Grouper local entities

This topic is discussed in the Advanced Topics training video.

Grouper users sometimes need to create and manage entities in Grouper that are not part of a central subject source. An example is where Grouper integrates with an external database that has schemas needed for access management. These schemas must be represented in Grouper so they can be assigned to Groups/Roles/Permissions. A "local entity" can be created in the folder structure.

Local entities are not intended to be used to represent people, those should be in your subject source.

Description

A local entity in Grouper is an object in the Grouper namespace (folder structure), that non-grouper-admins can create, manage, use. It is a Java interface in the API (Entity), which has:

- id - uuid, doesn't change
- extension - system name in the folder, shouldn't change
- display extension - display name in the folder, can change
- description - free form text documentation about the entity
- name - fully qualified (including parent folders) system name
- display name - fully qualified (including parent folders) display name
- subjectIdentifier attribute - if the identifier of the entity is not valid for the extension (e.g. if it could contain a colon, or other invalid character in the grouper extension namespace), then you can put any fully qualified (including folder names) identifier here. Note, no two entities can have the same subjectIdentifier. Also, this attribute is public, meaning anyone can read (if they can VIEW the entity), or update it (if the can ADMIN the entity). Note, this security to be maintained, this assumes a hierarchical security model for folders... i.e. you must trust the owners of parent folders where the entities are stored since they can have a subjectIdentifier with a colon inside. The attribute must start with the folder where the entity is stored. This is autocreated for you, depending on your config, might be here: etc:attribute:entities:entitySubjectIdentifier Assign this to the local entity (e.g. with UI), and give the string value which is the identifier. Note: the assignment to the local entity is done with a "group attribute assignment" not an "entity attribute assignment"

For web services, you can set a password on the local entity UUID, and use the UUID as username and password as password to authenticate to web services. You can also generate a JWT private key, and authenticate that way too though there is encryption involved so its a little more complicated

Local entity subjects

Grouper entities have a subject source different than the Grouper subject source (though similar). Since there is an optional subjectIdentifier attribute, queries for search or findByIdentifier will consider that value. Also, the following subject attributes exist in addition to the group subject attributes (name, extension, displayName, description, etc) :

Attribute name	Meaning
entityIdAttribute	if there is an entity id attribute assigned, this is the value
entityId	if there is an entity id attribute assigned, it is used, if not, then this is the name attribute
entityExtension	if there is an entity id attribute assigned, this is the suffix after the entity folder name and colon, if not, then this is the extension (not of attribute)

API

You can create a local entity with the EntitySave class:

```
Entity testEntity = new EntitySave(grouperSession).assignCreateParentStemsIfNotExist(true)
      .assignName("test:testEntity").save();
```

You can find local entities with the EntityFinder class (note a grouper session must be open, and the grouper session user must have VIEW or ADMIN on the entity to show the result):

```
Set<Entity> entities = new EntityFinder().addName("test:testEntity").findEntities();
```

dfs

Local entity typeOfGroup

The "Group" object in Grouper is close to what we need for entities, they are in the namespace, they have some privileges (only ADMIN and VIEW are needed), and they have UI/WS support. The implementation of this enhancement is to have a typeOfGroup option as entity. Currently for v2.1 the options are "group", "role", and "entity".

The implementation of groups in the database has entries in the grouper_group_set table for each of the possible "lists". The only grouper_group_sets for entities are: admins, viewers.

A local entity is modeled as a grouper group object, but you cannot ad members to it, and of course you cannot add role permissions to it. Though of course if it were a member of a role, you could add individual permissions in the context of that role.

Local entity privileges

There are only two privileges for local entities: VIEW and ADMIN.

- VIEW means you can see it, its name, description, etc. With VIEW you could add it to a group or assign permissions to it in a role.
- ADMIN means you can edit it, delete it, assign attributes to it, etc.

In the grouper.properties you can designate if entities are viewable by all by default. This occurs on local entity create, and can be unassigned. This defaults to false for security reasons

```
# if set to true, then the ALL subject will be granted view on new entities
entities.create.grant.all.view = false
```

If you try to assign READ, UPDATE, OPTIN, OPTOUT to a local entity, you will get an error

Note: when you assign privileges in the API you use the AccessPrivilege class, e.g. AccessPrivilege.VIEW

Local entity auditing, change log, point in time

Entities are auditing like groups, but the categories are: entity, and the actions are addEntity, updateEntity, and deleteEntity.

There are three change log types for entities: ENTITY_ADD, ENTITY_UPDATE, ENTITY_DELETE. All other actions will appear under groups. e.g. if you add a privilege to an entity it will appear like a privilege is added to a group.

The point in time information is available, similar to point in time information on groups.

Misc

For hooks, just use group hooks and check that typeOfGroup equals 'entity'

You cannot change from an object of type "group" or "role" to "entity", and you cannot change from type "entity" to "group" or "role"

Obviously you cannot make an entity into a composite, or add a local entity as a part of a composite

Web services

Note: all web service changes are also available in the Grouper client.

You can create a local entity (or edit, delete), with the group web services and typeOfGroup

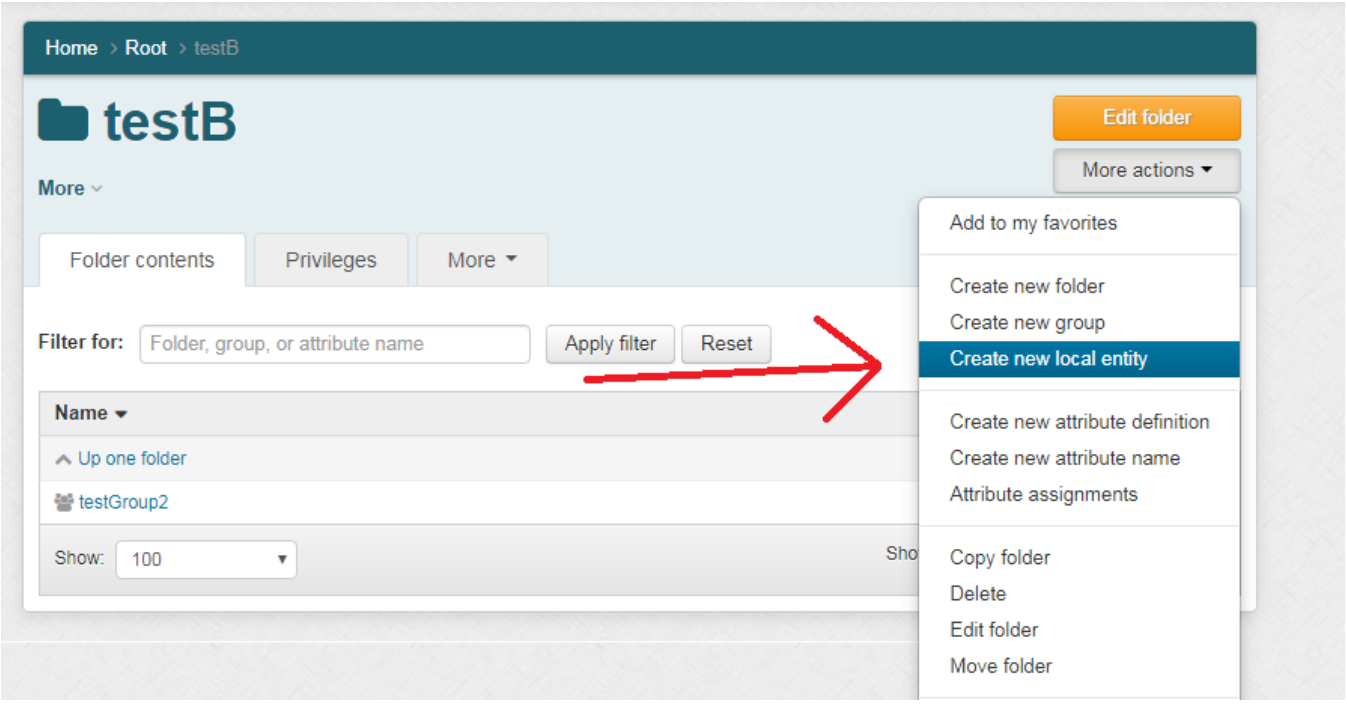
```
<WsRestGroupSaveRequest>
 <wsGroupToSaves>
  <WsGroupToSave>
   <wsGroupLookup>
    <groupName>aStem:newGroup4</groupName>
   </wsGroupLookup>
   <wsGroup>
    <typeOfGroup>entity</typeOfGroup>
    <displayExtension>newGroup4</displayExtension>
    <name>aStem:newGroup4</name>
   </wsGroup>
  </WsGroupToSave>
 </wsGroupToSaves>
</WsRestGroupSaveRequest>
```

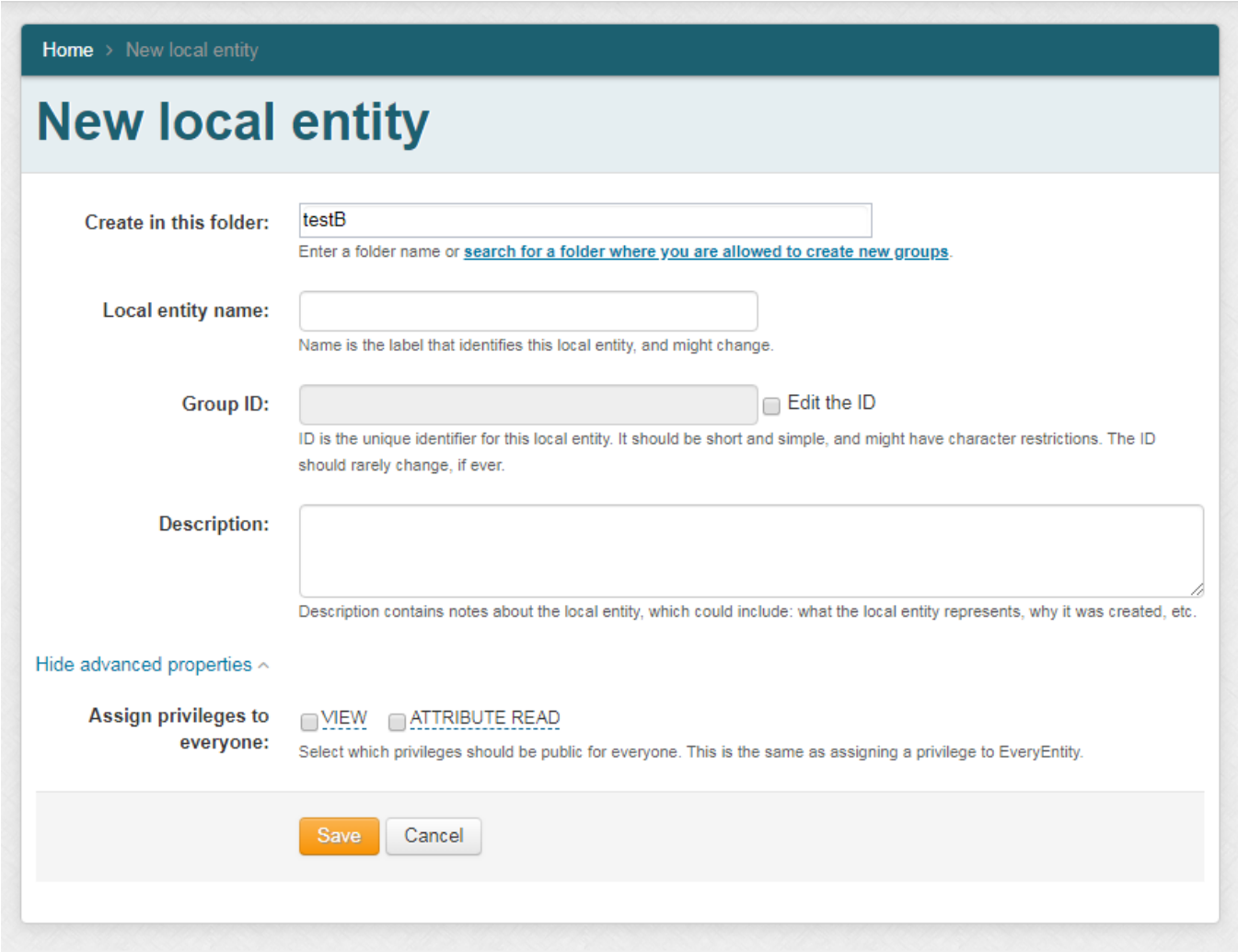
You can filter group searches by typeOfGroup also

```
<WsRestFindGroupsRequest>
 <wsQueryFilter>
  <typeOfGroups>entity</typeOfGroups>
  <queryFilterType>FIND_BY_GROUP_NAME_APPROXIMATE</queryFilterType>
  <groupName>aStem:aGroup</groupName>
  <stemName>aStem</stemName>
 </wsQueryFilter>
</WsRestFindGroupsRequest>
```

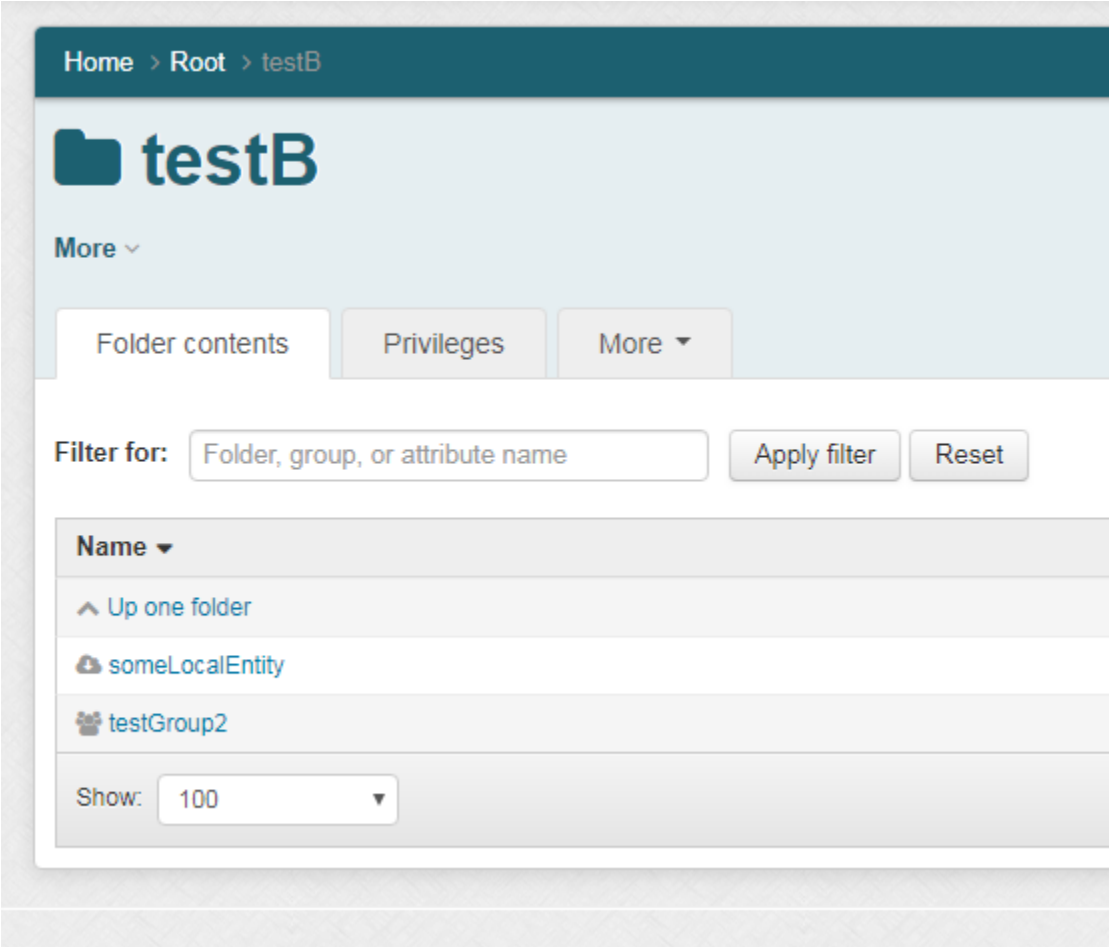
UI

You can create/edit/delete local entities on the UI in a folder you have CREATE on. In 2.4 UI patch #27 this is in the new UI

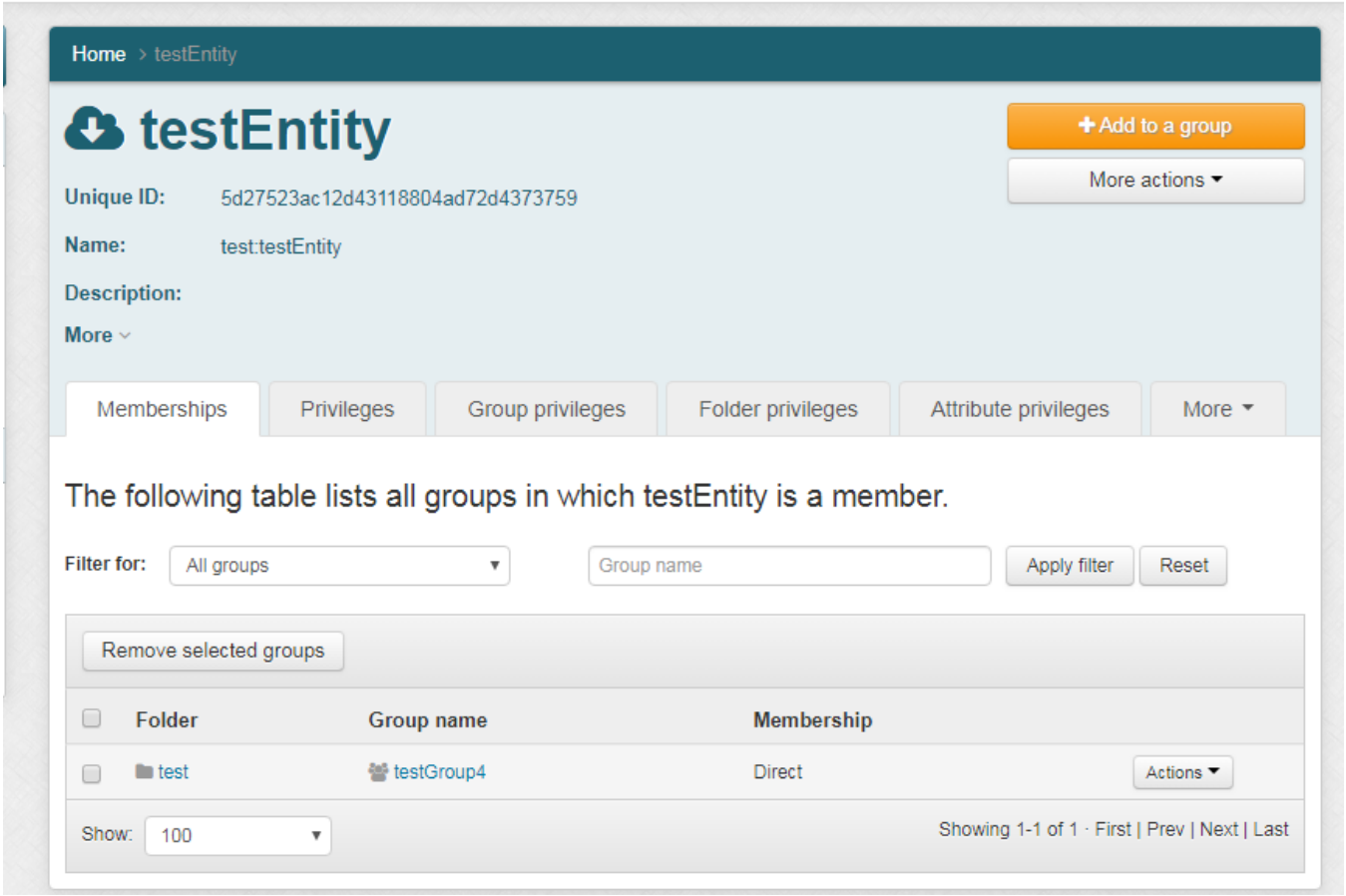




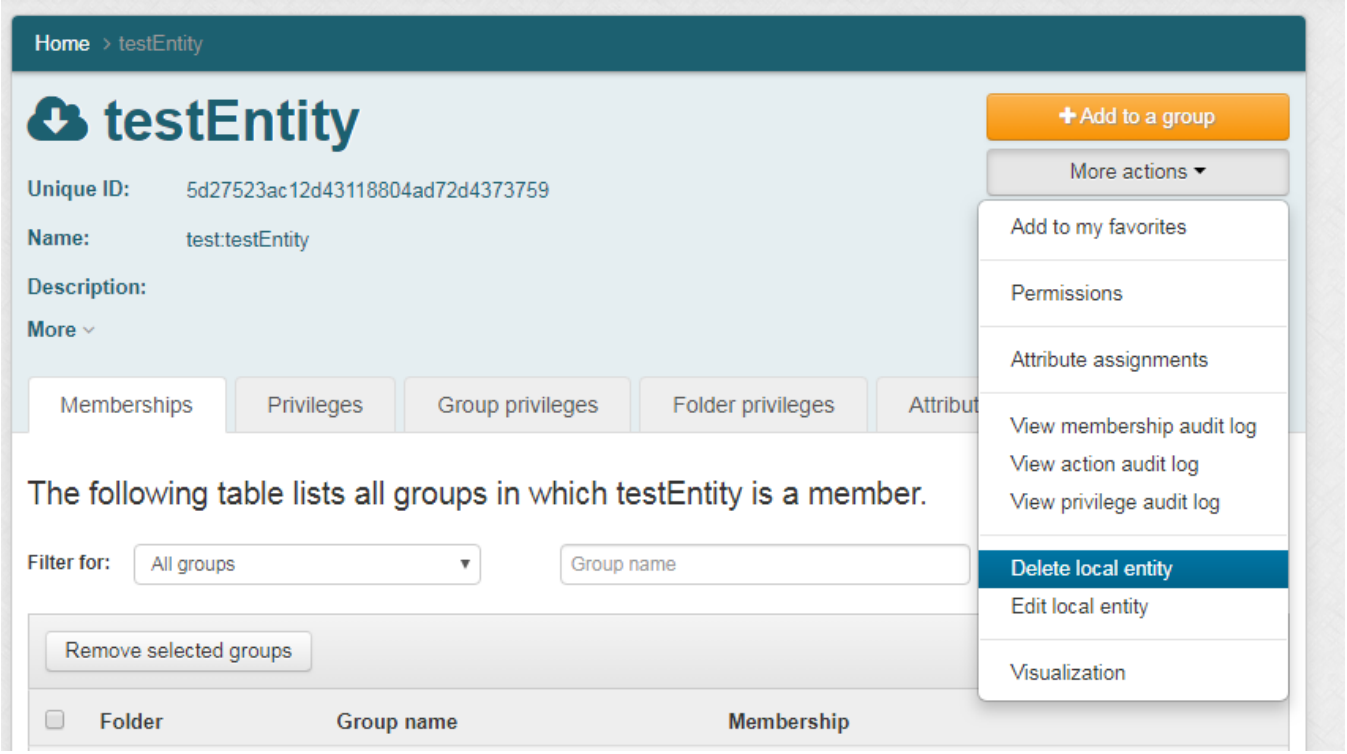
Local entity icon:



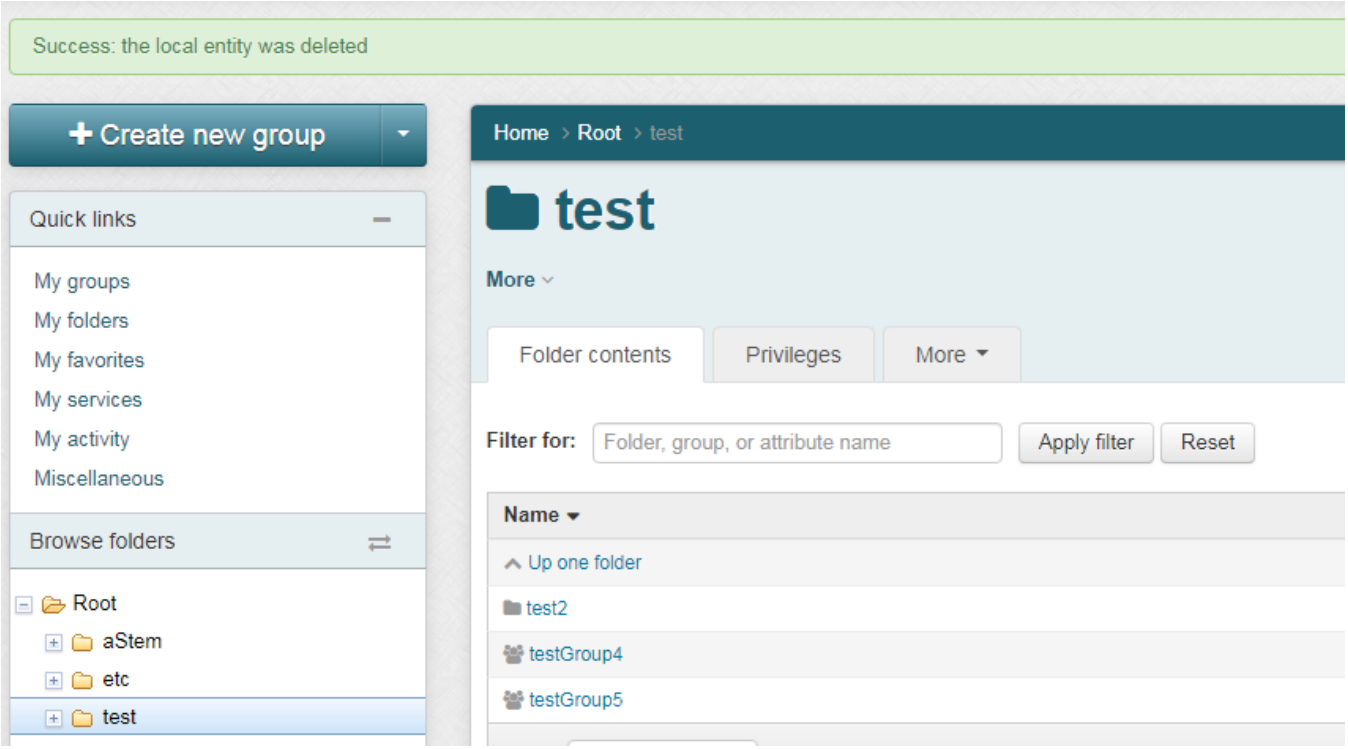
View an entity



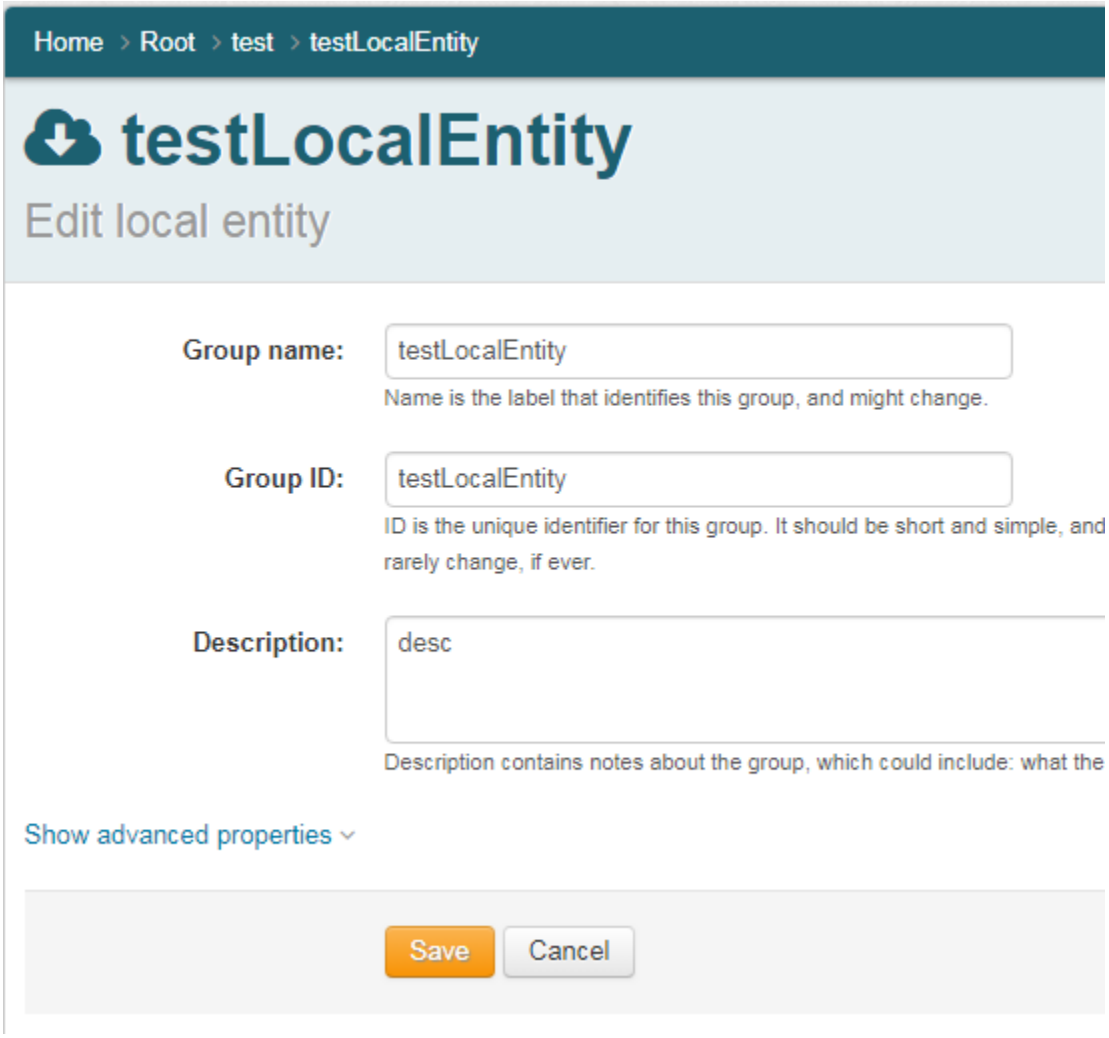
Menu has entity options



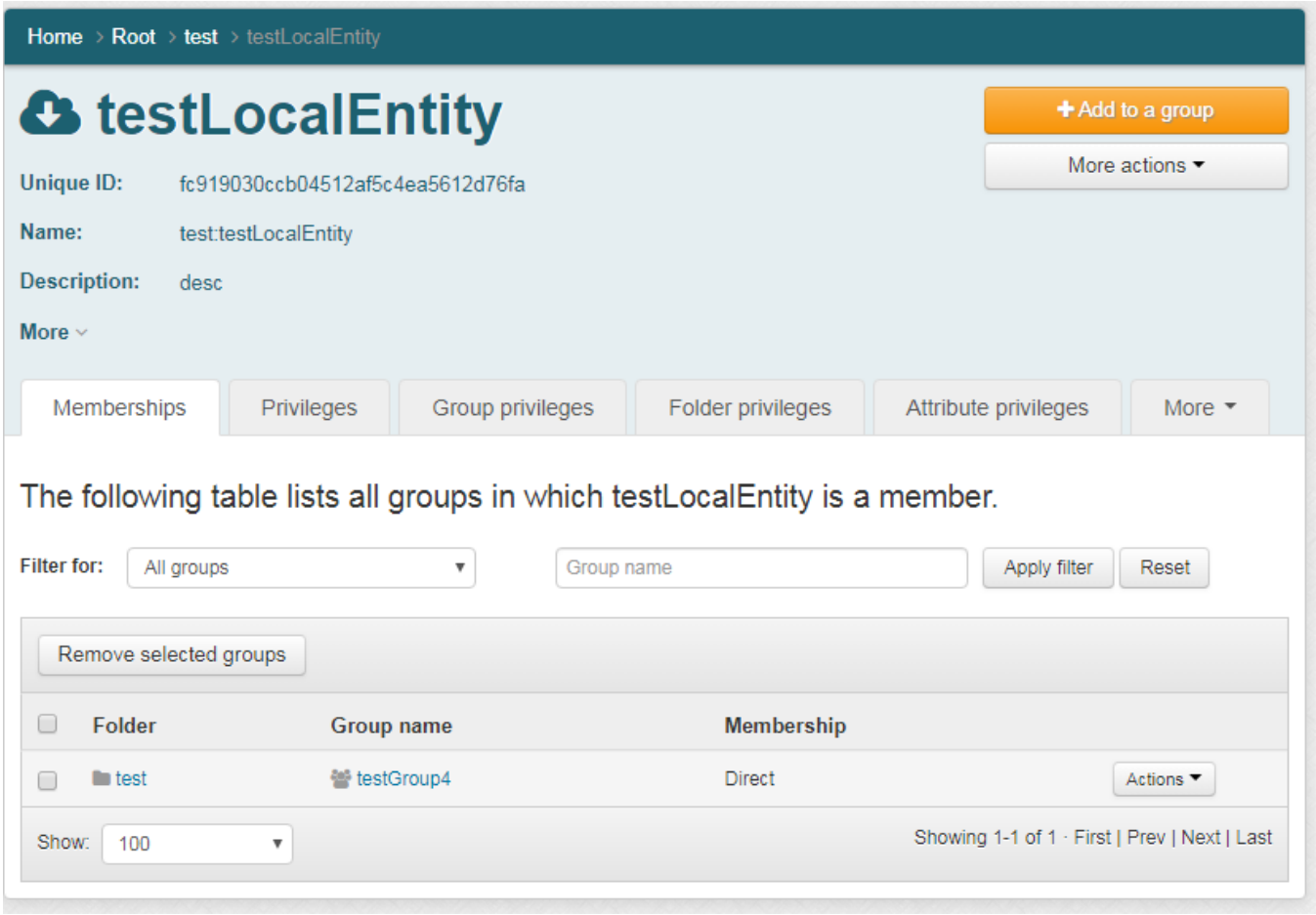
Delete a local entity



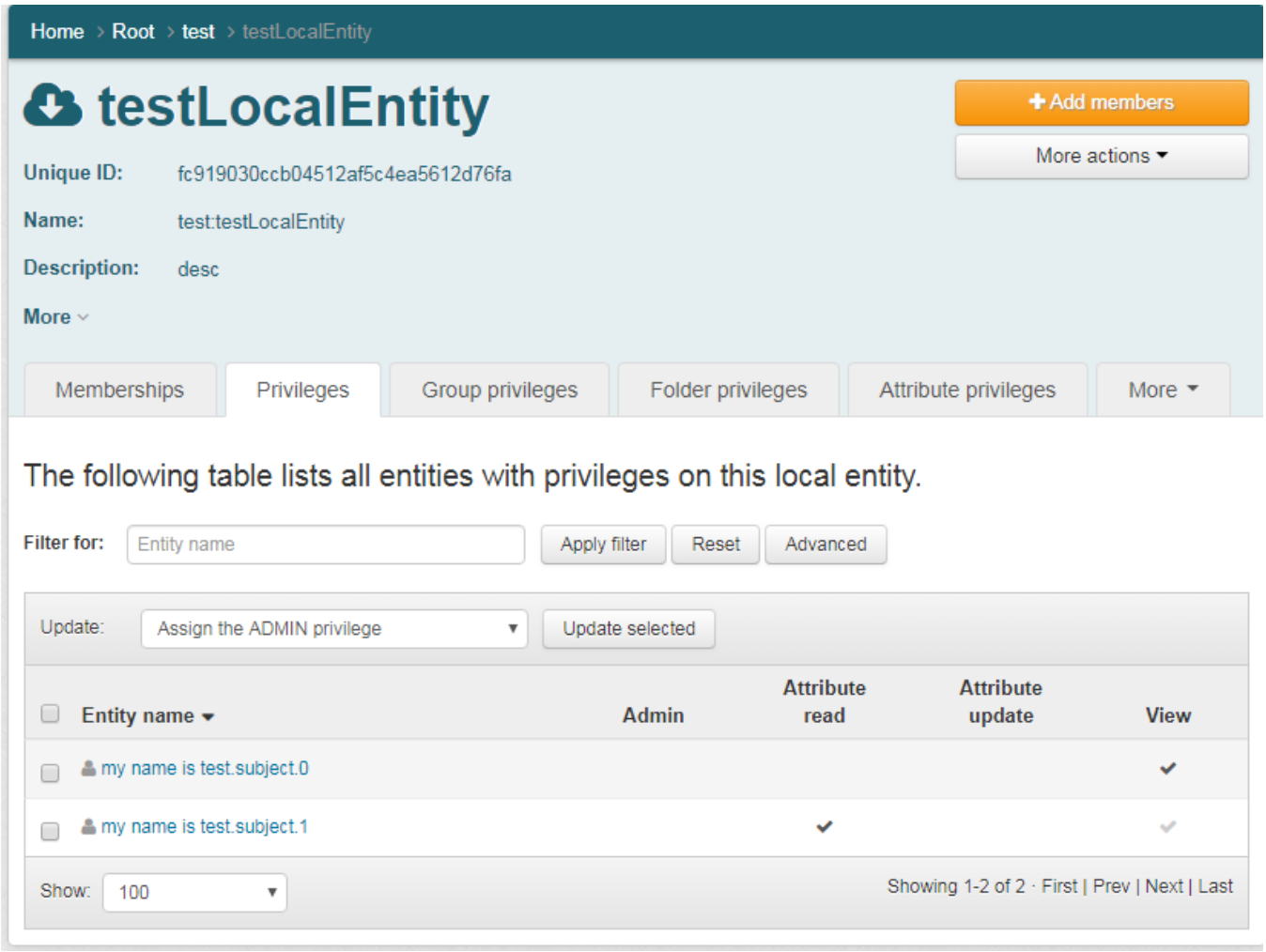
Edit a local entity



There is a privilege tab



Only entity privileges can be assigned



Limiting the scope of entities

The documentation of entities has this sentence "Entities are not intended to be used to represent people."

On the UI they should have a technical name, like "Service entity". Note: we changed the term on the UI from "Entity" to "Local entity"

LDAPPC should have a switch which defaults to off to provision these things as groups...