Infrastructure - Story #8737

Submitters and rights-holders with group permissions can't be granted authorization when the sync process validates sid

2018-10-26 22:08 - Jing Tao

Status:	New	**Start date:**	2018-10-26
Priority:	Normal	**Due date:**	
Assignee:	Jing Tao	**% Done:**	0%
Category:	d1_synchronization	**Estimated time:**	0.00 hour
Target version:			
Story Points:			

Description

During the sync process, CN has a check to avoid somebody hijacking a existing sid. In the V2TransferObjectTask class, the method validateSeriesId has the code:

```
SystemMetadata sidHeadSysMeta = getSystemMetadataHandleRetry(nodeCommunications.getCnRead(), sid);
            if (!AuthUtils.isAuthorized(
                    Collections.singletonList(sysMeta.getSubmitter()),
                    Permission.CHANGE_PERMISSION,
                    sidHeadSysMeta)) {

                if (logger.isDebugEnabled()) 
                    logger.debug("Submitter doesn't have the change permission on the pid "
                            + pid.getValue() + ". We will try if the rights holder has the permiss
ion.");
                if(!AuthUtils.isAuthorized(
                    Collections.singletonList(sysMeta.getRightsHolder()),
                    Permission.CHANGE_PERMISSION,
                    sidHeadSysMeta)) {
                    throw new NotAuthorized("0000", "Both the submitter and rightsHolder does not 
have CHANGE rights on the SeriesId as determined by"
                            + " the current head of the Sid collection, whose pid is: " + pid.getV
alue());
                }
            }
```

It passes the subject of the submitter or rights-holder to the method AuthUtils.isAuthorized to check the permission. However, this AuthUtils.isAuthorized doesn't look the group information at all. So the authorization depending on the permission fails.
We need to develop a new isAuthorized method which will take one single subject (person or group), then look the group information associated with the given subject (the depth may limit to 10) and do a comprehensive authorization process.

History

#1 - 2018-10-26 22:09 - Jing Tao

- Assignee changed from Rob Nahf to Jing Tao

#2 - 2018-10-26 23:05 - Matthew Jones

Agreed that we need to check the group information. We should have existing functions that do access control checks in our code base, so there should not be a need to write new code for this -- it should be covered by existing functions. In Metacat I think we do this in edu.ucsb.nceas.metacat.dataone.D1NodeService.isAuthorized(), which includes group checking given a Session that already is populated with Groups. Where is our current library for checking access policies for the CN components? Shouldn't such a function be in d1_libclient_java?

#3 - 2018-10-27 04:10 - Jing Tao

Maybe we don't really need a new method. In the AuthorUtil class which is in d1_common_java has some method to look group information. Maybe we can use those methods together.
Here is little different to our API- it just compare a subject object rather than a session object.

#4 - 2018-11-28 17:12 - Rob Nahf

AuthUtils.isAuthorized(...) is able to check groups, so long as they are in the List parameter. The List parameter should be populated with the Subjects from a getSubjectInfo request. Typically, a CILogon cert contains the session subject and SubjectInfo, which would contain equivalent identities and groups of the main certificate subject.