

First Steps

This page will guide you through first steps with [midPoint](#). It describes how to set up the first resource and how to do basic provisioning. It also provides pointers to other information in [midPoint wiki](#).

midPoint Installation

For instruction how to install midPoint please see the [Installation Guide](#) page.

midPoint comes with almost empty database after installation. It needs to be set up to get the full capabilities.

Logging in

midPoint administration user interface will be available (most likely) on the following URL:

<http://localhost:8080/midpoint/>

midPoint comes with one pre-configured administrator user after installation. You can use it to log in and setup midPoint:

Username	administrator
Password	5ecr3t

This is an (almost) ordinary user in the midPoint database. Therefore you can change it later, create more administrative users and so on.

Configuration Files Management

midPoint uses XML configuration files that are being parsed by internal validators. When validated and saved, the XML files you send will be updated with namespace notations that might make them harder to read and edit. Therefore, we suggest that you keep a copy of all XML files you import into midPoint and work on those files instead of the embedded XML editor.

It is generally a good practice to keep all your XML configuration files in a versioning system to keep track of the changes you apply to your system.

Basic User Management

You cannot do much in the initial state of midPoint after the installation. You can create *users* by using items in the `Users` menu. The users represent physical users of the system: employees, customers, persons, etc. midPoint provides basic set of attributes for users that are commonly found in identity management deployments. Properties from the default schema are displayed in the dialog. The full set of user properties is quite rich therefore only a small subset is displayed by default. You can see all the properties by clicking on a small "Show empty fields" button in the upper left corner of the dialog. If the default set of properties is not enough than the user schema definition can be extended with custom properties during midPoint customization.

Select Users->New user from the menu. Fill out the details for a user. Especially make sure to fill out "Name", this will be the user's login name. You must fill out all parameters marked by red asterisk. This will create a user in midPoint repository. This is a "master" user record for the provisioning system. No resources will be modified yet.

After you submit the form, you should see the user in a table. Feel free to create more users. You can return to this table anytime by selecting Users->List users from the menu.

Select a specific user by clicking on a user's name in a table row. It will open a "User details" dialog. You should see the user data organized in several panels. You cannot create any accounts because there is no resource definition yet.

Accounts and Resources

Users are stored locally in midPoint repository. But the goal of identity management is to manage *accounts*. Accounts are data structures representing users on *resources*. For example, a particular user may have his LDAP account, Active Directory account, Solaris account, etc. The systems that hold the accounts are known to midPoint as *resources*. midPoint needs to know quite a lot about a resource to be able to manage accounts in it. The information needed by midPoint include the type of the resource, hostnames, ports, administrative user identifiers and credentials, patterns and templates for new accounts, a definition of account attribute synchronization details and so on. Next section of this guide will provide an example how to set such a definition.

Terminology

	Represents	Stored in	Example
User	Physical person	midPoint Repository	Jane Smith John Doe
Account	Data records used to access a system	a Resource	jsmith123 uid=doejr,ou=people,dc=example,dc=com
Resource	Remote system that midPoint manages	N/A	LDAP server ds1.example.com Active Directory server over there in my server room

OpenDJ Resource Setup

This guide uses OpenDJ as a sample resource. It is a modern fully-featured LDAP Directory server, yet it is easy to use and works almost everywhere. We recommend to use OpenDJ directory server as a testing resource for midPoint.

Please follow the steps in chapter.

Installing OpenDJ

The first step to successfully test account interaction between midpoint and LDAP server is the actual installation of LDAP Directory server OpenDJ. Follow the instructions in 'Installation' subsection [here](#).

Importing Resource Definition

The resource definition is a piece of XML that defines resource parameters in midPoint. It contains definition of a *connector* that is used to access the resource and its parameters, such as hostname, port number, administrative user identifiers and credentials, etc. The parameters may be different for each resource type. The resource definition also contains more data and these will be described later. There needs to be a definition for every resource that midPoint connects to.

There are two sample resource definitions that can be used to connect to a OpenDJ instance that you have just installed. They can be found in our `samples/opendj` directory accessible here:

Current Development Snapshot (trunk)	http://git.evolveum.com/view/midpoint/master/samples/resources/opendj/
Latest Release	http://git.evolveum.com/view/midpoint/v2.2/samples/resources/opendj/

The resource definition XML file is full of in-line comments that explain individual configuration items used in the file. The `opendj-localhost-basic.xml` file in the samples directory is a basic, readable and understandable definition of an LDAP resource. However, for the following actions, we will use `opendj-localhost-resource-sync-advanced.xml` file. For description of other examples please see page

To import `opendj-localhost-resource-sync-advanced.xml` resource file, there is one thing you need to do first. Copy & paste file 'extension-whatever.xsd' from `midpoint/trunk/samples/schema` ([here](#)) to `<midpoint-homeFolder>/schema`. To apply these changes to midPoint, you will need to restart your application server (Tomcat).

Log in to the midPoint administration console using the following URL:
<http://localhost:8080/midpoint/>

Navigate to Configuration->Import xml page. Copy&paste the content of `opendj-localhost-resource-sync-advanced.xml` file into the text area. Press the "Import object" button. You should see green message "Operation successful".

If you are trying to reimport resource file after previous failed import, be sure to check 'Overwrite existing object' check button.

Account Management

Once the resource definition is in place, you can create accounts. Accounts cannot be created just by themselves, they need to be assigned to a specific user. That's the purpose of identity management. Therefore edit any existing user, switch to the "Accounts" tab and add a new account on the OpenDJ resource.

Select Users->List users from the menu. Select a specific user by clicking on his login in the table row. It will open a "User details" dialog. Left-hand side shows user's assignments and accounts. It should be empty now.

Click "Add (for Account)" button located in the bottom bar. Resource selection dialog will appear. You should see OpenDJ resource there. Select it and click on "Add resource(s)" button to create a new account.

A form describing details of a new account should be displayed in the accounts panel. Some values may be marked by blue asterisk. These values are generated by an expression defined in the resource and therefore do not usually need to be filled in. Fill in the remaining values into the form fields. Make sure that all the mandatory values (marked by a red asterisk) are filled in.

Click on "Save" button located in the bottom bar to apply changes and create the account. You should get a green message "Changes saved successfully."

Now the account should be listed in the user details page. You may also use the LDAP tools or OpenDJ control panel to make sure that the account was created in the directory server.

The account form is dynamically generated. The fields are based on the XSD resource schema defined in the "Localhost OpenDJ" XML object, i.e. the resource description you have imported. If you want to adapt the fields displayed here, you just need to change the XSD schema in the "Localhost OpenDJ" XML object, e.g. by using debug pages. This form is dynamically generated from the resource schema.

See Also

- [Examples Description](#)
- [Synchronization Examples](#)
- [Architecture and Design](#)

External links

- [What is midPoint Open Source Identity & Access Management](#)
- [Evolveum](#) - Team of IAM professionals who developed midPoint