MidPoint Deployment: First Steps [MID301]

Student Lab Guide - Module 9

Evolveum, s.r.o.

Revision 4.8-LTS-A.01, 2023-11-06

This lab guide is not a standalone document and should be used only for the purpose of this training. If there are any questions during the course related to the content of the training or this lab guide itself, do not hesitate to ask the instructor.

If there are any errors, typos or typographic convention mistakes, please report them to the instructor as well. Thank you.

All labs were tested with the midPoint version used during the training.

We assume you have already installed the prerequisites before this training (if there were any).

Disclaimer

The names, organizations and places portrayed in this training course are fictitious. No identification with actual persons (living or deceased), organizations, places or events is intended or should be inferred.

Table of Contents

Module 9: Overriding Incorrect Data

LAB 9-1: Overriding Malicious User Status

Estimated Time: 10 min.

In this lab, we will show how midPoint can be used to disable user/account even if data in HR system is incorrect.

Imagine the following situation: some malicious user must be deactivated immediately. We cannot wait until the user is properly deactivated using HR `status` attribute, we must act *now*

Let’s create the user first.

In your browser with HR application:

1. click **Register user** and fill in the following attributes:
   a. **First name**: `John`
   b. **Surname**: `Newman`
   c. **Employee number**: `9005`
   d. **Locality**: `Fast River City`
   e. **Job**: `107#Junior Consultant`
   f. **EmpType**: select/keep `FTE`
   g. **Status**: select/keep `In`
   h. click **Register user**
2. click **Export users to csv file**

In your browser with midPoint:

1. wait for the next regular reconciliation with HR resource
2. go to **Users › Persons**
3. check that user `jnewman` has been created in midPoint and has Active Directory resource account

Now we will override user’s status in midPoint because we have realized the user must be immediately deactivated.

In your browser with midPoint:

1. go **Users › Persons**

2. edit user **jnewman**
3. in user's summary panel, badge **Enabled** signalizes user's effective status, also `Lifecycle state` is: `Active`
4. click **Activation** menu item
5. click **Show empty fields** if **Administrative status** is not displayed
6. set the following properties:
    a. **Administrative status**: `Disabled`
7. click **Save**
8. edit user **jnewman** again
9. in user's summary panel, badge **Disabled** signalizes user's effective status (even if user's `Lifecycle state` is still `Active`)
10. click **Projections** menu item
11. click **AD** resource account
12. scroll down to **Activation** container
13. you should see the following:
    a. **Administrative status**: `Disabled`

Malicious user and his accounts are disabled and will be kept disabled regardless of state information coming from HR resource. Reconciliation execution will not change it. midPoint has overridden the HR data for user's status.

If the user is cleared from suspicion, administrator may reset the Administrative status to **Undefined** value again, allowing HR data to be used.

LAB 9-2: Overriding Incorrect HR Data

Estimated Time: 15 min.

In this lab, we will show how midPoint can be used to enable user/account or override incorrect attribute values provided by HR data.

Imagine the following situation: user has been created in HR with some errors, e.g. incorrect locality and even status. We cannot wait until the user is fixed in HR, because this is some VIP user who needs to work, but his account is kept inactive because of those errors.

Let's create the user first.

In your browser with HR application:

1. click **Register user** and fill in the following attributes:
   a. **First name**: `John`
   b. **Surname**: `Doe`
   c. **Employee number**: `9006`
   d. **Locality**: `Fat Rover City`
   e. **Job**: `999#CXO`
   f. **EmpType**: select/keep `FTE`
   g. **Status**: select `Long-term leave`
   h. click **Register user**
2. click **Export users to csv file**

In your browser with midPoint:

1. wait for the next regular reconciliation with HR resource
2. go to **Users › Persons**
3. edit user **jdoe**
   a. check user's **Lifecycle state**: it's `Suspended`
   b. check user's **Locality**: it contains incorrect locality `Fat Rover City` instead of `Fast River City`
   c. check the user has **AD** account disabled

Change of Administrative status won't work in this case; the current implementation allows Administrative status to override Lifecycle status only to deactivate the user. Any attempt to blindly fix `Locality` or `Lifecycle state` would not work: the user is synchronized from HR by scheduled reconciliation; we must create an exception.

1. go to **Resources › All resources**
2. edit **HR** resource
3. click **Accounts** menu item
4. search for account `9006`
5. click the context menu for the account and click **Add marks**
6. check **Invalid data** mark
7. click **Add**

`Invalid data` mark causes the HR account data to be ignored for account's owner during synchronization. We can override user data now.

1. while displaying content of **Accounts** menu item
2. click the **jdoe** in **Owner** column to edit the user
    a. edit the following properties:
        i. **Locality**: set `Fast River City`
        ii. **Lifecycle state**: select `Active`
    b. click **Save**

User and his AD account are now enabled.

Wait for the next scheduled reconciliation with HR to see that midPoint overrides have not been reverted. You can even test that any updates of HR data for `9006` are ignored.

In your browser with HR application:

1. click **Show users**
2. edit **John Doe** entry using **Modify**
3. update the following fields:
    a. **Locality**: `Fast River Chaty`
4. click **Modify user**
5. click **Export users to csv file**

In your browser with midPoint:

1. wait for the next scheduled reconciliation with HR
2. edit user **jdoe** again
3. check that users **Locality** is still set to: `Fast River City`

Before we correct HR data, let's see how midPoint dashboard can help us to see which account marks are in use.

In your browser with midPoint:

1. go to **Dashboards › Account marks**
2. dashboard indicates how many accounts are using the account marks
3. click **More info** in **Invalid data accounts** tile
    a. all accounts marked as `Invalid data` on any resource are displayed, in our particular case: account `9006` on HR resource
4. go to **Dashboards › Account marks**
5. click **More info** in **Protected accounts** tile
    a. all accounts marked as `Protected` on any resource are displayed
6. go to **Dashboards › Account marks**
7. click **More info** in **All marked accounts** tile
    a. all accounts marked with any mark on any resource are displayed

If you prefer a report, that can be used too.

In your browser with midPoint:

1. go to **Reports › Dashboard reports**
2. click ▶ icon to run **Account Marks Dashboard Report**
3. click **show task** link displayed at the top of the page
4. wait until the task completes
5. click **Download report** to save the report as HTML file
6. open the downloaded HTML file in browser to display the report

To access the report outputs later:

1. go to **Reports › Created reports**
2. download desired report

Let’s correct HR data now and set a different locality.

In your browser with HR application:

1. click **Show users**
2. edit **John Doe** entry using **Modify**
3. update the following fields:

   a. **Locality**: `White Stone City`

   b. **Status**: `In`

4. click **Modify user**
5. click **Export users to csv file**

The HR data will be ignored until we remove account's `Invalid data` mark.

In your browser with midPoint:

1. go to **Resources › All resources**
2. edit **HR** resource
3. click **Accounts** menu item
4. search for account `9006`
5. click the context menu for the account and click **Remove marks**
6. check **Invalid data** mark
7. click **Remove Marks**

Wait for the next scheduled reconciliation with HR.

1. while displaying content of **Accounts** menu item
2. click the **jdoe** in **Owner** column to edit the user

   a. check the following properties:

      i. **Locality**: `White Stone City`

      ii. **Lifecycle state**: `Active`

Synchronization works again for this user.

We have seen that not just user's status but also other user properties may be overridden by administrator if needed. In addition, we have tried how account marks can be reported using midPoint's dashboard and reports.

LAB 9-3: Overriding Username

Estimated Time: 5 min.

In this lab, we will try to override username in case the generated one is not *appropriate*.

There are situations when the generated username is or resembles an insulting or offensive word. Fortunately, even in these situations, midPoint can help.

Let’s create the user first.

In your browser with HR application:

1. click **Register user** and fill in the following attributes:
   a. **First name**: `Brenda`
   b. **Surname**: `Itchy`
   c. **Employee number**: `9007`
   d. **Locality**: `Fast River City`
   e. **Job**: `191#Accountant`
   f. **EmpType**: select/keep `FTE`
   g. **Status**: select/keep `In`
   h. click **Register user**
2. click **Export users to csv file**

In your browser with midPoint:

1. wait for the next regular reconciliation with HR resource
2. go to **Users › Persons**

Based on our naming convention, the username generator generated the following username: `bitchy`. For obvious reasons, user could complain about such username.

Fortunately, there is an easy way how to override midPoint username.

In your browser with midPoint:

1. go to **Users › Persons**
2. edit user **bitchy**
3. replace the following properties:
   a. **Name**: `britchy`

4. click **Preview changes** to display what will be saved to midPoint and target system account(s)
   a. midPoint is previewing the username change in midPoint and all target system accounts (in our case, only AD)
   b. change of **Login name** in Active Directory is indicated

> Preview changes is a simple functionality of doing simulation without talking about execution mode and configuration to use. It will simply preview what would be done and allow to save or cancel the operation.

5. click **Save**

As midPoint uses the username generator only *once* because of weak mapping in `Person Object Template`, overriding the username won't conflict with the mapping. Just be careful to not select any username that already exists; in such situation midPoint would show an error message during Preview changes and/or saving the user.

For example if you would try to change the username of `Brenda Itchy` to `cwhitehe`, an error similar to the following would be displayed:

```
Too many iterations (100) for focus(user:ac6cab9f-519c-4ea2-8354-fa84664a70f8(cwhitehe)):
cannot determine values that satisfy constraints: Found conflicting existing object with
property name = PP({.../common/common-3}name):[PPV(PolyString:cwhitehe)]:
user:3fd75c16-8697-4049-aa9c-bbfcd4ea8c2e(cwhitehe)
```

Username is never automatically changed if user's Given name or Family name change.

This concludes the Module 9 labs.