Deleted: 02/09/2003

Deleted: 02/09/2003

Deleted: 22/07/200303/07/2003

Subject: **INITIAL THOUGHTS ABOUT VOMS SETUP AND OPERATIONS**

Author(s): **I. Augustin, J. Closier, S. Burke, A. Fanfani, T. Wildish**

Distribution: **WP8 – TWG, finally ATWG and WP7/LCG Security Group**

1. VOMS

This note is supposed to provide an initial input for the setup of the first VOMS in EDG. However, thoughts and examples are not restricted to the scope of EDG. Some of the use cases might not be implemented within EDG. VOMS as a (relatively) stand-alone system to administrate users and their rights is likely to be used well beyond the lifetime of EDG.

2. SOME DEFINITIONS AS UNDERSTOOD BY WP8

Groups:

- Collections of people (or resources) with common capabilities

Role:

- Temporary extensions to group members' capabilities (e.g. Group admin, production manager, VO admin…)
- Must be declared within `voms-proxy-init`

Capabilities

- Grant access to resources (CASTOR write, priorities, number of concurrent jobs, etc.)
- Resources ALSO have capabilities (*have disk, will serve ALICE, but only if RAL based, Italians only on Mondays, Germans when hell freezes over*)

3. VOMS AS DYNAMIC ADMINISTRATION OF VOS

We see VOMS as a statement of the p*olicy* of the experiment. The VOMS enshrines the policy of the experiment about who is allowed to do what and how they are granted access to resources. This is not static, even if the population of the experiment were static. It is highly dynamic: policies may change daily.

Here are some not entirely ridiculous scenarios to try to clarify that:

1. QCD studies require a lot of data to be crunched, so the 'QCD Analysis Coordinator' role in VOMS is granted a lot of access to resources at high priority. Then someone in the 'EGamma Analysis Coordinator' role announces a hint of the Higgs has been found, but they need more studies to find it. The experiment policy shifts overnight, the 'QCD Analysis Coordinator' is told to take a holiday and the 'EGamma Analysis Coordinator' is given sweeping access to everything in sight.

2. 'Professor X' may have significant access to resources, because he’s important. His student, 'Student Y', has minimal access. The Professor wants an important piece of work done, and delegates to the student. 'Student Y' then must be given access to resources available to 'Professor X' for the duration of 'Task T'. How do we determine who can delegate which rights to whom?
3. VOMS should be able to influence the existing state of a grid. If 'Student Z' is running jobs for the 'QCD Analysis Coordinator' when 'Student Y' starts running the Higgs search for 'Professor X' then maybe 'Student Y' should have the rights to cause jobs belonging to 'Student Z’ to be killed or rescheduled to make way for his own jobs.
4. Analysis groups A and B meet once every two weeks, in rotation, (as done in CMS!). Immediately after group A meets we want to give higher priority to group B, so they can finish last minute things before their next meeting. After the group meeting, group A takes higher priority again. Think of it as passing a 'my-turn-now token' around the groups.
5. Analyses due to be presented at conferences are a special case of 4). The people preparing papers for the conference get higher priority in the last few weeks leading up to the conference.

We think this is enough to show that the experiment policy can be expected to change on an almost daily basis. If the VOMS does embody the policy (as we believe it should) then it must be able to react to such policy changes in less than a day.

This also means that a grid must be able to retroactively (?) adjust priorities according to new policies. E.g., take example 4: group B has its meeting, but minutes before it they submit one week of work to the grid. Group A will be starved for resources before their next meeting, which is not fair. We call this 'Policy inversion', by analysis of 'priority inversion' in non pre-emptible kernels.

4. USE CASES

4.1. REGISTRATION PROCEDURE

A VO manager expects to receive some information (by mail with link to a web page) when a new user wants to be part of a VO. This mail or web page should provide information like firstname, surname, phone, supervisor, institute and why this user wants to register to the VO.

Then we would like to use some tools (provided by EDG) which allow to enter this user in the VO in an easy way and not like today: now, one has to request the public key of the user, then transform this key in a ldif format, then import inside LDAP this info.

A good point would be to have the possibility to inform the grid that my VO status has changed and the gridmapfile should be re-created.

A user with a valid certificate released by a trusted Certification Authority works at a given institute. The institute responsible will guarantee to the CMS Collaboration that the user belongs to CMS, thus allowing the user to be accepted as a member of the CMS VO. In order to be added to a particular group (i.e "PRS/mu","MC Production",...) the responsible of the group should be contacted. The management of a group can be delegated.

It’s not clear how the information flow goes through the different players in the use case: a diagram would probably better clarify it

4.2. USE CASES AND PRIORITIES FOR VO GROUPS

4.2.1. VO Management use cases:

- add/remove users to/from the VO
- associate/remove groups and roles (I guess more than one should be allowed) to a user
- create a new group/subgroup
- remove group/subgroup
- delegate management of a group
- assign resources and permissions to a group
- modify the group information and privileges
- query ALL services about capabilities relevant to the querying VO

A requirement common to all the previous operations is that it must be possible to perform and propagate them quickly and reliably.

4.2.2. VO users/group use cases:

- reading and deleting files restricted to people with a particular group/role (high priority)
- restricting the ability to modify metadata associated to data written by a particular group (high priority)
- restrict access to Mass Storage to people with a particular role (high priority)
- restricting the ability to publish information about written data? (how?)
- support for users belonging to multiple groups
- control access on resources based on groups
- reserving cpu or disk storage.
- the experiment policy can be expected to change on an almost daily basis. VOMS must react to such policy changes in less than a day.
- query ALL services about capabilities relevant to the querying group/user

4.2.3. VO group/roles Information use cases:

- Retrieve information about the list of permissions related to a given group/role in order to know what a user belonging to that group is allowed to do.

5. SOME ADDITIONAL QUESTIONS

- On its own VOMS does nothing for you, the proxies just behave like the existing ones unless the system using them can interpret the VO information. At the moment there is nothing to do that, but some things would be fairly easy to add. The easiest is probably to control access to a gatekeeper, e.g. to restrict a CE to particular subgroups. There are also ideas about attaching Access Control Lists (ACLs) to files (or collections, in that case a hierarchic structure in the Data/MetaData Catalogue would help a lot), so that e.g. reading or deleting could be restricted to people with a particular role.

On the other hand there doesn't seem to be much of a concept for enforcing quotas, either on cpu or disk storage, which to me seems likely to be more important for HEP use. It isn't obvious to me how people would expect that to work even in a general way, e.g. would you have a quota across the entire grid or a separate quota for each site? Some of the ideas for Tier-2 centres seem to envisage higher priority or more quotas for "local" users, does that accord with what the experiments want? (it should go through the definition of a group to which local users belong. That functionality should be there, the way it will be used will depend on the VO(s) policy/ies)

- There is also the question of control on the information system. We suspect HEP doesn't care much about reading, but we probably want to be able to restrict the ability to publish information - but at what granularity?
- One question which came up in the WP8 meeting is whether we need support for multiple VOs. VOMS allows you to embed information from multiple VOs in one proxy, but the current software assumes that there is a unique VO, e.g. you can only have input files belonging to one VO in a given job. Is that adequate? (ALICE think the management of different VOs should be separated. However, we should be aware that the same user might belong to more than a VO)
- Another issue is management of the system. Managing the VOMS server (adding/removing users and changing groups and roles) should be fairly straightforward. However, managing the permissions on the resources seems less obvious, we suspect you don't want to have to edit individual ACLs on millions of files, probably not even programmatically (see comment on a hierarchical Catalogue). Management use cases (e.g. create a new subgroup and assign resources and permissions to it) are important. There's also a question of how much interaction is required with local system managers, who generally want to have the right to override the VO decision if necessary.

  Related issues:

  - Transfer of ACLs (from user to group or vice versa)
  - Should ACLs be role dependent?