



Centreon SSH Connector Documentation

Release 1.1.2

Centreon

November 24, 2015

Centreon SSH Connector is a free software from Centreon available under the Apache Software License version 2 (ASL 2.0). It speeds up execution checks over SSH when used along Centreon Engine.

Contents:

Release notes

1.1 Centreon SSH Connector 1.0

1.1.1 What's new

First release

Centreon SSH Connector allow you to manage and recycle ssh connection, to improve performance.

Installation

Centreon recommends using its official packages from the Centreon Enterprise Server (CES) repository. Most of Centreon's endorsed software are available as RPM packages.

Alternatively, you can build and install your own version of this software by following the *Using sources*.

2.1 Using packages

Centreon provides RPM for its products through Centreon Enterprise Server (CES). Open source products are freely available from our repository.

These packages have been successfully tested with CentOS 5 and RedHat 5.

2.1.1 Prerequisites

In order to use RPM from the CES repository, you have to install the appropriate repo file. Run the following command as privileged user

```
$ wget http://yum.centreon.com/standard/2.2/ces-standard.repo -O /etc/yum.repos.d/ces-standard.repo
```

The repo file is now installed. Don't forget to cleanup

```
$ yum clean all
```

2.1.2 Install

Run the following commands as privileged user

```
$ yum install centreon-connector-ssh
```

All dependencies are automatically installed from Centreon repositories.

2.2 Using sources

To build Centreon SSH Connector, you will need the following external dependencies:

- a C++ compilation environment.
- CMake (**>= 2.8**), a cross-platform build system.

- Centreon Clib, The centreon Core library.
- ssh2 library to use ssh functions.
- gcrypt library to secure connections.

This program is compatible only with Unix-like platforms (Linux, FreeBSD, Solaris, ...).

2.2.1 Prerequisites

CentOS

In CentOS you need to add manually cmake. After that you can install binary packages. Either use the Package Manager or the yum tool to install them. You should check packages version when necessary.

Package required to build:

Software	Package Name	Description
C++ compilation environment	gcc gcc-c++ make	Mandatory tools to compile.
CMake (>= 2.8)	cmake	Read the build script and prepare sources for compilation.
Centreon Clib (>= 1.0)	centreon-clib-devel	Core library used by Centreon Connector.
ssh2 library	libssh2-devel	SSH library.
gcrypt library	libgcrypt-devel	Gcrypt library.

1. Install basic compilation tools

```
$ yum install gcc gcc-c++ make libssh2-devel libgcrypt-devel
```

2. Install Centreon repository

You need to install Centreon Enterprise Server (CES) repos file as explained *Prerequisites* to use some specific package version.

3. Install cmake

```
$ yum install cmake
```

4. Install Centreon Clib

See the Centreon Clib [documentation](#).

Debian/Ubuntu

In recent Debian/Ubuntu versions, necessary software is available as binary packages from distribution repositories. Either use the Package Manager or the apt-get tool to install them. You should check packages version when necessary.

Package required to build:

Software	Package Name	Description
C++ compilation environment	build-essential	Mandatory tools to compile.
CMake (>= 2.8)	cmake	Read the build script and prepare sources for compilation.
Centreon Clib	centreon-clib-dev	Core library used by Centreon Connector.
ssh2 library	libssh2-1-dev	SSH library.
gcrypt library	libgcrypt11-dev	Gcrypt library.

1. Install compilation tools

```
$ apt-get install build-essential cmake libssh2-1-dev libgcrypt11-dev
```

2. Install Centreon Clib

See the Centreon Clib [documentation](#).

OpenSUSE

In recent OpenSUSE versions, necessary software is available as binary packages from OpenSUSE repositories. Either use the Package Manager or the zypper tool to install them. You should check packages version when necessary.

Package required to build:

Software	Package Name	Description
C++ compilation environment	gcc gcc-c++ make	Mandatory tools to compile.
CMake (>= 2.8)	cmake	Read the build script and prepare sources for compilation.
Centreon Clib	centreon-clib-devel	Core library used by Centreon Connector.
ssh2 library	libssh2-devel	SSH library.
gcrypt library	libgcrypt-devel	Gcrypt library.

1. Install compilation tools

```
$ zypper install gcc gcc-c++ make cmake libssh2-devel libgcrypt-devel
```

2. Install Centreon Clib

See the Centreon Clib [documentation](#).

2.2.2 Build

Get sources

Centreon SSH Connector can be checked out from GitHub at <https://github.com/centreon/centreon-connectors>. The SSH connector sources reside in the ssh subdirectory. On a Linux box with git installed this is just a matter of

```
$ git clone https://github.com/centreon/centreon-connectors
```

Or You can get the latest Centreon Connector's sources from its [download website](#). Once downloaded, extract it

```
$ tar xzf centreon-connector.tar.gz
```

Configuration

At the root of the project directory you'll find a ssh/build directory which holds build scripts. Generate the Makefile by running the following command

```
$ cd /path_to_centreon_connector/ssh/build
```

Your Centreon SSH Connector can be tweaked to your particular needs using CMake's variable system. Variables can be set like this

```
$ cmake -D<variable1>=<value1> [-D<variable2>=<value2>] .
```

Here's the list of variables available and their description:

Variable	Description	Default value
WITH_CENTREON_CLIB_INCLUDE_DIR	Set the include directory path of centreon-clib include.	auto detection
WITH_CENTREON_CLIB_LIBRARIES	Set the centreon-clib library to use.	auto detection
WITH_CENTREON_CLIB_LIBRARY_DIR	Set the centreon-clib library directory (don't use it if you use WITH_CENTREON_CLIB_LIBRARIES)	auto detection
WITH_KNOWN_HOSTS_CHECK	Check or disable Check hosts against user's known_hosts file.	OFF
WITH_LIBCRYPT_INCLUDE_DIR	Set the directory path of libcrypt include.	auto detection
WITH_LIBCRYPT_LIBRARIES	Set the libcrypt library to use.	auto detection
WITH_LIBCRYPT_LIBRARY_DIR	Set the libcrypt library directory (don't use it if you use WITH_LIBCRYPT_LIBRARIES)	auto detection
WITH_LIBSSH2_INCLUDE_DIR	Set the directory path of libssh2 include.	auto detection
WITH_LIBSSH2_LIBRARIES	Set the libssh2 library to use.	auto detection
WITH_LIBSSH2_LIBRARY_DIR	Set the libssh2 library directory (don't use it if you use WITH_LIBSSH2_LIBRARIES)	auto detection
WITH_PREFIX	Base directory for Centreon SSH Connector installation. If other prefixes are expressed as relative paths, they are relative to this path.	/usr/local
WITH_PREFIX_BINARY	Define specific directory for Centreon Connector SSH binary.	\${WITH_PREFIX}/bin
WITH_TESTING	Enable generation of unit tests. They can later be run by typing <i>make test</i> .	OFF

Example

```
$ cmake \
  -DWITH_PREFIX=/usr \
  -DWITH_PREFIX_BINARY=/usr/lib/centreon-connector \
  -DWITH_TESTING=0 .
```

At this step, the software will check for existence and usability of the rerequisites. If one cannot be found, an appropriate error message will be printed. Otherwise an installation summary will be printed.

Note: If you need to change the options you used to compile your software, you might want to remove the *CMake-Cache.txt* file that is in the *build* directory. This will remove cache entries that might have been computed during the last configuration step.

Compilation

Once properly configured, the compilation process is really simple

```
$ make
```

And wait until compilation completes.

2.2.3 Install

Once compiled, the following command must be run as privileged user to finish installation

```
$ make install
```

And wait for its completion.

3.1 Configuration

Centreon SSH Connector itself does not require any configuration. It should only be configured as a connector of Centreon Engine.

To execute SSH check over SSH with Centreon SSH Connector from Centreon Engine, one might configure commands that relates to SSH check (like `check_by_ssh`).

3.1.1 Binary arguments

These arguments are `centreon_connector_ssh` options.

Short name	Long name	Description
-d	-debug	If this flag is specified, print all logs messages.
-h	-help	Print help and exit.
-v	-version	Print software version and exit.

3.1.2 Check arguments

These arguments are checks options (like `check_by_ssh` options).

Short name	Long name	Description
-1	-proto1	This option is not supported.
-2	-proto2	Tell ssh to use Protocol 2.
-4	-use-ipv4	Enable IPv4 connection.
-6	-use-ipv6	Enable IPv6 connection.
-a	-authentication	Authentication password.
-C	-command	Command to execute on the remote machine.
-E	-skip-stderr	Ignore all or first n lines on STDERR.
-f	-fork	This option is not supported.
-h	-help	Not used.
-H	-hostname	Host name, IP Address.
-i	-identity	Identity of an authorized key.
-l	-logname	SSH user name on remote host.
-n	-name	This option is not supported.
-o	-ssh-option	This option is not supported.
-O	-output	This option is not supported.
-p	-port	Port number (default 22).
-q	-quiet	Not used.
-s	-services	This option is not supported.
-S	-skip-stdout	Ignore all or first n lines on STDOUT.
-t	-timeout	Seconds before connection times out (default 10).
-v	-verbose	Not used.
-V	-version	Not used.

Example:

```
define connector{
    connector_name centreon_connector_ssh
    connector_line /usr/bin/centreon-connector/centreon_connector_ssh
}
```

```
define command{
    command_name ssh_check_cpu
    command_line $USER1$/check_by_ssh -H $HOSTADDRESS$ -l $_HOSTUSER$ -a $_HOSTPASSWORD$ -C "$USER1$/ch
    connector centreon_connector_ssh
}
```

```
define command{
    command_name ssh_check_disk
    command_line $USER1$/check_by_ssh -H $HOSTADDRESS$ -l $_HOSTUSER$ -a $_HOSTPASSWORD$ -C "$USER1$/ch
    connector centreon_connector_ssh
}
```

3.2 Technical details

This article describes how Centreon SSH Connector allow much gain on SSH check execution.

One major CPU-intensive and long operation in a SSH environment is the key exchange and verification mechanism. This operation occurs when a SSH session is started between two hosts. After this step all exchange operations are using far less resources.

Centreon SSH Connector take advantage of this fact and maintain semi-permanent connection with hosts to which it had to connect to. This way if multiple checks are performed on the same host, where “check_by_ssh” opens one session for each check, Centreon Connector SSH only opens one session. However this does not limit the number of concurrent checks on a host, as the SSH protocol allows multiple channels to be opened on the same session. Therefore

if multiple checks are run on the same host simultaneously, they are executed concurrently but with separate execution environment.