Multi-Connector Resource

MidPoint 3.6 and later
This feature is partially implemented in midPoint 3.6. The completion work needs your support.

Introduction

Traditionally identity management systems are limited to one connector instance per resource. This makes sense in normal situations. There are resources accessible using LDAP protocol, CSV life, database interfaces and so on. One connector is all that is needed for simple scenarios. But things to get more complicated in a real world. Combination of several connectors may be needed.

Use Cases

This is best explained using examples. Therefore we provide a couple of use cases for illustration.

Semi-Manual Resources

MidPoint 3.6 has a basic support for manual and semi-manual resources. In these cases the operation is executed manual by a human operator. Therefore the operation takes some time and midPoint has a limited visibility to the operation effects. In case of semi-manual resources midPoint has a way how to (more or less) directly read the resource content. However, some deployments will prefer to use CSV-formatted exports for reading the data. Other deployments may prefer database tables. And yet other deployments may even use a completely custom method. We do not want to support all of these methods in our manual connector implementation. In fact, there may be several manual connector implementations: internal midPoint implementation and several ITSM integration plugins. Matching the manual connector implementations and the methods of reading the data would lead to a Cartesian situation - but not in midPoint. MidPoint has the ability to combine several connectors in one resource. In this case one manual resource for writing and one ordinary resource for reading.

Provisioning and Scripting Operations

Let's have a big LDAP server with "NIS" schema (RFC2307). MidPoint is an excellent choice to manage such a system. However, managing UNIX account in LDAP is usually not enough. The account will not be of much use without a home directory. Of course, Unix PAM can be set up to automatically create home directory. But what about deprovisioning? And control? What is usually needed is for midPoint to create home directory on some kind of file server when an account is created. And to delete or archive the directory when the account is deleted. SSH protocol is usually the preferred method. However, LDAP connector does not have SSH support. It is not its responsibility. The SSH support may be expected from the Unix connector. However, this connector does not support LDAP.

MidPoint can combine these two connectors together. The LDAP connector will be used as a primary connector. It will determine the schema and execute almost all the operations. Except for scripting operations. The scripting operations may be routed to the UNIX connector. Natural combination to support LDAP/Unix deployments.

Configuration Approach

There is always one primary connector for each resource. This is the connector that is supposed to do most of the operations. This connector defines the character of the resource.

In addition to that various *additional connectors* can be configured. These connectors add capabilities to the primary connector. Additional connectors may have their own configuration separate from the primary connector.

Limitations

There are some inherent limitations for multi-connector resource:

- Same schema: This is still one resource. All the connectors must be able to work with the same schema. The schema can be statically specified or dynamically discovered by any of the connectors. But once set the same schema is used for all the connectors.
- One connector for each operation: Each operation is executed by exactly one connector. Read operations may be executed by a different

connector than write operations and yet another connector can provide scripting capabilities. But there is no way how to merge search results from two connectors or how to route create operation to two different connectors. Each operation is executed by a single connector and that is where the operation ends.

- Connector capabilities may somehow overlap. It is OK to overlap capabilities of primary connector and additional connector. The capabilities of additional connectors always take precedence. However, take care when overlapping capabilities of additional connectors. If more then one additional connector has a certain capability then any of these two connectors may be used - and it is not guaranteed to be deterministic. The recommended approach is to disable the overlapping capabilities in all but one connector. There is one quite special case though: semi-manual connectors. Manual connectors have read capability that relies on midPoint caching. Manual connector can be combined with real connector to create a semi-manual connector. In that case it is OK to leave the caching-only read capability of manual connector enabled, the real read capability of the other connector will take precedence. In fact midPoint currently relies on that caching-only read capability to be enabled and disabling that is know to cause problems.

See Also

- Manual Resource and ITSM Integration