

Infrastructure - Bug #2572

Metacat MN replication fails when EML describing access control is replicated before the data

2012-04-03 23:09 - Ben Leinfelder

Status:	Closed	Start date:	
Priority:	High	Due date:	
Assignee:	Ben Leinfelder	% Done:	100%
Category:	Metacat	Estimated time:	0.00 hour
Target version:	Sprint-2012.13-Block.2.3	Story Points:	
Milestone:	CCI-1.0.0		
Product Version:	*		
Description			

History

#1 - 2012-04-03 23:18 - Ben Leinfelder

- Category set to Metacat
- Assignee set to Ben Leinfelder
- Priority changed from Normal to High

From the standalone Metacat perspective:

- 1) data file is inserted in Metacat
- 2) EML is inserted in Metacat and the EML describes access control rules for the data file
- 3) docid-guid mappings are created for the data file so that the docid of the data file (in EML) can be found in the Metacat DB and linked to the access control rules (stored per guid).

From the Metacat replication target perspective

- 1) the EML file is replicated from source to target
- 2) the data file's access control rules are parsed and written to the DB
- 3) this includes the data's docid->guid mapping
- 4) when there is an attempt to replicate the actual data file, it fails because the identifier is already "in use" but the actual object is not replicated.

This is tricky because we need that data docid->guid mapping in order for Metacat to function as it has been WRT data access control rules as specified in EML documents. But we can't go willy nilly ignoring the fact that an identifier is now in use when we attempt to replicate the data file.

#2 - 2012-04-09 23:02 - Ben Leinfelder

- Status changed from New to Closed

I think our recent change to NOT write EML-embedded access rules to the DB when handling EML in the D1 API will make this issue go away. The only reason we were getting these phantom docid-guid mappings was because we processed and wrote access control rules for data files that happened to be described in EML files.