Synchronization of roles/groups

sync, role, groups

Role synchronization works according to the same rules as identity synchronization. In this page we will described only extra behavior **specific** for this synchronization.

What is a role and a group?

A role in IdM is a basic entity that is used to define permissions in IdM. Therefore, if such a role is assigned to an identity, the identities obtain the permissions that are defined on the role. In addition to permissions in IdM, a role can define connections to a specific external system. This means that the identity to which the role will be assigned will in this case obtain an account on the external system.

As part of the synchronization of roles, we will primarily deal with the second case, ie the connection of external systems.

As part of role synchronization, we often talk about **group** synchronization. A group is a term from external systems, typically **MS AD**, where it is equivalent to a role in IdM. Ie. defines the **membership of users in a given group**, just as a role defines the **membership of identities in a given role in IdM**.

Role synchronization is very often used to synchronize **groups and their membership from MS AD**. Loading AD groups to IdM is usually done when you want to manage the group membership of the AD users by IdM. So connecting the system for managing AD users is a logical step before you start to synchronize the groups.

Specific configuration for role synchronization

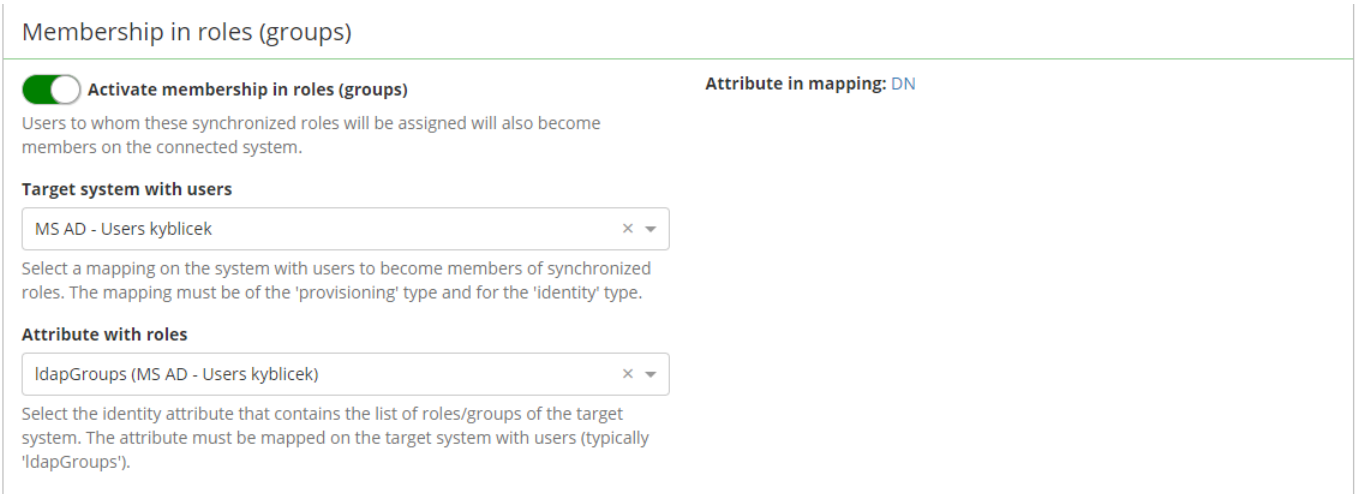
Membership in roles (groups)

The goal of membership management is to ensure that by assigning a role (created based on this synchronization) to a user, that role/group is provisioned to the end system with members.

For using this feature, we must:

- **Enable this feature** in sync configuration by the switch.
- **Choose a system** provisioning mapping that manages members. Typically, this is the system by which we manage users in MS AD.
- Next, we need to select the mapped attribute from the user system that provides role/group provisioning for the system. In the case of an MS AD connection, this is typically the "**ldapGroups**" attribute.
- Finally, you need to create an attribute in the synchronization mapping that **returns the role identifier** (*roleMembershipId*). This identifier will be used, in the provisioning of members.

By activating the membership management feature, each synchronized role in IdM will be linked to a system with members. In addition, an attribute is mapped to the role that determines what roles the user has on the membership system. As the output of the transformation to the system for this attribute, the value from the attribute identifying the role will be used. Typically this is the **DN of the role**.



Membership in roles (groups)

Activate membership in roles (groups)

Users to whom these synchronized roles will be assigned will also become members on the connected system.

Target system with users

MS AD - Users kyblicek

Select a mapping on the system with users to become members of synchronized roles. The mapping must be of the 'provisioning' type and for the 'identity' type.

Attribute with roles

ldapGroups (MS AD - Users kyblicek)

Select the identity attribute that contains the list of roles/groups of the target system. The attribute must be mapped on the target system with users (typically 'ldapGroups').

Attribute in mapping: DN

Assign roles to users

This feature will ensure that we assign synchronized roles to existing users in IdM. In other words, the external system becomes the authority for IdM.

Be careful, the roles are assigned only to the **primary** contracts of the users for now.

This operation should be **use only during** system initialization when we need to read the state from the external system (i.e. membership) into IdM.

This feature does **not support differential synchronization**. When this function is activated, difference synchronization will be **deactivated**!

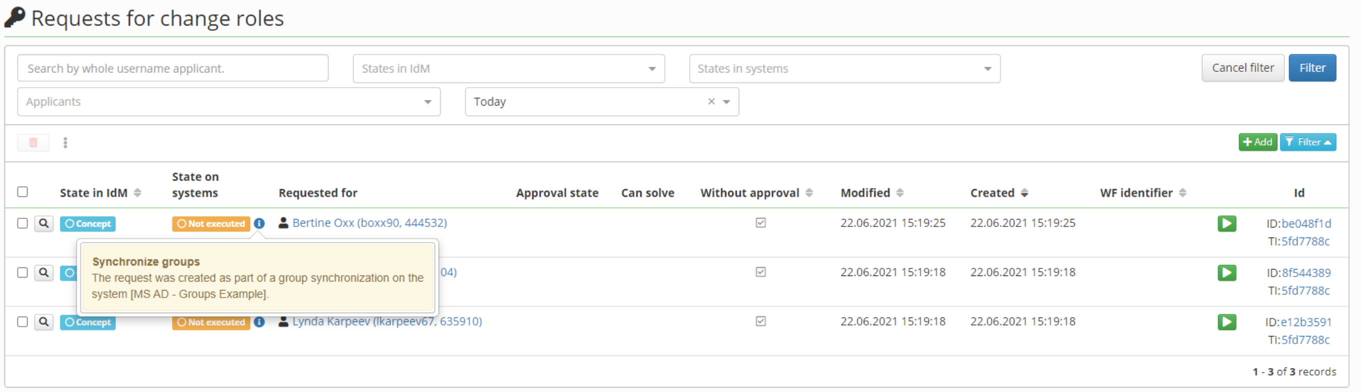
Compared to the previous solution, this operation does **not require any additional attribute** on identities containing an identifier (typically a **DN**). In order for this attribute not to be needed, the synchronization must perform a conversion between the identifier in the membership and the identifier of each user. That is, if one of the roles from the end system contains an attribute with members and those members are identified by, for example, a DN, then the corresponding identities must be found in IdM.

The problem with this task is that the identities may not contain this identifier. The solution is to **call the external system with members** by an identifier obtained from the role (**DN**). The system will return the member, including the primary identifier that is used in IdM to identify the account (typically **sAMAccountName**). This identifier is then used to **find the account and identity in the IdM**.

From the above, it follows that the end system call with members, thus takes **N** times, where **N** is the number of members. To avoid redundant calls to the system with members, the results are stored in **cache** for this case. This cache is cleared after the synchronization is complete.

If it is identified that a given role is to be assigned to an identity in IdM, then a request is created for that identity. For optimization reasons, such a request is not execute immediately, but only after the entire synchronization is complete. Thus, the goal is to create only **one request for one identity**, where this request will contain all the required role changes resulting from the synchronization.

> As you can see in the image below, role requests created during this synchronization during processing are marked with the result code **SYNC-OF-ROLES-COMMON-ROLE-REQUEST**.



Requests for change roles

Search by whole username applicant. | States in IdM | States in systems | Cancel filter | Filter
Applicants | Today

☐		State in IdM	State on systems	Requested for	Approval state	Can solve	Without approval	Modified	Created	WF identifier		Id
☐	🔍	○ Concept	○ Not executed	Bertine Oxx (boxx90, 444532)			☑	22.06.2021 15:19:25	22.06.2021 15:19:25		▶	ID:be048f1d TI:5fd7788c
☐	🔍	○		04)			☑	22.06.2021 15:19:18	22.06.2021 15:19:18		▶	ID:8f544389 TI:5fd7788c
☐	🔍	○ Concept	○ Not executed	Lynda Karpeev (lkarpeev67, 635910)			☑	22.06.2021 15:19:18	22.06.2021 15:19:18		▶	ID:e12b3591 TI:5fd7788c

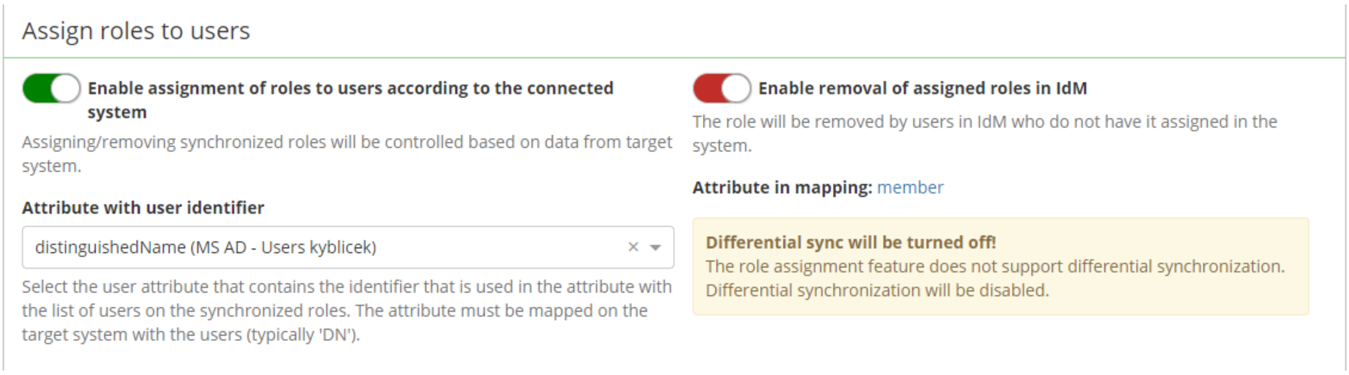
Synchronize groups
The request was created as part of a group synchronization on the system [MS AD - Groups Example].

1 - **3** of **3** records

For using this feature, we must:

- **Enable this feature** in sync configuration by the switch.
- **Select the user attribute** that contains the identifier that is used in the attribute with the list of members on the synchronized roles. The attribute must be mapped on the target system with the users (typically 'DN').
- Finally, you need to create an attribute in the synchronization mapping for returns **list of members** (roleMembers).

> You can enable **removal of assigned roles in IdM** by the switch. The role will be removed by users in IdM who do not have it assigned in the system.



Assign roles to users

Enable assignment of roles to users according to the connected system
Assigning/removing synchronized roles will be controlled based on data from target system.

Attribute with user identifier
distinguishedName (MS AD - Users kyblicek)
Select the user attribute that contains the identifier that is used in the attribute with the list of users on the synchronized roles. The attribute must be mapped on the target system with the users (typically 'DN').

Enable removal of assigned roles in IdM
The role will be removed by users in IdM who do not have it assigned in the system.

Attribute in mapping: member

Differential sync will be turned off!
The role assignment feature does not support differential synchronization. Differential synchronization will be disabled.

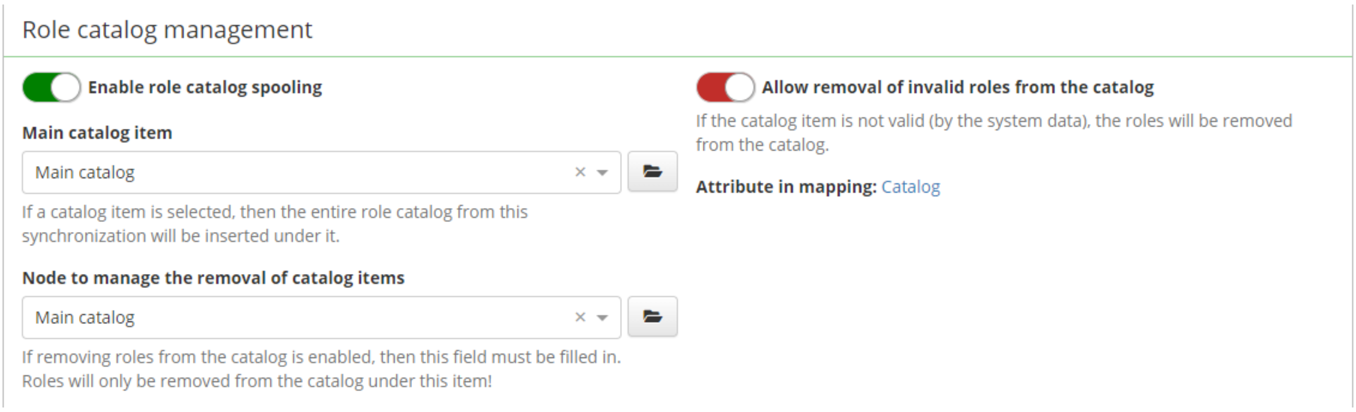
Role catalog management

Role catalog management **ensures that synchronized roles are included in the IdM role catalog**. How roles are cataloged depends on the transformation we choose in the attribute that builds the catalog.

For using this feature, we must:

- **Enable this feature** in sync configuration by the switch.
- Finally, you need to create an attribute in the synchronization mapping for **returns list of role catalog nodes** to which the role is to be assigned (roleCatalogue). If a more complex structure needs to be solved, then each node can contain (in embedded.parent) another catalog node.

You can select a **main role catalog** node. If a main catalog item is selected, then the entire role catalog from this synchronization will be **inserted under it**.



Role catalog management

Enable role catalog spooling

Main catalog item: Main catalog

If a catalog item is selected, then the entire role catalog from this synchronization will be inserted under it.

Node to manage the removal of catalog items: Main catalog

If removing roles from the catalog is enabled, then this field must be filled in. Roles will only be removed from the catalog under this item!

Allow removal of invalid roles from the catalog

If the catalog item is not valid (by the system data), the roles will be removed from the catalog.

Attribute in mapping: Catalog

The product contains two scripts addressing different role cataloging strategies:

- **Put all role catalog items under main catalog from a sync** - This script returns list of IdmRoleCatalogueDto with max one item. This DTO is main catalog from sync configuration. This script doesn't make any save or update.
- **Resolve a role catalogue by DN** - This script returns list of IdmRoleCatalogueDto with max one item. This DTO will contains whole org structure in embedded data. This script doesn't make any save or update.

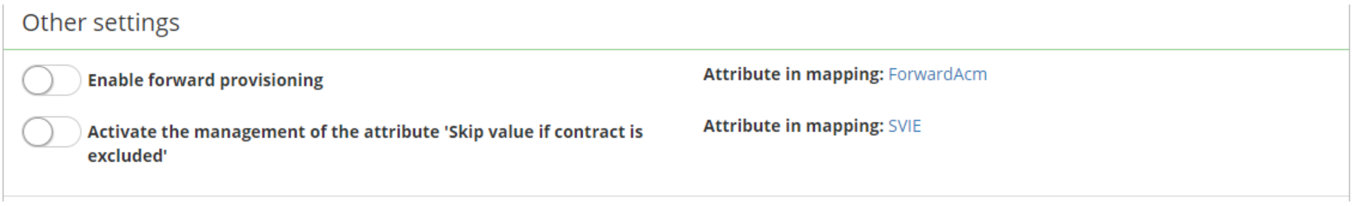
Other settings

You can activate other specific features, such as:

- **Enable forward provisioning** - Enable/disable forward provisioning in the role-to-system mapping. For properly work, you must create a mapped attribute in this system that will return a boolean value defining whether forward provisioning is enabled or disabled.
- **Activate the management of the attribute 'Skip value if contract is excluded'** - Enable/disable 'Skip value if contract is excluded' in the role-to-system attribute. For properly

work, you must create a mapped attribute in this system that will return a boolean value defining whether 'Skip value if contract is excluded' is enabled or disabled.

'**Skip value if contract is excluded**' is used if the contract is excluded, then that value will be skipped from the resulting merge of that attribute values.



Other settings

○ Enable forward provisioning — Attribute in mapping: ForwardAcm

○ Activate the management of the attribute 'Skip value if contract is excluded' — Attribute in mapping: SVIE

From:
https://wiki.czechidm.com/ - **IdStory Identity Manager**

Permanent link:
https://wiki.czechidm.com/devel/documentation/synchronization/dev/role-sync

Last update: **2023/03/28 07:34**