

Getting started with midPoint using Docker

Purpose: To help someone with no experience with midPoint be able to setup and run midPoint. Provide basic configuration to pull in users from a data source and sync that data to external target system such as LDAP.

Pull new Docker Image from Evolveum:

ON Linux VM:

add user to docker group (dont run as sudo)

```
docker run -d -p 8080:8080 --name midpoint evolveum/midpoint:latest
```

Start and Stop container

Once you have your container created use start and stop commands for starting and stopping it.

- Start not running container:

```
docker start midpoint
```

- Stop running container:

```
docker stop midpoint
```

Stop command will save your configuration until you remove midPoint container.

To Stop/Start only Tomcat: enter the midPoint container bash use:

```
docker exec -it midpoint bash
```

In Docker container fix midpoint.sh - change: `#!/bin/bash` to `#!/bin/sh`

Should be running here: `http://<VMname>:8080/midpoint/`

Login as Administrator with default password and changeit.

Create Incoming sync from Oracle DB

Copy Oracle Driver to VM.

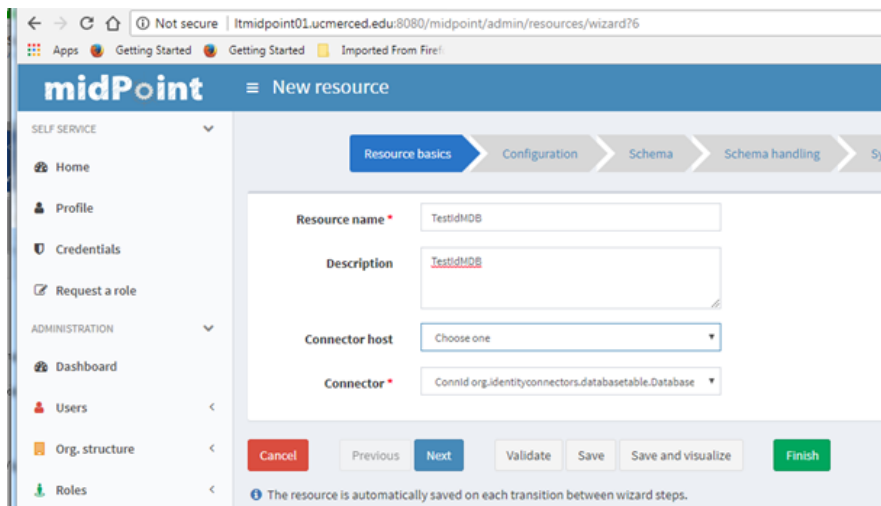
`/opt/midpoint/var/lib`

Go to: Resources -> New Resource

Resource Basics Tab:

Add Resource Name

Connector: ConnId org.identityconnectors.databasetable.DatabaseTableConnector v1.4.2 0



midPoint New resource

Resource basics Configuration Schema Schema handling Syn

Resource name * TestidMDB

Description TestidMDB

Connector host Choose one

Connector * Connid.org.identityconnectors.databasetable.Database

Cancel Previous Next Validate Save Save and visualize Finish

The resource is automatically saved on each transition between wizard steps.

Next:

Configuration Tab:

Host: <DB Server>

TCP Port: <DB Port>

User: <DB UserName>

User Password: <DB Pwd>

Database: <Oracle DB Name>

Table: <IdM Table Name>

Key Column: <Table Primary Key>

JDBC Driver: oracle.jdbc.driver.OracleDriver

Change Log Column: <for us> OPERATIONTIMESTAMP

Schema Tab:

Should automatically bring in AccountObjectClass with all DB columns. You can remove columns if you need to, but for us we accepted them all since it's just a view of columns we need in the DB anyway.

Next:

Schema handling Tab:

Click Add Object type to add mappings from DB to midPoint.

Kind: Account

Intent: default

Display name: Default Account

Make sure Default is selected.

Object class: AccountObjectClass

Add Attributes (Click plus + sign):

Select DB Attribute from drop down.

ri: NETID

ri: FIRSTNAME

ri: LASTNAME

Add Inbound mappings (Click plus + sign):

Select Authoritative.

Target: \$user/name

Target: \$user/givenName

Target: \$user/familyName

Next:

Synchronization Tab:

Click Add synchronization object to add Actions for syncs.

Name: Default Account

Kind: Account

Intent: default

Select Enabled.

Add Correlation (Click plus + sign):

Filter clause:

```
<q:equal xmlns:org="http://midpoint.evolveum.com/xml/ns/public/common/org-3">  
  <q:path>name</q:path>  
  <expression>  
    <path>$account/attributes/ri:ldapid</path>  
  </expression>  
</q:equal>
```

Add Reactions (Click plus + sign):

Choose Situation: Linked

Synchronize: True

Choose Situation: Deleted

Synchronize: True

Action: unlink

Choose Situation: Unlinked

Synchronize: True

Action: link

Choose Situation: Unmatched

Synchronize: True

Action: Add focus

Re-Select Enabled if it disappeared.

Next.

Capabilities Tab:

Finish.

Create the Import Sync for the Resource

Go to the Resource Details Page

Click on Accounts Tab:

Click the Import Button bottom left -> Create New

TaskName: IdMImportSync

Type: Importing accounts

Kind: Account

Intent: default

Object class: AccountObjectClass

Check Recurring task

Schedule interval (seconds): 300

Save.

Should now have users in midPoint

When it runs after 5 mins.

Go to Users -> List users

Users from IdM DB should be listed.

Create Export sync to LDAP

For us, it is Oracle DS

Go to: Resources -> New Resource

Resource Basics Tab:

Add Resource Name

Connector: ConnId com.evolveum.polygon.connector.Ldap.LdapConnector v1.5.1

The screenshot shows the 'New resource' wizard in midPoint. The title bar is 'New resource'. The wizard has four steps: 'Resource basics' (active), 'Configuration', 'Schema', and 'Schema handling'. The 'Resource basics' tab contains the following fields:

- Resource name ***: A text input field with the value 'ldapTest02'.
- Description**: A text area with the value 'ldapTest02'.
- Connector host**: A dropdown menu with the value 'Choose one'.
- Connector ***: A dropdown menu with the value 'ConnId com.evolveum.polygon.connector.Ldap.LdapConnector v1.5.1'.

At the bottom of the form, there are five buttons: 'Cancel' (red), 'Previous' (disabled), 'Next' (blue), 'Validate' (disabled), and 'Save' (disabled). To the right of these buttons is a 'Finish' button (green). Below the buttons, there is a note: 'The resource is automatically saved on each transition between wizard steps.'

Next:

Configuration Tab:

Host: <LDAP Server>
TCP Port: < LDAP Port>
Bind DN: < LDAP BindDN>
Bind Password: <LDAP Pwd>
Connect timeout: 300000
Maximum number of attempts: 5
Base context: <LDAP base context>
Paging strategy: auto
Paging block size: 1000
VLV sort attribute: uid
Primary identifier attribute: uid

Schema Tab:

This will bring in all ObjectClasses from LDAP server automatically.

You have to edit XML to remove objectClasses that are not needed. I just downloaded to eclipse to modify then re-uploaded it.

Next:

Schema handling Tab:

Click Add Object type to add mappings from midPoint to LDAP.

Kind: Account

Intent: default

Display name: Default Account

Make sure Default is selected.

Object class: inetOrgPerson (for us)

Add Attributes (Click plus + sign):

Select LDAP Attribute from drop down.

ri: dn

ri: uid

ri: givenName

ri: cn

ri: sn

Add Outbound mappings (Click plus + sign):

Select Authoritative.

Strength **Strong**

Source: \$user/name

Expression type: **Script**

Language: **Groovy**

Expression:

```
<script xmlns:org="http://midpoint.evolveum.com/xml/ns/public/common/org-3">
```

```
<code>
```

```
'uid=' + name + ',ou=People,dc=<campus>,dc=edu'
```

</code>

</script>

Source: \$user/name

Source: \$user/givenName

Source: \$user/fullName

Source: \$user/familyName

Make sure Default is **Still** selected.

Next:

Synchronization Tab:

Click Add synchronization object to add Actions for syncs.

Name: Default Account

Kind: Account

Intent: default

Select Enabled.

Add Correlation (Click plus + sign):

Filter clause:

```
<q:equal xmlns:org="http://midpoint.evolveum.com/xml/ns/public/common/org-3">
```

```
<q:path>c.name</q:path>
```

```
<expression>
```

```
<path>declare namespace ri="http://midpoint.evolveum.com/xml/ns/public/resource/instance-3";
```

```
$account/attributes/ri:uid
```

```
</path>
```

```
</expression>
```

```
</q:equal>
```

Add Reactions (Click plus + sign):

Choose Situation: Linked

Synchronize: True

Choose Situation: Deleted

Synchronize: True

Action: unlink

Choose Situation: Unlinked

Synchronize: True

Action: link

Re-Select Enabled if it disappeared.

Next:

Capabilities Tab:

Finish.

Create LiveSync for the Resource

Go to the Resource Details Page

Click on Accounts Tab:

Click the Live Sync Button bottom left -> Create New

TaskName: LdapExportSync

Type: Live synchronization

Resource reference: <Resource Name>

Kind: Account

Intent: default

Object class: inetOrgPerson

Select: Recurring task

Schedule interval (seconds): 300

IdM users should be synced: Oracle -> midPoint -> LDAP

It took 2-3 days to initially import 100k users from our test LDAP, so we will work on performance tuning next.