



CVMFS server is operating in **TEST** mode. You are encouraged to use it and provide feedbacks, comments and requests.

CVMFS client for LUIS cluster software repository

The article describes how to setup the CVMFS([CERN Virtual Machine File System](#)) client for LUIS HPC cluster software repository. CVMFS is a distributed *read-only* POSIX file system in user space(FUSE) based on HTTP protocol. Files and file meta-data are cached and downloaded on demand. CVMFS was originally developed for the LHC experiments at CERN to deploy software on a globally distributed computing infrastructure, replacing the software package management at local sites.

Please note that when installing CVMFS, only the software already available on the LUIS HPC cluster is provided. Service requests outside the HPC Cluster to install new software and/or update existing software will not be accepted at this time.

CVMFS requirements

To access the cluster software stack using the CVMFS client, on an individual system(laptop, desktop or a single compute node) you will need:

- Outbound HTTP access (direct or via your local proxy server) to the Leibniz University network
- Linux operating system with a Kernel 2.6.32 or newer
- x86-64 compatible CPU supporting at least one of AVX, AVX2 or AVX512 instruction sets
- Approximately 50 GB of available local storage, for the cache

If many CVMFS clients are deployed, for example in a cluster, it is recommended that in addition you deploy forward caching HTTP proxy servers (such as Squid).

Install CVMFS client, autofs and Lmod

Execute commands below as system(root) user

RedHat/CentOS

```
[root@server ~]# yum install  
https://ecsft.cern.ch/dist/cvmfs/cvmfs-release/cvmfs-release-latest.noarch.r  
pm  
[root@server ~]# yum install  
https://docs.cluster.uni-hannover.de/files/cvmfs-config-luiscluster-1.0-1.no
```

arch.rpm

Fedora

Download and install the CVMFS client RPM for your Fedora release from

<https://cernvm.cern.ch/portal/filesystem/downloads>

```
[root@server ~]# yum install
https://docs.cluster.uni-hannover.de/files/cvmfs-config-luiscluster-1.0-1.no
arch.rpm
```

Debian/Ubuntu (newer than 16.04)

```
[root@server ~]# cd /tmp
[root@server ~]# wget
https://ecsft.cern.ch/dist/cvmfs/cvmfs-release/cvmfs-release-latest_all.deb
[root@server ~]# dpkg -i cvmfs-release-latest_all.deb
[root@server ~]# apt update
[root@server ~]# wget
https://docs.cluster.uni-hannover.de/files/cvmfs-config-luiscluster_1.0-1_al
l.deb
[root@server ~]# apt install -f ./cvmfs-config-luiscluster_1.0-1_all.deb
```

Note concerning AutoFS: make sure the autofs config file /etc/auto.master contains the option +dir:/etc/auto.master.d

CVMFS repository configuration

Execute commands below as system(root) user.

Apply the initial client setup:

```
[root@server ~]# cvmfs_config setup
```

Create the file /etc/cvmfs/default.local and add the lines below:

```
CVMFS_REPOSITORIES="cvmfs-config.cluster.uni-hannover.de,sw.cluster.uni-
hannover.de"
CVMFS_CACHE_BASE=/var/lib/cvmfs
CVMFS_QUOTA_LIMIT=5000
CVMFS_HTTP_PROXY=http://your_proxy_server:port
```

- CVMFS_REPOSITORIES is a comma-separated list of the repositories to use (**order is important**).
- CVMFS_QUOTA_LIMIT is the amount of local cache space under CVMFS_CACHE_BASE in MB for CVMFS to use. Set it to about 15% less than the size of your local cache filesystem.
- If you have many CVMFS clients at your site, it is recommended that you deploy forward caching HTTP proxy servers (such as Squid) and specify them with CVMFS_HTTP_PROXY. See [the documentation](#) about this parameter.

Validate the configuration:

```
[root@server ]# systemctl restart autofs
[root@server ]# cvmfs_config chksetup
```

Make sure to address any warnings or errors that are reported

Test your setup

Execute commands below as normal user.

Check that the repositories are OK:

```
[user@server ]$ cvmfs_config probe
```

In case of problems, this [debugging guide](#) may help.

To list available software, first activate LUIS cluster software environment in your session

```
[user@server ]$ source /sw-eb/apps/activate.sh
```

.. and then execute the command:

```
[user@server ]$ module avail

----- Core software: compilers, toolchains -----
Eigen/3.3.7                                iccifort/2020.1.217
(D)
GCC/8.3.0                                ifort/2018.3.222-GCC-7.3.0-2.30
GCC/9.3.0                                (D) ifort/2019.3.199-GCC-8.3.0-2.32
(D)
foss/2019b                                (D) iimpi/2018b
foss/2020a                                iimpi/2019b
(D)
gomp/2019b                                iimpi/2020a
gomp/2020a                                (D) intel/2018b
icc/2018.3.222-GCC-7.3.0-2.30             intel/2019b
(D)
icc/2019.3.199-GCC-8.3.0-2.32             (D) intel/2020a
iccifort/2018.3.222-GCC-7.3.0-2.30        lmod
iccifort/2019.5.281                       settarg

----- AIS: Architecture Independent Software -----

ABAQUS/2019-hotfix-1939                   Gaussian/g09.E01
Mathematica/12.1.1
ABAQUS/2020                               (D) Gaussian/g16.B01           (D)
Miniconda2/4.7.10
ANSYS/2019.3                             Gurobi/9.0.1
```

VTune/2019_update3			
ANSYS/2020.2	(D)	Inspector/2019_update3	ipp/2019.3.199
ANSYSEM/20.2		Java/1.8.0_92	itac/2019.3.032
Advisor/2019_update3		MATLAB/2019b	tbb/2019.2.187
COMSOL/5.5		MATLAB/2020b	(D)
GaussView/6.0.16		Maple/2020.1	

From:

<https://docs.cluster.uni-hannover.de/> - **Cluster Docs**

Permanent link:

https://docs.cluster.uni-hannover.de/doku.php/resources/accessing_cvmfs

Last update: **2021/09/07 09:34**

