Table of Contents

File-transfer Account setup

Introduction

NetYCE supports with the introduction of version 7.1.x the file-transfer protocols **SCP**, **SFTP**, **FTP** and **TFTP**. These file-transfers concern primarily the transfers between the NetYCE servers and the network devices but could be applied to other sources if desired.

Where the previous NetYCE releases used primarily TFTP for file transfers, this protocol is now considered outdated, insecure and too slow for modern operations. The FTP protocol is considered a step up in security since it requires authentication, but does not offer encryption. The SCP and SFTP are based on SSH and offer both authentication and encryption.

With the release of version 7.0.x support of the SFTP and FTP protocols was optionally available and was used selectively by some of the vendor modules. Now all four protocols are supported consistently throughout all vendor modules, provided the vendor supports it.

Several requirements apply to the server in order to support the file-transfers in the way the vendor modules expect it. They reflect authentication and security aspects when dealing with incoming transfer requests:

- A dedicated functional (transfer)user with (non-expiry) password
- that has the SAME file permissions as the 'yce:nms' functional user
- and is allowed to access the SCP, SFTP and FTP services ONLY (no shell login).
- The SCP, SFTP and FTP services will be contacted by the devices to connect INTO the NetYCE server to pull or push the files.
- These incoming transfer sessions are initiated by the NetYCE change jobs, never 'live' users. The Userid and password of the functional (transfer)user are hidden from the NetYCE users and not commonly shared.
- SCP, SFTP and FTP sessions must be able to both read and write files. File ownership for 'yce:nms' must be guaranteed.
- The SCP, SFTP and FTP directory trees will be shared with the TFTP service.
- All transfer sessions will be locked within the SCP/SFTP/FTP/TFTP directory tree using `chroot`.

For the server to securely offer these four protocols, several services need to be installed, activated and configured properly. This article describes the steps required to do this.

Definitions

NetYCE will support SCP, SFTP and FTP and TFTP:

SCP - SSH Copy A variant of BSD `rcp` utility that transfers files over SSH session. The SCP protocol has been mostly superseded by the more comprehensive SFTP protocol and some implementations of the “scp” utility actually use SFTP instead.

SFTP - SSH File Transfer Protocol SFTP runs over an SSH session. A file transfer protocol that has nothing in common with original FTP and has been around since late 1990s. SFTP is sometimes called “Secure FTP” which leads to a common confusion with FTPS (which is called “Secure FTP” too). SFTP is very secure because of its encrypted authentication and session encrypting.

FTP - File Transfer Protocol The plain old FTP protocol that has been around since 1970s. Uses unencrypted authentication and sessions.

TFTP - Trivial File Transfer Protocol This protocol uses no authentication, is udp based and only low-level error recovery. Due to its lack of security and its poor performance (only one 512 byte frame in transit) it should have been obsoleted long ago. It has not though. Mostly because ALL network devices support it.

NetYCE will *not* support Secure FTP over SSL:

Secure FTP Plain FTP over TLS/SSL channel. This name is the most confusing, because it is used to refer to either of the two different protocols. Whenever this name is used, it is necessary to specify whether the SSH-based or SSL-based file transfer protocol is meant.

Ycicle user

The dedicated **`ycicle`** user created for SCP, SFTP and FTP access will be a local Linux user without login privileges (MySecureShell and cpsh wrapper) that shares its userid and group id with the **`yce`** user. Its password may not expire and must be made available in the local yce configuration file (`/opt/yce/etc/<server>_yce.conf`).

The SCP, SFTP and FTP configurations will ensure the dedicated `ycicle` user is locked within the `/var/opt/shared/` directory tree using a 'chroot' jail.

The `ycicle` file-transfer user must be a local user and like the `yce` user included in the `/etc/passwd` file. In the `/etc/passwd` file, the `ycicle` user must appear below the `yce` user in order to share its user-id and group-id without conflict.

A sample of the relevant lines from `/etc/passwd`. The shell of the `yce` user must be `/bin/bash`, the shell of the `ycicle` user must be `/bin/MySecureShell` and its home `/var/opt/shared`.

Example of `/etc/passwd` entries using uid=8010 and gid=1000:

```
yce:x:8010:1000:NetYCE functional usr:/home/yce:/bin/bash
ycicle:x:8010:1000:NetYCE file-transfers:/var/opt/shared:/bin/MySecureShell
```

The default password for the `ycicle` user is set at installation time to `ycicle`. It will be retrieved from the local yce configuration file each time it is required. The password will be stored in encrypted format and can be encoded or decoded using the cli tool `/opt/yce/system/api_crypt.sh`:

```
$ ./api_crypt.sh
missing userid
usage: api_crypt.sh (-e|-d) <userid> <password>
  -e   encrypts the password
  -d   decrypts the password

$ ./api_crypt.sh -e ycicle ycicle
encrypting password for 'ycicle':
U2FsdGVkX18v5ZAiL69H/wDSWjXVr8bV
```

```
$ ./api_crypt.sh -d ycicle U2FsdGVkX18v5ZAiL69H/wDSWjXVr8bV
decrypting password for 'ycicle':
ycicle
```

The created encrypted password is then entered in the `/opt/yce/etc/<server>_yce.conf` file:

```
our $sftp_user = "ycicle";
our $sftp_passwd = "U2FsdGVkX18v5ZAiL69H/wDSWjXVr8bV";
```

Firewall settings

Since it is common to position a firewall between the network devices and the NetYCE servers, care should be taken to install the appropriate rules to allow the file transfers using the various protocols.

In all cases the file transfers will be initiated by the devices when contacting the NetYCE servers. Normally a ssh/telnet session is established from the NetYCE server to the device first, then the device cli is used to retrieve the desired files. The file transfer authentication is therefore entered on the device by the NetYCE job session.

The (default) port numbers and protocol(s) the various services use:

Service	proto	port-number(s)
Telnet	tcp	23
SSH	tcp	22
SCP	tcp	22
SFTP	tcp	22
FTP	tcp	20, 21
TFTP	udp	69

Required packages

Two additional Rhel/Centos packages are required to support SCP, SFTP and FTP.

MySecureShell

To enforce a jail for SCP and SFTP sessions to lock the transfers within the /var/opt/shared akin to a 'chroot', the package `mysecureshell` is needed.

RHEL6.x / CentOS6.x: Install the package **`mysecureshell-1.33-1.x86_64`** using `yum` or `rpm`.
RHEL7.x / CentOS7.x: Install the package **`mysecureshell-1.33-1.x86_64`** using `yum` or `rpm`.

The rpm package can be downloaded from Sourceforge

To install using `yum`, a yum repository must be added in the file `/etc/yum.repos.d/mysecureshell.repo`. Add the following as the content of this file:

```
[mysecureshell]
```

```
name=MySecureShell
baseurl=http://mysecureshell.free.fr/repository/index.php/centos/6.4
enabled=1
gpgcheck=0
```

The yum install command then becomes (as root):

```
# yum install mysecureshell
```

This will install the binary `/bin/MysequreShell` that is assigned as the shell for `ycicle`. In combination with the NetYCE wrapper `/opt/yce/cpsh.pl` both the SCP and SFTP sessions are enforced in a chroot jail. The SCP/SFTP services themselves are incorporated in the SSH daemon.

VsFtpd

The FTP protocol requires a separate daemon to offer the service. NetYCE uses the commonly used package “vsftpd” (for “very secure ftp daemon”).

RHEL6.x / CentOS6.x: Install the package **`vsftpd-2.2.2-24.el6.x86_64`** using `yum` or `rpm`.
RHEL7.x / CentOS7.x: Install the package **`vsftpd-3.0.2-22.el7.x86_64`** using `yum` or `rpm`.

The rpm files can be downloaded from centos.pkgs.org for Centos 6.x or Centos 7.x

The yum install command (as root):

```
# yum install vsftpd
```

Ftp and Sftp clients

For testing purposes it can be very useful to install ftp and sftp clients. Again `yum` or `rpm` is used.

SCP and SFTP Setup

The SFTP implementation uses the Secure FTP server that is part of `sshd` and does not rely on additional installations. For secure file transfers between the NetYCE servers and the network devices specific setup is required.

These requirements are met using:

- an '**ycicle**' user that is identical (uid, gid, home) to 'yce' with the default password '**ycicle**'
- ensuring that the 'ycicle' user appears *below* the 'yce' user in the /etc/passwd file
- changing the tftp root from `/var/opt/tftp` to `/var/opt/shared/public` where the directory `shared` is 'root' owned.
- The entire directory tree **public** must 'yce:nms' owned (and use 0755 permissions)
- change the sshd configuration in `/etc/ssh/sshd_config`:

```
# NetYCE 2019

Protocol 2

SyslogFacility AUTHPRIV

PasswordAuthentication yes

ChallengeResponseAuthentication no

GSSAPIAuthentication yes
GSSAPICleanupCredentials yes

UsePAM yes
PermitRootLogin no

AcceptEnv LANG LC_CTYPE LC_NUMERIC LC_TIME LC_COLLATE LC_MONETARY
LC_MESSAGES
AcceptEnv LC_PAPER LC_NAME LC_ADDRESS LC_TELEPHONE LC_MEASUREMENT
AcceptEnv LC_IDENTIFICATION LC_ALL LANGUAGE
AcceptEnv XMODIFIERS

X11Forwarding no
IgnoreRhosts yes
PrintLastLog yes

# Set Ciphers and MACs
Ciphers aes128-ctr,aes192-ctr,aes256-ctr
MACs hmac-ripemd160,hmac-sha2-256,hmac-sha2-512,hmac-ripemd160@openssh.com

# no reverse lookups
UseDNS no

# use default subsystem, not the internal-sftp
# Subsystem       sftp    internal-sftp
Subsystem         sftp    /usr/libexec/openssh/sftp-server

# 'ycicle' user will be chroot-ed using MySecureShell and cpsh
# do not use the chroot or forcecommand options.
Match User ycicle
     AllowTCPForwarding no
     X11Forwarding no
#    ChrootDirectory /var/opt/shared
#    ForceCommand internal-sftp
#    ForceCommand /opt/yce/bin/cpsh.pl
```

> Note: NetYCE patch nr '14081902' will create the `ycicle` user and ssh modifications automatically. This patch is normally NOT executed because of policy constraints.

This patch can be executed (as 'yce') using:

```
$ cd /opt/yce/system/patches
$ /opt/ycelib/perl/bin/perl 14081902 -F -d 1
```

MySecureShell setup

MySecureShell is an essential part of the SCP and SFTP configuration in that it allows to create chrooted environments for both SSH and SFTP without interfering with the FTP setup.

Its configuration file is `/etc/ssh/sftp_config`.

Create the following in **`/etc/ssh/sftp_config`**
The limited download speeds (100mbps global and 10mbps per session) are intended as guidelines to prevent multiple OS-file transfers to consume too much bandwidth. These values can be adjusted to suit server and network capabilities.

```
# MySecureShell Configuration File ##
# NetYCE 2019

# Default rules for everybody
<Default>
        GlobalDownload          100m    #total speed download for all
clients
                                        # o -> bytes   k -> kilo bytes   m
-> mega bytes
        GlobalUpload            0       #total speed upload for all clients
(0 for unlimited)
        Download                10m     #limit speed download for each
connection
        Upload                  0       #unlimited speed upload for each
connection
        StayAtHome              true    #limit client to his home
        VirtualChroot           true    #fake a chroot to the home account
        LimitConnection         50      #max connection for the server sftp
        LimitConnectionByUser   50      #max connection for the account
        LimitConnectionByIP     50      #max connection by ip for the
account
        Home                    /var/opt/shared/
        Shell                   /opt/yce/bin/cpsh.pl
        IdleTimeOut             30m     # disconnect idle client after 30
min
        ResolveIP               false   #resolve ip to dns
        IgnoreHidden            true    #treat all hidden files as if they
don't exist
        DirFakeUser             true    #Hide real file/directory owner
(just change displayed permissions)
        DirFakeGroup            true    #Hide real file/directory group
(just change displayed permissions)
        DirFakeMode             0400    #Hide real file/directory rights
```

```
(just change displayed permissions)
                                        #Add execution right for directory
if read right is set
        HideNoAccess            true    #Hide file/directory which user has
no access
#       MaxOpenFilesForUser     20      #limit user to open x files on same
time
#       MaxWriteFilesForUser    10      #limit user to x upload on same time
#       MaxReadFilesForUser     10      #limit user to x download on same
time
        DefaultRights           0640 0750       #Set default rights for new
file and new directory
#       MinimumRights           0400 0700       #Set minimum rights for
files and dirs
        ShowLinksAsLinks        false   #show links as their destinations
        ConnectionMaxLife       2h      #limits connection lifetime to 2
hours
#       Charset                 "ISO-8859-15"   #set charset of computer
</Default>

<User ycicle>
        Shell                   /opt/yce/bin/cpsh.pl
        Home                    /var/opt/shared/
        VirtualChroot           true
        ResolveIP               false
        IgnoreHidden            true
        ShowLinksAsLinks        false
</User>

#Include /etc/my_sftp_config_file       #include this valid configuration
file
```

FTP Setup

FTP access can be setup to function in conjunction with SFTP. It depends on the `vsftpd` package (“very secure ftp daemon”).

The vsftpd package uses the configuration directory `/etc/vsftpd`. The relevant files:

```
-rw-r--r-- 1 root root   25 Mar 24  2017 chroot_list
-rw------- 1 root root  125 Mar 22  2017 ftpusers
-rw------- 1 root root  361 Mar 22  2017 user_list
-rw------- 1 root root  547 Mar 24  2017 vsftpd.conf
```

The configuration file `vsftpd.conf` is different for RHEL6/CentOS6 and RHEL7/CentOS7.

vsftpd.conf RHEL6/CentOS6

```
# NetYCE 2018

anonymous_enable=NO
local_enable=YES
write_enable=YES
local_umask=002
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
xferlog_file=/var/opt/yce/logs/ftpxfer.log
vsftpd_log_file=/var/opt/yce/logs/ftplog.log
xferlog_std_format=YES
chroot_list_enable=YES
listen=YES

pam_service_name=vsftpd
userlist_enable=YES
tcp_wrappers=YES

local_root=/var/opt/shared
secure_chroot_dir=/var/opt/shared
chown_username=yce.nms
guest_enable=NO

force_dot_files=NO
hide_file={.yce_prop}
delete_failed_uploads=YES
log_ftp_protocol=NO
reverse_lookup_enable=NO
```

vsftpd.conf RHEL7/CentOS7

```
# NetYCE 2018
# vsftpd IPv4 + IPv6, binds all addresses

anonymous_enable=NO
local_enable=YES
write_enable=YES
local_umask=002
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
xferlog_file=/var/opt/yce/logs/ftpxfer.log
vsftpd_log_file=/var/opt/yce/logs/ftplog.log
xferlog_std_format=YES
chroot_list_enable=YES
listen=NO
listen_ipv6=YES
```

```
pam_service_name=vsftpd
userlist_enable=YES
tcp_wrappers=YES

local_root=/var/opt/shared
secure_chroot_dir=/var/opt/shared
chown_username=yce.nms
guest_enable=NO
force_dot_files=NO
hide_file={.yce_prop}
delete_failed_uploads=YES
log_ftp_protocol=NO
```

The setup refers to the `chroot_list` file, containing the users forced to chroot in their home directories. Modify this file to read:

```
# NetYCE 2018

ycicle
ftp
```

The `ftpusers` and `user_list` files can remain unchanged.

The vsftpd is like the other daemons and application processes under control of yce_psmon.

> Note: The logfile `/var/opt/yce/logs/ftpxfer.log` will be created root-owned. For rotation and maintenance purposes, this should be chown-ed to yce.nms. This log will then be maintained by bin/log_maint.pl

download scp-config files

This tgz contains MysecureShell rpm file and the config files shown above

scp-files.tgz