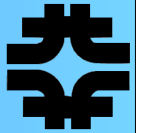


Privilege Project

June 28, 2004



Privilege Project Goals



The goal of the Privilege and Authorization Project is to enable finer grained authorization to grid enabled services

- ➔ Areas of authentication management have had longer to develop
 - Verification of user identity using a grid certificate and a trusted certificate authority
- ➔ How to establish and enforce policy for what that user can do and how many resources can be utilized is not well implemented
 - In Grid2003 users were mapped to group accounts
 - Mapping was many to one and static
 - Access was controlled on a very coarse level using the UNIX access writes of the group account the user is mapped to

We would like to enable the site to define and enforce finer grained auth.

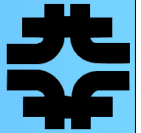
- ➔ Enforce quotas, provide access to specific services based on activity

Allow users to define their roles and activities dynamically

- ➔ Obtain different authorization for different activities



Slides from the first work plan



We started to develop improved authorization on the processing services

- ➔ We expect the Globus gatekeeper callout can be extended to storage
 - Needs to be somewhat richer
- ➔ Development systems with latest version of middleware are installed
 - Trying to run example code this week
- ➔ Getting VOMS services up
- ➔ Establishing project

infrastructure

CVS Repositories

Mailing Lists

Accounts for off-site people



Starting with Processing

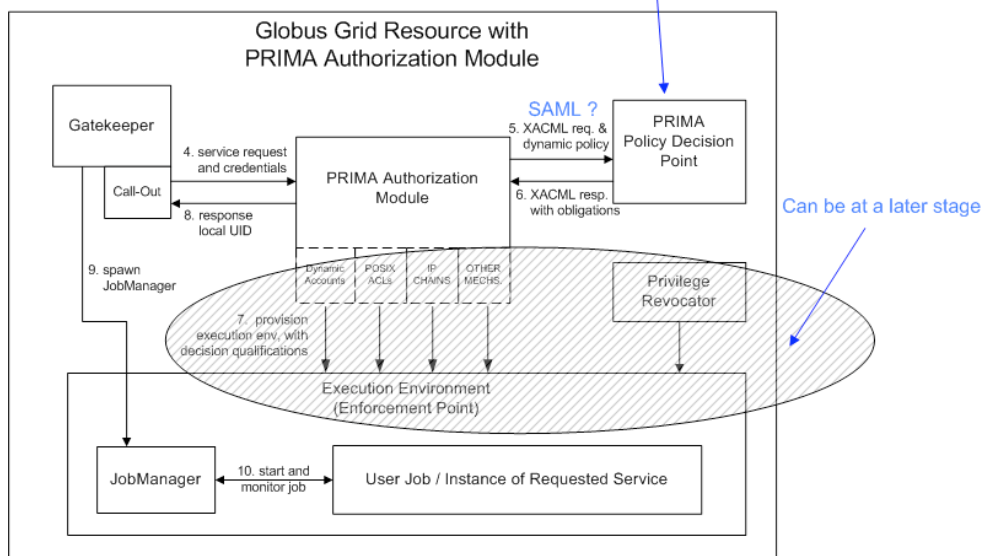
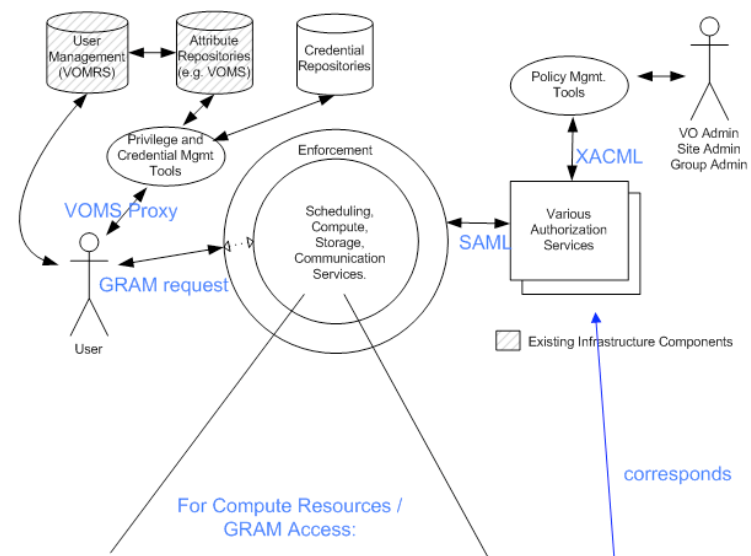
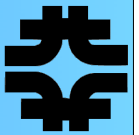


US-CMS will start testing authorization with processing testbed

- ➔ I hope the modules in question are easily applicable to storage

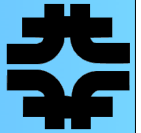
Proposed Infrastructure Setup

- ➔ Setup Development System
 - System is available (gyoza7)
- ➔ Install Globus 3.2
 - John will get started on this next week
- ➔ Start working with the callout and the globus examples
 - Make sure it works like we expect
- ➔ Contact our friends in VOMSRS to make sure we can get a VOMS extended proxy with information we can make selections with
- ➔ Practice submitting fork jobs into the testbed system
- ➔ Setup common cvs server





Currently Activities



Adapting prototype authorization module to work with current gatekeeper

- ➔ Working on enabling authorization module to work with storage gatekeeper
 - Also investigating other techniques for storage plugin

Adapting prototype authorization module to parse extended proxies

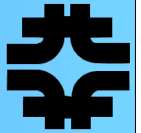
- ➔ Enabling VOMRS infrastructure for end to end testing
 - Register Users
 - Assign group memberships
 - Enable users to create extended proxies that indicate membership and role
 - Allow processing and storage gatekeepers to change authorization and policy based on extended proxy

Developing prototype policy decision point

- ➔ Policy communicated through developing standard XACML and SAML
- ➔ Working with our BNL colleagues to have GUMS interface with this



Current Steps



1. Install Testbed gateway, verify Globus callout

- ➔ Success defined as two people with the same credentials can be mapped differently when the extended proxy is changed

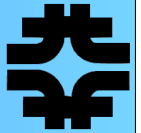
2. Provide interface to Policy Decision Point

- ➔ Success defined as policy decision point installed on system separate from authorization model
 - Secure protocol connection

3. Provide ability to change user mapping based on credential and extended proxy.

- ➔ Success defined as being able to map some users to local accounts, some users to a pool of accounts with leases, some users to a pool of accounts dynamically, and some users to group accounts
- ➔ Change mapping with extended proxy changes

At the moment we are almost through step one, so there is plenty to do



Computing:

Even after step 3, we only have the user account mapping issues addressed

- ➔ How to use the authorization module to authorize more advanced services?
- ➔ General communication with authorization module

Storage:

Implement Authorization Module for use in storage gateway

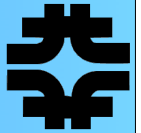
- ➔ Problem is similar
 - Does not solve issues of directory access or permissions to update and delete, but would solve initial owner creation issues

More discussion about pushing storage authorization down to OS or up to policy system

- ➔ Advantages and problems associated with both



User centric vs. VO centric



Up to now we have officially had a User centric model

- ➔ Sites responsible for individual authentication and authorization

In reality there has only been a very small number of users doing a predictable set of tasks on behalf of the VO

- ➔ As we try to enable more users and a more diverse set of activities on the resources, we need to improve our local authorization ability

In a purely VO centric authorization model some of the authorization issues are simplified

- ➔ Site trusts the VO, if a VO job asks for access to files the user must have had access, service requests are within a predefined agreement, misuse is blamed on the VO directly
 - Not a new concept, this is how CMS used SRB
- ➔ Some elements of both user centric and VO centric are needed
 - Big sites probably need to trust but verify with the VO
 - Local non-grid enabled activities need to be transparent