Access on opportunistic resources with cvmfsexec

Dave Dykstra, dwd@fnal.gov

pre-GDB – Software deployment

5 May 2020

Fermilab

CernVM File system

SINGULARITY

CVMFS and singularity with minimal host support

- WLCG VOs depend heavily on cvmfs and singularity for performance, a common program environment, and security
  - A major impediment for use of opportunistic 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

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)
2. cvmfsexec on RHEL 7.6 or 7.7
   - requires fusermount and additionally unprivileged user namespaces enabled
   - maps /cvmfs without singularity, can run singularity under it
   - unmounts repos 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:

```
$ git clone https://github.com/cvmfs-contrib/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 to add or remove mounted repositories
  - use through $CVMFSMOUNT and $CVMFSUMOUNT
  - recommend closing the communication file descriptor before running any user payload jobs

    ```
    exec {CVMFSEXEC_CMDFD}>&-
    ```
- can also use mountrepo/umountrepo separate from cvmfsexec, with fusermount and your own singularity

singcvmfs

- drop-in replacement for singularity exec, shell, run, and version commands
  - ideal for older systems that have setuid singularity, such as 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

Production use case

- CMS is using mountrepo/umountrepo + locally installed singularity 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

Final thoughts

- other than at TACC, these haven't been used in production yet, but GlideinWMS is working on using cvmfsexec from pilots
  - I think it likely that after that it will be promoted to HTCondor
- I'm concerned that the default 4096 file descriptors will become a problem for general opportunistic case, especially if sharing mountpoints between jobs
  - likely will need to pay attention to where cache is stored (default is in dist/var/cache, limit of 4000 MB)
  - a pilot may want to share an unmanaged cache between separate jobs with their own mountpoints by using cvmfs alien cache feature
- https://github.com/cvmfs-contrib/cvmfsexec