

Infrastructure - Bug #7927

Archive operation not permitted for V1 readonly MNs

2016-11-09 12:24 - Dave Vieglais

Status:	Closed	Start date:	2016-11-09
Priority:	High	Due date:	
Assignee:	Jing Tao	% Done:	100%
Category:	Metacat	Estimated time:	0.00 hour
Target version:	CCI-2.3.1	Story Points:	
Milestone:	None		
Product Version:	*		
Description			
While attempting to archive content for SEAD in the production environment using a CN certificate to authenticate:			
d1_common.types.exceptions.ServiceFailure: name: ServiceFailure errorCode: 500 detailCode: 4972 description: Couldn't determine the authoritative member node storage version for the pid seadva-HsuLeslie029090a9-11b8-4fc1-bf76-bb5a8153363f			
The request is authenticated OK, but fails when the CN attempts to determine the version of the Storage API provided by the MN.			
In this case, the MN is a Tier 1 node using the V1 API.			
The problem lies in: /edu/ucsb/nceas/metacat/dataone/CNodeService.java at around line 601.			
If the returned version is null then the node does not implement the storage API. In this case, the CN should check the version of the read API. If that is version 1, then the archive request should proceed since the CN is authoritative for that sysmeta. If the read API is V2, then the request should fail because the MN is authoritative for the sysmeta.			
PIDs to be archived include:			
seadva-HsuLeslie029090a9-11b8-4fc1-bf76-bb5a8153363f seadva-nonee903e476-9bdd-4332-823a-aabea162acd6 seadva-EssawyBakinam066de0b8-a0c9-4724-913a-9060f82148f1 seadva-EssawyBakinamc8e53366-8745-4009-bb38-786ee49cd6fe seadva-EssawyBakinam70c6e869-5518-4c75-8d09-6a808bb41fb3 seadva-ZhouQuane9e0f510-1599-4311-a6ed-ecf803f3481f			
Related issues:			
Related to Member Nodes - Task #7929: Archive content for SEAD		In Progress	2016-11-09
Related to Member Nodes - Task #7930: verify obsolescence in tDAR sandbox		In Progress	2016-11-09

History

#1 - 2016-11-09 13:41 - Dave Vieglais

- Description updated

#2 - 2016-11-09 16:32 - Jing Tao

For v1 read-only MNs, we can allow CNs to change the system metadata object. I can imagine a v2 read-only MN maybe needs us to help to archive an object as well. In this approach:

If the returned version is null then the node does not implement the storage API. In this case, the CN should check the version of the read API. If that is version 1, then the archive request should proceed since the CN is authoritative for that sysmeta. If the read API is V2, then the request should fail because the MN is authoritative for the sysmeta

CNs can't help it.

#3 - 2016-11-09 17:11 - Dave Vieglais

- Related to Task #7929: Archive content for SEAD added

#4 - 2016-11-09 23:53 - Jing Tao

CN.setAccessPolicy
CN.setObsoletedBy
CN.setReplicationPolicy
CN.setRightsHolder
have the same issue.

After I looked at code, I found we are using the version of "MNStorage" as the version of the MN. It seems wrong. We should just use "MNRead" to determine the version of the MN.

#5 - 2016-11-10 00:15 - Dave Vieglais

Either MNRead or MNCORE. Those must be supported by any MN and should always be able to determine version.

#6 - 2016-11-16 23:57 - Jing Tao

- % Done changed from 0 to 100
- Status changed from New to Closed

We are using MNRead to determine the version. I tested it and it worked.

#7 - 2016-12-01 23:48 - Dave Vieglais

- Related to Task #7930: verify obsolescence in tDAR sandbox added

#8 - 2017-01-07 02:35 - Rob Nahf

- Status changed from Closed to In Progress
- % Done changed from 100 to 30

This is not really a bug, because it is newly designed behavior when we implemented V2. Synchronization accepts system metadata changes from v1 read-only MNs as if they were V2 nodes, as was discussed and agreed upon. Checking for v1 MNStorage service was the correct way to do this.

If we want to be able to assist MNs who don't want to (or can't) make changes to their system metadata, we should limit this special access to only CN/MN administrators.

The v1.archive implementation is broken, however, in that it delegates authorization to the v2.archive implementation, which only allows CN and MN administrators. This effectively disallows object rightsHolders from updating their systemMetadata objects.

#9 - 2017-01-09 19:47 - Jing Tao

The issue is how we determine the version of a member node. We used the storage version as the version of the member node. However, for read-only node which doesn't support the storage, the mechanism will fail since it doesn't have it. So we change to the version of the MN.READ which every member node should support.

#10 - 2017-01-11 21:09 - Jing Tao

- % Done changed from 30 to 100

- Status changed from In Progress to Closed

Create a new ticket to address the permission issue about cn v1.archive:

<https://redmine.dataone.org/issues/7964>