



Published on *ROOT* (<http://root.cern.ch/drupal>)

[Home](#) > [Printer-friendly PDF](#) > Printer-friendly PDF

Installing xrootd

- [3.1.0](#) [1]
- [cmake](#) [2]
- [install](#) [3]
- [script](#) [4]
- [setxrd](#) [5]
- [xrootd](#) [6]

Xrootd source tarballs and binaries for some platforms can be downloaded from the [xrootd web site](#) [7]. Version 3.1.0 (or newer) can be built and installed using the dedicated script `installXrootd.sh` distributed with ROOT under `build/unix`. Note that, starting with version 3.1.0, xrootd support has been restricted to unix platforms only, i.e. Windows support has been dropped.

The following of this section is structured in:

- [The \\$ROOTSYS/build/unix/installXrootd.sh script](#)
 - [Additional configuration switches](#)
- [Configuring ROOT with an external XROOTD](#)
- [Setting the environment to run with an external XROOTD](#)
- [Installing the RPMs from the XROOTD official side](#)

Using the \$ROOTSYS/build/unix/installXrootd.sh script

The `build/unix/installXrootd.sh` script understands many switches but the only one which is mandatory is the installation directory. This directory must be writable by the user running the script. If you chose to install under a system directory, e.g. `/opt/xrootd`, you have two options:

- run the script under 'sudo'

```
sudo build/unix/installXrootd.sh /opt/xrootd
```

(or acquire privileges with 'sudo su');

- change the ownership of the chosen directory according to your identity. For example, to install under `/opt/xrootd` make sure to run the following before running the script:

```
sudo mkdir -p /opt/xrootd
sudo chown $USER /opt/xrootd
build/unix/installXrootd.sh /opt/xrootd
```

Additional configuration switches

The `installXrootd.sh` script allows to a few other inputs relevant to the build process. The only mandatory argument is the directory where to install the binaries, libraries, man pages and utility stuff. This *installdir* is not prefixed by any switch. The full list of additional, optional switches can be obtained with '-h' or '--help':

Script to **install** a given version of Xrootd/Scalla

Syntax:

```
./installXrootd.sh <installdir> [-h|--help] [-d|--debug] [-o|--optimized]
                        [-v <version>|--version=<version>]
                        [-t <tarball>|--tarball=<tarball>]
                        [-b <where-to-build>|--builddir=<where-to-build>]
                        [--xrdopts="<opts-to-xrootd>"]
```

where

```
<installdir>: the directory where the bin, lib, include/xrootd, share and
              man directories will appear
              (see also --vers-subdir)
-b <where-to-build>, --builddir=<where-to-build>
  directory where to build; default /tmp/xrootd-<version>
-d,--debug    build in debug mode (no optimization)
-h,--help     print this help screen
-o,--optimized build in optimized mode without any debug symbol
-t <tarball>, --tarball=<tarball>
  full local path to source tarball
-v <version>, --version=<version>
  version in the form x.j.w[-hash-or-tag] ;
  current default 3.1.0
--xrdopts=<opts-to-xrootd>
  additional configuration options to xrootd
  (see xrootd web site)
--vers-subdir [=<version-root>]
  install in <installdir>/<version-root><version> instead of
  <installdir> directly; helps separating different versions
  under the same <root-installdir>; if <version-root> is not
  specified, 'xrootd-' is used.
```

When relevant, the script uses 'wget' ('curl' on MacOSX) to retrieve the tarball

The switches have the following meaning:

- -v, --version=*version-string*
Define the version to be built; this is used to determine the name of the tarball to be downloaded if not specified otherwise;
- -t, --tarball=*tarball-path*
Define the tarball to be built; it can be the path to a local file or the URL to download;

- -d, --debug
Build xrootd in debug mode;
- -b, --builddir=*build-directory*
Directory where to configure and compile; default is /tmp/xrootd-;
- -h, --help
Print the help screen;
- --xrdopts="*xrd-opt1 xrd-opt2 ...*"
List of additional options to be passed to 'cmake' when configuring the built.
- --vers-subdir=*version-root*
Install in *installdir/version-rootversion-string* instead of *installdir* directly; helps separating different versions under the same *root-installdir*; if *version-root* is not specified, 'xrootd-' is used, i.e. by passing '--vers-subdir' the installation will be in *installdir/xrootd-version-string* .

Configuring ROOT with an external XROOTD

Once the script finished successfully you have to tell ROOT where to find xrootd, both for the configure/ make steps and for running. To build against the chosen xrootd you have three options:

- Use the '--with-xrootd' switch in ROOT configure:

```
./configure [architecture] --with-xrootd=<xrootd-install-dir> <other-configure-options>
```

- If the xrootd is installed in standard paths (/opt/xrootd, /usr/local, /usr) or if the environment variable XRDSYS points to the installation directory, then you can just run

```
./configure [architecture] <other-configure-options>
```

- Use the '--with-xrootd-incdir=' and '--with-xrootd-libdir=' switches in ROOT configure when the chosen installation paths are such that they cannot be derived from a common one in a simple way (this may be the case of experiment specific installations):

```
./configure [architecture] --with-xrootd-incdir=<dir-with-XrdVersion.hh> --with-xrootd-libdir=<dir-with-libXrdClient> <other-configure-options>
```

The include directory to be given is the one containing XrdVersion.hh ; the library directory is the one containing libXrdClient (the libraries needed to build depend on the version and are checked by the configure script).

Setting the environment to run with an external XROOTD

To run the paths pointing to the xrootd binaries and libraries need to be available in PATH and LD_LIBRARY_PATH (and DYLD_LIBRARY_PATH on MacOSX). To help doing this two scripts have been made available under \$ROOTSYS/bin, setxrd.sh and setxrd.csh for bash and csh shells, respectively. The scripts are to be sourced with argument the installation directory:

```
source bin/setxrd.sh <xrootd-installation-dir>
```

Note that these scripts set also the variable XRDSYS to point to the installation directory. Therefore, if sourced before running configure (e.g. just after running installXrootd.sh), then configure should pickup the wanted xrootd automatically, as mentioned above.

Installing the RPMs from the XROOTD official side

The XROOTD team provides binaries under RPM form for the SLC5 and SLC6 platforms. The relevant repo files to be used with YUM are available from the [download page](#) [8]. In order to build correctly the ROOT components requiring XROOTD, i.e. netx, proofx and proofd, installation of the following RPMs are required:

```
xrootd-client-devel
xrootd-client
xrootd-libs-devel
xrootd-libs
xrootd-server
```

© 1995-2015 The ROOT Team

Source URL: <http://root.cern.ch/drupal/content/installing-xrootd>

Links:

- [1] <http://root.cern.ch/drupal/category/package-context/310>
- [2] <http://root.cern.ch/drupal/category/package-context/cmake>
- [3] <http://root.cern.ch/drupal/category/package-context/install>
- [4] <http://root.cern.ch/drupal/category/package-context/script>
- [5] <http://root.cern.ch/drupal/category/package-context/setxrd>
- [6] <http://root.cern.ch/drupal/category/package-context/xrootd>
- [7] <http://xrootd.slac.stanford.edu/>
- [8] <http://xrootd.slac.stanford.edu/dload.html>