Glexec, LCAS, LCMAPS:
Status update and progress

Oscar Koeroo

Gerben Venekamp

index

- **Glexec**
  - What is it? What does it do? Why does it do what it does?
  - Usage of glexec
    - All the ins and outs
- **Status & Todo**

What is glexec?

glexec

a thin layer
to change unix credentials
based on grid identity and attribute information

you can think of it as:

- **‘a replacement for the gatekeeper’**
- **‘a *griddy* version of Apache’s `suexec(8)`’**
- **‘a program wrapper around LCAS, LCMAPS or GUMS’**

What glexec does

Input

1. a certificate chain, possibly with VOMS extensions
2. a user program name & arguments to run

Action

1. check authorization (LCAS, SAZ)
   - user credentials, proper VOMS attributes, executable name
2. acquire local credentials (LCMAPS, GUMS)
   - local (uid, gid) pair, possibly across a cluster
3. enforce the local credential on the process

Result

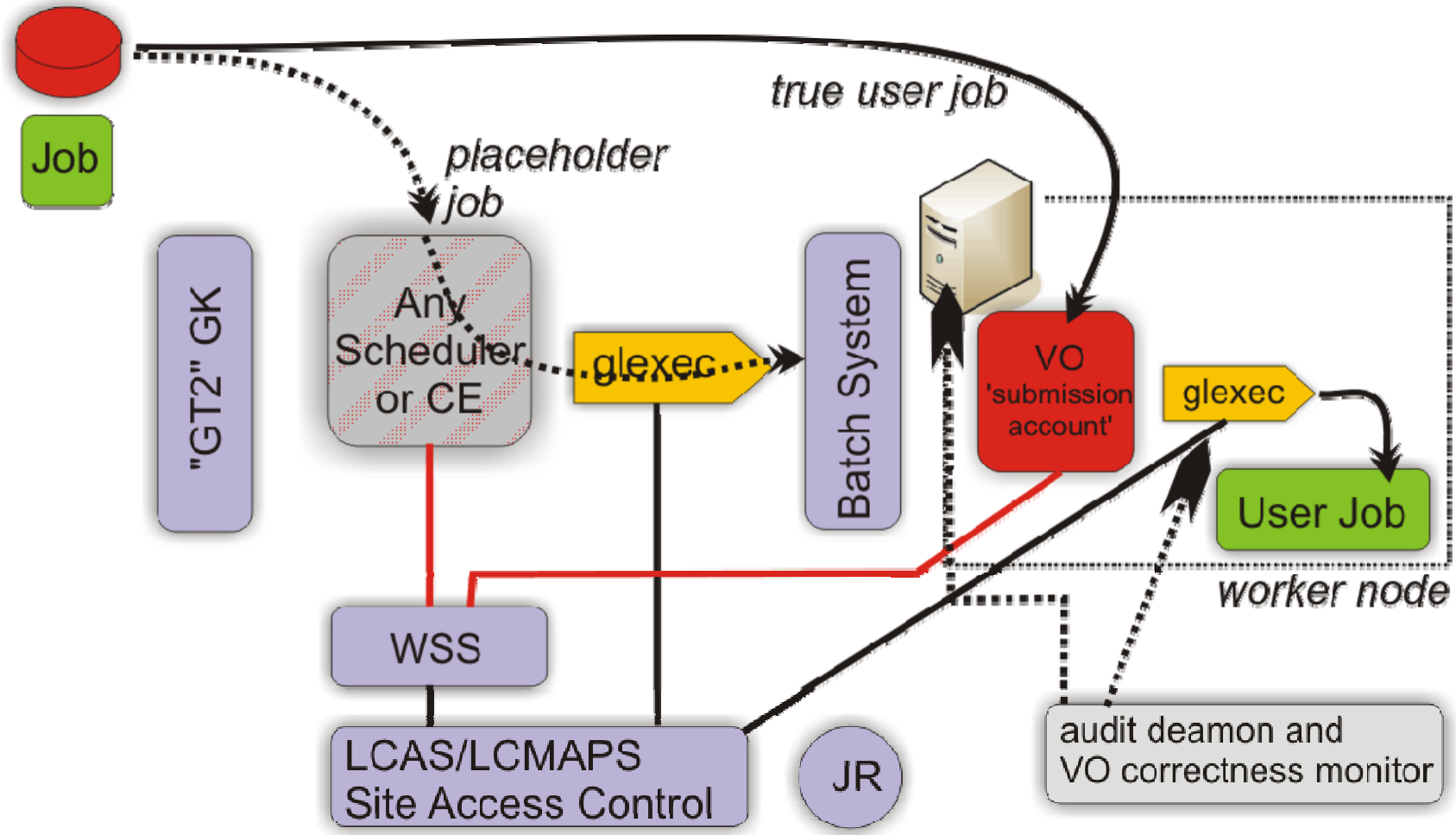
1. user program is run with the mapped credentials

Why was glexec devised?

- **gatekeeper and other schedulers are complex, and need not be run with root privileges all the time**
  - take an example from Apache httpd, where user *cgi* scripts can be run under their own identity, but without the web server itself having to run as root
  - to accomplish this, a small, program is needed with setuid(2) power: ‘suexec(8)’
- **variety in grid job submission systems is increasing**
  - need a common way of obtaining and enforcing site policy and credential mapping
  - without the need to modify each and every system
  - as such, glexec in this deployment mode is an alternative to having authorization and mapping *call-out*s in each system

VO pilot job on the node

- **On success: the site will set the uid/gid of the new user's job**
- **On failure: glexec will return with an error, and pilot job can terminate or obtain other job**



Note: proper uid change by Gatekeeper or Condor-C/BLAHP on head node **should remain default**

Glexec as *suexec* wrapper

Thin layer with root privileges will replace gatekeeper

- **Intended for identity-switching services:**
  - *condor, gridsite, globus gram, cream.*
- **Internal**
  - Uses LCAS as authorization mechanism
  - Uses LCMAPS as credential mapping mechanism
  - Executes the requested command with local credentials
- **External interface**
  - Should be usable by C, java (, perl?) services: program executable
  - A (user) credential should be passed to ***glexec***.
  - In- and outgoing pipes, file descriptors should be preserved as much as possible.

Usage of *glexec* (1)

Two run-modes depending on setuid bit settings:

1. *glexec* **is setuid-root: setuid()/setgid() to local user in** *glexec* **code and execute the program**

```
-r-s--x--- 1 root apache /usr/sbin/glexec
```

In the next release:

1. *glexec* **runs as special user:** *glexec* **uses** ***sudo*** **for identity switching and program execution:**

```
-r-s--x--- 1 glexec glexec /opt/glite/sbin/glexec
```

   - ***sudo*** preserves ***only*** stdin, stdout, stderr
   - ***sudo*** can be configured to allow the user "**glexec**" to run a predefined set of programs (**blahp, qsub**)

Usage of *glexec* (2)

Configuration:

- **glexec**
  - Hardcoded glexec configuration file path
    - `/opt/glite/etc/glexec.conf`
  - Contains:
    - locations of lcas and lcmaps config files
    - locations of lcas and lcmaps log files
    - loglevels for LCAS and LCMAPS
- **lcas and lcmaps use the following (default) config files:**
  - **`/opt/glite/etc/lcas/lcas-glexec.db`**
  - **`/opt/glite/etc/lcmaps/lcmaps-glexec.db`**

Usage of *glexec* (3)

- **Environment variables to be set before calling glexec**
  - GLEXEC_MODE:
    - "lcmaps_verify_account": glexec <uid> <gid> <command+args>
    - "lcmaps_get_account": glexec <command+args>
  - SSL_CLIENT_CERT and SSL_CLIENT_CERT_<n>: PEM-encoded strings containing the proxy cert and chain components. ***Deprecated***.
  - GLEXEC_CLIENT_CERT: location of the (PEM-encoded) proxy cert and chain components.
    - In addition to the SSL_CLIENT_CERT
  - GLEXEC_SOURCE_PROXY: location of the (delegated) proxy to be used by the user job..
  - GLEXEC_TARGET_PROXY: location where the proxy should be copied to. If not specified ~/.glexec/proxy is used.
  - GLEXEC_ID (optional): unique job id to be used as an index for the jobrepository:

- **All other environment variables are cleared**
  - Unless you've allowed specific environment variables to be preserved in the glexec.conf file

- **$HOME is set to the mapped user's $HOME (according to the system) within the execution of glexec**

Usage of *glexec* (4)

- **White listing the env-vars**
- **White listing the users that may execute glexec**

```
#
#  Glexec
#
[glexec]
silent_logging               = no
log_level                    = 5
user_white_list              = glexec*, venekamp, root
preserve_env_variables       =

lcmaps_db_file               = /home/venekamp/EGEE/glexec-tst/lcmaps-glexec.db
lcmaps_log_file              = /home/venekamp/EGEE/glexec-tst/lcas_lcmaps.log
lcmaps_debug_level           = 5
lcmaps_get_account_policy    = glexec_get_account
lcmaps_verify_account_policy = glexec_verify_account

lcas_db_file                 = /home/venekamp/EGEE/glexec-tst/lcas-glexec.db
lcas_log_file                = /home/venekamp/EGEE/glexec-tst/lcas_lcmaps.log
lcas_debug_level             = 1
```

suid-less glexec

- **Site admins can choose not to set the set-uid bit.**
  - glexec will adapt its functionality
    - logging works
    - user banlist works
    - certificate chain checks work
    - mapping is disabled verification only
      - *needs root privileges*
  - real user job gets run with **pilot job identity**
    - The discrimination will only be expressed in the log files

Status & Todo

Status today…

- **Getting glexec up to speed on a CE and on the WN scenario's**
  - Being tested by the CREAM CE dev team
  - glexec-on-WNs scenario is being tested by Fermilab's CDF VO
    - They have a strong requirement to enable setuid bit
  - ***Just*** got hold of sufficient privileges to install glexec-on-WNs in the Preview Testbed environment
    - No progress good have been made on the PTB

- ***Note! We'd like to encourage the development of LCAS or LCMAPS plugins (like the GPbox plugin) outside of our scope***
  - *Yes, we can still help and assist (where needed…)*

- **Restrictions of glexec**
  - policy should be located on local posix-accessible file systems
  - policy transport should be 'trustworthy'

- **The GT4 Mapping interface to LCMAPS is being created**
- **The GT4 AuthZ interface to LCAS will follow after that**

Our Todo List

- **Needed specifically for the –on-WN model**
  - make the credential acquisition process (LCAS/LCMAPS) work with a *site-central policy engine*
    - Look like GUMS but is powered by LCAS and LCMAPS
    - One site centrally LCAS and LCMAPS instance to support all installed instances of LCAS and LCMAPS enabled services throughout a site
    - enforcement will have to stay local
  - Non-setuid bit enabled execution of glexec needs to be tested in more various deployment configurations
- **Use of sudo to perform the setuid magic**
- **The lcmaps_verify-proxy will also gain the possibility to perform VOMS credential TTL checking**
  - Still not implemented because of various interrupts, but will be finished soon
- **Logging improvements**
  - Move to syslog
  - This logging rework is slipstreamed with the "*Draft MW security logging guidelines*"

?