VOMS ACLs in StoRM

Luca Magnoni - INFN CNAF

CERN - Pre-GDB

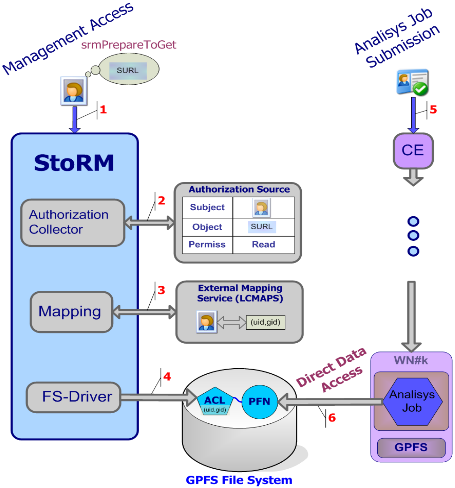
05 June 2007

Outline

- Authorization operations sequence diagram.
- VOMS attributes in StoRM.
- Approchable rules.
- Authorization sources.
- ACL enforcing: JiT and AoT.

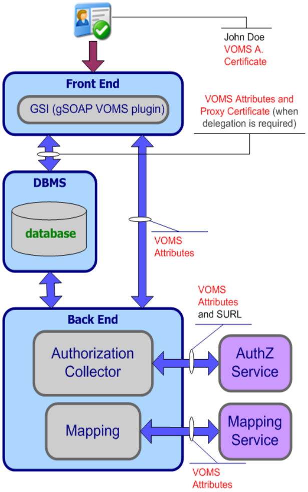
Authorization operations sequence diagram

- Sequence diagram for authorization operations in case of a file access request.



VOMS attributes in StoRM

- The VOMS attributes are retrieved from the user proxy by the StoRM Frontend through the CGSI_GSOAP plugin.
- The attributes are stored into dedicated tables in the StoRM catalog, to represent the user credential and the requestor identity.
- The attributes are evaluated for **authorization** operation by the StoRM Backend.



Approachable rules

- The Glue Schema defines the Access Control Base Rule (ACBR) per Storage Area (SA).
  - ACBR represents the **authorization rule** for the SA, or in other words which VO can approach (or view) the SA.
- StoRM is able to represents this information by a configuration features named **approachable rule**.
  - Approachable rules are defined per Storage Area and they are expressed by **regular expression in terms of FQAN**.
  - The SURLs within a request will be considered well formed if and only if the requestor is compliant with the specific app-rule.
  - Currently the default value for app-rules is **".*"**, so all FQAN can approach every SA.

Authorization source

StoRM is able to interact with external authorization services (named **authorization sources**) to perform the authorization decision.

- OGSA AuthZ WG is defining a standardized AuthZ Query Interface
- We suppose that external service answer question like *"Can USER perform the ACTION on this RESOURCE"*.

We distinguish two kind of authorization sources:

- **Local**: based on configuration file or local information.
- **Global**: external service.

Local authorization source

Local authorization source holds AuthZ policies valid only for storage area or for the entire storage element.

- Basic policy: Permit/Deny All .
- **Regular expression per path** (permissions equals on the same path) A simple XML file which defines AuthZ policies on the bases of directories (i.e. files within a directory will hold the same ACL).

Global authorization source

Global authorization source holds AuthZ policies valid for all storage resources accessible by VO-users.

- Currently StoRM can uses ECAR, a client for the **LFC catalogue**, to retrieve AuthZ information based on the ACLs on LFN.
- In the future version StoRM will have a Policy Enforcement Point (PEP) bound with external tools as the **G-PBox**. (G-PBox is an highly distributed policy management and evaluation framework).

ACL files and directories.

StoRM uses **ACL on file and directories** to enforce authorization decision.
StoRM interacts with the **LCMAPS** service to retrieve the **local uid** and **gid**. Two approach for ACL enforcing:

- **Just in Time (JiT)**
- **Ahead of Time (AoT)**

ACL Enforcing approach: Just in Time and Ahead of Time.

Just In Time

- Guarantee a secure local access also in scenario where users on the same pool account does not share the same permission on file access (E.g. Grid financial analysis).
- ACL enforced for **local user_id** when a SrmPtP/SrmPtG request come.
- ACL removed when the SrmPutDone/SrmReleaseFiles operation take place.

Ahead of Time

- Approach for scenario where the users on the same pool account share the file access permission.
- ACL enforced for **local group_id**.
- ACL will remain in place until the file or directory exists and the policy still remain valid.

Conclusion

StoRM provides:

- A layered security mechanism.
- Interaction with external authorization services (as the LFC catalogue).
- Enforcement of physical ACLs on file and directory for the local user identity corresponding to Grid credentials.

StoRM

StoRM
Storage Resource Manager

http://storm.forge.cnaf.infn.it

INFN GRID

Antonia Ghiselli

Alberto Forti
Luca Magnoni
Riccardo Zappi

egrid

Ezio Corso

Massimo Sponza