rclone Cloud file sync

CONCEPTS

Paths are remote:path. A path without a colon is local. Test any sync with --dry-run first.

Source local or remote
↓ copy / sync / move
rclone filters, checks
↓ backend API
Destination remote:path

sync makes the destination identical to the source and DELETES extra files there. Swapped arguments delete files you wanted to keep.

Config lives in rclone.conf (find it with rclone config file). It holds tokens and secrets: keep it private.

REMOTES AND CONFIG

rclone config
Interactive remote setup

rclone listremotes
List configured remotes

rclone config show *<remote>*
Show a remote's settings

rclone config file
Print config file path

rclone config create *<name>* *<type>* **[key=value]**
Create a remote non-interactively

rclone config delete *<name>*
Remove a remote

COPY, SYNC AND MOVE

rclone copy *<src>* *<dst>* **-P**
Copy new/changed files, never delete

rclone sync *<src>* *<dst>* **-P**
Mirror src to dst, deleting extras

rclone move *<src>* *<dst>*
Copy then delete from source

rclone copyto *<src>* *<dst>*
Copy one file, renaming it

rclone moveto *<src>* *<dst>*
Move or rename one file

COPY, SYNC AND MOVE (cont.)

rclone rcat *<remote>*:*<file>*
Upload from stdin

rclone cat *<remote>*:*<file>*
Print a remote file

LISTING AND INFO

rclone ls *<remote>*:*<dir>*
Size and path of every file

rclone lsd *<remote>*:*<dir>*
List directories only

rclone lsf *<remote>*:*<dir>*
Plain names, script friendly

rclone lsf -R --files-only *<remote>*:
Recursive file names

rclone lsjson *<remote>*:*<dir>*
List as JSON

rclone tree *<remote>*:*<dir>*
Tree view

rclone size *<remote>*:*<dir>*
Total size and file count

rclone ncdu *<remote>*:
Interactive disk usage

rclone about *<remote>*:
Quota and free space

VERIFY AND CLEAN UP

rclone check *<src>* *<dst>*
Compare sizes and hashes

rclone check *<src>* *<dst>* **--one-way**
Only check source files exist

rclone delete *<remote>*:*<dir>*
Delete files, keep directories

rclone purge *<remote>*:*<dir>*
Delete directory and everything in it

rclone mkdir *<remote>*:*<dir>*
Create a directory or bucket

FLAGS

-n / --dry-run
Show what would happen

-P / --progress
Live transfer stats

FLAGS (cont.)

-v / -vv
More logging

-i / --interactive
Ask before each change

--transfers *<n>*
Parallel file transfers

--bwlimit *<rate>*
Limit bandwidth (e.g. 5M)

-u / --update
Skip files newer on destination

--ignore-existing
Never overwrite existing files

-c / --checksum
Compare by hash, not mod-time

--backup-dir *<dir>*
Move overwritten/deleted files aside

--max-delete *<n>*
Abort if sync would delete more

--log-file *<file>* **--log-level INFO**
Log to a file

FILTERS

--include "*<glob>***"**
Only matching files

--exclude "*<glob>***"**
Skip matching files

--exclude-from *<file>*
Read exclude patterns from file

--filter-from *<file>*
Rules in a file (+ / - lines)

--max-age *<dur>*
Only files newer than e.g. 7d

MOUNT AND SERVE

rclone mount *<remote>*: *<dir>*
Mount as a filesystem

rclone mount *<remote>*: *<dir>* **--vfs-cache-mode writes**
Make writes reliable

rclone mount ... --daemon
Run in the background

fusermount -u *<dir>*
Unmount on Linux