Service Account Management Improvements

Planned feature

This page describes a feature planned for future midPoint versions.

This feature is roughly designed and it was evaluated as feasible. However, there is currently no specific plan when it will be implemented because there is no funding for this development yet. In case that you are interested in supporting development of this feature, please consider activating midPoint Platform subscription.

Introduction

See Service Account Management

Limitations of Service Account Management

There are still some limitations and missing pieces in Service Account Management:

- There is a concept of service ownership. This concept is using the same mechanism as role ownership. However, the service-specific processes are not implemented.
    - E.g. there is no process how to distribute new service password to all the owners. Good security policy would be to generate the password randomly (ensuring complexity criteria) and the distribute such random password to owners. So owners would not be able to choose weak password, choose the same password for several services and so on.
    - The process of re-generating the password after the change in ownership is missing.
- There is no mechanism to prevent accidental changes to service accounts. Of course, service accounts may have different mappings than ordinary accounts. However, as account type (*intent*) is determined very early in account processing there may be change that mappings from person accounts may be accidentally used on service accounts that are mis-detected.
- There is an issues with assigning vs linking. Service accounts are often just linked to the services and not assigned. The processes that handle those transitions for service accounts are missing.
- Service itself can be subject to certification processes. But the process is not tailored specifically for service accounts and there may be missing pieces.
- Account reclassification is supported only in fully automated mode, without any GUI support.
    - We need reclassification action that will "reset intent" and run whole kind/intent classification from scratch.
    - The reclassification should be manual or automatic. Currently there is only automatic reclassification. There should be some kind of condition that should only reclassify account in "unmatched" situation.
    - GUI support for account reclassification is needed

All of this can be improved. Management of service accounts is perfectly aligned with midPoint architecture and design. Just some implementation pieces are missing. And those gaps may be filled in if needed - assuming a funding for this work is provided. MidPoint subscription is the method to provide the funding.

Account Reclassification

There is no need to explicitly mark service accounts in any special way. The concept of *intent* is good enough. Therefore all the service account will have a service intent. They may be linked to service objects. If there will be no outbound mappings for service intent then the service accounts will not be modified by midPoint. In addition to that we can utilize per-object-type capabilities to disable write capabilities for service intent altogether.

The problem is how to set the service intent. Synchronization code will probably set ordinary intent (`default`) for all unmatched shadows. However, we need to change this to service intent for those accounts that are (manually) identified as service accounts. Currently (midPoint 4.0) there is no GUI functionality for this. What is more important is that there is no definition that would specify that `default` intent may be switched to `service` intent. The `objectSynchronization` seems like a good place for this, e.g.:

```
<objectSynchronization>
    <intent>default</intent>
    ...
    <reclassification>
        <targetIntent>service</targetIntent>
    </reclassification>
    ...
</objectSynchronization>
```

Therefore the GUI can use this definition to properly render the *change owner* functionality. In this case the default accounts will have two options to change owner: *change owner (user)* and *change owner (service)*. The latter option would actually execute two operations: change shadow intent to `service` and then link the shadow to service object.

There is a Synchronization Sorter mechanism that can be used for automatic reclassification. But support for manual (GUI-based) reclassification is still missing.

Note: there is almost no chance to destroy account data even if intent is mis-detected as long as the account ends up in `unmatched` situation. In that case the account is not linked therefore no mappings are applied. MidPoint will not change the account unless the account is changed manually from the GUI.

In addition to this there are expected changes in the UI for service objects. There are expected bugfixes and improvements as this part of the UI is rarely used. The projection enforcement mode needs to be applied on a per-object-type basis to resolve the assigning vs linking issue (service accounts will be linked but not assigned). Additional improvements to other part of GUI, meta-roles and other mechanisms may also be needed.

See Also

- Services
- Generic Synchronization
- Protected Accounts
- Synchronization Sorter