EUROPEAN MIDDLEWARE INITIATIVE

Security aspects of the CREAM-CE

Cristina Aiftimiei

On behalf of the gLite job management product team

CREAM

- CREAM service: Computing Resource Execution And Management service
- Service for job management operations at the Computing Element (CE) level
  - Allows to submit, cancel, monitor, … jobs
- Web service interface
- Possibility to interact directly with the CE or via a higher level service (WMS, Condor)
- Implemented and maintained so far in the context of the EGEE projects
  - Work is continuing now under the EMI hat

CREAM security mechanisms (now)

- Authentication managed by trustmanager
- Authorization to the service managed by a “custom” gJAF (Grid Java Authorization Framework)
  - Possibility to grant authorization based on VOMS attributes or DNs
- When authorized, glexec used to get the local user id mapped to that Grid user
  - glexec uses LCAS and LCMAPS
- Sudo (using the local user id) is then used to perform the needed operations on behalf of that user (CREAM sandbox dir creation, submission to the batch system, etc.)
- Gridftpd also used in the CREAM CE
  - It also uses LCAS and LCMAPS, but specific configuration files are needed (i.e. not the same conf files used for glexec)

CREAM security mechanisms (future)

- CREAM CE is being integrated with Argus
  - Integration foreseen in CREAM CE 1.7 and ARGUS 1.2
- Scenario in CREAM-CE integrated with Argus:
  - Authentication still managed by trustmanager
  - Authorization to the service managed by ARGUS
    - Argus, besides telling if the operation for that user is authorized, will also return the local user mapped to that Grid user
    - Sudo (using the local user id) still used to perform the needed operations (CREAM sandbox dir creation, submission to the batch system, etc.)
  - Gridftpd also integrated with Argus
- At configuration time, admin will have the possibility to choose between Argus based authorization and the old mechanisms

The following slides refer to the implementation now in production (i.e. not to the CREAM CE integrated with Argus)

Installation and Configuration

- Supported installation method is yum
  - As all the other gLite services
- Supported configuration method is yaim
  - As all the other gLite services
  - Something different being adopted in EMI ?
- Some sites are using quattor for installation and/or configuration
- Once the CREAM-CE is installed and configured
  - You can check some relevant part of the configuration running a script
    - http://grid.pd.infn.it/cream/field.php?n=Main.CheckYourCREAMCEConfiguration
    - Script updated whenever needed (i.e. typical configuration problems are checked by this script)
  - You can also perform the tests suggested at http://grid.pd.infn.it/cream/field.php?n=Main.HowToCheckAndTestYourCREAMCE

Configuration files

- Configured by yaim, but afterwards they can be manually further modified if needed (but these changes will be reset if case of reconfiguration)
- CREAM configuration file
  - /opt/glite/etc/glite-ce-cream/cream-config.xml
- BLAH configuration file
  - /opt/glite/etc/blah.config
- BLparser configuration file (only for the old blparser)
  - /opt/glite/etc/blparser.conf
- glexec configuration file
  - /opt/glite/etc/glexec.conf
- sudo configuration file
  - /etc/sudoers
- LCAS and LCMAPS configuration files used by glexec
  - /opt/glite/etc/lcas/lcas-glexec.db and /opt/glite/etc/lcmaps/lcmaps-glexec.db
- LCAS and LCMAPS configuration files used by gridftpd
  - /opt/glite/etc/lcas/lcas.db and /opt/glite/etc/lcmaps/lcmaps.db

All details in: http://grid.pd.infn.it/cream/field.php?n=Main.ConfigurationFiles

Log files

- CREAM log file
  - /opt/glite/var/log/glite-ce-cream.log
- Tomcat and trustmanager log files
  - /usr/share/tomcat5/logs/*
- Glexec logs on syslog by default, but possibility to configure it to log on specific files
- Sudo by default logs via syslog, but possibility to configure it to log on specific file
- BLparser log files
  - For the old parser: /opt/glite/var/log/glite-[lrms]parser.log
  - For the new parser: /opt/glite/var/log/glite-ce-bupdater.log and /opt/glite/var/log/glite-ce-bnotifier.log
- Gridftpd log files: /var/log/globus-gridftp.log and /var/log/gridftp-session.log

All details in: http://grid.pd.infn.it/cream/field.php?n=Main.RelevantLogFiles

User mapping

- For pool accounts, mapping between Grid users and local users can be found in /etc/grid-security/gridmapdir
- It contains a file per each pool account
- The file for an unused account has a hard link count of 1

```
# cd /etc/grid-security/gridmapdir

# ls -li ops034

2467593 -rw-r--r-- 1 root root 0 Dec 15 2006 ops034
```

- The file for a used account has a hard link count of 2

```
# ls -li ops022

2467581 -rw-r--r-- 2 root root 0 Sep 14 03:57 ops022
```

- The other link’s name encodes the DN and VOMS UNIX groups

```
# ls -li | grep ^2467581

2467581 -rw-r--r-- 2 root root 0 Sep 14 03:57 %2fdc%3dch%2fdc%3dcern%2fcn%3darthur
%20dent:ops

2467581 -rw-r--r-- 2 root root 0 Sep 14 03:57 ops022
```

User mapping (cont.ed)

- If glexec and gridftpd maps you to different local accounts, there is a bug or misconfiguration !
  - This is one of the issue addressed with the integration with Argus
- lcg-expiregridmapdir cron job recycles pool accounts as needed
  - When usage exceeds a threshould (80 %)
  - The oldest accounts are recycled until the usage falls again below the thresholds
  - An account can only be recycled if idle for a certain period (default: 48 hours)
  - Recycling of pool accounts should happen rarely
  - Check the status in /var/log/lcg-expiregridmapdir.log

Network usage

	From node	From port	To node	To port	Variable
CREAM service	{UI, WMS}	C	CREAM-CE	8443	
Gridftp control	{UI, WMS, WN}	C	CREAM-CE	2811	
Gridftp data	{UI, WMS, WN}	C	CREAM-CE	C	
Notifications	{WN, BLParser host}	C	CREAM-CE	9091	LRMS_EVENT_LISTENER_PORT
LB locallogger	WN	C	CREAM-CE	9002	
LB locallogger	CREAM-CE	C	LB server	9001	
mysql	CREAM-CE	*	Mysql server	3306	
bdii	BDII	*	CREAM-CE	2170	
Old BLParser listening port	CREAM-CE	*	Blparser host	33333	BLP_PORT
Old blparser CREAM listening port	CREAM-CE	*	Blparser host	56565	CREAM_PORT

http://grid.pd.infn.it/cream/field.php?n=Main.PortsUsedInACREAMCE

Start/stop

- Start
  - CREAM service is started starting the container:
    - /etc/init.d/tomcat start
  - In case the new BLAH blparser is used, this will also start it (if not already running)
  - If for some reason it is necessary to explicitly (re)start the new BLAH blparser:
    - /opt/glite/etc/init.d/glite-ce-blahparser (re)start
  - If instead the old BLAH blparser is used, before starting tomcat it is necessary to start it on the blparser node:
    - /opt/glite/etc/init.d/glite-ce-blparser start
- Stop
  - CREAM service is stopped stopping the container:
    - /etc/init.d/tomcat start

Banning

- How to ban a VO
  - Simplest way: reconfigure without that VO
- How to ban a user
  - Insert the DN of the user to be banned in /opt/glite/etc/glite-ce-cream/banned.lst (for CREAM service) and /opt/glite/etc/lcas/ban_users.db (for glexec and gridftpd)
  - DN must be in quotes

CREAM administrators

- An administrator of a CREAM CE is a user with special privileges
  - Can manage (check status, cancel, etc.) jobs submitted to that CREAM CE by other users
  - Can perform some special operations (e.g. disable new job submissions)
- How to define a CREAM administrator
  - Insert the DN of that user (in quotes) in /etc/grid-security/admin-list

How to trace a specific job

- Get the CREAM jobid of the job
- If the job was submitted through the WMS and you have the WMS (Grid) jobid, you can get its CREAM jobdid doing:
  - `glite-wms-job-logging-info -v 2 <gridjobdid> | grep "Dest jobid"` (if the job is yours)
  - `grep <gridjobid> /opt/glite/var/log/glite-ce-cream.log*`
- Grep the "numeric part" of the CREAMjobid in the CREAM log file
  - E.g. https://cream-07.pd.infn.it:8443/CREAM383606450 → CREAM383606450
  - `grep CREAM383606450 /opt/glite/var/log/glite-ce-cream.log*`
- This will return all the information relevant for this job
  - CREAM jobid, WMS jobid, batch system jobid, DN, local user, issued operations, client hostname, status changes, etc.

Some security recommendations

- Close port 9091 (LRMS_EVENT_LISTENER_PORT in the CREAM conf file) to all nodes, expect the WNs and the one running the BLparser
- Use pool accounts instead of static accounts where possible
- Configure enough pool accounts so that recycling occurs rarely
- Releasing accounts should be an exception to the rule and should happen only rarely. However, if this happens, it should be taken care of, that really all jobs of that account are finished

Some security recommendations (cont.ed)

- Ensure that each VO software area is only writable by the software manager for that VO (group writable only for the "sgm" accounts group, not the VO)
- Do not mount the VO software areas on the CREAM CE node, but only on the WNs
- If there are more than one computing element at a site, that access the same worker nodes, it`s recommended to set up individual, unique accounts for all CEs (such as "user001" ... "user100" on CE1 and "user101" ... "user200" on CE2 and so on) or to centrally mount gridmapdir, in order to guarantee, that no user is able to read the data of any other users.

Troubleshooting

- Check if the error message is documented in http://grid.pd.infn.it/cream/field.php?n=Main.ErrorMessagesReportedByCREAMToClient
  - Page updated quite often, whenever needed
  - Common error messages along with their meaning and possible solutions, are documented
- Check if something useful is reported in the CREAM log files
- Open a GGUS ticket
- Contact the developers: cream-support [at] lists [dot] infn [dot] it

Other info

- Check the CREAM web site http://grid.pd.infn.it
- In particular the links in the "Administrator guides" section
  - Cream control mechanisms
  - Daemons running on a CREAM CE
  - Configuration files
  - Ports
  - Cron jobs
  - Troubleshooting
  - …

EUROPEAN MIDDLEWARE INITIATIVE

Thank you!

EMI is partially funded by the European Commission under Grant Agreement RI-261611