Getting Started With OpenIDM

Version 4

Mike Jang

ForgeRock AS
201 Mission St., Suite 2900
San Francisco, CA 94105, USA
+1 415-599-1100 (US)
www.forgerock.com

Abstract

Guide to installing and evaluating OpenIDM. The OpenIDM project offers flexible, open source services for automating management of the identity life cycle.

Table of Contents

Preface

This guide shows you how to install and get started with OpenIDM.

1 Who Should Use This Guide

This guide is written for identity management developers and administrators who build, deploy, and maintain OpenIDM services for their organizations. This guide covers the tasks you need to quickly get OpenIDM running on your system.

As you read this guide, you will see how OpenIDM reconciles customer identity data to ensure accurate information across disparate resources within an organization.

You will also read about what else OpenIDM can do, in the areas of provisioning, self-service workflows, and password management. You will also read about how OpenIDM connects to a variety of remote data stores, with links to detailed documentation.

For example, engineers might access their systems through Active Directory accounts. Those same engineers might need to update their information in a Human Resources database, stored in a separate LDAP directory. With OpenIDM, you can keep those user identities synchronized, so each engineer only has to update their data once.

2 Accessing Documentation Online

ForgeRock core documentation, such as this document, aims to be technically accurate and complete with respect to the software documented.

Core documentation therefore follows a three-phase review process designed to eliminate errors:

- Product managers and software architects review project documentation design with respect to the readers' software lifecycle needs.
- Subject matter experts review proposed documentation changes for technical accuracy and completeness with respect to the corresponding software.
- Quality experts validate implemented documentation changes for technical accuracy, completeness in scope, and usability for the readership.

The review process helps to ensure that documentation published for a ForgeRock release is technically accurate and complete.

Fully reviewed, published core documentation is available at http://backstage.forgerock.com/. Use this documentation when working with a ForgeRock Enterprise release.

You can find pre-release draft documentation at the online community resource center. Use this documentation when trying a nightly build.

3 Joining the ForgeRock Community

Visit the Community resource center where you can find information about each project, download nightly builds, browse the resource catalog, ask and answer questions on the forums, find community events near you, and of course get the source code as well.

Chapter 1

Getting Started With OpenIDM

Whenever you need access to important information, administrators need to know who you are. They need to know your identity, which may be distributed in multiple accounts.

As a user, you might have several accounts even within your own company, for functions such as:

- Email
- Human Resources
- Payroll
- Engineering, Support, Accounting, and other functions

Each of these accounts may be stored in different resources, such as Active Directory, OpenDJ, OpenLDAP, and more. Keeping track of user identities in each of these resources (also known as data stores) can get complex. OpenIDM simplifies the process, as it reconciles differences between resources.

With situational policies, OpenIDM can handle discrepancies such as a missing or updated address for a specific user. OpenIDM includes default but configurable policies to handle such conditions. In this way, OpenIDM ensures consistency and predictability in an otherwise chaotic resource environment.

OpenIDM can make it easier to track user identities across these resources. OpenIDM has a highly scalable, modular, readily deployable architecture that can help you manage workflows and user information.

1.1 What Can You Do With OpenIDM?

With OpenIDM, you can simplify the management of identity, as it can help you synchronize data across multiple resources. Each organization can maintain control of accounts within their respective domains.

OpenIDM works equally well with user, group, and device identities.

You can also configure workflows to help users manage how they sign up for accounts, as part of how OpenIDM manages the life cycle of users and their accounts.

You can manage employee identities as they move from job to job. You will make their lives easier as OpenIDM can automatically register user accounts on different systems. Later, OpenIDM will increase productivity when it reconciles information from different accounts, saving users the hassle of entering the same information on different systems.

1.2 What You Will See In This Document

In this guide, you will see how OpenIDM reconciles user data between two data stores. We will look at a department that is adding a third engineer, Jane Sanchez.

Your Human Resources department has updated their data store with Jane Sanchez's information. You want to use OpenIDM to update the internal Engineering data store. But first, you have to start OpenIDM.

1.3 What You Need Before Starting OpenIDM

This section covers what you need to have on your system before running OpenIDM:

- Operating System: Windows or UNIX/Linux.
- Java: Java Runtime Environment (JRE) Standard Edition (Java SE) 7, update 6 or later, or Java 8. Alternatively, you can use the same version of the Java Development Kit (JDK). On Linux, you may also install the OpenJDK package native to your updated Linux distribution.
- At least 250 MB of free disk space.
- At least 1 GB of free RAM.
- If your operating system includes a firewall, make sure that it allows traffic through (default) ports 8080 and 8443.

We provide this document, *Getting Started with OpenIDM*, for demonstration purposes only.

With this document, we want to make it as easy as possible to set up a demonstration of OpenIDM. To that end, we have written this document for installations on a desktop operating system, Microsoft Windows 7.

For a list of software that we support in production, see Chapter 2, "Before You Install OpenIDM Software" in the *OpenIDM Release Notes*.

1.3.1 Java Environment

On Windows systems, after installing Java, set the `JAVA_HOME` environment variable. To do so on Windows 7, take the following steps:

1. Locate your JRE or JDK installation directory. For a default installation of Java 8 on Windows 7, you should find the directory here: `C:\Program Files\Java\jre-`*`version`*.
2. Select Start > Control Panel > System and Security > Advanced System Settings to open a System Properties window.
3. Select Advanced > Environment Variables.
4. Set the value of `JAVA_HOME` to match the JRE or JDK installation directory.

1.4 Downloading and Starting OpenIDM

This procedure assumes that you are downloading and starting OpenIDM as a regular (not administrative) user named `user`.

1. Download enterprise software releases through the ForgeRock BackStage site. ForgeRock enterprise releases are thoroughly validated builds for ForgeRock customers who run OpenIDM in production deployments, and for those who want to try or test with release builds.

   For more information on the contents of the OpenIDM binary package, see Appendix A, "File Layout" in the *OpenIDM Integrator's Guide*.
2. Extract the contents of the OpenIDM binary file to your user's `Downloads` directory. The process should unpack the contents of OpenIDM to the `Downloads/openidm` subdirectory.
3. Navigate to the `Downloads/openidm` subdirectory:
   - In Microsoft Windows, use Windows Explorer to navigate to the `C:\Users\`*`user`*`\Downloads\openidm` directory.

Double-click the `getting-started(.bat)` file. Do not select the `getting-started.sh` file, as that is intended for use on UNIX/Linux systems.

- In Linux/UNIX, open a command-line interface and run the following commands:

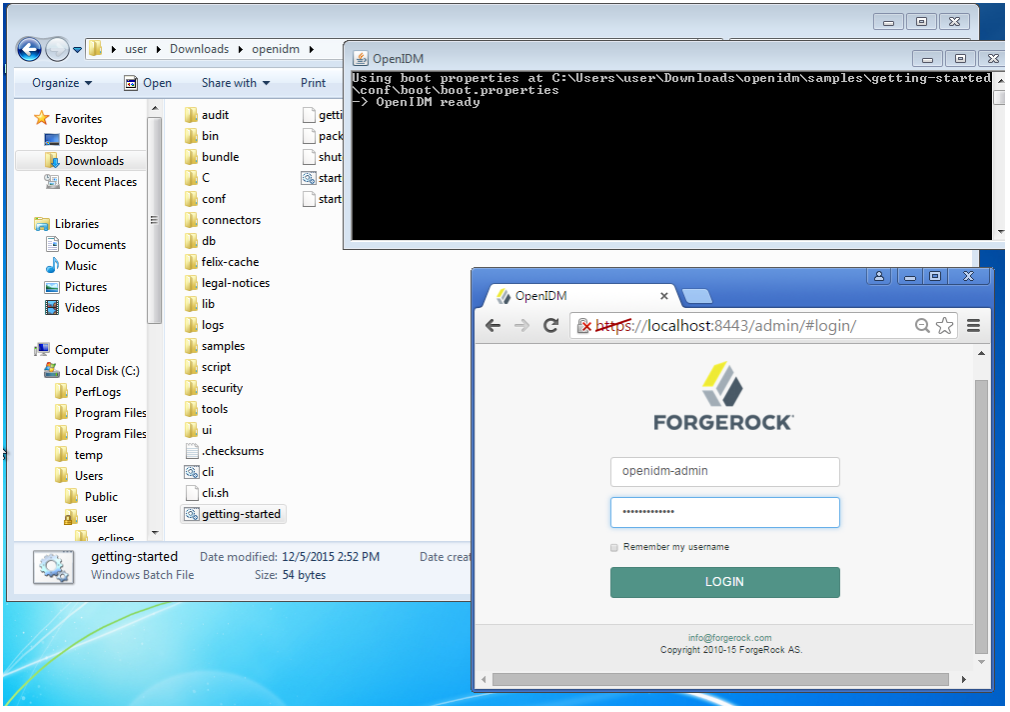
```
$ cd /home/user/Downloads/openidm
$ ./getting-started.sh
```

4. When OpenIDM is ready, you will see the following message:

```
-> OpenIDM ready
```

Once OpenIDM is ready, you can administer it from a web browser. To do so, navigate to `http://localhost:8080/admin` or `https://localhost:8443/admin`. If you have installed OpenIDM on a remote system, substitute that hostname or IP address for `localhost`.

Figure 1.1. Starting OpenIDM in Microsoft Windows



Note

We recommend that you connect to OpenIDM via the default secure port, 8443, and import a signed certificate into the OpenIDM truststore, as discussed in Section 16.1, "Accessing the Security Management Service" in the *OpenIDM Integrator's Guide*.

Until you install that certificate, you will see a warning in your browser at least the first time you access OpenIDM over a secure port.

The default username and password for the OpenIDM Administrator is `openidm-admin` and `openidm-admin`.

When you log into OpenIDM at a URL with the `/admin` endpoint, you are logging into the OpenIDM Administrative User Interface, also known as the Admin UI.

Warning

The default password for the OpenIDM administrative user, `openidm-admin`, is `openidm-admin`. To protect your deployment in production, change this password.

All users, including `openidm-admin`, can change their password through the Self-Service UI, at `http://localhost:8080/` or `https://localhost:8443/`. Once logged in, click Profile > Password.

1.5 The Getting Started Data Files

In a production deployment, you are likely to see resources like Active Directory and OpenDJ. But the setup requirements for each are extensive, and beyond the scope of this document.

For simplicity, this guide uses two static files as data stores:

- `hr.csv` represents the Human Resources data store. It is in CSV format, commonly used to share data between spreadsheet applications.

- `engineering.xml` represents the Engineering data store. It is in XML format, a generic means for storing complex data that is commonly used over the Internet.

You can find these files in the OpenIDM binary package that you downloaded earlier, in the following subdirectory: `openidm/samples/getting-started/data`.

Chapter 2

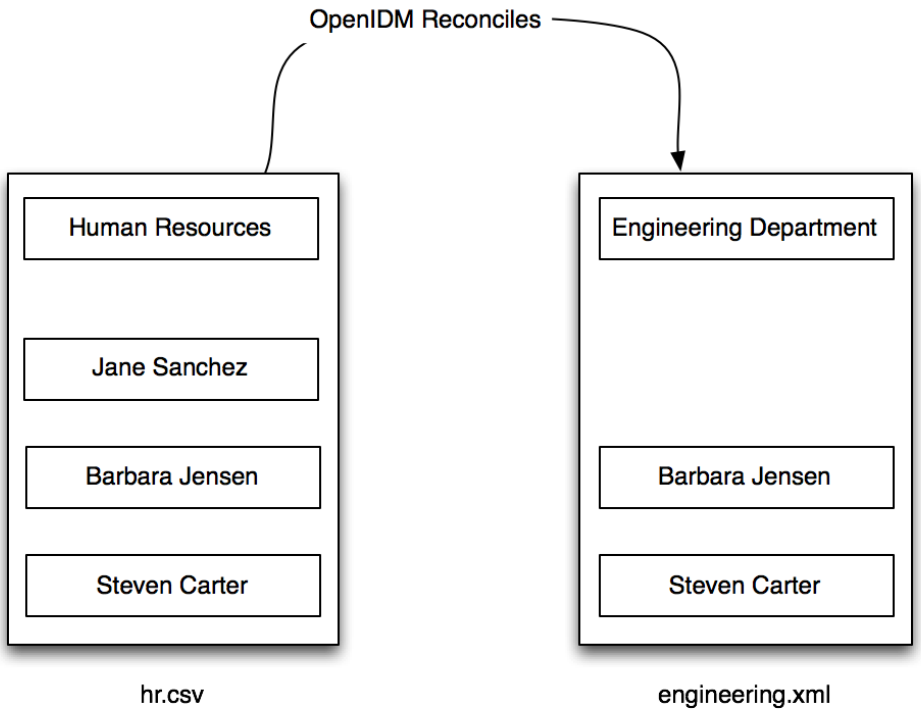
Reconciling Identity Data

Now that you have installed OpenIDM with a "Getting Started" configuration, you will learn how OpenIDM reconciles information between two data stores.

While the reconciliation demonstrated in this guide uses two simplified data files, you can set up the same operations at an enterprise level on a variety of resources.

Return to the situation described earlier, where you have Jane Sanchez joining the engineering department. The following illustration depicts what OpenIDM has to do to reconcile the differences.

Figure 2.1. OpenIDM can reconcile differences between data stores



2.1 Using OpenIDM to Reconcile Differences

A central feature of OpenIDM is reconciliation. In other words, OpenIDM can compare the contents of two data stores, and make decisions on what to do, depending on the differences.

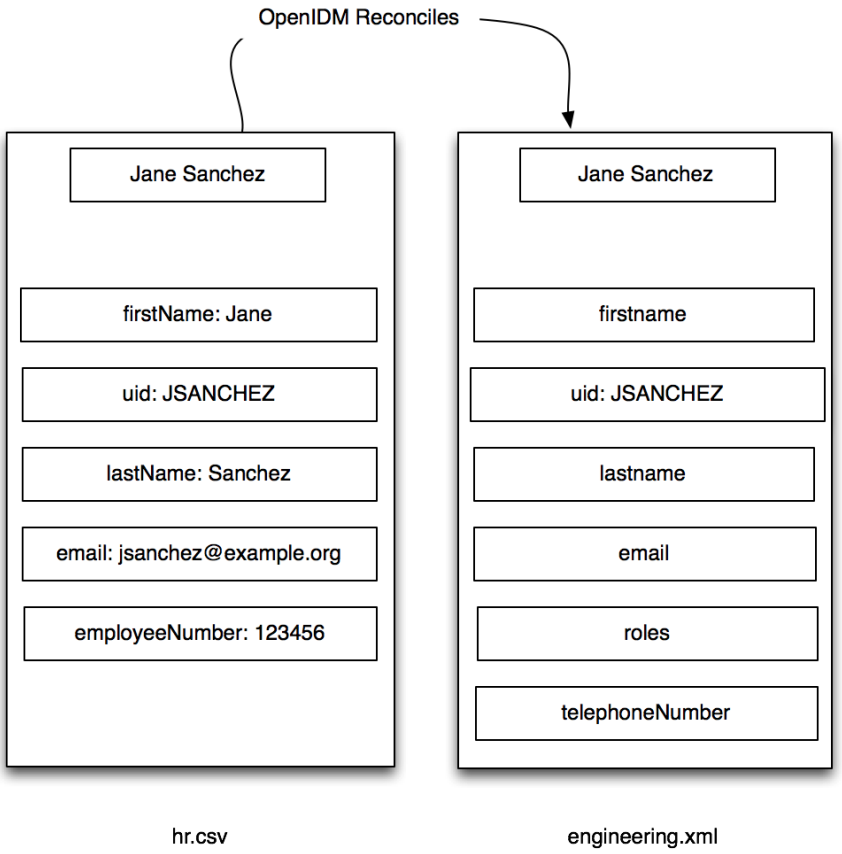
This scenario is based on two data files:

- `hr.csv`, which represents the Human Resources data store
- `engineering.xml`, which represents the Engineering data store

OpenIDM will modify the Engineering data store by adding the newly hired Jane Sanchez. As suggested by the following illustration, it will also address detailed

differences between Jane's Human Resources account and the Engineering data store.

Figure 2.2. Data Stores Can Have Different Categories of Data



OpenIDM includes configuration files that map detailed information from the Human Resources data store to the Engineering data store. For example, the OpenIDM configuration maps the `firstName` entry in Human Resources to the `firstname` entry in Engineering.

Note

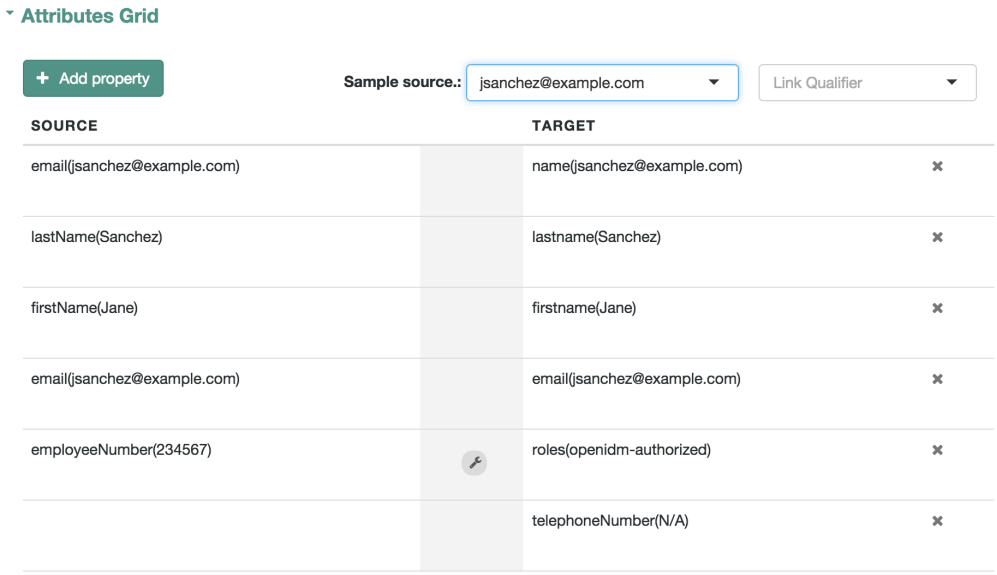
Mapping between data stores may require additional configuration. You should find two `provisioner.openicf-*.json` files in the `/path/to/openidm/samples/getting-started/conf` subdirectory. The provisioner files configure connections to external resources, such as Active Directory, OpenDJ or even the `engineering.xml` and `hr.csv` files used in this guide. For more information, see Chapter 11, "Connecting to External Resources" in the *OpenIDM Integrator's Guide*.

In the Admin UI, you can see how OpenIDM reconciles the different categories for user Jane Sanchez. Log in to the Admin UI at `https://localhost:8443/admin`. The default username is `openidm-admin` and default password is `openidm-admin`.

Select Configure > Mappings > `HumanResources_Engineering` > Properties.

In the Sample Source text box, enter `Sanchez`. You should see a drop-down entry for Jane Sanchez that you can select. You should now see how OpenIDM would reconcile Jane Sanchez's entry in the Human Resources data store into the Engineering data store.

Figure 2.3. Reconciling Differences for an Account



Scroll back up the same page. Select Reconcile Now.

When you reconcile the two data stores, OpenIDM will make the change to the Engineering data store.

For those of you who prefer the command-line interface, you can see how the mapping works in the `sync.json` file, in the `/path/to/openidm/samples/getting-started/conf` directory.

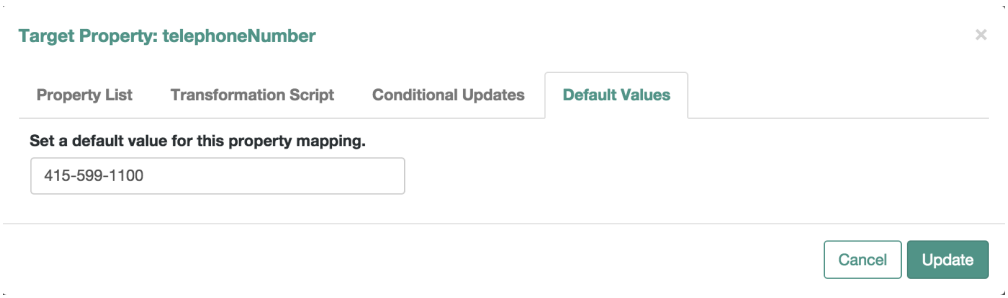
2.2 Reconciling Identity Data After One Update

Now that you have used OpenIDM to reconcile two data stores, try something else. Assume the Engineering organization wants to overwrite all user telephone numbers in its employee data store with one central telephone number.

For this purpose, you can set up a default telephone number for the next reconciliation.

In the HumanResources_Engineering page, scroll down and select `telephoneNumber` > Default Values.

Figure 2.4. Set A New Default Telephone Number



When you select Update, and Save Properties, OpenIDM changes the `sync.json` configuration file. The next time OpenIDM reconciles from Human Resources to Engineering, it will include that default telephone number for all employees in the Engineering group.

Chapter 3

Where To Go From Here

OpenIDM can do much more than reconcile data between two different sources. In this chapter, you will read about the key features of OpenIDM, with links to additional information about each feature.

3.1 Integrating Business Processes and Workflows

A business process begins with an objective and includes a well-defined sequence of tasks to meet that objective. In OpenIDM, you can configure many of these tasks as self-service workflows, such as self-registration, new user onboarding, and account certification.

With OpenIDM, you can automate many of these tasks as a workflow.

Once you configure the right workflows, a newly hired engineer can log into OpenIDM and request access to manufacturing information.

That request is sent to the appropriate manager for approval. Once approved, the OpenIDM provisions the new engineer with access to manufacturing.

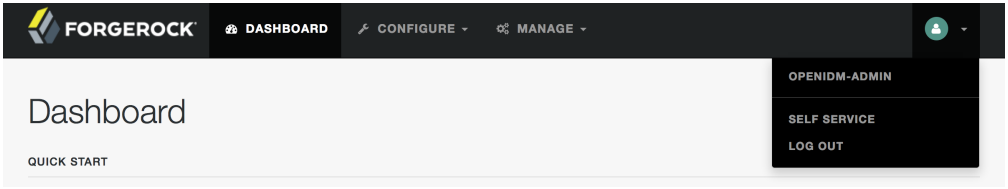
OpenIDM supports workflow-driven provisioning activities, based on the embedded *Activiti* Process Engine, which complies with the *Business Process Model and Notation 2.0* (BPMN 2.0) standard.

OpenIDM integrates additional workflows such as new user onboarding, orphan account detection, and password change reminders. For more information, see Chapter 11, "Workflow Samples" in the *OpenIDM Samples Guide*.

3.2 Managing Passwords

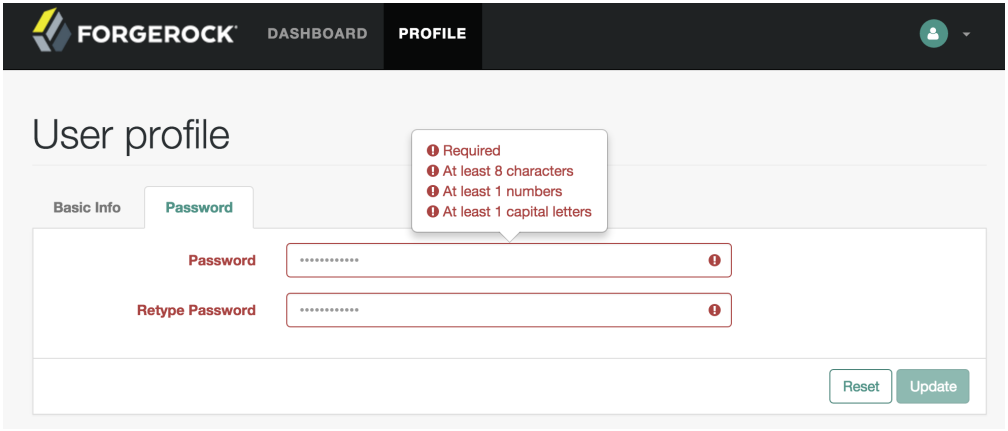
You can manage passwords from the Self-Service User Interface, also known as the Self-Service UI. From the Admin UI, click on the icon in the upper-right corner. In the menu that appears, click Self-Service:

Figure 3.1. Access the Self-Service User Interface



You should now be in the Self-Service UI. Click Profile > Password. You can now change your password, subject to the policy limits shown.

Figure 3.2. Changing Your Password



As you can see, OpenIDM supports a robust password policy. You can modify the rules shown, or add more rules such as the following:

- Elements that should not be a part of a password, such as a family name
- Password expiration dates
- Password histories, to prevent password reuse

For more information, see Chapter 14, "Managing Passwords" in the *OpenIDM Integrator's Guide*.

3.3 Managing User Roles

Some users need accounts on multiple systems. For example, insurance agents may also have insurance policies with the company that they work for. In that situation, the insurance agent is also a customer of the company.

Alternatively, a salesperson may also test customer engineering scenarios. That salesperson may also need access to engineering systems.

In OpenIDM, each of these user scenarios is known as a *role*. OpenIDM allows you to set up a consolidated set of attributes associated with each role. To do so, you would configure custom roles to assign to selected users. For example, you may assign both *insured* and *agent* roles to an agent, while assigning the *insured* role to all customers.

In a similar fashion, OpenIDM allows you to assign both *sales* and *engineering* roles to the sales engineer.

You can then synchronize users with those roles into appropriate data stores.

For more information, see Section 8.4, "Working With Managed Roles" in the *OpenIDM Integrator's Guide*. For a sample of how you can configure external roles within OpenIDM, see Chapter 7, "Roles Samples - Demonstrating the OpenIDM Roles Implementation" in the *OpenIDM Samples Guide*.

3.4 Connecting to Remote Data Stores

You can use OpenIDM to connect to a substantial variety of user and device data stores, on premise and in the cloud. While OpenIDM can connect to some connectors dedicated to a few data stores, OpenIDM can also connect to many more data stores using a scripted connector framework.

OpenIDM includes support for connectors to the following external resources:

- Google Web Applications (see Section 11.5.9, "Google Apps Connector" in the *OpenIDM Integrator's Guide*).
- Salesforce (see Section 11.5.8, "Salesforce Connector" in the *OpenIDM Integrator's Guide*).
- Any LDAPv3-compliant directory, including OpenDJ and Active Directory (see Section 11.5.1, "Generic LDAP Connector" in the *OpenIDM Integrator's Guide*).
- CSV Files (see Section 11.5.3, "CSV File Connector" in the *OpenIDM Integrator's Guide*).

- Database Tables (see Section 11.5.5, "Database Table Connector" in the *OpenIDM Integrator's Guide*).

If the resource that you need is not on the list, you should be able to use one of the OpenIDM scripted connector frameworks to connect to that resource:

- For connectors associated with Microsoft Windows, OpenIDM includes a PowerShell Connector Toolkit that you can use to provision a variety of Microsoft services, including but not limited to Active Directory, SQL Server, Microsoft Exchange, SharePoint, Azure Active Directory, and Office 365. For more information, see Section 11.5.7, "PowerShell Connector Toolkit" in the *OpenIDM Integrator's Guide*. OpenIDM includes a sample PowerShell Connector Toolkit configuration, described in Chapter 5, "Samples That Use the PowerShell Connector Toolkit to Create Scripted Connectors" in the *OpenIDM Samples Guide*.

- For other external resources, OpenIDM includes a Groovy Connector Toolkit that allows you to run Groovy scripts to interact with any external resource. For more information, see Section 11.5.6, "Groovy Connector Toolkit" in the *OpenIDM Integrator's Guide*. Chapter 4, "Samples That Use the Groovy Connector Toolkit to Create Scripted Connectors" in the *OpenIDM Samples Guide* includes samples of how you might implement the scripted Groovy connector.

3.5 Reconciliation

OpenIDM supports reconciliation between two data stores, as a source and a target.

In identity management, reconciliation compares the contents of objects in different data stores, and makes decisions based on configurable policies.

For example, if you have an application that maintains its own user store, OpenIDM can ensure your canonical directory attributes are kept up to date by reconciling their values as they are changed.

For more information, see Chapter 12, "Synchronizing Data Between Resources" in the *OpenIDM Integrator's Guide*.

3.6 Authentication Modules Available for OpenIDM

OpenIDM has access to several different authentication modules that can help you protect your systems. For more information, see Section 15.1.2, "Supported Authentication and Session Modules" in the *OpenIDM Integrator's Guide*.

3.7 Finding Additional Use Cases

OpenIDM is a lightweight and highly customizable identity management product.

The OpenIDM documentation includes additional use cases. Most of them are known as *Samples*, and are described in Chapter 1, "Overview of the OpenIDM Samples" in the *OpenIDM Samples Guide*.

These samples include step-by-step instructions on how you can connect to different data stores, customize product behavior using JavaScript and Groovy, and administer OpenIDM with ForgeRock's commons RESTful API commands.

3.8 How OpenIDM Can Help Your Organization

Now that you have seen how OpenIDM can help you manage users, review the features that OpenIDM can bring to your organization:

- *Web-Based Administrative User Interface*

  Configure OpenIDM with the Web-Based Administrative User Interface. You can configure many major components of OpenIDM without ever touching a text configuration file.

- *Self-Service Functionality*

  User self-service features can streamline onboarding, account certification, new user registration, username recovery, and password reset. OpenIDM self-service features are built upon a *BPMN 2.0*-compliant workflow engine.

- *Role-Based Provisioning*

  Create and manage users based on attributes such as organizational need, job function, and geographic location.

- *Backend Flexibility*

  Choose the desired backend database for your deployment. OpenIDM supports MySQL, Microsoft SQL Server, Oracle Database, IBM DB2, and PostgreSQL. For the supported versions of each database, see Chapter 2, "Before You Install OpenIDM Software" in the *OpenIDM Release Notes*.

- *Password Management*

  Set up fine-grained control of passwords to ensure consistent password policies across all applications and data stores. Supports separate passwords per external resource.

- *Logging, Auditing, and Reporting*

  OpenIDM logs all activity, internally and within connected systems. With such logs, you can track information for access, activity, authentication, configuration, reconciliation, and synchronization.

- *Access to External Resources*

  OpenIDM can access a generic scripted connector that allows you to set up communications with many external data stores.

3.9 Stopping and Removing OpenIDM

Follow these steps to stop and remove OpenIDM.

1. To stop OpenIDM, return to the console window where you saw the following message:

   ```
   -> OpenIDM ready
   ```

   Press Return, and enter the following command:

   ```
   -> shutdown
   ```

2. OpenIDM is self-contained. After you shut down OpenIDM, you can choose to delete the files in the `/path/to/openidm` directory. OpenIDM includes no artifacts in system registries or elsewhere.

We hope that you want to continue exploring OpenIDM.

To do so, review the OpenIDM documentation available on ForgeRock's BackStage site.

Index