provisioning process (before the OpenDJ data store organization has been populated with entries), this is the most likely situation for AD user entries. Entries are unlikely to be in this state if a scheduled synchronization or liveSync process has been configured.

- *Conflicting Match*, meaning that more than one potential match exists in OpenDJ. Entries in this category should be manually assigned to the correct OpenDJ user.

- *Other Users* are all entries in either Active Directory or OpenDJ that are not candidates for synchronization. This category normally indicates "orphan entries" in either the source or target data stores. Other users can be one of the following:
- *Unresolved AD Users* are user accounts that exist in Active Directory but either have no match in OpenDJ or the potential OpenDJ match has already been associated with another Active Directory user account.

Most commonly, entries fall into this category when an Active Directory entry that was previously linked to an OpenDJ entry, has lost its corresponding entry, but the link was not removed. Alternatively, if a manual user assignment has already been made to the corresponding OpenDJ entry, this link can prevent the correct Active Directory entry from being mapped.

- *Unresolved FR Users*, are user accounts that exist only in OpenDJ and not in Active Directory.

Generally, the corresponding Active Directory entry is missing, either because it never existed, or because it was removed and the change has not yet been picked up by synchronization or liveSync.

- *Ignored Users*, are user accounts for which no match exists but which are not cause for concern.

The existence of these users in only one data source (either Active Directory or OpenDJ) is expected, and the accounts are ignored in future synchronization reports and reconciliation runs.

For example, certain OpenDJ administrative entries might be required only in the context of OpenDJ administration and have no use in Active Directory. These *known unresolved entries* can be flagged so that they appear in a separate list in future synchronization runs. Separating the ignored entries from the unresolved entries ensures that the list of unresolved entries remains a priority for cleanup.

In general, unresolved Active Directory and OpenDJ entries are cleaned up during synchronization. Entries that exist only in Active Directory are created in OpenDJ. Entries that exist only in OpenDJ are deactivated (in the event that they have been deleted from Active Directory) or must be moved manually to the list of Ignored Users (if they have never existed in Active Directory and are not candidates for synchronization).

Inactive OpenDJ users are filtered out of the reconciliation process. However, they are still visible in Identity Bridge and can therefore be manually linked to Active Directory entries and reactivated.

6.2. Managing User Associations

When a reconciliation operation finds a matching target entry, a *link* is created between the source and the target entry. This link is referred to in Identity Bridge as a *User Association*. User associations serve two purposes - they speed up future reconciliation operations, and they serve as a record of a source or target entry's existence.

For example, a target entry might be deleted at some point, but if an association to the source entry still exists, there is evidence that the target entry once existed. This functionality is useful for auditing purposes. If there is conflicting data between two matched entries, the reconciliation operation might be unable to associate the entries. In this case, the entries can be associated manually.

The first time you configure synchronization, Identity Bridge performs a blank reconciliation run. In this initial operation, no records are changed in OpenDJ. The reconciliation report that is generated enables you to assess the consistency of the entries stored in Active Directory with those stored in OpenDJ, by automatically associating user entries, wherever possible. Based on this report, you can change the Default User Association Rules, where necessary. Note that the Default Association Rules only apply before the first real synchronization operation is performed (that is, before any links exist between entries in Active Directory and entries in OpenDJ). You should review the associations carefully before running a real synchronization operation for the first time because it is more difficult to isolate or fix inaccurate associations after the data has been synchronized.

Entries that could still not be associated automatically, after the association rules have been finalized, can be associated manually, prior to performing a real synchronization operation. In the ideal scenario, all entries are either associated, or have been marked as "Ignored".

6.2.1. Changing User Associations Manually

Clean Data refers to all entries in the source and target system being matched and associated, with no conflicts in the entry data, and known unmatched entries being marked as ignored. The reconciliation report indicates the percentage of data that is *clean*. For more information, see [Running Reconciliation Reports](#).

Identity Bridge provides a mechanism to clean up data by working through any unmatched, unassociated, or conflicting entries found during the reconciliation run. The following list describes the data cleanup process for each category into which user entries fall after a reconciliation run.

1-to-1 Match

Generally, valid AD Users with a 1-to-1 Match do not require manual intervention. However, there might be specific entries whose user associations have been made incorrectly by the automatic association mechanism. In this case, you can manually disassociate these user entries and reassociate them to the correct entry.

To change a user association manually, see [Procedure 6.1, “To Create or Change User Associations Manually”](#).

No Match Found

For valid AD Users for whom no OpenDJ match was found, you can use manual association to link the entry to an existing OpenDJ account or move the entry to the list of ignored entries. Ignored entries will not appear in the list of unmatched entries in any future reconciliation reports. If you do not make any manual association on an unmatched account, the account is created in OpenDJ when the data is synchronized.

To locate a match in OpenDJ manually, see [Procedure 6.1, “To Create or Change User Associations Manually”](#). The manual association will be used during future synchronization runs. To move the entry to the list of Ignored Users, see [Procedure 6.2, “To Move Users to the Ignored List”](#).

Conflicting Matches

Conflicting matches are user entries for which more than one potential match exists. To resolve conflicting matches, specify a match for the user manually, as described in [Procedure 6.1, “To Create or Change User Associations Manually”](#).

Unresolved AD Users

These users exist only in Active Directory, with no known match in OpenDJ. You can either find a match in OpenDJ manually (see [Procedure 6.1, “To Create or Change User Associations Manually”](#)) or move the entry to the list of Ignored Users (see [Procedure 6.2, “To Move Users to the Ignored List”](#)).

Unresolved SF Users

These users exist only in OpenDJ, with no known match in Active Directory. You can either find a match in Active Directory manually (see [Procedure 6.1, “To Create or Change User Associations Manually”](#)) or move the entry to the list of Ignored Users (see [Procedure 6.2, “To Move Users to the Ignored List”](#)).

Ignored Users

Only users that you have explicitly moved to the ignored list appear here. If you have moved a user to the ignored list in error, select that user and click Change User Association to move the user out of the ignored list and manually associate it with its matching entry (see [Procedure 6.1, “To Create or Change User Associations Manually”](#)).

Procedure 6.1. To Create or Change User Associations Manually

To create a user association manually, or to change an association that was created automatically, follow these steps:

1. On the Sync page, select the tab and category for the user entry that you want to associate.

Select the entry that you want to match, or whose association you want to change, and click Change User Association.

2. In the Change User Association window, select an item by which to search for the correct user, either in Active Directory, or in OpenDJ. Select either Alias, Email, First Name or Last Name.

Enter the required value for this field and click Search.

3. All entries that match your search are displayed underneath the Search button. Select the correct entry to be matched, and click Link Account.
4. The user is now associated with the account that you have selected, rather than with the account that was identified during automatic association.

This manual association will override the automatic association during future synchronization runs.

Procedure 6.2. To Move Users to the Ignored List

To move a user to the list of ignored users, follow these steps:

1. On the Sync page, select the tab and category for the user entry or entries that you want to ignore.
2. Select the entry or entries and click Ignore User(s).

Users that have been moved to this category are displayed in the list of Ignored Users on the Other Users tab.

Procedure 6.3. To Test the Mapping for a Specific Entry

After you have completed the mapping configuration and established an association between two entries, either automatically or manually, test the configuration for a specific user as follows:

1. On the Sync page, select the tab and category for the user entry that you want to test.
2. Select the entry and click Sync Selected Record.

The Single User Data Synchronization panel displays the record in Active Directory, the current corresponding record in OpenDJ, if there is one, and what the record will look like in OpenDJ after a synchronization operation has been run.

Click **Sync Now** to perform the synchronization operation on that particular record. If the synchronization is unsuccessful, an error indicating the reason for the failure is displayed at the top of the screen. You can use this information to correct any errors in the mapping, and attempt the synchronization test again.

Note that this step synchronizes the data of this particular user - it is not merely a validity check. Synchronizing a single user enables you to test your mapping before applying it to the entire OpenDJ data store.

The following image shows an Active Directory new user entry that did not exist in OpenDJ until the Sync Now operation was launched.

The screenshot shows a window titled "Single User Data Synchronization" with a close button in the top right. It contains two main sections: "Active Directory Properties" and "ForgeRock Properties".

Active Directory Properties

Current values
c
cn
co
company
countryCode
department
description

ForgeRock Properties

Before sync	After sync
givenName	givenName
familyName	familyName
emailAddress	emailAddress
displayName	displayName
userStatus	userStatus
uid	uid
telephoneNumber	telephoneNumber

Values shown in the "After sync" column:

- givenName: Stephanie
- familyName: Conroy
- emailAddress: stephanie.conroy@example.c
- uid: sconroy

A "Sync Now" button is located in the top right corner of the panel.

After you have changed the automatic user associations and moved any entries to the Ignored Users list, click **Analyze Associations Now** to run the reconciliation report again. Verify that all the user associations are correct before enabling synchronization for the entire data store.

To synchronize data immediately, after you have verified the user associations, click **Sync Now**.

6.2.2. Association Rules

Association Rules are the criteria by which user accounts are mapped between OpenDJ and Active Directory. By default, users are linked if one of the following situations is true:

- Their email addresses match
- Their first name and last name match and either their phone, mobile phone, or title match.

You can change the way in which these matches are made by changing the Association Rules on the Sync Page.

Click Change Association Rules and add or remove fields to compile your own set of association rules.

6.3. Configuring the Synchronization Schedule

Data synchronization enables you to specify when and how often Active Directory data changes are pushed to the OpenDJ data store. Data can be synchronized according to a defined schedule, or automatically, as soon as changes are made in Active Directory.

The Live Updates mechanism is intended to react quickly to changes as they happen. Live Updates are, however, a best effort mechanism that can miss changes in certain situations. In addition, if a system is down when an update occurs on Active Directory, that change might not be propagated to OpenDJ when the system comes back online.

Scheduled Updates are more thorough. The Scheduled Updates mechanism recognizes system error conditions and catches changes that might be missed by the Live Updates mechanism.

It is recommended that you enable both Scheduled Updates and Live Updates in production.

To configure the synchronization schedule, follow these steps:

1. In the Scheduled Data Synchronization area, check *Schedule Updates* to specify a regular schedule for synchronization.
2. Select an update interval from the drop down list. Selecting minute, hour, day, and so on, specifies that updates are scheduled once every minute, hour or day. Selecting (n) days, (n) hours, and so on, enables you to specify the precise number of days, hours or minutes between each update.
3. Select *Live Updates* to indicate that data should be synchronized as soon as changes are made in Active Directory.
4. Click *Save* to save the synchronization configuration.

DRAFT

Chapter 7. Configuring Single Sign-On

Identity Bridge enables you to set up single sign-on (SSO) using the Security Assertion Markup Language (SAML). With SSO configured, when a user accesses a defined resource, he is redirected to the Identity Bridge user interface (at `https://hostname.domain.com:8443/connect/`). In this sample deployment, logging in to this interface routes the user directly to OpenAM, which acts as the service provider.

The service provider does not validate the user's password. Identity Bridge validates the user's credentials (with a simple username/password check or by using Kerberos) and generates an assertion that is sent back to OpenAM in an HTTP POST request. OpenAM then verifies the assertion and allows single sign-on if the assertion is true.

To configure single sign on, you must install and configure OpenAM as a hosted service provider and remote identity provider. The steps to configure SSO with OpenAM are described in this chapter.

7.1. Setting Up OpenAM

In this sample deployment, users access Identity Bridge for SSO to OpenAM. OpenAM is configured as a hosted service provider. This procedure shows how to get OpenAM up and running. The following procedure shows how to configure OpenAM as the sample Service Provider. Complete installation instructions for OpenAM are provided in the [OpenAM Installation Guide](#).

Procedure 7.1. To Configure OpenAM

1. Prepare your `/etc/hosts` file.

OpenAM requires that you use fully qualified domain names when protecting web resources. This is because OpenAM uses HTTP cookies to keep track of sessions for single sign on, and setting and reading cookies depends on the server name and domain.

For this sample deployment, give your system the aliases `openam.example.com` and `www.example.com` by editing your hosts file.

Add the aliases to your hosts file using your preferred text editor.

```
$ sudo vi /etc/hosts
Password:

### Edit /etc/hosts ###

$ cat /etc/hosts | grep openam
127.0.0.1    localhost openam.example.com www.example.com
```

-
2. Install Apache Tomcat.
3. Create a file named to set the Tomcat environment variables.

The file must be named `setenv.sh` and located in the `$CATALINA_HOME/bin` directory. Add the following line to this file:

```
export CATALINA_OPTS="-server -XX:MaxPermSize=512m -Xms2048m -Xmx2048m"
```

Change the file permissions so that the script is executable.

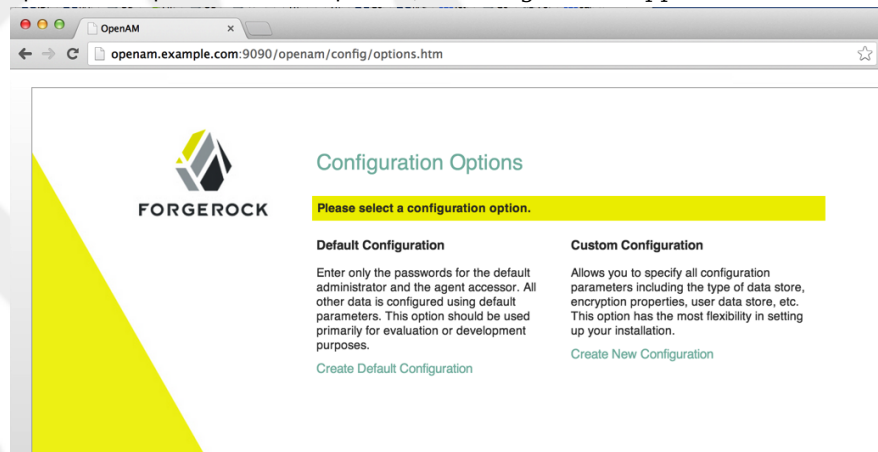
```
$ chmod +x $CATALINA_HOME/bin/setenv.sh
```

-
-
-
4. Download the OpenAM .war (web archive) file either from the ForgeRock [Enterprise Downloads](#) page, or from the project nightly [Builds](#) page.
5. Deploy the .war file in Tomcat as `openam.war`.

```
$ mv ~/Downloads/openam*.war /path/to/tomcat/webapps/openam.war
```

Tomcat deploys OpenAM under the `/path/to/tomcat/webapps/openam/` directory. You can access the web application in a browser at `http://openam.example.com:8080/openam/`.

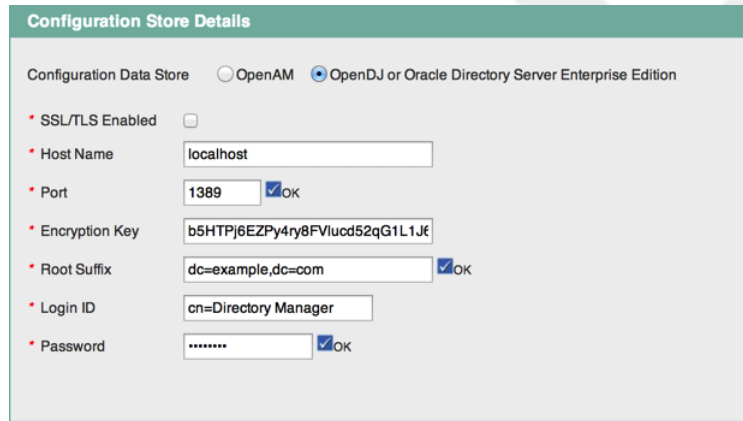
-
-
-
-
-
6. Browse to OpenAM where it is deployed in Tomcat, for example `http://openam.example.com:8080/openam/`, to configure the application.



-
-
-
-
-
-
7. On the OpenAM home page, click the link to Create Custom Configuration.

8. Enter a password for the default user (amAdmin), confirm the password and click Next.
9. On the Server Settings page, accept the default server settings and click Next.
10. On the Configuration Store Details page, select OpenDJ as the configuration store.

You can use the OpenDJ instance that you set up previously. Enter the server details for this instance and click Next.



Configuration Store Details

Configuration Data Store ☐ OpenAM ☒ OpenDJ or Oracle Directory Server Enterprise Edition

* SSL/TLS Enabled ☐

* Host Name

* Port ☒ OK

* Encryption Key

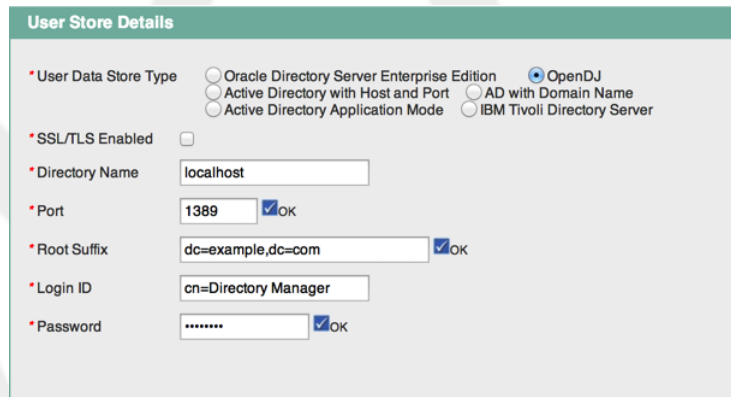
* Root Suffix ☒ OK

* Login ID

* Password ☒ OK

11. On the User Store Details page, select OpenDJ as the data store.

You can use the OpenDJ instance that you set up previously. Enter the server details for this instance and click Next.



User Store Details

* User Data Store Type ☐ Oracle Directory Server Enterprise Edition ☒ OpenDJ
☐ Active Directory with Host and Port ☐ AD with Domain Name
☐ Active Directory Application Mode ☐ IBM Tivoli Directory Server

* SSL/TLS Enabled ☐

* Directory Name

* Port ☒ OK

* Root Suffix ☒ OK

* Login ID

* Password ☒ OK

12. On the Site Configuration page, select No and click Next.

- On the Policy Agent User page, enter a password for the policy agent user. This password must not be the same as the one you entered for the default user.

In this example, we used agentPassword as the password.

- Review the Summary details and click Create Configuration.

Configurator Summary Details

Configuration Store Details edit...

SSL/TLS Enabled	No
Host Name	localhost
Listening Port	1389
Root Suffix	dc=example,dc=com
User Name	cn=Directory Manager
Directory Name	/Users/openam

User Store Details edit...

SSL/TLS Enabled	No
Host Name	localhost
Listening Port	1389
Root Suffix	dc=example,dc=com
User Name	cn=Directory Manager
User Data Store Type	OpenDJ

Site Configuration Details edit...

This instance is not setup behind a load balancer

Procedure 7.2. To Configure OpenAM as a Service Provider

- Create two xml metadata files from the following samples. Replace the string openam.example.com:8080 in each of these files with the host on which your OpenAM instance is installed and the port on which Tomcat is listening.

The files are named sp.xml and sp-extended.xml and provide the configuration for the service provider.

sp.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<EntityDescriptor entityID="bridge-sp" xmlns="urn:oasis:names:tc:SAML:2.0:metadata">
  <SPSSODescriptor AuthnRequestsSigned="false" WantAssertionsSigned="false"
    protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
    <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
      Location="http://openam.example.com:8080/openam/SPSloRedirect/metaAlias/sp"
      ResponseLocation="http://openam.example.com:8080/openam/SPSloRedirect/metaAlias/sp"/>
    <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
      Location="http://openam.example.com:8080/openam/SPSloPOST/metaAlias/sp"
      ResponseLocation="http://openam.example.com:8080/openam/SPSloPOST/metaAlias/sp"/>
    <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
      Location="http://openam.example.com:8080/openam/SPSloSoap/metaAlias/sp"/>
    <ManageNameIDService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
```

```

    Location="http://openam.example.com:8080/openam/SPMniRedirect/metaAlias/sp"
    ResponseLocation="http://openam.example.com:8080/openam/SPMniRedirect/metaAlias/sp"/>
  <ManageNameIDService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Location="http://openam.example.com:8080/openam/SPMniPOST/metaAlias/sp"
    ResponseLocation="http://openam.example.com:8080/openam/SPMniPOST/metaAlias/sp"/>
  <ManageNameIDService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="http://openam.example.com:8080/openam/SPMniSoap/metaAlias/sp"
    ResponseLocation="http://openam.example.com:8080/openam/SPMniSoap/metaAlias/sp"/>
  <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:transient</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:WindowsDomainQualifiedName</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName</NameIDFormat>
  <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:persistent</NameIDFormat>
  <AssertionConsumerService index="0" isDefault="true"
    Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact"
    Location="http://openam.example.com:8080/openam/Consumer/metaAlias/sp"/>
  <AssertionConsumerService index="1" isDefault="true"
    Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Location="http://openam.example.com:8080/openam/Consumer/metaAlias/sp"/>
  <AssertionConsumerService index="2" isDefault="false" Binding="urn:oasis:names:tc:SAML:2.0:bindings:PAOS"
    Location="http://openam.example.com:8080/openam/Consumer/ECP/metaAlias/sp"/>
</SPSSODescriptor>
</EntityDescriptor>

```

Take note of the last HTTP-POST Location URL in this file `http://openam.example.com:8080/openam/Consumer/metaAlias/sp`). You will need this URL when you set up the Identity Provider in Identity Bridge.

sp-extended.xml

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<EntityConfig entityID="bridge-sp" hosted="true" xmlns="urn:sun:fm:SAML:2.0:entityconfig">
  <SPSSOConfig metaAlias="/sp">
    <Attribute name="appLogoutUrl">
      <Value/>
    </Attribute>
    <Attribute name="spAdapterEnv"/>
    <Attribute name="useIntroductionForIDPPProxy">
      <Value>false</Value>
    </Attribute>
    <Attribute name="spAdapter">
      <Value/>
    </Attribute>
    <Attribute name="intermediateUrl">
      <Value/>
    </Attribute>
    <Attribute name="spAccountMapper">
      <Value>com.sun.identity.saml2.plugins.DefaultSPAAccountMapper</Value>
    </Attribute>
    <Attribute name="signingCertAlias"/>
    <Attribute name="useIDPFinder"/>
    <Attribute name="enableIDPPProxy">
      <Value>false</Value>
    </Attribute>
    <Attribute name="encryptionCertAlias"/>
  </SPSSOConfig>
</EntityConfig>

```

```

<Attribute name="spAuthncontextMapper">
  <Value>com.sun.identity.saml2.plugins.DefaultSPAAuthnContextMapper</Value>
</Attribute>
<Attribute name="idpProxyCount">
  <Value>0</Value>
</Attribute>
<Attribute name="wantAttributeEncrypted">
  <Value/>
</Attribute>
<Attribute name="cotlist">
  <Value>cot</Value>
</Attribute>
<Attribute name="ECPRequestIDPListFinderImpl">
  <Value>com.sun.identity.saml2.plugins.ECPIDPFinder</Value>
</Attribute>
<Attribute name="relayStateUrlList"/>
<Attribute name="idpProxyList">
  <Value/>
</Attribute>
<Attribute name="wantLogoutResponseSigned">
  <Value/>
</Attribute>
<Attribute name="saeSPLlogoutUrl"/>
<Attribute name="basicAuthUser">
  <Value/>
</Attribute>
<Attribute name="wantPOSTResponseSigned">
  <Value/>
</Attribute>
<Attribute name="alwaysIdpProxy"/>
<Attribute name="spSessionSyncEnabled">
  <Value>>false</Value>
</Attribute>
<Attribute name="spAuthncontextClassrefMapping">
  <Value>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport|0|default</Value>
</Attribute>
<Attribute name="assertionTimeSkew">
  <Value>300</Value>
</Attribute>
<Attribute name="spDoNotWriteFederationInfo"/>
<Attribute name="wantAssertionEncrypted">
  <Value/>
</Attribute>
<Attribute name="basicAuthOn">
  <Value>>false</Value>
</Attribute>
<Attribute name="useNameIDAsSPUserID">
  <Value>>false</Value>
</Attribute>
<Attribute name="attributeMap">
  <Value>*</Value>
</Attribute>
<Attribute name="autofedAttribute">
  <Value/>
</Attribute>
<Attribute name="saml2AuthModuleName">
  <Value/>
</Attribute>
<Attribute name="defaultRelayState">
  <Value/>
</Attribute>
<Attribute name="wantNameIDEncrypted">

```

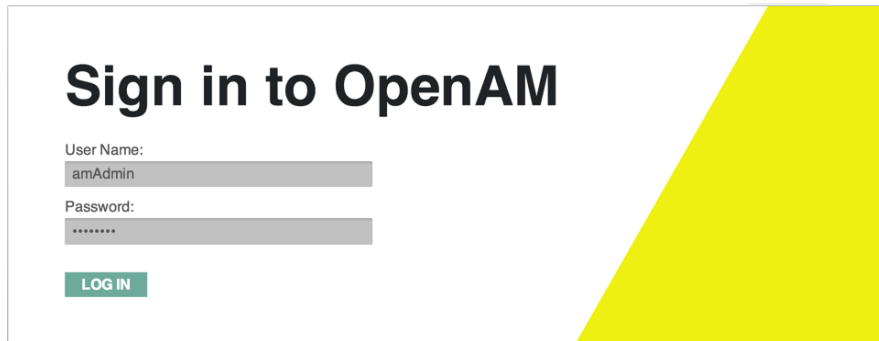


```

        <Value/>
      </Attribute>
      <Attribute name="responseArtifactMessageEncoding">
        <Value>URI</Value>
      </Attribute>
      <Attribute name="saeAppSecretList"/>
      <Attribute name="localAuthURL">
        <Value/>
      </Attribute>
      <Attribute name="saeSPURL">
        <Value>http://openam.example.com:8080/openam/spsaehandler/metaAlias/sp</Value>
      </Attribute>
      <Attribute name="transientUser">
        <Value>anonymous</Value>
      </Attribute>
      <Attribute name="autofedEnabled">
        <Value>>false</Value>
      </Attribute>
      <Attribute name="wantMNIResponseSigned">
        <Value/>
      </Attribute>
      <Attribute name="wantLogoutRequestSigned">
        <Value/>
      </Attribute>
      <Attribute name="ECPRequestIDPLISTGetComplete">
        <Value/>
      </Attribute>
      <Attribute name="spAuthncontextComparisonType">
        <Value>exact</Value>
      </Attribute>
      <Attribute name="basicAuthPassword">
        <Value/>
      </Attribute>
      <Attribute name="wantArtifactResponseSigned">
        <Value/>
      </Attribute>
      <Attribute name="spAttributeMapper">
        <Value>com.sun.identity.saml2.plugins.DefaultSPAttributeMapper</Value>
      </Attribute>
      <Attribute name="ECPRequestIDPLIST">
        <Value/>
      </Attribute>
      <Attribute name="wantMNIRequestSigned">
        <Value/>
      </Attribute>
      <Attribute name="metaAlias"/>
    </SPSSOConfig>
  </EntityConfig>

```

2. Log in to OpenAM with the default user name (amAdmin) and password.



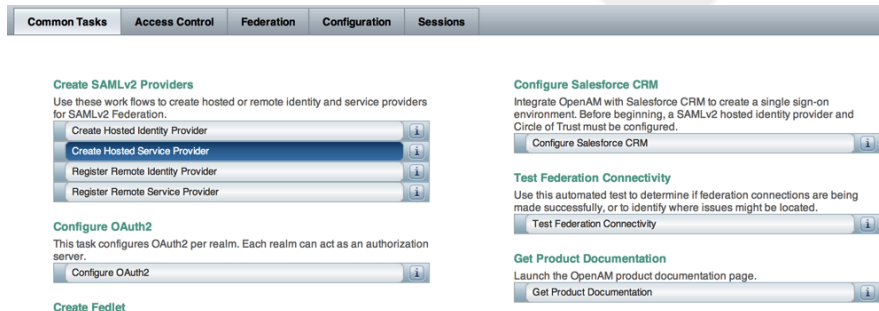
Sign in to OpenAM

User Name:
amAdmin

Password:

LOG IN

- On the Common Tasks tab, select Create Hosted Service Provider.



- On the SAML v2 Service Provider page, select Yes to indicate that you have metadata for this provider.

For the metadata file, click Upload, and browse for the sp.xml file you create previously.

For the extended data file, click Upload, and browse for the sp-extended.xml file you created previously.

- Enter bridge-sp in the New Circle of Trust field and click Configure.
- Select No when you are requested to create a remote identity provider. You will configure the remote identity provider at a later stage.

The service provider configuration is now complete.

7.2. Setting up an Identity Provider in Identity Bridge

This section describes how to configure Identity Bridge to use OpenAM as the Identity Provider. The section assumes you have installed and configured Identity Bridge, with the exception of the SSO configuration.

1. Log into the Identity Bridge administration console (<https://hostname.domain:8443/admin/>) and click on the SSO tab.
2. In the SAML Login URL field, enter the HTTP-POST Location URL from the metadata file you saved in the previous procedure (<http://openam.example.com:8080/openam/Consumer/metaAlias/sp>).

SAML Configuration

SAML Login URL

IDP.xml

```
<?xml version='1.0' encoding='UTF-8' standalone='no' ?>
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:entitydescriptor" >
  <NameIDFormat urn:oasis:names:tc:SAML:2.0:nameid-format:transient />
  <NameIDFormat urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress />
  <NameIDFormat urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified />
  <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
    Location="https://localhost:8443/connect/SSORedirect/metaAlias/idp" />
  <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
    Location="https://localhost:8443/connect/SSOPOST/metaAlias/idp" />
  <SingleSignOnService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="https://localhost:8443/connect/SSOSoap/metaAlias/idp" />
  <NameIDMappingService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="https://localhost:8443/connect/NIMSoap/metaAlias/idp" />
  <AssertionIDRequestService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="https://localhost:8443/connect/AIDReqSoap/IDPRole/metaAlias/idp" />
  <AssertionIDRequestService Binding="urn:oasis:names:tc:SAML:2.0:bindings:URI"
    Location="https://localhost:8443/connect/AIDReqUri/IDPRole/metaAlias/idp" />
</EntityDescriptor>
```

Save

3. Copy the contents of the IDP.xml text box and paste them into a text file. Save the file as IDP.xml.
4. Click Save to save the SAML Login URL.

7.3. Creating the Identity Provider in OpenAM

The IDP.xml file that you created in the previous procedure will be used to create the Identity Provider. Before you start, edit the IDP.xml file, as follows:

- Add a trailing slash to the entityID URL, for example:

```
entityID="https://localhost:8443/connect/"
```

- Make sure there is a line break between the certificate tags and the certificate content.
- Remove the -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- lines from the certificate.

The complete certificate should look something like this:

```
<ds:X509Certificate>
MIIDUDCCAjigAwIBAgIIFMvEFfjvrpIwDQYJKoZIhvcNAQENBQAwADEyMDAGA1UE
AwwpaHR0cHM6Ly9sb2NhbGhvc3Q6ODQ0My9jb25uZWNoL2luZGV4Lmhh0bWwxDAL
BgNVBAsMBE5vbmUxZDAsBgNVBAoMC05vbmUgTD10b25lMQ0wCwYDVQQGEwR0b25l
MB4XDTEzMTEyMDgyN1oXDTIzMTIxODEzMdgyN1owaDEyMDAGA1UEAwwpaHR0
cHM6Ly9sb2NhbGhvc3Q6ODQ0My9jb25uZWNoL2luZGV4Lmhh0bWwxDALBgNVBAsM
BE5vbmUxZDAsBgNVBAoMC05vbmUgTD10b25lMQ0wCwYDVQQGEwR0b25lMIIIBjAN
BgkqhkiG9w0BAQEEFAAOCAQ8AMIIBCGKCAQEAfNegVrqR2hAZEc5p89KPtq0tYLIB
Hrb0p7v/TT0Unw5PjiMxqGvJSRUvNkoddq/wvtuZtqrHsb529IOXMq+lbpgmNme
mx4wclSFCOKSeUxvv/RkQDEvyPq65UZILjFPrCAZK0MXt06Nz8+Wa5Sw6FrtsBW3
JhrxQdjYU3ml1SauU0ph643Jt0kV34q9NSF0ENp4fmpn5hDGvYyo0W9aUBvr3dKA
o6GVujUXSHj9tjc+tvFDcw9iKDCIoNRdfQstCLUQE6HJnl6cmsuoP0cPFZL5ZrpV
kT9/cQ4kiUBhJ+P7lvpJ3DCSXjU556+Fwr9ehnpHp90WRkVv4F86B9hvYQIDAQAB
MA0GCSqGSIb3DQEBDQUAA4IBAQBxAT1jJWKJPpasFyKLfnpJTe4/7tF0+EkSd2lX
tJjWajFREGrreN+mZMvUjEj5yug9SkcRb50A0Fmoszj+PtoeZg143rzi87jKI9X
EPiFkmHAcAJ5L0qGjdbSkaIvU7sd8bkzkUuUS9RD00d564piwXc/QpbbFnnKkZdU
52yNrf1+iCPdv05FUNW6UMcU7LFVbIyv5soRZBzRDbBr60U1ZJVG8b8f588YyNIU
jg9uLJNwksEpJJsANXTPhsHdZmGqNxYgJEK9qxFdhgJtqXeeWCVKWJuy0tUs6//c
GoxiZysLLY9FvHt/Lycn/epjaeK/wMP4bt1F9/4aTu8MR0N
</ds:X509Certificate>
```

After you have edited the xml configuration file, create the Identity Provider in OpenAM, as follows:

1. Log in to OpenAM with the default user name (amAdmin) and password.
2. Select Register Remote Identity Provider.
3. Select File as the location where the metadata resides.
4. Click Upload, browse for the IDP.xml file you created in the previous procedure, and click Upload file.
5. Under Circle of Trust, select Add to existing, and select the bridge-sp circle of trust.
6. Click Configure.

The remote identity provider is now created.

7. To configure the identity provider, select the Federation tab on the OpenAM home page.

Common Tasks Access Control Federation Configuration Sessions

Circle of Trust Configuration SAML 1.x Configuration

Circle of Trust Configuration

This section can be used to configure the properties for a Circle of Trust. The Entities table can be used for managing entity providers including importing and exporting of providers. Entities can be added to a Circle of Trust after they are created in the Entities table.

Circle of Trust (1 Item(s))

New... Delete

Name	Entities	Realm	Status
<input checked="" type="checkbox"/> bridge-sp	bridge-sp/saml2 http://openam.example.com:9080/openam/saml2 https://localhost:8443/connect/saml2	/	Active

Entity Providers (3 Item(s))

New... Delete Import Entity...

Name	Protocol	Type	Location	Realm
<input checked="" type="checkbox"/> bridge-sp	SAMLv2	SP	Hosted	/
<input type="checkbox"/> http://openam.example.com:9080/openam	SAMLv2	SP	Hosted	/
<input type="checkbox"/> https://localhost:8443/connect/	SAMLv2	IDP	Remote	/

- Under Entity Providers, click on the URL on which Identity Bridge is running (<https://hostname.domain:8443/connect/>).
- In the NameID Format list, edit the order of the entries so that `urn:oasis:names:tc:SAML:2.0:nameid-format:transient` is at the top of the list. You can do this by removing `urn:oasis:names:tc:SAML:2.0:nameid-format:persistent` and then adding it again so that it appears at the bottom of the list.

Do not remove any entries without readding them to the list.

NameID Format

NameID Format List

Current Values

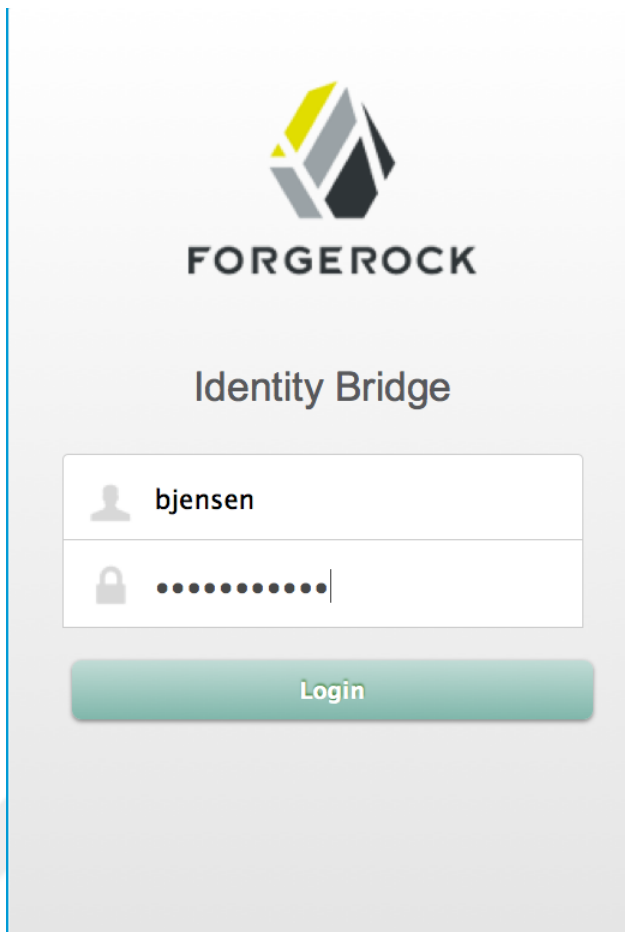
`urn:oasis:names:tc:SAML:2.0:nameid-format:transient`
`urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress`
`urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified`
`urn:oasis:names:tc:SAML:2.0:nameid-format:persistent`

New Value

List of nameid formats the requestor will use to contact. Order listed shows the order of preference


- Click Save to update the Identity Provider configuration.


Single sign on is now configured. To test the configuration, log in to the Identity Bridge user interface (<https://hostname.domain:8443/connect/>) as a regular user in Active Directory.



FORGEROCK

Identity Bridge

 bjensen



Login

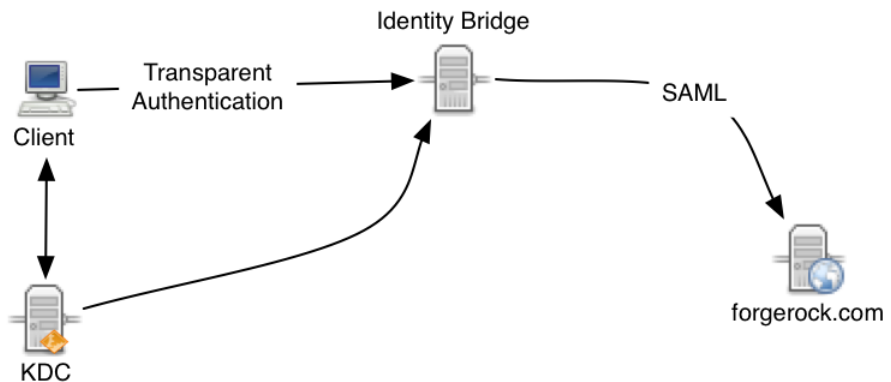
The user is routed directly to the OpenAM console.

Chapter 8. Configuring Identity Bridge for Integrated Windows Authentication

You can configure Identity Bridge so that clients use Integrated Windows Authentication (IWA) to authenticate, rather than authenticating by providing a username and password.

This chapter describes the steps required to use IWA with Identity Bridge. The chapter assumes that you are familiar with the principles of IWA, Kerberos and SPNEGO.

The following figure outlines the components involved when Identity Bridge is configured for IWA. This example assumes that the client and the Key Distribution Center are in the same Active Directory domain. This might not always be the case. The examples at the end of this section illustrate additional scenarios, when the client is not part of the domain.



The following sections describe the process for configuring Identity Bridge to use IWA. If you encounter problems during this process, see the [Troubleshooting the Integrated Windows Authentication Configuration](#).

8.1. Preparing Identity Bridge for IWA

8.1.1. Creating a Specific User for IWA

To authenticate Identity Bridge to the Key Distribution Centre (KDC) you must create a specific user entry in Active Directory whose credentials will be used for the authentication.

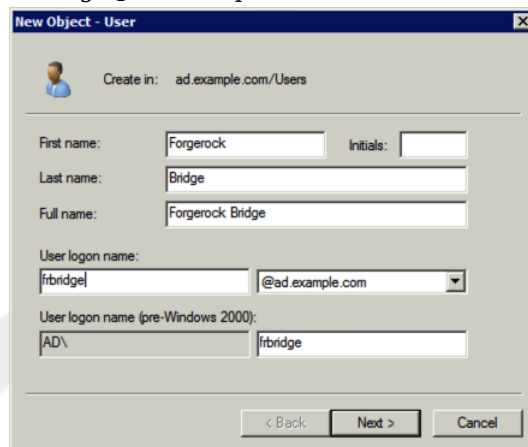
Note

It is recommended that this IWA user account is not used for anything else, that is, that you have separate user accounts for Kerberos and for the Active Directory connector.

The IWA user account is used to generate the Kerberos keytab. Therefore, if you change the password of the IWA user after you have set up IWA, you must update the keytab accordingly.

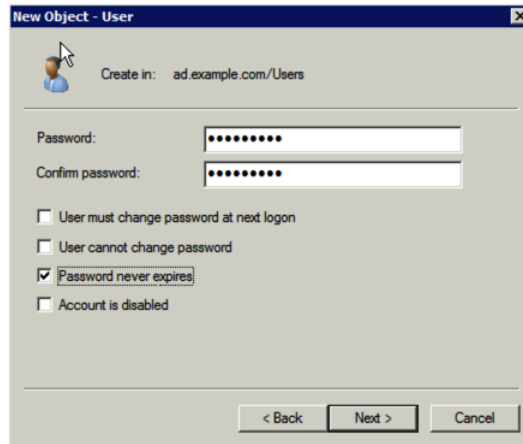
Create a new user in Active Directory as follows:

1. Provide a login name for the user that reflects its purpose, for example, `frbridge@ad.example.com`.



The screenshot shows the 'New Object - User' dialog box. At the top, it says 'Create in: ad.example.com/Users'. Below this are several input fields: 'First name' with 'Forgerock', 'Last name' with 'Bridge', and 'Full name' with 'Forgerock Bridge'. There is also an 'Initials' field which is empty. Below these is the 'User logon name' section, which has a text field containing 'frbridge' and a dropdown menu showing '@ad.example.com'. Below that is the 'User logon name (pre-Windows 2000)' section, which has two text fields: the first contains 'AD\' and the second contains 'frbridge'. At the bottom of the dialog are three buttons: '< Back', 'Next >', and 'Cancel'. The 'Next >' button is highlighted with a blue border.

2. Enter a password for the user. Check the *Password never expires* option and leave all other options unchecked.



3. Click Finish to create the user.

8.1.2. Linking the User Account to the Service Principal and Creating the Keytab File

Kerberos authentication uses Service Principal Names (SPNs) to identify specific services to which clients have access. The first time a client makes a request for authentication, the client must include the SPN of the ForgeRock service in its request. To do so, the client user account must be identified with the SPN of the service.

In addition to linking the user account to the SPN, you must create a Kerberos keytab file (krb5.keytab) for the host on which Identity Bridge is running. The keytab file enables Identity Bridge to validate the Kerberos tickets that it receives from client browsers.

You can link a user account to an SPN in two ways:

- Using the **ktpass** command. For more information, see the [ktpass documentation](#).
- Using the **setspn** command. For more information, see the [setSPN documentation](#).

When you use **ktpass** to set the SPN, and then create the keytab file in a second step, inconsistencies can arise with the key version number (kvno). If a "checksum" error occurs during the initial validation of the Identity Bridge credentials against the KDC, it is likely that the key version number is inconsistent between the user account entry and the keytab. To prevent this situation, link the user account to the SPN and create the keytab in a single command.

Use the **ktpass** tool, included in the Windows Server toolkit, to link the user account and create the keytab file. The correct crypto parameter to use with **ktpass** depends on your version of the Windows toolkit. Run **ktpass -?** to determine the appropriate crypto parameter for your version.

The **ktpass** command must be run on the Active Directory domain controller.

The following example links the user account to the SPN, and creates a keytab file (named `identityConnect.HTTP.keytab`) for Identity Bridge.

```
C:\Users\Administrator>ktpass
-princ HTTP/frbridge.ad.example.com@AD.EXAMPLE.COM
-mapUser AD\frbridge
-mapOp set
-pass Password1
-crypto RC4-HMAC-NT
-pType KRB5_NT_PRINCIPAL
-kvno 0
-out identityConnect.HTTP.keytab

Targeting domain controller: host.ad.example.com
Using legacy password setting method
Successfully mapped HTTP/frbridge.ad.example.com to frbridge.
Key created.
Output keytab to identityConnect.HTTP.keytab:
Keytab version: 0x502
keysize 69 HTTP/frbridge.ad.example.com@AD.EXAMPLE.COM ptype 1 (KRB5_NT_PRINCIPAL)
vno 0 etype 0x17 (RC4-HMAC) keylength 16 (0x5858d47a41e40b40f294b3100bea611f)
```

The command takes the following options:

- **-princ** specifies the service principal name in the format *service/host-name@realm*

In this example (`HTTP/frbridge.ad.example.com@AD.EXAMPLE.COM`), the client browser constructs an SPN based on the service name (`HTTP`), the FQDN of the URL (`frbridge.ad.example.com`) and the realm name (`AD.EXAMPLE.COM`). This example assumes that users will access Identity Bridge at the URL `https://frbridge.ad.example.com:8443/connect/`.

The service name for SPNEGO web authentication must be `HTTP`. The service name for Kerberos authentication can be any strings that are permitted by the Key Distribution Center. The host name must be a fully qualified host name. The realm name must be in upper case.

- **-mapUser** specifies the name of the user account to which the principal should be mapped (the user account that you created in the previous section).
- **-mapOp** specifies how the account is linked. Use `set` to set the first user name to be linked. The default (`add`) adds the value of the specified local user name if a value already exists.

- `-pass` specifies a password for the principal user name. Use "*" to prompt for a password.
- `-crypto` Specifies the cryptographic type of the keys that are generated in the keytab file. The correct `crypto` parameter to use depends on your version of the Windows toolkit. Run **ktpass -?** to determine the appropriate `crypto` parameter for your version.
- `-ptype` Specifies the principal type. The recommended principal type is `KRB5_NT_PRINCIPAL`.
- `-kvno` specifies the key version number. Set the key version number to 0.
- `-out` specifies the name of the keytab file that will be generated. Use `identityConnect.HTTP.keytab`.

Note that the keys that are stored in the keytab are similar to user passwords. You must therefore protect the Kerberos keytab file in the same way that you would protect a file containing passwords.

For more information about the **ktpass** command, see the [ktpass reference](#) in the Windows server documentation.

If your service is behind a load balancer, the service principal name that is configured by **ktpass** might not be the same as the alias that clients use to request the service. In this case, you can use **setspn** to link the account to whatever alias will be used by the clients.

1. Use the **setspn -L** command to check which SPNs are currently registered for that user account:

```
C:\Users\Administrator>setspn -L frbridge
Registered ServicePrincipalNames for CN=Identity Bridge,CN=Users,DC=ad,DC=example,DC=com:
HTTP/frbridge.ad.example.com
```

2. Use the **setspn -S** command to link a new SPN to the user account. The `-S` option also verifies that there are no duplicate SPNs.

```
setspn -S service name/host name user account
```

For example:

```
C:\Users\Administrator> setspn -S HTTP/forgerockbridge.example.com@AD.EXAMPLE.COM frbridge
Checking domain DC=ad,DC=example,DC=com
Registering ServicePrincipalNames for CN=Identity Bridge,CN=Users,DC=ad,DC=example,DC=com
HTTP/forgerockbridge.example.com
Updated object
```

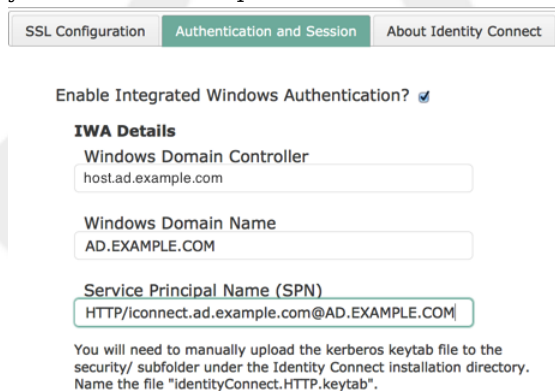
3. Use the **setspn -L** command again to verify that the new SPN has been registered correctly:

```
C:\Users\Administrator>setspn -L frbridge
Registered ServicePrincipalNames for CN=Identity Bridge,CN=Users,DC=ad,DC=example,DC=com:
HTTP/forgerockbridge.example.com
HTTP/frbridge.ad.example.com
```

8.1.3. To Enable IWA in Identity Bridge

IWA is disabled by default. Follow this procedure to enable IWA and to configure the authentication filter.

1. Log in to the Identity Bridge administrative interface (for example `https://hostname.domain:8443/admin`).
2. Click Settings in the top right corner and select the Authentication and Session tab.
3. Select Enable Integrated Windows Authentication.
4. Enter the following information:
 - *Windows Domain Controller*. Enter the FQDN or IP address of the Key Distribution Center (KDC), for example, `host.ad.example.com`. The value of this field defaults to the host machine on which your Active Directory server is located.
 - *Kerberos Realm*. Enter the name of the Kerberos realm, for example, `AD.EXAMPLE.COM`.
 - *Windows Domain Name*. Enter SPN of the user account created specifically for IWA, for example, `HTTP/frbridge.ad.example.com@AD.EXAMPLE.COM`.
 - *Upload Kerberos Keytab File*. Click Browse to locate the keytab file that you created in the previous section.



SSL Configuration Authentication and Session About Identity Connect

Enable Integrated Windows Authentication? ☒

IWA Details

Windows Domain Controller
host.ad.example.com

Windows Domain Name
AD.EXAMPLE.COM

Service Principal Name (SPN)
HTTP/forgerockbridge.ad.example.com@AD.EXAMPLE.COM

You will need to manually upload the kerberos keytab file to the security/ subfolder under the Identity Connect installation directory. Name the file "identityConnect.HTTP.keytab".

Note

Name resolution must be valid for KDC (the server that is specified in the Windows Domain Controller field). If this is not the case, Identity Bridge will be unable to contact the KDC. In a typical Windows environment, the KDC is part of the DNS record. This might not be the case if Identity Bridge is located inside a DMZ.

8.2. Configuring Your Browser for SPNEGO

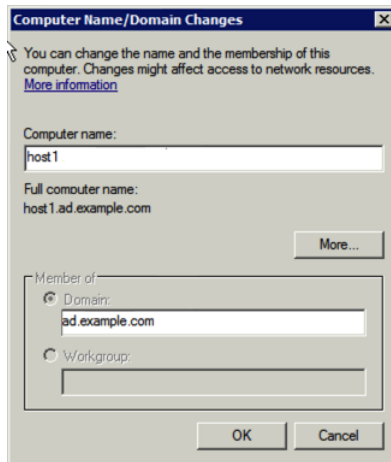
Identity Bridge uses the Simple and Protected GSSAPI Negotiation Mechanism (SPNEGO), to negotiate authentication to the remote server. You must have a browser installed that supports SPNEGO authentication. Most modern browsers support SPNEGO but require some additional configuration to make it work.

The configuration required varies, depending on the operating system of the client from which you will access Identity Bridge.

8.2.1. To Configure Internet Explorer for SPNEGO on a Windows Client

For Windows clients, the easiest way to configure Internet Explorer is for the Windows client to join the Active Directory domain (ad.example.com in our example). To join a Windows client to the Active Directory domain, follow these steps:

1. From the Control Panel, select System Properties > Advanced > Computer Name > Change.
2. In the *Member of* panel, select *Domain* and enter the name of the Active Directory domain.



To join the domain, you will need to provide the administrator's credentials.

3. After you have joined the domain, reboot your Windows client.
4. Launch Internet Explorer and check that the Identity Connect URL is included in the list of "Trusted Sites", as follows:
 - From the Tools menu, select Internet Options and select the Security tab.
 - Select Trusted Sites and click the Sites button.
 - In the list of Websites, check that the URL for Identity Bridge appears.
 - Click the Custom Level... button.
 - In the Settings pane, scroll down to User Authentication.

Select Automatic logon with current username and password.

5. Select the Advanced tab and scroll down to the list of Security settings.
Make sure that Enable Integrated Windows Authentication is selected.
6. You should now be able to access Identity Bridge through your Internet Explorer browser.

It is advisable to check, at this point, that the *Identity Provider Login URL* in your ForgeRock single sign-on settings matches the Identity Bridge login URL.

For information about changing your single sign-on settings, see the chapter on [Configuring Single Sign-On](#).

8.2.2. To Configure Firefox for SPNEGO on a Windows Client

Firefox supports SPNEGO, but it is disabled by default. To enable SPNEGO, follow these steps:

1. Enter the URL `about:config` in the address bar.

Click past the warnings about your warranty.
2. At the top of the page, search for `negotiate-auth` to filter the results.

Double-click on `network.negotiate-auth.trusted-uris`.
3. In the dialog box, enter the Identity Bridge URL and click OK.

You should now be able to access Identity Bridge through your Firefox browser.

8.2.3. To Enable Kerberos Authentication Mac OS

On a Mac OS client, there are two ways to enable Kerberos authentication.

- Join the Mac OS client to the Active Directory domain.
- Edit the `krb5.conf` to generate tickets by using **kinit**.

8.2.3.1. Joining a Mac OS X Client to an Active Directory Domain

Before you attempt to join an Active Directory domain from a Mac OS client, ensure that the Mac has the following basic networking configuration:

- An IP address and a subnet mask
- A DNS hostname
- A connection to a Windows DNS server

When you join the Mac to the domain, you will need to use the credentials of the domain administrator, or of a user account with the required privileges.

On your Mac client, follow these steps to join the AD domain. These instructions are for Mac OS X Lion. You might need to adjust these instructions for your particular Mac OS version.

1. Select System Preferences > Users and Groups > Login Options

Click the Lock icon to enable you to change these settings.

2. Click the Join button next to Network Account Server.
3. Enter the name of the KDC server (ad.example.com in our example)
4. Enter the credentials of the administration user for the KDC server and click OK.

After you have added the Mac to the domain, launch your browser and edit the list of network.negotiate-auth.trusted-uris, as described in the previous section.

8.2.3.2. Using a Kerberos Configuration File to Generate Tickets With kinit

The Kerberos configuration file (krb5.conf) contains configuration information that is required by the Kerberos library, including the default Kerberos realm and the location of the Kerberos Key Distribution Center (KDC).

1. Create a file named krb5.conf and place it in the /etc directory. The file contents must include the Kerberos information specific to your site, including the permitted encryption types. The following example shows the Kerberos configuration file for the example described previously.

```
$ more /etc/krb5.conf
[libdefaults]
    default_realm = AD.EXAMPLE.COM
    default_tkt_enctypes = des3-cbc-sha1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc
    default_tgs_enctypes = des3-cbc-sha1 arcfour-hmac aes256-cts des-cbc-md5 des-cbc-crc

[realms]
    AD.EXAMPLE.COM = {
        admin_server = 192.0.2.0
        kdc = 192.0.2.0
        kpasswd = 192.0.2.0
    }

[domain_realm]
    .yourdomain.com = AD.EXAMPLE.COM
    localhost = AD.EXAMPLE.COM
```

When the Kerberos configuration file is in place, you can generate the initial TGT Kerberos ticket that will be used by Safari, Chrome and Firefox to request additional tickets.

2. Use **kinit** to generate the initial ticket:

```
$ kinit admin@AD.EXAMPLE.COM
admin@AD.EXAMPLE.COM's Password: *****
```


The format in which the user name is entered depends on how your client machine is configured (so might be simply **\$ kinit admin** in your case).

3. Run the **klist** command to verify that the ticket has been created.

```
$ klist -v
  Credentials cache: API:501:68
    Principal: admin@AD.EXAMPLE.COM
    Cache version: 0

  Server: krbtgt/frbridge.AD.EXAMPLE.COM@AD.EXAMPLE.COM
  Client: admin@AD.EXAMPLE.COM
  Ticket etype: aes256-cts-hmac-sha1-96, kvno 2
  Ticket length: 1051
  Auth time: Jun 4 16:10:43 2013
  End time: June 5 02:09:01 2013
  Ticket flags: pre-authent, initial, proxiable, forwardable
  Addresses: addressless
```

After the ticket has been generated, launch your browser and edit the list of `network.negotiate-auth.trusted-uris`, as described in the previous section. You should now be able to access Identity Bridge through your browser.

DRAFT

Chapter 9. Customizing the Identity Bridge Interface

This chapter describes how to customize various aspects of the Identity Bridge User Interface.

9.1. Changing the Password Reset Link

You can reroute password reset in the event that a user has forgotten his password, by specifying an external URL to which password reset requests are sent.

To set an external URL to handle password resets:

1. Click Settings at the top right of the Identity Bridge window.
2. Select the Authentication and Session tab.
3. In the *Reset Password Link* field, enter the URL to which password reset requests should be sent.

9.2. Changing the Session Timeout

By default, an Identity Bridge UI session times out after 30 minutes of inactivity, or 120 minutes after the initial login, whichever happens first.

To adjust the login session length:

1. Click Settings at the top right of the Identity Bridge window.
2. Select the Authentication and Session tab.
3. In the *Maximum session lifetime* field, enter the maximum number of minutes that a session can be live, after the initial login.
4. In the *Session idle timeout* field, enter the maximum number of minutes that a session can be idle, before the user is logged out automatically.

Note

Changes to the session timeout settings only take effect for a new session, that is, you need to log out and log back in to see the effect of the change.

DRAFT

Chapter 10. Securing an Identity Bridge Deployment

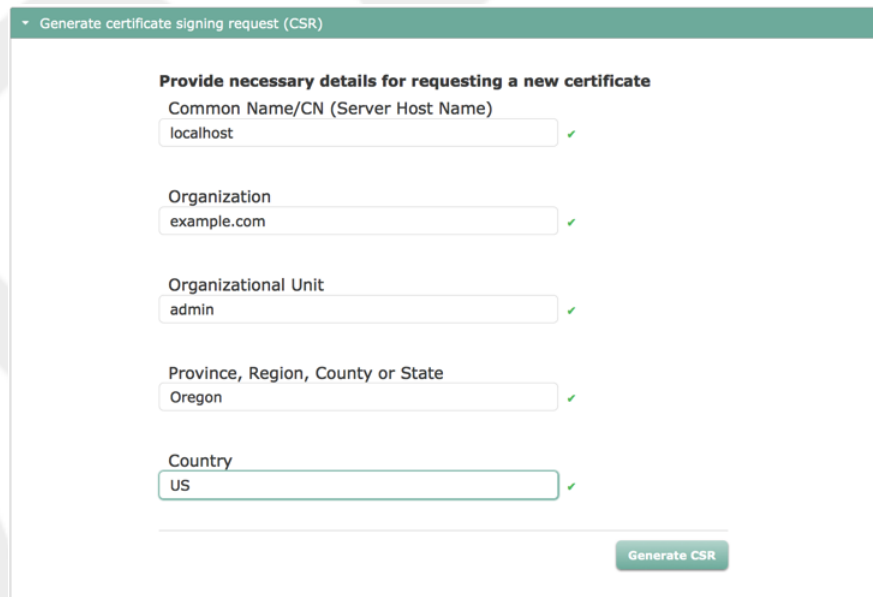
This chapter describes how to manage keys and certificates to establish trust between users and Identity Bridge.

10.1. Managing SSL Certificates

Identity Bridge provides a self-signed certificate for evaluation purposes. In production systems, it is recommended that you use a certificate that has been signed by a certificate authority to establish trust between the users and Identity Bridge. A CA-signed certificate will prevent users from seeing the certificate warning when they log in to their corporate application via Identity Bridge.

The following procedure shows how to generate a certificate signing request (CSR) and to import the signed certificate into Identity Bridge's keystore.

1. Log into the Identity Bridge administration interface, (for example, `https://hostname.domain:8443/admin`).
2. Click Settings in the top right corner.
3. On the SSL Configuration tab, enter the details of the CSR, then click Generate CSR.



Generate certificate signing request (CSR)

Provide necessary details for requesting a new certificate

Common Name/CN (Server Host Name)
localhost ✓

Organization
example.com ✓

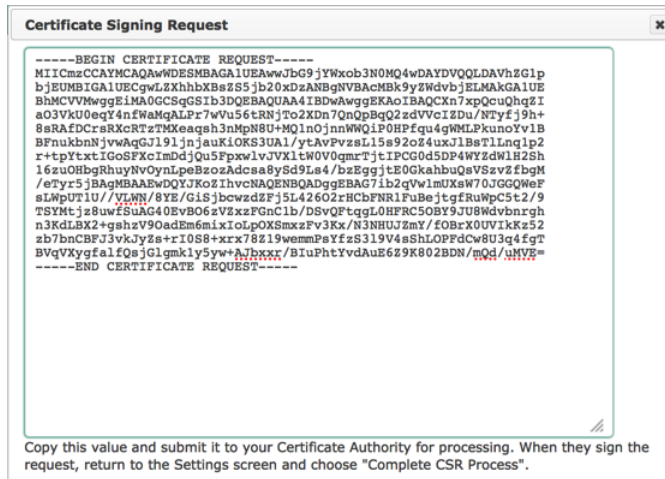
Organizational Unit
admin ✓

Province, Region, County or State
Oregon ✓

Country
US ✓

Generate CSR

- The resulting CSR is displayed in a new window. Copy the contents of the CSR and submit it to your Certificate Authority for signing.



- When the signed certificate is returned from your certificate authority, click Settings again, then click Complete CSR Process on the SSL Configuration tab.
- Copy and paste the contents of the CA-signed certificate (PEM file) into the first text box.

Complete CSR process

If you have started a previous CSR process, enter the signed certificate you received from your Certificate Authority (PEM format):

```

-----BEGIN CERTIFICATE-----
MIICmCCAYMCAQAwHDESMBAQIUEAwWJbG9jYWxob3N0MQ4wDAYDVQQLEDAVh2G1p
b1EUMBIGA1UECgwLZXhhbXBsZS5jb20xZDZANBgNVBACMBk9yZWdvbjELMAkGA1UE
BhMCVVwggE1MA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCXn7xpQcuQhgzI
aO3Vku0eqY4nfWmQALPr7wVu56tRNjTo2XDn7QnQpBgQ2zdVVCIZDu/NTyfj9h+
8sRAfDCrRxcRTzTMXeaqsh3nMpN8U+MQ1nOjnnWwQIP0HPfqu4gWMLPkunoYv1B
BFnukbnNjvwAgGj191jnJauKiOKS3UA1/ytAvPvzsL15s92oZ4uxJ1BsTLLnq1p2
r+tpYtXtIGoSfKcImDdJQu5Fpxw1vJVXitW0V0gmrtjtIPCG0d5DP4WYzdW1H2Sh
16zuOBbgRhuyNvOynLpeBzoZAdca8ySd9Le4/bzEggjtEOGkahbuQsVszvZfBgM
/eTyr5jBAqMBAAEwdQYJKozThvcNAQENBQADggEBAG71b2qVwlmUXeW70JGGQWwF
sLWpUT1U//VLWN/8YE/GisjbcwzdZFj5L426O2rHCbFNr1FuBeJtgfRuWpC5t2/9
TSYmtjz8uwfsuAG40EvBO6zVZxzFGnClb/DSvQFtqgL0HFRFC5OBY9JU8Wdvbnrgh
n3KdLBX2+gshzV9OadEm6mIXIoLpOXSmxzFv3Kx/N3NHUJZmY/fOBrX0UVIKKz52
zb7bnCBFJ3vkJyZs+rIO88+xxr78Z19wemmPeYfzS319V4sShLOPFdCw8U3q4fgT
BVqVXygfalfQsJGlgmkly5yw+A7bxxcr/BIuPhtYvdAuE6Z9K802BDN/mQd/uMVE=
-----END CERTIFICATE-----

```

If your CA provided intermediate or root certificates along with your signed certificate, upload those as well:

Include Additional Certificate

Note: you may need to restart your browser after uploading the new certificate, as the old certificate may be cached.

Upload Signed Certificate

- If your CA has provided an intermediate or root certificate, click Include Additional Certificates and copy and paste the contents of those certificates.
- Click Upload Signed Certificate to import the CA certificate and the corresponding certificate chain into the keystore.
- Restart your browser after you have uploaded the new certificates, because the old self-signed certificate might be cached.

10.2. Configuring Identity Bridge for Client Certificate Authentication

By default, client certificate authentication is disabled in Identity Bridge. If you want to use mutual authentication, you must adjust the web server SSL settings to enable client certificate authentication.

To enable client certificate authentication, edit the `/path/to/forgerockBridge/conf/jetty.xml` file as follows:

```
<Set name="wantClientAuth">true</Set>
```

```
<Set name="needClientAuth">true</Set>
```

Chapter 11. Installing an Alternative Repository

By default, Identity Bridge stores its configuration in an internal OrientDB repository. (User entries are not stored in the internal repository.) In certain situations, you might want to use your own SQL database to store the server configuration (for example, if you are setting up a clustered Identity Bridge deployment, or if you have an existing SQL database that you would prefer to use).

Identity Bridge supports the use of [MySQL](#) as a repository. For details of the supported versions, see [Supported Repositories](#) in the *Release Notes*.

Procedure 11.1. To Set Up Identity Bridge With MySQL

Set up Identity Bridge to use the MySQL repository, as described in the following steps. This procedure assumes that:

- Identity Bridge has been downloaded and unzipped *but not configured*, that is, the setup process has not been run.
 - MySQL has been installed, either on the host on which Identity Bridge will run, or on a host that is accessible to the Identity Bridge instance.
1. Download MySQL Connector/J, version 5.1 or later from the MySQL website. Unpack the delivery, and copy the .jar into the forgerockBridge/bundle directory.

```
$ cp mysql-connector-java-version-bin.jar /path/to/forgerockBridge/bundle/
```

2. Remove the default OrientDB configuration file (/path/to/forgerockBridge/conf/repo.orientdb.json) from the configuration.

```
$ cd /path/to/forgerockBridge/conf/  
$ rm repo.orientdb.json
```

3. Copy the MySQL JDBC configuration file (/path/to/forgerockBridge/db/scripts/mysql/repo.jdbc-mysql.json) to the forgerockBridge/conf directory and rename it repo.jdbc.json.

```
$ cd /path/to/forgerockBridge/conf  
$ cp ../db/scripts/mysql/repo.jdbc-mysql.json repo.jdbc.json
```

4. Import the data definition language script for Identity Bridge into MySQL.

```
$ cd /path/to/mysql
```

```
$ ./bin/mysql -u root -p < /path/to/forgerockBridge/db/scripts/mysql/openidm_forgerockBridge-MySQL.sql
Enter password:
$
```

Enter a root password that you will use in future to protect access to the MySQL database.

This step creates a database named `openidm` for use as the internal repository, and a user `openidm` with password `openidm` who has all the required privileges to update the database.

Load the `openidm` database and verify that you can display the default Identity Bridge tables.

```
$ cd /path/to/mysql
$ ./bin/mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 5.5.19 MySQL Community Server (GPL)
...
mysql> use openidm;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_openidm |
+-----+
| auditaccess        |
| auditactivity      |
| auditrecon         |
| clusterobjectproperties |
| clusterobjects     |
| configobjectproperties |
| configobjects      |
| genericobjectproperties |
| genericobjects     |
| internaluser       |
| links              |
| managedobjectproperties |
| managedobjects     |
| objecttypes        |
| orphanarium        |
| orphanariumproperties |
| schedulerobjectproperties |
| schedulerobjects   |
| uinotification     |
+-----+
19 rows in set (0.00 sec)
```

The table names are similar to those used with the embedded OrientDB repository.

5. Update `forgerockBridge/conf/repo.jdbc.json` as necessary, to reflect the details of your MySQL deployment.

```
"connection" : {  
  "dbType" : "MYSQL",  
  "jndiName" : "",  
  "driverClass" : "com.mysql.jdbc.Driver",  
  "jdbcUrl" : "jdbc:mysql://localhost:3306/openidm?characterEncoding=utf8",  
  "username" : "openidm",  
  "password" : "openidm",  
  "defaultCatalog" : "openidm",  
  "maxBatchSize" : 100,  
  "maxTxRetry" : 5,  
  "enableConnectionPool" : true,  
  "connectionTimeoutInMs" : 30000  
},
```

Specifically, make sure that the username and password used to access the database are correct, and that the `jdbcUrl` reflects the location of the database. The MySQL database can be either local (that is, running on the same host as the Identity Bridge instance) or remote (running on a different host to the Identity Bridge instance). If the database is on a remote host, adjust the `jdbcUrl` property accordingly, for example:

```
"jdbcUrl" : "jdbc:mysql://remoteMySQLServer.domain.com:3306/openidm?characterEncoding=utf8"
```

When you have set up MySQL for use as the Identity Bridge internal repository, start Identity Bridge and check the output log (`logs/console.out`), to make sure that the startup has been successful.

```
$ cd /path/to/forgerockBridge  
$ nohup ./startup.sh > logs/console.out 2>&1&
```

```
$ tail -f logs/console.out  
Executing ./startup.sh...  
Using OPENIDM_HOME: /path/to/forgerockBridge  
Using OPENIDM_OPTS: -Xmx1024m -Xms1024m  
Using LOGGING_CONFIG: -Djava.util.logging.config.file=  
/path/to/forgerockBridge/conf/logging.properties  
Using boot properties at /path/to/forgerockBridge/conf/boot/boot.properties  
->
```

Log in to the Identity Bridge administration console (<https://hostname.-domain:8443/admin/>) to confirm that you can access the UI and continue the configuration with the MySQL repository.

DRAFT

Chapter 12. Deploying Identity Bridge for High Availability

To ensure availability of the service, you can deploy multiple Identity Bridge instances. In a highly available configuration, only the *primary* instance includes a database. Additional *secondary* instances point to the database of the primary instance. The secondary instances maintain a cached copy of the configuration, and of the list of ignored users, in memory. (For more information about ignored users, see [Overview of the Synchronization Process](#).)

Each secondary instance also contains its own local keystore. The required security certificates are copied into the keystore of the secondary instance from the shared (primary instance) repository when the secondary instance is first brought online.

In the event of the primary instance failing, the secondary instances continue to serve login requests until the primary instance comes back online. Reconciliation and synchronization operations are suspended until the primary instance is back online.

Specific configuration changes must be made to configure multiple instances that use a shared repository. These configuration changes are described in this chapter.

The configuration differs slightly, depending on whether you are using the default embedded OrientDB repository, or an external MySQL repository. For information about setting up a MySQL repository, see [Installing an Alternative Repository](#).

12.1. Configuring High Availability With OrientDB

Procedure 12.1. To Configure the Primary Instance for High Availability

1. Unpack the Identity Bridge zip file, as described in [Downloading, Installing, and Starting Identity Bridge](#), but do not set up Identity Bridge.
2. In your preferred text editor, edit the `conf/boot/boot.properties` file as follows:
 - a. Change the IP address on which the embedded database listens. The address (`openidm.embeddeddb.ip`) should be changed from the default localhost (`127.0.0.1`) to all configured IPv4 addresses on all interfaces (`0.0.0.0`).

```
$ grep openidm.embeddeddb.ip /path/to/forgerockBridge/conf/boot/boot.properties  
openidm.embeddeddb.ip=0.0.0.0
```

- b. Specify a unique identifier for this Identity Bridge instance by setting the `openidm.node.id`. For example:

```
$ grep openidm.node.id /path/to/forgerockBridge/conf/boot/boot.properties  
openidm.node.id=IdentityConnect1
```

- c. Specify the instance type in the cluster by adding the following line to the `boot.properties` file:

```
openidm.instance.type=clustered-first
```

3. Set up and configure Identity Bridge, as described in [Downloading, Installing, and Starting Identity Bridge](#).

Procedure 12.2. To Configure Additional Instances for High Availability

1. Unpack the contents of the .zip file into the install location on the secondary host, but do not set up Identity Bridge.
2. In your preferred text editor, edit the `conf/boot/boot.properties` file as follows:
 - a. Change the `openidm.repo.orientdb.dburl` property to point Identity Bridge to the OrientDB database of the primary instance, in the format `primary-node-IP:primary-node-dbport/db/openidm`. For example, if the primary host URL is `host1.example.com`, the value would be as follows:

```
$ grep openidm.repo.orientdb.dburl /path/to/forgerockBridge/conf/boot/boot.properties  
openidm.repo.orientdb.dburl=remote:host1.example.com:2424/db/openidm
```

The DB port for the primary node is 2424, unless you have changed it.

- b. Specify a unique identifier for this Identity Bridge instance by setting the `openidm.node.id`. For example:

```
$ grep openidm.node.id /path/to/forgerockBridge/conf/boot/boot.properties  
openidm.node.id=IdentityConnect2
```

- c. Specify the instance type in the cluster by adding the following line to the `boot.properties` file:

```
openidm.instance.type=clustered-additional
```

3. Edit the `conf/system.properties` file, to prevent Identity Bridge from reading its configuration from the configuration files, by uncommenting the line `openidm.fileinstall.enabled=false`.

This forces Identity Bridge to read its configuration only from memory or from the repository, in this case, the repository of the primary instance.

```
$ grep openidm.fileinstall /path/to/forgerockBridge/conf/system.properties
openidm.fileinstall.enabled=false
```

4. Copy the `truststore` file from the primary instance to the secondary instance. For example, run the following command *on the primary instance*, substituting the hostname or IP address of the secondary instance:

```
$ cd /path/to/forgerockBridge
$ scp security/truststore admin@host2.example.com:/home/testuser/forgerockBridge/security/
```

5. Set up the second Identity Bridge instance.

```
$ cd /path/to/forgerockBridge
$ ./setup.sh
```

The second instance reads the configuration from the first instance. Therefore, pointing your browser to `https://host2.example.com:8443/-admin` should show the existing Identity Bridge configuration from the first host.

6. Follow this same procedure, specifying a unique ID, for each additional Identity Bridge instance.

12.2. Configuring High Availability With MySQL

This procedure describes how to configure multiple Identity Bridge instances for high availability, when you are using a remote MySQL database, instead of the default embedded OrientDB repository.

1. On each host that will contain an Identity Bridge instance, unpack the contents of the .zip file into the install location, but do not set up Identity Bridge.
2. Configure each Identity Bridge instance for a remote MySQL database, as described in [Installing an Alternative Repository](#).

When you edit the `/path/to/forgerockBridge/repo.jdbc.json` file (Step 5 of this procedure), set the `"jdbcUrl"` property to point to the remote MySQL server. For example:

```
"jdbcUrl" : "jdbc:mysql://server-ip:3306/openidm?characterEncoding=utf8"
```

where *server-ip* is the IP address of the server on which the MySQL server is located.

3. On each Identity Bridge instance, edit the `conf/boot/boot.properties` file, as follows:

- a. Specify a unique identifier for the instance.

For example, on the primary instance:

```
$ grep openidm.node.id /path/to/forgerockBridge/conf/boot/boot.properties  
openidm.node.id=Bridge1
```

On subsequent instances, the `openidm.node.id` can be set to `Bridge2`, `Bridge3`, and so forth. You can choose any value, as long as they are unique.

- b. Specify the instance type in the cluster.

On the primary instance, add the following line to the `boot.properties` file:

```
openidm.instance.type=clustered-first
```

On subsequent instances, add the following line to the `boot.properties` file:

```
openidm.instance.type=clustered-additional
```

4. On all instances except the primary instance, edit the `conf/system.properties` file, to prevent Identity Bridge from reading its configuration from the configuration files, by uncommenting the line `openidm.fileinstall.enabled=false`.

This forces Identity Bridge to read its configuration only from the repository, in this case, the repository of the primary instance.

```
$ grep openidm.fileinstall /path/to/forgerockBridge/conf/system.properties  
openidm.fileinstall.enabled=false
```

5. Start up the primary Identity Bridge instance and configure it.

Make sure that when you access initially access this Identity Bridge instance, you use the FQDN (`https://host.domain:8443/admin`) and not `localhost`.

6. Copy the truststore file from the primary instance to the secondary (and additional) instances. For example, run the following command *on the primary instance*, substituting the hostname or IP address of each secondary instance:

```
$ cd /path/to/forgerockBridge
$ scp security/truststore admin@host2.example.com:/home/testuser/forgerockBridge/security/
```

7. Start up the secondary (and additional) instances.

Additional instances read the configuration from the first instance, so requests to `https://host2.example.com:8443/connect` should read the existing Identity Bridge configuration from the first host.

12.3. Configuring a Load Balancer

After you have configured multiple Identity Bridge instances to work together in a cluster, you can configure a load balancer of your choice to distribute client connections between the instances.

Identity Bridge has been tested with Nginx Version 1.2.9, but any load balancer that supports sticky session configuration should be adequate.

If you configure a load balancer for Identity Bridge, you must specify that the logout from your application be directed to the load balancer.

The load balancer should not send clients to the secondary hosts for administration or configuration of Identity Bridge. Configuration should be handled only by the primary host. Therefore, `https://host2.example.com:8443/admin`, for example, should not be accessible through the load balancer.

DRAFT

Chapter 13. Advanced Configuration

This chapter provides additional information about the Identity Bridge setup. The information in this chapter is not required for you to be able to get Identity Bridge up and running, but might help you to understand some of the lower level configuration, and might provide some assistance when troubleshooting an installation. Additional troubleshooting information is provided in the chapter on [Troubleshooting](#).

13.1. Querying the Internal Repository

Identity Bridge is provided with an internal noSQL database, OrientDB, for use as the internal repository out of the box.

If you want to query the internal noSQL database, you can download OrientDB (version 1.5.0) from <https://github.com/orientechnologies/orientdb/wiki/Download>. You will find the shell console in the bin directory. Start OrientDB console using either **console.sh** or **console.bat**, and then connect to the running Identity Bridge instance, with the **connect** command. The default Identity Bridge database name is `openidm` and the default username and password are `admin` and `admin`.

```
$ /path/to/orientdb-graphed-/bin/console.sh
OrientDB console v.1.5.0 (build @BUILD@) www.orientechnologies.com
Type 'help' to display all the commands supported.

Installing extensions for GREMLIN language v.2.4.0-SNAPSHOT

orientdb> connect remote:localhost/openidm admin admin
Connecting to database [remote:localhost/openidm] with user 'admin'...OK

orientdb>
```

When you have connected to the database, you might find the following commands useful.

info

Shows classes and records

select * from managed_group

Shows the groups configured in the Active Directory.

select * from audit_recon

Shows all reconciliation audit records

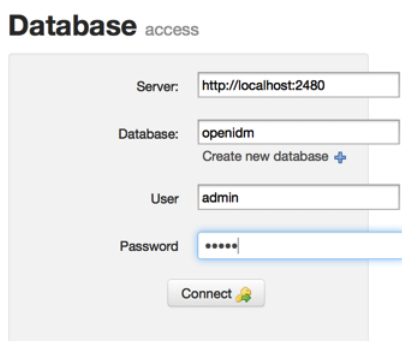
This table is created when you run reconciliation.

You can also use OrientDB Studio to query the OrientDB repository. Access to OrientDB Studio is disabled by default. After you have installed and started Identity Bridge, enable access to OrientDB Studio by setting the `studioUi` enabled property to `true` in the `/path/to/forgerockBridge/conf/repo.orientdb.json` file.

```
"embeddedServer" : {  
  "enabled" : true,  
  "studioUi" : {  
    "enabled" : true  
  },  
}
```

Restart Identity Bridge, as described in [Stopping and Restarting Identity Bridge](#).

Point your browser to <http://localhost:2480/studio/>. In the initial Authentication Required window, click Cancel, then enter your authentication details in the Database access pane.



Database access

Server:

Database:
[Create new database](#)

User:

Password:

The default database name is `openidm`. The default administrator username and password are `admin` and `admin`. The `admin` user has unrestricted access to all database functions.

Click **Connect** to connect to the repository.

For more information about OrientDB and OrientDB Studio, see the [OrientDB Studio documentation](#).

13.1.1. Changing the OrientDB Admin Password

When you are logged into the repository as the `admin` user, you have unlimited access to all tables and functions. It is therefore recommended that you change the `admin` user password in a production system.

You can change the `admin` user password in the OrientDB console, or in OrientDB Studio, as shown in the following examples.

1. To change the admin password in the OrientDB console, make sure that Identity Bridge is running, then follow these steps.
 - a. Launch OrientDB console and connect to the database, as described in the previous section.
 - b. Run the following query:

```
orientdb> update ouser set password='password123' where name='admin'  
Updated 1 record(s) in 0.002000 sec(s).
```

This query changes the admin password to password123.

To change the admin password in OrientDB Studio, make sure that Identity Bridge is running, then follow these steps.

- a. Open OrientDB Studio, as described in the previous section.
- b. Select Query and enter the following query:

```
update ouser set password='password123' where name='admin'
```

This query changes the admin password to password123.

- c. Click Execute.
2. After you have changed the admin password, shut down Identity Bridge , as described in [Stopping and Restarting Identity Bridge](#).
 3. In a text editor, edit the repo.orientdb.json file to add a password property, with the new value of the password.

```
$ more repo.orientdb.json  
{  
  "dbUrl" : "&{openidm.repo.orientdb.dburl}",  
  "user" : "admin",  
  "password" : "password123",  
  ...}
```

4. Restart Identity Bridge, as described in [Stopping and Restarting Identity Bridge](#).
5. Check that you can access the OrientDB database with the new password, or monitor the log files to ensure that Identity Bridge is able to access the database.

13.2. Identity Bridge Log Files

When you set up Identity Bridge by using the **setup.sh** or **setup.bat** scripts, any startup messages that would be output to the OSGi console are output to the file `/path/to/forgerockBridge/logs/console.out`. If you encounter problems while you are configuring Identity Bridge, check this file for an indication of what might have gone wrong in the setup process.

During configuration and authentication, Identity Bridge log messages are output to files named `/path/to/forgerockBridge/logs/openidm0.log.0`, with the integers being incremented with each successive Identity Bridge startup, and after log rotation, when the file size exceeds the configured limit. Check these log files for additional information if you are experiencing problems with Identity Bridge.

Log levels and maximum log file sizes are defined in the file `/path/to/forgerockBridge/conf/logging.properties`. You can adjust the log level in order to provide more or less information. The default configuration rotates log files when the size reaches 5 MB, and retains up to 5 files.

You can adjust the general log level by changing `.level=INFO` to one of the following, in the `logging.properties` file.

```
SEVERE (highest value)
WARNING
INFO
CONFIG
FINE
FINER
```

You can also set specific log levels for individual components. For example, the following setting will provide the maximum output for log messages from the reconciliation process:

```
org.forgerock.openidm.recon.level = FINEST
```

13.3. Tuning the Performance of the OrientDB Repository

By default, the embedded OrientDB repository assumes an environment with unreliable (non-RAID) hardware. These settings might not be appropriate in other environments.

To improve performance in a deployment that runs on reliable (RAID) hardware, change the following settings in the OrientDB configuration file (`/path/to/forgerockBridge/conf/repo.orientdb.json`):

```
"transactionCommitSynch" : false,  
"transactionLogSynch" : false,  
"nonTransactionRecordUpdateSynch" : false
```

By default, these parameters are all set to true, which implies the following:

transactionCommitSynch - The storage is synchronized after each transaction commit.

transactionLogSynch - A disk synch is executed for each log entry, which slows down transactions but guarantees transaction reliability on non-reliable drives.

nonTransactionRecordUpdateSynch - A disk synch is executed at every record operation. This slows down record updates but guarantee reliability on unreliable drives.

DRAFT

Chapter 14. Troubleshooting an Identity Bridge Installation

This chapter describes common problems that might occur during the installation and configuration of Identity Bridge, and how these problems can be resolved.

14.1. Troubleshooting the Integrated Windows Authentication Configuration

This section describes problems that might occur during the configuration and use of Integrated Windows Authentication (IWA) with Identity Bridge. The IWA configuration process is described in [Configuring Identity Bridge for Integrated Windows Authentication](#).

IWA Configuration Issues

The following section highlights common errors that occur during the IWA configuration. These errors might result in some fairly cryptic messages being output. The errors can usually be resolved by updating the IWA configuration in the Identity Bridge Administration interface, or by updating the configuration files in the `path/to/forgerockBridge/security` directory.

Missing or incorrectly named keytab file

If the keytab file is absent or does not match the default keytab file name (`identityConnect.HTTP.keytab`), an error similar to the following is output to the `openidm0.log.*` files:

```
====  
Sep 26, 2013 3:35:02 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO verifyAttributes  
SEVERE: IWA WDSSO: Key Tab File does not exist  
=====
```

This error should be resolved when you copy the keytab file to the `path/to/forgerockBridge/security` directory (or when you rename the keytab file with the correct name). The new keytab file will be picked up automatically - there is no need to restart Identity Bridge.

Incorrect KDC server name

If the name of the Key Distribution Center (KDC) server is incorrect, an error similar to the following is output to the `openidm` log files:

```
====  
Sep 26, 2013 3:41:32 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin  
SEVERE: IWA WDSSO: Service Login Error: server-name: Name or service not known  
Sep 26, 2013 3:41:32 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin  
SEVERE: IWA WDSSO: Stack trace:  
javax.security.auth.login.LoginException: server-name: Name or service not known  
at com.sun.security.auth.module.Krb5LoginModule.attemptAuthentication(Krb5LoginModule.java:763)
```

```

.....
Caused by: java.net.UnknownHostException: server-name: Name or service not known
  at java.net.Inet6AddressImpl.lookupAllHostAddr(Native Method)
  at java.net.InetAddress$1.lookupAllHostAddr(InetAddress.java:894)
  ===

```

This error should be resolved when you specify the correct name for the KDC server.

In the Identity Bridge administration interface, click Settings and select the Authentication and Session tab. Enter the correct KDC server name in the Windows Domain Controller field.

Incorrect SPN (Service Principal Name)

If the SPN that is specified in the Identity Bridge configuration does not match the SPN that is provided in the keytab, Identity Bridge is unable to acquire its login information. The following error is output to the openidm log files:

```

===
Sep 26, 2013 3:51:37 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin
SEVERE: IWA WDSSO: Service Login Error: Unable to obtain password from user

Sep 26, 2013 3:51:37 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin
SEVERE: IWA WDSSO: Stack trace:
javax.security.auth.login.LoginException: Unable to obtain password from user

  at com.sun.security.auth.module.Krb5LoginModule.promptForPass(Krb5LoginModule.java:852)
  at com.sun.security.auth.module.Krb5LoginModule.attemptAuthentication(Krb5LoginModule.java:715)
  at com.sun.security.auth.module.Krb5LoginModule.login(Krb5LoginModule.java:580)
  at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
  at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
  at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
  at java.lang.reflect.Method.invoke(Method.java:606)
  at javax.security.auth.login.LoginContext.invoke(LoginContext.java:784)
  at javax.security.auth.login.LoginContext.access$000(LoginContext.java:203)
  at javax.security.auth.login.LoginContext$4.run(LoginContext.java:698)
  at javax.security.auth.login.LoginContext$4.run(LoginContext.java:696)
  at java.security.AccessController.doPrivileged(Native Method)
  at javax.security.auth.login.LoginContext.invokePriv(LoginContext.java:695)
  at javax.security.auth.login.LoginContext.login(LoginContext.java:594)
  at org.forgerock.jaspi.modules.iwa.wdsso.WDSSO.serviceLogin(WDSSO.java:601)
  at org.forgerock.jaspi.modules.iwa.wdsso.WDSSO.initWindowsDesktopSSOAuth(WDSSO.java:560)
  at org.forgerock.jaspi.modules.iwa.wdsso.WDSSO.process(WDSSO.java:139)
  at org.forgerock.jaspi.modules.iwa.IWAModule.validateRequest(IWAModule.java:107)
  at org.forgerock.openidm.jaspi.modules.IWAModule.validateRequest(IWAModule.java:105)
  at org.forgerock.openidm.jaspi.modules.IWAPassthroughModule.validateRequest(IWAPassthroughModule.java:13)
  at org.forgerock.openidm.jaspi.modules.IDMServerAuthModule.validateRequest(IDMServerAuthModule.java:13)
  at org.forgerock.jaspi.container.ServerAuthContextImpl.validateRequest(ServerAuthContextImpl.java:168)
  at org.forgerock.jaspi.filter.AuthNFilter.doFilter(AuthNFilter.java:161)
  ===

```

The message in the stack trace can be confusing. It indicates, however, that during the module initialization, the `promptForPass()` method of the `Krb5LoginModule.java` module fails while attempting to validate the principal (SPN), by using the keytab.

This error should be resolved when you provide an SPN in the Identity Bridge configuration that matches the keytab.

In the Identity Bridge administration interface, click Settings and select the Authentication and Session tab. Enter the correct SPN name in the Service Principal Name (SPN) field.

Missing login configuration errors on older Java versions

On systems using JDK versions prior to 1.7.0, the following error might appear in the openidm logs:

```

===
SEVERE: IWA WDSSO: Service Login Error: Unable to locate a login configuration
Sep 24, 2013 1:11:38 PM org.forgerock.jaspi.modules.iwa.wdss0.WDSSO serviceLogin
SEVERE: IWA WDSSO: Stack trace:
java.lang.SecurityException: Unable to locate a login configuration
    at com.sun.security.auth.login.ConfigFile.<init>(ConfigFile.java:93)
    at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
    at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:39)
    at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:27)
    at java.lang.reflect.Constructor.newInstance(Constructor.java:513)
    at java.lang.Class.newInstance0(Class.java:355)
    at java.lang.Class.newInstance(Class.java:308)
    at javax.security.auth.login.Configuration$3.run(Configuration.java:247)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.login.Configuration.getConfigurations(Configuration.java:242)
    at org.forgerock.jaspi.modules.iwa.wdss0.WDSSO.serviceLogin(WDSSO.java:586)
    at org.forgerock.jaspi.modules.iwa.wdss0.WDSSO.initWindowsDesktopSSOAuth(WDSSO.java:560)
    at org.forgerock.jaspi.modules.iwa.wdss0.WDSSO.process(WDSSO.java:139)
    at org.forgerock.jaspi.modules.iwa.IWAModule.validateRequest(IWAModule.java:107)
    at org.forgerock.openidm.jaspi.modules.IWAModule.validateRequest(IWAModule.java:105)
    at org.forgerock.openidm.jaspi.modules.IWAPassthroughModule.validateRequest(IWAPassthroughModule.java:114)
    at org.forgerock.openidm.jaspi.modules.IDMServerAuthModule.validateRequest(IDMServerAuthModule.java:137)
===

```

This error typically occurs when the WDSSO module cannot find the `jaas-repo.json` file. Although this file is not used, issues occur if the JDK fails to find it. To resolve this issue, the following property has been included in the startup scripts:

```
-Djava.security.auth.login.config="$OPENIDM_HOME/security/jaas-repo.conf"
```

Keytab and SPN Issues

The keytab and the configuration of a dedicated Active Directory user for the service are critical elements of a Kerberos configuration. Before describing potential problems with these elements, it is helpful to have an understanding of the main Kerberos components involved in the authentication process.

- Kerberos distinguishes between two types of principals (accounts) - User Principal Name (UPN), and Service Principal Name (SPN). Both of these are essentially unique identifiers for the security identity of a user or of a computer. UPNs are of the format *userID@DNS domain name* while SPNs are of the format *serviceClass/host:port/serviceName*.

Both UPNs and SPNs are registered in the Active Directory Domain Controller (DC) for the user account that the Identity Bridge instance will use.

- The Key Distribution Center (KDC) comprises two elements - the Authentication Service and the Ticket Granting Service. Identity Bridge uses its keytab to authenticate against the Authentication Service (AS) and obtains a ticket from the Ticket Granting Service (TGS) for the specified Service Principal Name (SPN).
- The AS uses the SPN to locate the service user entry in Active Directory and to retrieve the account password to establish a session key.

The following misconfigurations can cause errors at this point.

Incorrect UPN

If the user account uses a UPN that does not match the SPN that Identity Bridge uses (and the SPN that is defined in the keytab), an error similar to the following output:

```

===
Sep 26, 2013 4:33:52 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin
SEVERE: IWA WDSSO: Service Login Error: Client not found in Kerberos database (6)
Sep 26, 2013 4:33:52 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO serviceLogin
SEVERE: IWA WDSSO: Stack trace:
javax.security.auth.login.LoginException: Client not found in Kerberos database (6)
    at com.sun.security.auth.module.Krb5LoginModule.attemptAuthentication(Krb5LoginModule.java:759)
    at com.sun.security.auth.module.Krb5LoginModule.login(Krb5LoginModule.java:580)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at javax.security.auth.login.LoginContext.invoke(LoginContext.java:784)
    at javax.security.auth.login.LoginContext.access$000(LoginContext.java:203)
    at javax.security.auth.login.LoginContext$4.run(LoginContext.java:698)
    at javax.security.auth.login.LoginContext$4.run(LoginContext.java:696)
    at java.security.AccessController.doPrivileged(Native Method)
    at javax.security.auth.login.LoginContext.invokePriv(LoginContext.java:695)
    at javax.security.auth.login.LoginContext.login(LoginContext.java:594)
    at org.forgerock.jaspi.modules.iwa.wdsso.WDSSO.serviceLogin(WDSSO.java:601)
...
Caused by: KrbException: Client not found in Kerberos database (6)
    at sun.security.krb5.KrbAsRep.<init>(KrbAsRep.java:76)
    at sun.security.krb5.KrbAsReqBuilder.send(KrbAsReqBuilder.java:319)
    at sun.security.krb5.KrbAsReqBuilder.action(KrbAsReqBuilder.java:364)
    at com.sun.security.auth.module.Krb5LoginModule.attemptAuthentication(Krb5LoginModule.java:731)
    ... 61 more
Caused by: KrbException: Identifier doesn't match expected value (906)
    at sun.security.krb5.internal.KDCRep.init(KDCRep.java:143)
    at sun.security.krb5.internal.ASRep.init(ASRep.java:65)
    at sun.security.krb5.internal.ASRep.<init>(ASRep.java:60)
    at sun.security.krb5.KrbAsRep.<init>(KrbAsRep.java:60)
    ... 64 more
===

```

The message does, in fact, identify the issue, which occurs during the authentication attempt. In this case, the SPN name that is used by Identity Bridge was not found in the Kerberos database (Active Directory).

The UPN, or user logon name, is the crucial attribute in this error. The UPN *must* match the SPN that Identity Bridge uses.

Inconsistent Key Version Number (kvno)

When you create a keytab without specifying a key version number (using the **ktpass** command without the **kvno** option), the **msDS-KeyVersionNumber** is automatically incremented in Active Directory. You can obtain the current key version number by using the **klist** command, for example:

```
$ klist -ke -t security/identityConnect.HTTP.keytab
Keytab name: FILE:security/identityConnect.HTTP.keytab
KVNO Timestamp Principal
-----
5 01/01/70 00:00:00 HTTP/connect.forgerock.com@ad.example.com
```

If the key version number is incorrect, an error similar to the following is observed:

```
===
Sep 26, 2013 6:03:49 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO process
SEVERE: IWA WDSSO: Authentication failed with GSSException. Failure unspecified at GSS-API level
(Mechanism level: Specified version of key is not available (44))
=====
```

Single SPN associated with multiple accounts

If the same SPN is associated with multiple Active Directory accounts, the Kerberos exchange will fail, because the Key Distribution Center is unable to determine which entry to use.

The error message might be confusing but is generally an indication that multiple accounts have been associated with the SPN:

```
===
Sep 26, 2013 6:21:36 AM org.forgerock.jaspi.modules.iwa.wdsso.WDSSO process
SEVERE: IWA WDSSO: Authentication failed with GSSException. Defective token detected
(Mechanism level: GSSHeader did not find the right tag)
=====
```

DRAFT

Identity Bridge Glossary

reconciliation

The process of analyzing data on a target system to determine its consistency with the data on a source system.

synchronization

The process of modifying data on a target system to maintain consistency with the data on a source system.

DRAFT

Index

C

connections, 17

G

Getting started, 7

M

mapping data, 25

P

Prerequisites, 5

R

Repository database

SQL database, 67

Table names, 67

DRAFT