

rclone

1) Installation.....	1
2) Configuration.....	1
2.1) Server setup.....	1
2.2) Client setup.....	2
2.3) Server setup - part 2.....	2
2.4) Client verification.....	3
2.5) rclone - part 1.....	3
2.6) rclone - part 2.....	4
3) Backup configuration.....	5
4) Usage.....	5

1) Installation

<https://rclone.org/install/>

Script installation

To install rclone on Linux/macOS/BSD systems, run:

```
# cd /opt
```

...

```
# curl https://rclone.org/install.sh | sudo bash
```

Show the rclone version:

```
$ rclone version
```

```
rclone v1.55.0
- os/type: linux
- os/arch: amd64
- go/version: go1.16.2
- go/linking: static
- go/tags: cmount
```

2) Configuration

Source: <https://access.redhat.com/solutions/2399571>

How to configure an sftp server with restricted chroot users with ssh keys

2.1) Server setup

```
# useradd user1
```

```
# passwd user1
```

2.2) Client setup

Copy the ssh key from the client to the server (The user does not have to exist on the client)

```
$ ssh-copy-id user1@server-3
```

```
...
```

```
$ ssh user1@server-3
```

```
...
```

```
$ exit
```

```
logout
```

```
Connection to server closed.
```

```
$
```

Verify that your sftp connection works without a password prompt

```
$ sftp user1@server-3
```

```
Connected to server
```

```
sftp> quit
```

```
$
```

Without making any changes, user1 has full access and can ssh or sftp and change to any directory. We'll now make the necessary changes to chroot user1 and keep them jailed and locked down to a specified directory.

2.3) Server setup - part 2

Create a new group to add all your jailed chroot users on the server

```
# groupadd sftpusers
```

Create a common directory for all of your jailed chroot users

```
# mkdir /sftp
```

Create a subdirectory for each individual user that you want to chroot

```
# mkdir /sftp/user1
```

Create the "home" directory for the user

```
# mkdir /sftp/user1/home
```

Modify the user to add them to the new group you created

```
# usermod -aG sftpusers user1
```

Change permission for the users chrooted "home" directory only. It's important to leave everything else with the default root permissions.

```
# chown user1:sftpusers /sftp/user1/home/
```

Modify the /etc/ssh/sshd_config file and add the following lines:

```
# vim /etc/ssh/sshd_config
```

```
...
```

```
Subsystem sftp internal-sftp -d /home
```

```
Match Group sftpusers  
ChrootDirectory /sftp/%u
```

Restart the sshd service:

```
# systemctl restart sshd
```

2.4) Client verification

From the client, verify that everything is working now

```
ssh user1@server-3
```

```
Last login: ... from ...  
Could not chdir to home directory /home/user1: No such file or directory  
/bin/bash: No such file or directory  
Connection to server closed.
```

```
$
```

The user can no longer connect via ssh. Let's try sftp

```
$ sftp user1@server-3
```

Connected to server.

```
sftp> pwd
```

Remote working directory: /home

```
sftp> cd /etc
```

```
Couldn't canonicalize: No such file or directory  
Ok!
```

2.5) rclone - part 1

```
...  
$ rclone config  
n  
name> server_3_user1  
Storage> 32  
host> server-3  
user> user2  
port>  
y/g/n>  
key_pem>  
key_file>  
...  
y/g/n> g  
...  
pubkey_file>  
key_use_agent>  
use_insecure_cipher>
```

```
disable_hashcheck>
Edit advanced config? (y/n)
n
```

```
Remote config
```

```
-----
[server_3_user1]
type = sftp
host = server-3
user = user1
pass = *** ENCRYPTED ***
-----
```

```
...
$ rclone mkdir server_3_user1:scratch
```

```
...
$ rclone sync -vvv --bwlimit 50M ~/scratch/ server_3_user1:scratch
```

```
...
$ rclone check --bwlimit 50M ~/scratch/ server_3_user1:scratch
```

```
...
$ rclone ls server_3_user1:scratch
```

2.6) rclone - part 2

```
...
$ rclone config
```

```
n) New remote
name> server_3_user1_crypt
```

```
Storage> 11 - Encrypt/Decrypt a remote - "crypt"
** See help for crypt backend at: https://rclone.org/crypt/ **
```

```
remote> server_3_user1:crypt
filename_encryption> 1
directory_name_encryption> 1
Password or pass phrase for encryption.
Bits> 128
<see Password above>
Password or pass phrase for salt.
Bits> 128
```

```
...
Remote config
-----
[server_3_user1_crypt]
type = crypt
remote = server_3_user1:crypt
filename_encryption = standard
directory_name_encryption = true
password = *** ENCRYPTED ***
password2 = *** ENCRYPTED ***
```

3) Backup configuration

```
...  
$ rclone config file
```

```
...  
$ tail -n 8 ...
```

4) Usage

<https://rclone.org/>

Remote: server_3_user1_crypt:scratch

```
...  
$ rclone listremotes
```

```
...  
$ rclone mkdir server_3_user1_crypt:scratch
```

```
...  
$ rclone sync -vvv --progress --bwlimit 50M ~/scratch/  
server_3_user1_crypt:scratch
```

```
...  
$ rclone check ~/scratch/ server_3_user1_crypt:scratch
```

```
...  
$ rclone ls server_3_user1_crypt:scratch
```

```
...  
$ rclone tree server_3_user1_crypt:scratch
```

```
...  
$ rclone ncd� server_3_user1_crypt:scratch
```

```
...  
$ rclone ncd� server_3_user1_crypt:
```