Unprivileged CernVM-FS with cvmfsexec

Dave Dykstra, dwd@fnal.gov

CernVM Workshop

1 February 2021

Fermilab

CernVM
File system

SINGULARITY

CVMFS (and singularity) completely unprivileged

- High Throughput Computing (HTC) depends heavily on cvmfs and singularity for performance, a common program environment, and security
- A major impediment for our use of opportunistic resources, especially supercomputer (HPC) resources, is a lack of cvmfs installed by system administrators
- The cvmfsexec package makes it easy to use cvmfs without requiring installation by system administrators
  - 4 different ways to use it
  - designed with HTC pilot systems in mind

4 ways to use cvmfsexec package

1. mountrepo/umountrepo only
   - requires fusermount; mounts in user space
   - map /cvmfs in container with singularity –bind (not run from cvmfs because of path)
   - user must manage umountrepo, which can be a problem if job killed with kill -9
2. cvmfsexec on RHEL 7.6 or 7.7 or OpenSUSE 15
   - requires fusermount and additionally unprivileged user namespaces enabled
   - maps /cvmfs without singularity, can run singularity under it unprivileged
   - unmounts repos automatically on exit, but not with kill -9
3. cvmfsexec on RHEL >= 7.8
   - no fusermount needed, and cleans up mounts even with kill -9
   - still needs unprivileged user namespaces enabled; that is default on RHEL 8
4. singcvmfs on any system with singularity >= 3.4.0
   - drop-in replacement for singularity; set environment variable with cvmfs repos to mount
   - requires container image to already be present (not read from cvmfs)
   - requires setuid-root singularity except when RHEL >= 7.8 and and singularity >= 3.6.0

makedist

- makedist downloads cvmfs software to send to job
  - will create default, osg, or egi cvmfs configuration
- Example with method 2 or 3:

```
$ git clone https://github.com/cvmfs/cvmfsexec
$ cd cvmfsexec
$ makedist osg
$ cvmfsexec grid.cern.ch atlas.cern.ch -- ls /cvmfs
atlas.cern.ch config-osg.opensciencegrid.org grid.cern.ch
```

Self-extracting distribution script

- After running makedist, use makedist -o to make self-extracting script including the cvmfs distribution

```
makedist –o /tmp/cvmfsexec
```

- Send /tmp/cvmfsexec to a job, and when it is executed it will extract the cvmfsexec and cvmfs distribution into a .cvmfsexec subdirectory and run from there

What about squids?

- cvmfs requires local squid cache to work well at scale
- Between makedist and makedist -o you can edit configuration
- Default configuration uses WLCG Web Proxy Auto Discovery (WPAD) servers at CERN & FNAL
  - following WLCG standard, first looks for local http://grid-wpad/wpad.dat or http://wpad/wpad.dat services
  - if those are not found, http://cernvm-wpad.cern.ch/wpad.dat or http://cernvm-wpad.fnal.gov/wpad.dat are consulted
    - if squids are known for the requesting GeoIP organization, they are returned
    - if no squids are known, connects DIRECT to openhtc.io Cloudflare aliases
    - if many requests from same org with no squid within 15 minutes, directs to monitored fallback squids at CERN or FNAL
- frontier-squid can auto-register itself with WLCG WPAD (via shoal)

mountrepo/umountrepo

- Can use mountrepo/umountrepo within cvmfsexec (methods 2 & 3) to add or remove mounted repositories
  - use through $CVMFSMOUNT and $CVMFSUMOUNT
  - they work by sending a message to parent cvmfsexec process
  - I recommend closing the communication file descriptor before running any user payload jobs

```
exec {CVMFSEXEC_CMDFD}>&-
```

- Same mountrepo/umountrepo commands work separate from cvmfsexec, with fusermount and singularity (method 1)

singcvmfs

- Drop-in replacement for singularity exec, shell, run, and version commands
  - ideal for older systems that have setuid singularity, as is the case on many HPCs
  - uses singularity >= 3.4.0 --fusemount option and fuse3 pre-mount feature
- Use makedist -s to create dist, and makedist -s -o to create a self-extracting script (the latter will store files in .singcvmfs)
- Example:

```
$ makedist -s osg
$ makedist -s -o /tmp/singcvmfs
$ cd /tmp
$ export SINGCVMFS_REPOSITORIES="grid.cern.ch,atlas.cern.ch”
$ ./singcvmfs -s exec -cip docker://centos:7 ls /cvmfs
atlas.cern.ch  config-osg.opensciencegrid.org  grid.cern.ch
```

- Also works unprivileged with RHEL >= 7.8 and singularity >= 3.6.0

Cache considerations

- Cache has to be managed carefully with production use
  - by default, mountrepo (cvmfsexec modes 1 to 3) just allocates some space (4GB) under its dist subdirectory, shared between the repositories mounted
  - multiple jobs on the same machine can’t easily share the cache
    - works best if controlled by pilots allocated with as large a portion of a worker node as possible
- Problem gets worse with singcvmfs, because then every invocation on a machine by default starts its own cache manager and needs its own cache
  - could perhaps use shared cache with CVMFS alien cache mode but then something has to manage making sure it doesn’t grow too big and gets cleaned up
- Avoid putting the cache on shared filesystems

File descriptor considerations

- Production use of cvmfs tends to use a lot of file descriptors
  - the default RHEL limit per process of 4096 may be a problem especially if cache is shared on a worker node between a lot of independent jobs
  - the standard cvmfs install increases that limit by default to 8192 and many nodes with lots of CPU cores have to increase it further
  - may need to trade reduced sharing (increasing cache space) in order to stay within limit, or ask system admin for an increase

Production use cases

- CMS is using mountrepo/umountrepo + locally installed singularity (method 1) successfully on Stampede2 at TACC
  - RHEL7 & fusermount but without unprivileged user namespaces
  - using a locally installed script, wrapping the pilot
  - whole-node pilots, so don’t worry about kill -9
  - 200 nodes, almost 20k cores
  - cvmfs cache configured to be on local disk (in /tmp)
  - large number of file descriptors (256k) available per process
- CMS is using cvmfsexec on OpenSUSE 15 (method 2) on Theta at Argonne
  - Have experienced some trouble with leftover mountpoints in /tmp after jobs killed, planning to at least move them to /dev/shm

Final thoughts

- Regular cvmfs installed by system administrators is still best, but cvmfsexec is an alternative in many cases
- Pilot systems are encouraged to use it
- Up next: cvmfsexec's integration into GlideinWMS
- https://github.com/cvmfs/cvmfsexec