Centre for Cosmology, Particle Physics and Phenomenology

UCLouvain

Tier2 Computing Centre UCLouvain

Andres Tanasijczuk

Virgo Computer Centres Workshop

28-29 November 2019

UCLouvain - CP3

UCLouvain

7 sites distributed across
Wallonia and Brussels (Belgium)

Centre for Cosmology, Particle Physics and Phenomenology

THE physics institute at UCLouvain

- 15 professors/researchers
- ~20 postdocs
- ~20 PhD students
- 5 master students
- 3 IT experts

300 m

CP3 cluster at UCLouvain's Data Center III

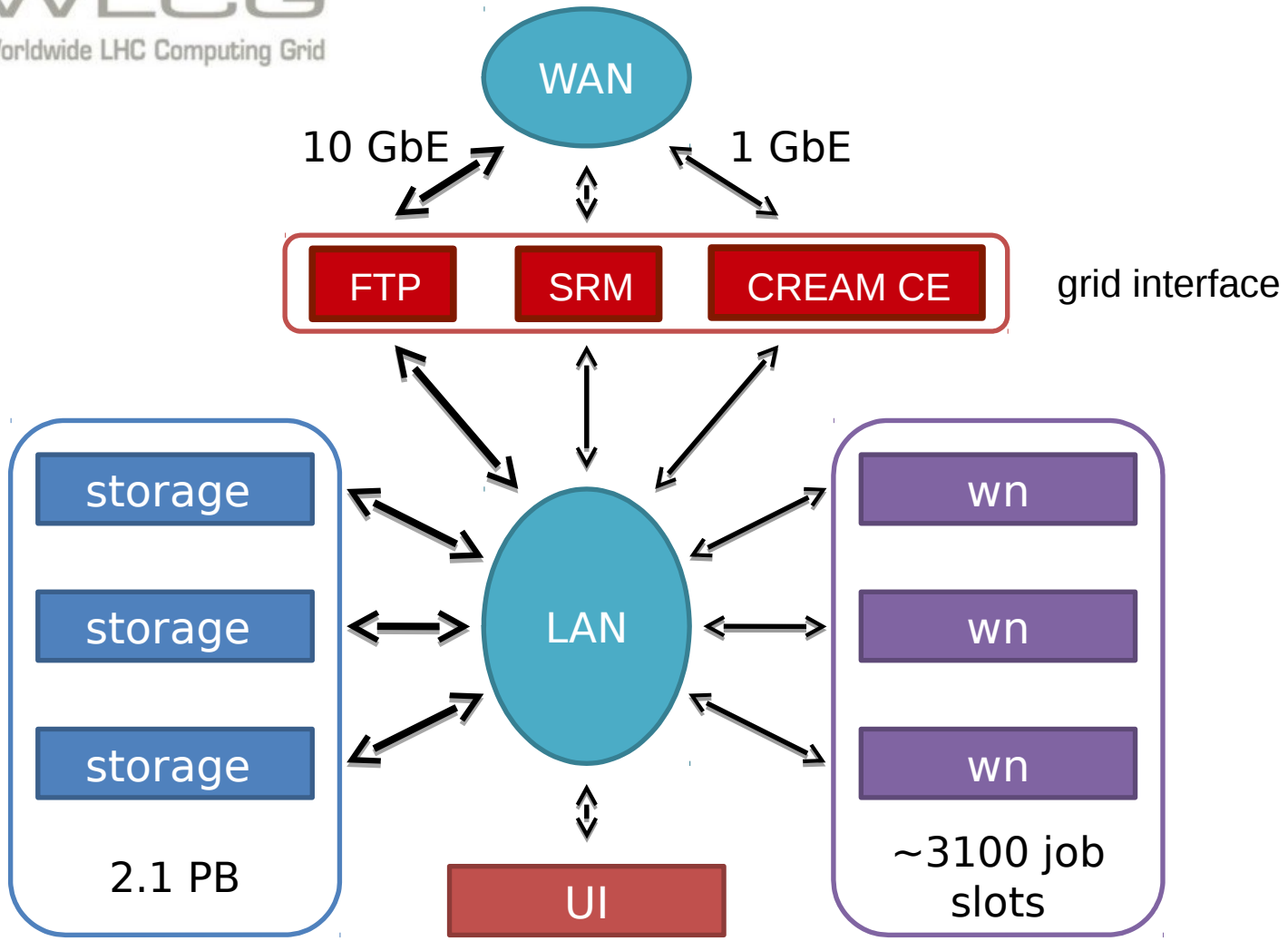
CP3 computing cluster

Administrated by UCLouvain's

Centre for HPC and Mass Storage

- Operating system is CentOS 7
- Slurm batch system
- Software distributed via CVMFS
- Singularity 2.6.1
- No shared filesystem
- User scratch & store areas mounted via NFS
- Storage (RAID6) mounted via NFS (readonly)
- User Interface (frontend) configured exactly same as WNs
- Second User Interface with CentOS 6

CP3 computing cluster – a WLCG Tier2 centre



CP3 computing cluster users

Mainly three groups of local users:

- users working for CMS experiment
- users working for Na62 experiment
- users working in particle physics phenomenology using MadGraph, Pythia, Mathematica

Grid jobs:

- CMS (GlideinWMS)
- NA62 (Dirac ?)
- new LIGO/Virgo via OSG (GlideinWMS)

Grid jobs limited to 2000 simultaneous jobs

Virgo group at UCLouvain

2 big events !!

JULY 2018

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Virgo group at UCLouvain

2 big events !!

JULY 2018

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Belgium's best finish in a FIFA World Cup

Virgo group at UCLouvain

2 big events !!

JULY 2018						
SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Belgium's best finish in a FIFA World Cup

UCLouvain joined Virgo collaboration

Virgo group at UCLouvain

Current members involved in Virgo:

- 2 experimentalists professors (G. Bruno, K. Piotrzkowski, both at 50%)
- 1 IT “specialist” (A. Tanasijczuk)
- 1 PhD (A. Depasse)
- 1 postdoc starting in January 2020
- 2 other postdoc positions still open https://cp3.irmp.ucl.ac.be/jobs

Also associated with Virgo:

- 1 theory professor (Ch. Ringeval) at 10%
- 1 theory postdoc (S. Clesse) at 10%
- 1 theory PhD (C. Joana)

Also closely collaborating with ULiège Virgo team:

- 1 professor (J.R. Cudell)
- 1 postdoc

Enabling LIGO/Virgo grid jobs at UCLouvain

What we had to do…

Issues we had to solve…

Everything in this OSG ticket:

https://support.opensciencegrid.org/public/tickets/47d99dd9e17e070de646488cfd4431a71789ed3c16ef7000d440da4ecb23186c

1) CVMFS configuration

Original configuration

cvmfs.x86_64 *2.6.0-1.el7* *@cernvm*
cvmfs-x509-helper.x86_64 *1.1-1.osg34.el7*
cvmfs-config-default.noarch *1.7-1* *@cernvm*

+ */etc/cvmfs/default.d/70-osg.conf* pointing to configuration repository */cvmfs/config-osg.opensciencegrid.org*
+ */etc/cvmfs/config.d/*.local* configuration files for each repository

↳ ligo.osgstorage.org.local:

```
CVMFS_SERVER_URL="http://cvmfs-s1bnl.opensciencegrid.org:8000/cvmfs/@fqrn@;
                  http://cvmfs-s1fnal.opensciencegrid.org:8000/cvmfs/@fqrn@;
                  http://cvmfs-s1goc.opensciencegrid.org:8000/cvmfs/@fqrn@"
CVMFS_KEYS_DIR=/etc/cvmfs/keys/opensciencegrid.org
CVMFS_USE_GEOAPI=yes
CVMFS_CONFIG_REPO_REQUIRED=yes
CVMFS_FALLBACK_PROXY=http://cvmfsbproxy.cern.ch:3126;http://cvmfsbproxy.fnal.gov:3126
X509_CERT_DIR=/cvmfs/oasis.opensciencegrid.org/mis/certificates
```

Worked with proxy in default location; didn't work with X509_USER_PROXY env variable

New configuration

Use EGI's CVMFS configuration (UMD repo http://repository.egi.eu/sw/production/umd/4/)

cvmfs.x86_64 *2.6.0-1.el7* *@cernvm*
cvmfs-x509-helper.x86_64 *1.1-1.osg34.el7*
cvmfs-config-egi.noarch *2.0-1.el7.centos* *@UMD-updates*

Brings */etc/cvmfs/default.d/60-egi.conf* which points to configuration repository */cvmfs/config-egi.egi.eu*

No more */etc/cvmfs/config.d/ligo.osgstorage.org.local* file

2) CVMFS auth data and the need for `setsid`

CVMFS auth data

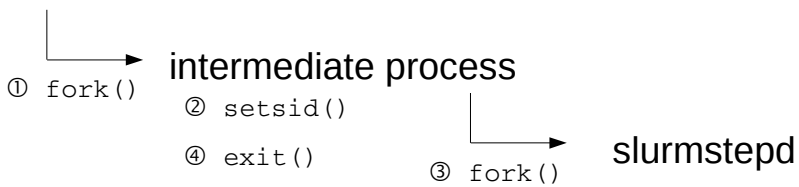
CVMFS limits access to those processes with an acceptable VOMS proxy.

Every process within the same linux session shares the VOMS proxy permissions.

When a process tries to access authenticated data, CVMFS searches information of the process that is the session leader (assumes the session leader process hasn't exited).

How Slurm runs jobs

The *slurmd* daemon running on the worker node spawns a *slurmstepd* daemon:



① fork intermediate process

② make intermediate process a group / session leader

③ fork slurmstepd from intermediate process

④ exit intermediate process

Problem

The intermediate process in Slurm is the session leader and it has exited.

Solution

Run commands that access authenticated data in CVMFS in a new session (`setsid`).

2) CVMFS auth data and the need for `setsid`

How the session-leader-not-existing error can be identified in the logs

CVMFS log:

(authz) Session key not found in cache, getting information from OS [09-17-2019 15:05:21 CEST]
(authz) Failed to open status file /proc/23447/stat: (errno=2) No such file or directory [09-17-2019 15:05:21 CEST]
(authz) Authorization for session 23447 disappeared [09-17-2019 15:05:21 CEST]

/var/log/messages:

cvmfs2: (ligo.osgstorage.org) Authorization for session 17062 disappeared

When there is no error the logs look like this:

CVMFS log:

(authz) Session key not found in cache, getting information from OS [09-30-2019 08:34:24 CEST]
(authz) Lookup key 30532/189721919; sid=30344, bday=189702998 [09-30-2019 08:34:24 CEST]
(authz) starting authz helper /cvmfs/config-egi.egi.eu/libexec/authz/cvmfs_x509_helper [09-30-2019 08:34:24 CEST]

/var/log/messages:

cvmfs2: (ligo.osgstorage.org) starting authz helper /cvmfs/config-osg.opensciencegrid.org/libexec/authz/cvmfs_x509_helper
cvmfs_x509_helper: (ligo.osgstorage.org) Support for Globus authz is enabled.
cvmfs_x509_helper: (ligo.osgstorage.org) Support for VOMS authz is enabled.
cvmfs_x509_helper: (ligo.osgstorage.org) x509 authz helper invoked, connected to cvmfs process 11714

2) Where is `setsid` needed

In OSG's LIGO CVMFS test script

There is a OSG script that checks for LIGO frames (and containers) in CVMFS:
https://github.com/opensciencegrid/osg-ligo-fe/blob/master/scripts/node-check/ligo-cvmfs-storage-check.sh

The script advertises the machine classads HAS_CVMFS_LIGO_STORAGE, HAS_LIGO_FRAMES and HAS_CVMFS_LIGO_CONTAINERS.

Tests done by the script didn't use `setsid` originally.

Changed the script to use `setsid` when Glidein entry (site) name is UCLouvain's Virgo entry.

In HTCondor

2 years old GlideinWMS ticket: https://cdcvs.fnal.gov/redmine/issues/17662

To couple with the CVMFS way of checking for VOMS proxy permissions, each job started by Condor has to run in a new session.

New HTCondor config parameter: USE_PROCESS_GROUPS (default: True)

- condor_starter uses `setsid()` system call before starting a job if USE_PROCESS_GROUPS = True

3) Bug in GlideinWMS Factory

In HTCondor

2 years old GlideinWMS ticket: https://cdcvs.fnal.gov/redmine/issues/17662

To couple with the CVMFS way of checking for VOMS proxy permissions, each job started by Condor has to run in a new session.

New HTCondor config parameter: USE_PROCESS_GROUPS (default: True)

↳ condor_starter uses `setsid()` system call before starting a job if USE_PROCESS_GROUPS = True

GlideinWMS uses Singularity to separate user jobs running under the same pilot.

In GlideinWMS, USE_PROCESS_GROUPS inherits the value of HAS_SINGULARITY machine classadd.

Bug in GlideinWMS Factory code (already fixed in 3.4.6) was causing to advertise HAS_SINGULARITY = False.

Pilots submitted by

production	CERN Production Factory	code OK ✅
integration test-bed	CERN ITB Dev Factory	code buggy ❌

3) Bug in GlideinWMS Factory

In HTCondor

2 years old GlideinWMS ticket: https://cdcvs.fnal.gov/redmine/issues/17662

To couple with the CVMFS way of checking for VOMS proxy permissions, each job started by Condor has to run in a new session.

New HTCondor config parameter: USE_PROCESS_GROUPS (default: True)

- condor_starter uses `setsid()` system call before starting a job if USE_PROCESS_GROUPS = True

GlideinWMS uses Singularity to separate user jobs running under the same pilot.

In GlideinWMS, USE_PROCESS_GROUPS inherits the value of HAS_SINGULARITY machine classadd.

Bug in GlideinWMS Factory code (already fixed in 3.4.6) was causing to advertise HAS_SINGULARITY = False.

Pilots submitted by

production	CERN Production Factory	code OK ✅
integration test-bed	CERN ITB Dev Factory	code OK ✅

updated after we found reported the problem

LIGO/Virgo glideins at UCLouvain

LIGO/Virgo glideins (pilots) running since exactly 2 weeks.

Most efficient configuration (as per Edgar Fajardo): 2 CPUs and 4GB max memory per CPU.

Pilots time limit = 3 days.

Old machines started to fail (will be replaced beginning of 2020).

↳ From the ~3100 job slots only ~2200 are available at this moment.

2000 CPUs pledged to CMS.

No machines bought yet with Virgo-related budget.

⟹ Might need to reduce LIGO/Virgo pilots priority w.r.t. CMS and Na62 pilots.

~50 job slots are used by LIGO/Virgo pilots (BayesWave pipeline -Parameter Estimation-)

Planned

March: invest ~50-100K € from Virgo-related budget in computing nodes (or GPUs?).

↳ At that point will give LIGO/Virgo pilots same priority as CMS and Na62 pilots.

Expect to increase LIGO/Virgo jobs throughput.

That’s all…

Thank you!