VOMS in DPM

Jean-Philippe Baud, IT-GD, CERN

October 2006

Agenda

- Main ideas
- Virtual Ids
- VOMS/LCAS/LCMAPS integration
- Access Control Lists

Main ideas

- All the DPM components are deployed in secure mode only (GSI or Kerberos 5)
- Pool accounts and local Unix accounts are not used for files managed by the DPM
- Virtual UIDs/GIDs are used instead

Virtual ids in non VOMS world

- DNs are mapped to virtual UIDs: the virtual uid is created on the fly the first time the system receives a request for this DN (no pool account)
- VO names are obtained using the DN and a special grid-mapfile
- VO names are mapped to virtual GIDs which are also automatically created when seen for the first time
- The mapping table (in the catalogue DB) can also be used for reverse mapping (display of ownership)
- Administrative tools are offered to update the DB mapping table

VOMS/LCAS/LCMAPS integration

- DNs are mapped to virtual UIDs: the virtual uid is created on the fly the first time the system receives a request for this DN (no pool account)
- VO names and VOMS roles are mapped to virtual GIDs
- A given user may have one DN (one UID) and several groups/roles (GIDs)
- Only the primary (first) group/role is currently used
- Integration with CSEC (for socket interface) and CGSI (for Web services like SRM)
- Use of LCAS: black listing (not in place yet)
- There is no plan to use LCMAPS

Access Control Lists

- DPM offers Posix Access Control Lists
- Access ACLs can be set on directories and files
- Default ACLs can be set on directories and propagated to sub-directories and files
- ACLs are based on these virtual UIDs/GIDs
- These ACLs can be set using either the proprietary socket interface (C API, CLI, Python) or the SRM v2 interface