Taking the C out of CVMFS

Samuel Skipsey[1], Thomas Doherty[1], A G D Turner[2]

[1] Department of Physics and Astronomy, University of Glasgow, G12 8QQ
[2] University of Leeds, East Building, School of Geography, Leeds, LS2 9JT

E-mail: Samuel.Skipsey@glasgow.ac.uk

Abstract. The Cern Virtual Machine File System is most well known as a distribution mechanism for the WLCG VOs' experiment software; as a result, almost all the existing expertise is in installing clients mount the central Cern repositories. We report the results of an initial experiment in using the cvmfs server packages to provide Glasgow-based repository aimed at software provisioning for small UK-local VOs. In general, although the documentation is sparse, server configuration is reasonably easy, with some experimentation. We discuss the advantages of local CVMFS repositories for sites, with some examples from our test VOs, vo.optics.ac.uk and neiss.org.uk.

1. Introduction

The Cern Virtual Machine File System (CVMFS)[1] was originally developed as an underlying filesystem for the Cern Virtual Machine project. However, it has become vastly more successful as the current medium for software distribution on the grid, used by ATLAS[2] and LHCb, and shortly by CMS. The properties that make CVMFS useful as a filesystem for LHC software distribution also lend themselves to distribution of common software in general. In order to do this, however, the software in question must be hosted in a properly configured CVMFS repository. While there is plentiful documentation on the configuration of CVMFS *clients*, provided by the developers[3], regions[4] and the experiments [5], information on the server-side is much sparser, being limited to the final section of one chapter of the technical report[6]. This paper discusses the experience of installing and configuring a central CVMFS server for small local VOs at the UKI-SCOTGRID-GLASGOW site. For information on CVMFS itself, we direct the reader to Jakob Blomer's talk in this proceedings.

2. Use case and Virtual Organisations

For the purposes of our test installation, we chose two smaller VOs supported locally by UKI-SCOTGRID-GLASGOW (and a small number of other UK sites). The first VO, vo.optics.ac.uk, exists to support the pool of UK academics making common use of the Lumerical software package for microoptical simulation. Software distribution for this VO consists mainly of the Lumerical software installation itself, but also the distribution of licence files providing individual sessions with the site's installed copy. Licence file distribution clearly cannot be solved by CVMFS, as licences are necessarily local to the individual site.

The second VO is neiss.org.uk, the virtual organisation for the National E-Infrastructure for Social Simulation. NEISS manages several projects, but the application supported on the grid is a population demographics simulation called GENESIS. Traditionally, the project has shipped the java bytecode for the simulation with the jobs themselves, but the payload is close to the size of the maximum input sandbox, and may become larger with time. As with vo.optics.ac.uk, there is very little free effort to manage structured software distribution, which has been a factor in hampering the growth of the VO.

3. Server Configuration

The CVMFS server packages are mutually incompatible with the CVMFS client, and are currently dependant on the (unmaintained) redirfs[10] kernel module. This latter dependency will go away in the next major release, but at present it prevents installation of the CVMFS server on RHEL6/SL6 systems, as the module is incompatible with newer kernels. It is possible to recompile the redirfs source against the stock RHEL6 kernel, but issues with the cvmfs modules which interface with it prevent the solution from working. We installed our CVMFS server on a 64-bit SL5 virtual machine, hosted on a VMware instance. The host being a member of a pool of virtual machines maintained at Glasgow has the semi-anonymous hostname of `vm004.gla.scotgrid.ac.uk`. As the documentation is sparse on CVMFS server configuration, we will take some time here to go over the installation process for the benefit of others.

The CVMFS server packages are mutually incompatible with the CVMFS client, but are hosted in the same repository, available at `http://cvmrepo.web.cern.ch/cvmrepo/yum/`. With the CVMFS repositories set up, the initial installation is managed by yum

```
yum install cvmfs-server
```

which installs cvmfs-server, and should also pull in the redirfs and cvmfsflt packages as a dependancy. Configuration of the server begins by creating the working repository (each cvmfs server can only manage a single repository for release).

```
cvmfs_server mkfs local.gla.scotgrid.ac.uk
```

Two directory structures are created in the process of making the repository filesystem:

```
/srv/cvmfs/local.gla.scotgrid.ac.uk/
```

which contains the ”release” repository, as a tree of files arranged by file hash, and various other support files which represent the repository state and configuration, and

```
/cvmfs/local.gla.scotgrid.ac.uk
```

which is a virtual filesystem managed by the cvmfsflt module, via redirfs. Writes to this filesystem are redirected and used to build a set of diffs against the current repository when updates are committed. Essentially, this the inverse process to that performed by the clients in creating the mounted fs of the same path from the published repository.

This process also generates the signing credentials for the repository, which are placed in `/etc/cvmfs/keys`.

The public part of the credential needs to be distributed to all clients wishing to mount the repository.

The process for installing (or updating) the contents of the repository is straightforward. Simply install the precise directory structure required for your software into the `/cvmfs/local.gla.scotgrid.ac.uk` filesystem. In our case, as we wished to support multiple VOs, but had only one test server, we installed the software side-by-side in the same repository. For a more robust solution, multiple repository servers collated by a single publishing web server or top tier squid cache would be superior.

When the distribution is updated to your satisfaction, the final step is

```
cvmfs_server publish
```

The publish command commits the changes made to the /cvmfs/ directory structure since the last publication. On a new server, there will be nothing available to clients until the first publish invocation pushes the initial configuration into the underlying filesystem. Errors are not always explicitly reported by the publishing tool, but can be inferred by a failure of the published packages to update. In general, we found that permissions on the configured shadow filesystem at `/srv/cvmfs/`were an initial cause of failures; making sure that the entire directory structure is owned by the cvmfs user and group prevents this.

4. Client Configuration

In order to mount the new repository on clients, a few changes are necessary to the client configuration. All of the following changes are described relative to the cvmfs configuration root `/etc/cvmfs`. Firstly, the repository must be added to the `$CVMFS_REPOSITORIES` environment variable by suitable configuration in `default.local`. eg.

```
CVMFS_REPOSITORIES=atlas .cern.ch , atlascondb.cern.ch, lhcb.cern.ch,
cms.cern.ch, geant4.cern.ch, sft.cern.ch, local.gla.scotgrid.ac.uk
```

Secondly, and most significantly, a new file is required in the `domain.d/` directory to configure the domain that the repository is based in. This will probably be the only new domain configuration file in a "standard" CVMFS client install, as the standard repositories are all hosted in the default cern.ch domain. The file must be named for the repository, and configure the `$CVMFS_SERVER_URL` to point at the http endpoint for the repository, and `$CVMFS_PUBLIC_KEY`pointing at the public key for the repository.

For example, in our case:

```
cat /etc/cvmfs/domain.d/local.gla.scotgrid.ac.uk.conf
CVMFS_SERVER_URL=${CERNVM_SERVER_URL:=
http://vm004.gla.scotgrid.ac.uk/cvmfs/local.gla.scotgrid.ac.uk/}
CVMFS_PUBLIC_KEY=/etc/cvmfs/keys/local.gla.scotgrid.ac.uk.pub
```

Finally, as mentioned before, the relevant public key must be made available to the clients, installed in the `/etc/cvmfs/keys` directory on each.

5. Testing

Limited time and free resources at the VOs in question prevented a large scale test of the implementation for either entity. However, some representative tests were performed to validate the functionality of the software on the new mount points, and the use cases considered.

5.1. neiss.org.uk

For the NEISS VO, a number of standard jobs were run against worker nodes mounting the CVMFS copy of the Java libraries comprising the GENESIS software. The jobs completed successfully, in a time similar to that required for the "standard" data staging mode of operation, within margin of error.

5.2. vo.optics.ac.uk

A simple test job utilising the Lumerical software was run at both the Glasgow and Durham grid sites, using the distribution of Lumerical hosted in the CVMFS filespace. For the purpose of this test, the licence keys were also distributed via CVMFS; in a production system, another mechanism would be needed to provide individual copies.

All jobs, albeit short, were successful. The test node at the Durham site was subsequently deconfigured to prevent the licenses from being used by others.

6. Conclusion and Comments

Although the documentation is sparse, configuring a cvmfs server is not particularly arduous, with some experimentation. We have demonstrated and tested the use of the standard cvmfs server package in software provisioning for two small VOs, across two sites. What remains to be demonstrated is a secure scheme for enabling VO members to update their own software releases on the repository provided for them. The authors consider that this problem is soluble by existing standard policies for allowing remote access and authentication, however (in the worst case, gsissh and VOBOX-like functionality on the repository servers).

The use of proprietary software requiring licence files or other files that are site specific is problematic with CVMFS. Arguably, this is not a use-case for which the software framework was intended, and we do not consider it's inability to address the problem to be a deficiency. Nonetheless, we cannot recommend VOs depending on proprietary software to use CVMFS as their distribution platform.

References

[1] P Buncic et al 2010 *J. Phys.: Conf. Ser.* **219** 042003

[2] De Salvo, A, et al Software installation and condition data distribution via CernVM FileSystem in ATLAS *J. Phys.: Conf. Ser.* THIS PROCEEDINGS CHEP 2012

[3] http://cernvm.cern.ch/portal/cvmfs/examples

[4] https://www.gridpp.ac.uk/wiki/UK_CVMFS_Deployment

[5] https://twiki.cern.ch/twiki/bin/view/Atlas/CernVMFS

[6] https://cernvm.cern.ch/project/trac/downloads/cernvm/cvmfstech-2.0-6.pdf

[7] Lumerical Inc. http://www.lumerical.com

[8] Birkin, M., et al (2010): *Phil. Trans. R. Soc.* A, 368:3797-3812; doi:10.1098/rsta.2010.0145. http://www.neiss.org.uk

[9] Turner, A.G.D. 2010 *Data Driven e-Science: Use Cases and Successful Applications of Distributed Computing Infrastructures* ed Simon C. Lin and Eric Yen (to be published, Springer) chapter 1

[10] http://www.redirfs.org/tiki-index.php