Multitenancy

- Tenant Organization
- Tenant References
    - Multi-tenant Authorizations
- Limitations and Future Improvements
- See Also

MidPoint supports *partial multitenancy*. In fact, midPoint multitenancy is mostly just an extension of organizational structure features.

Partial multitenancy

The term *partial multitenancy* does not means that multitenancy features are only partially supported in midPoint. Although there is always something that can be improved, multitenancy is supported as a first-class citizen in midPoint. The word *partial* means, that the tenant isolation may not be complete. There are systems that provide only a complete tenant isolation, usually provided by the means of virtualization. In such systems there may be no overlap between tenants. Of course, midPoint can be deployed in a virtualization-based multi-tenant arrangement as this approach is not application-specific. No special features are required to support such deployment. Some deployments would be completely happy with that. But there are deployments that need tenants to **overlap**. For example, we may need administrators that can manage several tenants. We may need global administrators or auditors that can access all the tenants. We may need system-wide reports that cross tenant boundaries. And so on, and so on. MidPoint has features that can be used to support *partial tenant isolation*.

Tenant Organization

Tenant is defines as an org object that has `tenant` flag set to true:

```
<org oid="521d55d8-b734-11e8-9ad4-7797b5085ca0">
    <name>Atreides</name>
    <subtype>tenancy</subtype>
    <assignment>
       <targetRef oid="00000000-8888-6666-a000-000000000000" type="OrgType"/> <!-- organizational structure
root -->
    </assignment>
    <displayName>House Atreides</displayName>
    <identifier>200</identifier>
    <locality>Caladan</locality>
    <tenant>true</tenant>
</org>
```

There only a few constraints on how the tenant-based organizational structure may look like. Tenants may be placed in organizational structure of any depth. And there may be organizational structure inside the tenants themselves. Perhaps the only limitation is that there must be no dispute into which tenant an object belongs. Therefore every object has to belong to at most one tenant. This may seem obvious, but midPoint organizational structure is very flexible and multiple membership in organizational units is quiet common. But the cases where an object should be part of two or more tenants will not work. The configuration or operational procedures must make sure that this will not happen. Also, nested tenants are not supported (yet). However, it might be possible to support nested tenants or even multiple tenancy in the future.

Tenant References

MidPoint 3.9 and later

This feature is available only in midPoint 3.9 and later.

All the objects that belong to a particular tenant will contain special tenant reference (`tenantRef`):

```
<user oid="c7e44dc0-b735-11e8-a95a-c33713563f97">
    <name>paul</name>
    <tenantRef oid="521d55d8-b734-11e8-9ad4-7797b5085ca0"/>
    <givenName>Paul</givenName>
    <familyName>Atreides</familyName>
    <fullName>Paul Atreides</fullName>
    <assignment>
        <targetRef oid="521d55d8-b734-11e8-9ad4-7797b5085ca0" type="OrgType"/> <!-- House Atreides -->
    </assignment>
</user>
```

The tenant reference is automatically set by midPoint. When an object is recomputed, midPoint will analyze all the assignments and organizational structure. MidPoint will also determine a tenant affiliation at than time: it will look for an organizational unit that is marked with a `tenant` flag. Therefore tenant references work regardless of the shape and depth of tenant organizational structure. This tenant reference may be used as a convenience in search filters and similar applications. But is it critical for correct behavior of multi-tenant authorizations (see below).

Multi-tenant Authorizations

> **MidPoint 3.9 and later**
>
> This feature is available only in midPoint 3.9 and later.

Until midPoint 3.9 the multitenancy deployments were managed in almost the same way as delegated administration. However, there were some limitations. This was improved in midPoint 3.9 where dedicated multi-tenant authorizations were introduced.

Multi-tenant authorization can be used to limit actions within a tenant. From the technical point of view such authorization apply only if `tenantRef` of the subject and `tenantRef` of object/target are the same:

```
<role>
        <name>Tenant Admin Role</name>
        <authorization>
            <name>tenant admin autz</name>
            <action>http://midpoint.evolveum.com/xml/ns/public/security/authorization-model-3#read</action>
            <action>http://midpoint.evolveum.com/xml/ns/public/security/authorization-model-3#add</action>
            <action>http://midpoint.evolveum.com/xml/ns/public/security/authorization-model-3#modify</action>
            <action>http://midpoint.evolveum.com/xml/ns/public/security/authorization-model-3#delete</action>
            <object>
                <tenant>
                    <sameAsSubject>true</sameAsSubject>
                    <includeTenantOrg>false</includeTenantOrg>
                </tenant>
            </object>
            <exceptItem>tenant</exceptItem>
            <exceptItem>tenantRef</exceptItem>
        </authorization>
    </role>
```

This authorization works only if both subject and object are multi-tenant. I.e. it will not work if subject does not have tenant (no `tenantRef`) or in case that the object does not have tenant. Ordinary (non-tenant) authorizations should be used for those cases. See Authorization Configuration page for more details.

Limitations and Future Improvements

No system is perfect and there is always something that can be improved. That also applies to midPoint multitenancy features. There are several improvements that could be made:

- Workitems currently do **not** support multi-tenancy. They are not bound to specific tenants. Approval processes may still be used in some multi-tenant deployments as workitems are bound to specific users or organizations that have tenancy support. But caution should be exercised in such deployments. This can be later solved by migrating approvals to use midPoint's internal *case* objects.
- Resource and task do not support assignments (they are not "focal objects"). Resources and tasks can still be placed inside organizational structure, but there are limitations. Support for the concept of assignment for resources and tasks may be one of possible solutions here.
- Objects created by a tenant user could be automatically part of that tenant. E.g. tasks started by user or new resources created by that user might be automatically assigned to the same tenant. This is not implemented yet. This is partially related to planned Archetypes feature.
- The concept of multi-tenancy has almost no support in midPoint user interface. Ideally, the tenant administrators should be constrained to their tenants, global administrators should be able to switch between tenant and global views and so on. However, current user interface support for tenancy is the same as support for organizational structure. This is sufficient for many multi-tenant deployments. But there is definitely a room for improvement.

In case that you are interested in any of those improvements you can use midPoint subscription to place those improvements on midPoint roadmap.

See Also

- Organizational Structure
- Authorization
- Authorization Configuration