gLExec and MyProxy integration in PANDA

Jose Caballero

on behalf the BNL Computing Team

OSG Annual Users Meeting

BNL, Jun 2008

The pilot jobs model

Pilot jobs can probe the environment on the remote worker node before pulling down the payload job from the server and executing it. Such a design allows for improved logging and monitoring capabilities, and the pilot becomes a ″smart wrapper″ for the payload job.

Two modes of operation:

- Single user mode: the production users submit production jobs which run using their own proxy.
- Multiuser mode: pilots are submitted under the credentials of a privileged VO member and execute jobs for one or more different users.

  - How to determine which user was responsible for each job?
  - Should users be able to use proxies they do not own?

Possible solution: gLExec

gLExec

gLExec can log the user credentials when the pilot starts to run the job.

gLExec can operate also in setuid mode: the uid is changed to the user’s who runs the job. gLExec performs this uid/gid change:

- It is a Grid version of suexec program.
- It runs as setuid process on the CE.
- It performs switch based on results from LCAS/LCMAPS mapping.

This require having the privilege to run as root. Many site admins do not like to allow users to run as root. However, in the setuid mode the security behavior is the same one as for normal submission. When the setuid mode is not used, all users jobs are run under the same uid, having access to each other processes and files (including the proxies).

gLExec can be configured to restrict the number of users allowed to invoke it.

Job flow

Flow of a grid job through the Grid Computing Service and the LRMS for the pilot jobs scenario

Pilot jobs/gLExec need access to the user credentials

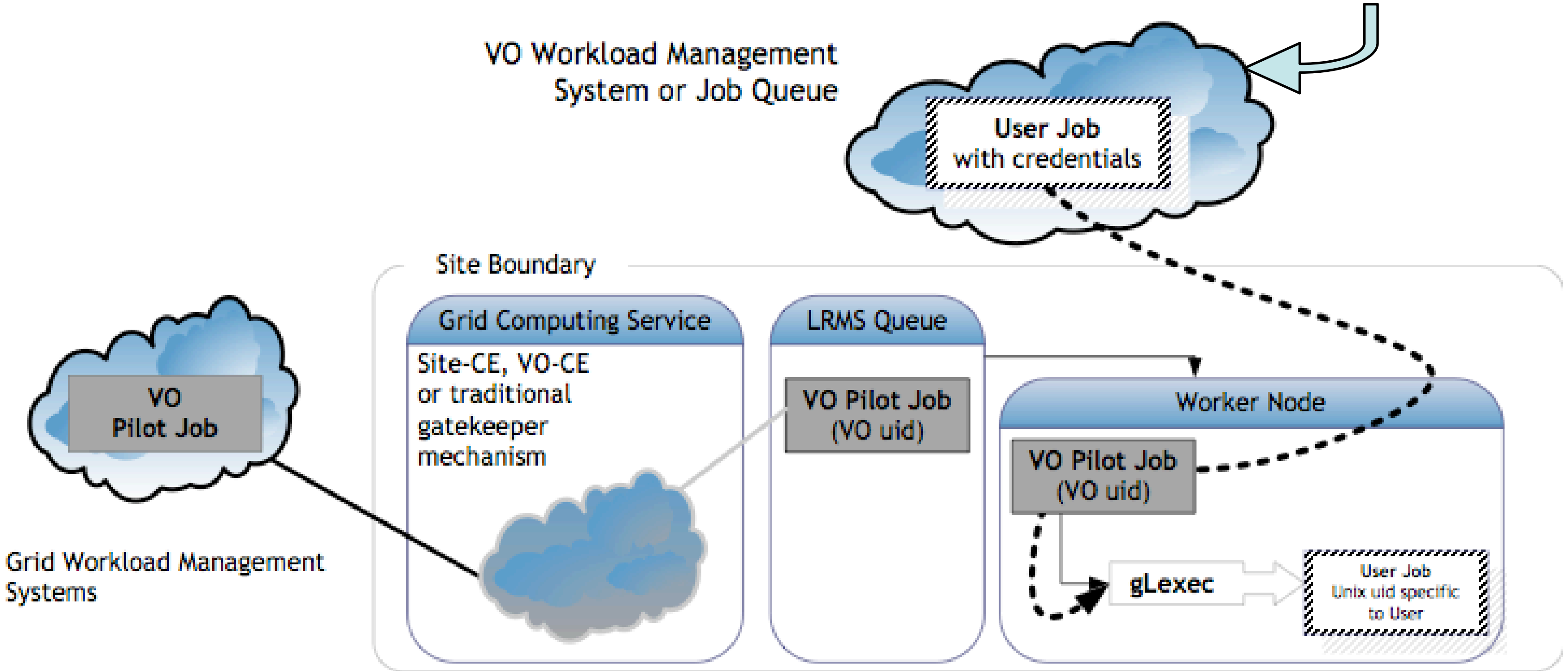
VO Workload Management System or Job Queue
User Job with credentials
Site Boundary
VO Pilot Job
Grid Workload Management Systems
Grid Computing Service
Site-CE, VO-CE or traditional gatekeeper mechanism
LRMS Queue
VO Pilot Job (VO uid)
Worker Node
VO Pilot Job (VO uid)
gLexec
User Job Unix uid specific to User

Image courtesy of the gLExec group

LCAS & LCMAPS

LCAS (Local Centre Authorization Service) is the authorization engine that adds access control to the gatekeeper.

- Framework of independent authorization pluggable modules. The final decision is the logical “AND” of all the invoked modules, depending of the specific policy:
  - Valid and banned-users lists inspection module.
  - Wall-time limiting module.
  - Comparison against a site-local access control list: VOMS module.
    The VO, the group, and the role are examined.
- The final decisions are based on the requested resource (RSL), requestor identity, and authorization credentials in the proxy certificate.

LCMAPS (Local Credential MAPping Service) maps Grid users to local uid/gid Unix accounts.

- Framework of independent pluggable modules:
  - Mapping to local Unix accounts and groups module.
  - Statistics module.
  - Mapping to local pool accounts module.
  - VOMS support module: VOMS groups and roles are mapped to local accounts
    ➡ e.g. the pilot jobs owner.
  - Mapping between DN and local AFS and Kerberos tokens module.
  - ...

External parties may add their own modules.

Payload jobs under final user identity

Jobs are run by a NEW user:

- ✤ The execution moves to a new $HOME directory:
  - ➢ Files are not in the new working area.
  - ➢ Does the new user have permissions to read/write/execute in the original working area? Usually not:
    - ✓ Copy all files to a different location before invoking gLExec. Slow process. Two copy operations: before and after execution. Second copy has to be removed after execution for security.
    - ✓ Change the Unix permissions to the pilot working dir. Security risk?
- ✤ Does the new user have permissions to read/write the input/output files from/to the SE?
- ✤ The pilot environment vanishes. In particular, the LD_LIBRARY_PATH disappears.

  Permitting user control over dynamically linked libraries would be disastrous for setuid/setgid programs if special measures weren't taken. Therefore, in the GNU loader (which loads the rest of the program on program start-up), if the program is setuid or setgid these variables (and other similar variables) are ignored or greatly limited in what they can do.

  http://tldp.org/HOWTO/Program-Library-HOWTO/shared-libraries.html

- ➢ Modify the /etc/ld.so.conf file (not a good idea ?)

Integration in PANDA

✓ A new VOMS role and Unix account have been created. Users with "role=pilot" are mapped to the new "atlasplt" pool account. Only atlasplt has the privilege to invoke gLExec in setuid mode and run jobs under a new uid. By default, $OSG_GRID/glexec_osg/etc/glexec.conf has a setting like

```
"user_white_list = *"
```

To modify /etc/ld.so.conf is not a good idea to recover the pilot environment. It is architecture depending and root privileges are needed.

✓ Intermediate wrapper to recreate the vanished environment. A wrapper script recreates the environment and runs the payload job. The pilot invokes this wrapper script.

✓ Users credentials are pre-allocated in a MyProxy credential caching service.

✓ Now BNL has a running server: pandaprx.usatlas.bnl.gov

```
$ myproxy-init -s pandaprx.usatlas.bnl.gov -x -Z /DC=org/DC=doegrids/OU=People/CN=Pilot Owner 123456 -d --voms atlas
```

Myproxy server

Allows the specified entity to retrieve credentials w/o password

Uses the certificate subject (DN) as username Instead of $LOGNAME env var

VOMS attributes are needed to invoke gLExec

❖ Pilot jobs retrieve the right user proxy before invoking gLExec.

```
$ myproxy-logon -s pandaprx.usatlas.bnl.gov --no_passphrase  -l  UserName --out /tmp/x509_new
```

User logname.
It should be his own DN

Retrieved proxy path

Integration in Panda (cont’d)

✓ pathena (the interface between athena and Panda) has been instrumented to perform the proxy delegation. A lifetime check has been incorporated to avoid that the user is requested for the password each time a new job is created. The existence of a valid proxy in the server is verified, and its timeleft is checked. Only when there is no proxy, or it is short time, a new one is created and delegated.

```
$ myproxy-info -s pandaprx.usatlas.bnl.gov -l UserName
username: /DC=org/DC=doegrids/OU=People/CN=Jose Caballero 511275
owner: /DC=org/DC=doegrids/OU=People/CN=Jose Caballero 511275
trusted retrieval policy: /DC=org/DC=doegrids/OU=People/CN=Jose Caballero 511275
timeleft: 167:57:52  (7.0 days)
```

✓ gLExec usage will be a site attribute, rather than a user/job attribute, included in PANDA configuration DB.

✓ Who is in charge of the automatic periodic renewal of the user proxies? The job submission systems (Condor, WMS/RB in gLite…) only can manage one proxy: the pilot one

- ✓ A daemon running in the background checks the timeleft of the retrieved proxy and renews it when it is close to expire.
- ✓ Maybe a future version of Condor can manage automatically this issue?

✓ Security check: verification that the DN and the logname mach. Otherwise, a malicious user could delegate a proxy using a different logname from his and runs jobs that he should not.

Integration in Panda (cont’ d)

✓ These new issues bring new problems. A new error code for easy diagnosis and troubleshooting has been incorporated to the current error list. The new error codes are propagated until the DB also.

2100	MyProxyError: server name not specified
2101	MyProxyError: voms attributes not specified
2102	MyProxyError: user DN not specified
2103	MyProxyError: pilot owner DN not specified
2104	MyProxyError: invalid path for the delegated proxy
2105	MyProxyError: invalid pilot proxy path
2106	MyProxyError: no path to delegated proxy specified

gLExec only works fine when the proxy is in the local machine, e.g. the /tmp directory

2200	MyProxyError: myproxy-init not available in PATH
2201	MyProxyError: myproxy-logon not available in PATH
2202	MyProxyError: myproxy-init version not valid
2203	MyProxyError: myproxy-logon version not valid

Not all versions support delegation without password

2300	MyProxyError: proxy delegation failed
2301	MyProxyError: proxy retrieval failed

2400	MyProxyError: security violation. Logname and DN do not match

Security checks

References

Panda: https://twiki.cern.ch/twiki/bin/view/Atlas/Panda

MyProxy: http://grid.ncsa.uiuc.edu/myproxy/

gLExec: http://www.nikhef.nl/grid/lcaslcmaps/glexec/

LCAS/LCMPAS: http://www.nikhef.nl/grid/lcaslcmaps/

Backup

gLExec installation

BNL's Linux Farm has worker nodes which are kept very static: it is not possible to roll out RPMs/updates on a whim. And our OSG client software installation resides in NFS, for easy management and to avoid having to alter worker nodes. So having gLExec installed in NFS was a necessity. However, as a security application, gLExec rightly takes several steps to restrict usage. Most important is that the gLExec executables may not reside on an NFS-mounted partition. They must be local. These constraints led us to use a mixed arrangement requiring some customization.

✓ On the Worker Nodes: symbolic link to NFS for the configuration directory. This will allow global config to be altered without touching each Worker Node:

```
/etc/glexec -> /usatlas/OSG/osg_wn_client/current/glexec-osg/etc
```

Places local copies of executables on each WN, which include the version string. That way multiple executables can be available in case a rollback is necessary.

✓ On the NFS OSG installation dir: symbolic links from NFS OSG installation to local files:

```
/usatlas/OSG/osg_wn_client/X.Y.Z/glxec-osg/sbin/
   glexec -> /usr/libexec/glexec-X.Y.Z
   glexec_fork -> /usr/libexec/glexec_fork-X.Y.Z.
```

This approach is very appealing because:

i. It keeps everything under $VDT_LOCATION.

ii. It does not require coordinating both and NFS-located pacman install and a local pacman install.

iii. It does not require even a medium-weight gLExec RPM install.

iv. It allows for smoother transitions between OSG versions usable on a Worker Node.

gLExec installation (cont’d): GUMS

Any site using gLExec needs to configure GUMS to allow the worker nodes to make mapping calls.

In a sense they are now acting as gatekeepers. So a host-to-group mapping needs to be established for all worker nodes (acas∗.usatlas.bnl.gov in our case).

The contents of this mapping should be identical to that for the Globus gatekeeper, i.e. a user should be mapped on a worker node to precisely the same account as they would have been on the gatekeeper.

Nothing is stopping a gLite site from downloading GUMS from the DVT, in which case it is easy to install. GUMS is usually installed on a separate machine, so there would be no port conflict issues:

1. Creating a GUMS configuration template for gLite sites.
2. Need to add CAs for CRLs for VOMS servers within the GUMS configuration.
3. GUMS caches names from VOMS servers, which may be a privacy concern. If generic grid proxies were not used, this would not be necessary.