kDrive

Tier 2 added in v1.76

Paths are specified as `remote:path`

Paths may be as deep as required, e.g. `remote:directory/subdirectory`.

Configuration

The initial setup for kDrive involves getting an OAuth2 token which you need to do in your browser. `rclone config` walks you through it.

kDrive uses the PKCE OAuth2 flow (RFC 7636), which means no `client_secret` is required. The `client_id` is optional and defaults to rclone's built-in application credentials.

Here is an example of how to make a remote called `remote`. First run:

```
rclone config
```

This will guide you through an interactive setup process:

```
No remotes found, make a new one?
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
name> remote
Type of storage to configure.
Choose a number from below, or type in your own value
[snip]
XX / Infomaniak kDrive
   \ "kdrive"
[snip]
Storage> kdrive

Option client_id.
OAuth Client Id.
Leave blank normally.
Enter a value. Press Enter to leave empty.
client_id>

Option drive_id.
ID of the kDrive to use.
When showing a folder on kdrive, you can find the drive_id here:
https://ksuite.infomaniak.com/{account_id}/kdrive/app/drive/{drive_id}/files/...
Enter a value. Press Enter to leave empty.
drive_id> 0654321

Use web browser to automatically authenticate rclone with remote?
 * Say Y if the machine running rclone has a web browser you can use
 * Say N if running rclone on a (remote) machine without web browser access
If not sure try Y. If Y failed, try N.
y/n> y

Waiting for code...
Got code

Configuration complete.
Options:
- type: kdrive
- drive_id: 0654321

Keep this "remote" remote?
y) Yes this is OK (default)
e) Edit this remote
d) Delete this remote
y/e/d> y
```

Once configured you can then use `rclone` like this (replace `remote` with the name you gave your remote):

List directories in top level of your rclone

```
rclone lsd remote:
```

Note that by default the top level directory of a private kDrive storage is your "Private" folder. When used with a Pro account, a second folder "Common documents" is also available, which can be used as the root by setting `root_folder_id` to `common`. *These top-level folders cannot be modified.*

List all the files in your kDrive

```
rclone ls remote:
```

To copy a local directory to a kDrive directory called backup

```
rclone copy /home/source remote:backup
```

Modification times and hashes

kDrive allows modification times to be set on objects accurate to 1 second. These will be used to detect whether objects need syncing or not.

kDrive supports the XXH3 checksum hash, so you can use the `--checksum` flag. XXH3 is only available for files uploaded as a single chunk: files uploaded in chunks (by rclone when larger than `--kdrive-chunk-size`, or by the kDrive apps which chunk files at their own sizes) have a "nested" hash which can't be compared with a plain XXH3, and rclone reports no hash for those files.

Restricted filename characters

In addition to the default restricted characters set the following characters are also replaced:

CHARACTER	VALUE	REPLACEMENT
LeftSpace	0x20	␠
RightSpace	0x20	␠

Invalid UTF-8 bytes will also be replaced, as they can't be used in JSON strings.

Overwriting files

When rclone overwrites an existing file, kDrive keeps the replaced content as a new version of the file. Old versions count against the drive quota, so repeatedly overwriting large files grows the storage usage. Manage old versions from the kDrive web app.

Uploading files of unknown size

kDrive can't upload files of unknown size, so `rclone rcat` and streaming uploads are not supported.

Deleting files

Deleted files will be moved to the trash. Your subscription level will determine how long items stay in the trash. `rclone cleanup` can be used to empty the trash.

Root folder ID

You can set the `root_folder_id` for rclone. This is the directory (identified by its numeric `Folder ID`) that rclone considers to be the root of your kDrive drive.

Normally you will leave this blank and rclone will use your private directory. You can also use one of the convenience shortcuts:

- `private` : your user private directory (default)
- `common` : the kDrive common directory, shared among users

Or the numeric `Folder ID` of any directory in your drive. The root of the drive only contains the `Private` and `Common documents` folders, so a numeric ID is needed to use any other directory as root. In order to find the `Folder ID` , list the parent directory of the folder you wish rclone to use with `rclone lsf` : the first column of the output is the folder ID:

Example:

```
$ rclone lsf --dirs-only -Fip --csv remote:Documents
456,Sub Folder/
789,Pdfexos/
```

So if the folder you want rclone to use is "Photos/", then use the returned id from `rclone lsf` command (ex. `789`) as the `root_folder_id` variable value in the config file.

Standard options

Here are the Standard options specific to kdrive (Infomaniak kDrive).

--kdrive-client-id

OAuth Client Id.

Leave blank normally.

Properties:

- Config: client_id
- Env Var: RCLONE_KDRIVE_CLIENT_ID
- Type: string
- Required: false

--kdrive-drive-id

ID of the kDrive to use.

When showing a folder on kdrive, you can find the drive_id here: https://ksuite.infomaniak.com/{account_id}/kdrive/app/drive/{drive_id}/files/...

Properties:

- Config: drive_id
- Env Var: RCLONE_KDRIVE_DRIVE_ID
- Type: string
- Required: false

Advanced options

Here are the Advanced options specific to kdrive (Infomaniak kDrive).

--kdrive-token

OAuth Access Token as a JSON blob.

Properties:

- Config: token
- Env Var: RCLONE_KDRIVE_TOKEN
- Type: string
- Required: false

--kdrive-auth-url

Auth server URL.

Leave blank to use the provider defaults.

Properties:

- Config: auth_url
- Env Var: RCLONE_KDRIVE_AUTH_URL
- Type: string
- Required: false

--kdrive-token-url

Token server url.

Leave blank to use the provider defaults.

Properties:

- Config: token_url
- Env Var: RCLONE_KDRIVE_TOKEN_URL
- Type: string
- Required: false

--kdrive-encoding

The encoding for the backend.

See the encoding section in the overview for more info.

Properties:

- Config: encoding
- Env Var: RCLONE_KDRIVE_ENCODING
- Type: Encoding
- Default: Slash,Del,Ctl,LeftSpace,RightSpace,InvalidUtf8,Dot

--kdrive-root-folder-id

The folder to use as root.

You may use either one of the convenience shortcuts [private or common] or an explicit folder ID.

- private: your user private directory
- common: the kDrive common directory, shared among users

Properties:

- Config: root_folder_id
- Env Var: RCLONE_KDRIVE_ROOT_FOLDER_ID
- Type: string
- Required: false

--kdrive-chunk-size

Files above this size are uploaded in chunks of this size.

The kDrive API accepts chunks between 1 MB and 1 GB, up to 10,000 chunks per file and a total file size of 1 TB. The chunk size is raised automatically for very large files to stay within the 10,000 chunks limit.

Properties:

- Config: chunk_size
- Env Var: RCLONE_KDRIVE_CHUNK_SIZE
- Type: SizeSuffix
- Default: 20Mi

--kdrive-upload-concurrency

Concurrency for chunked uploads.

Properties:

- Config: upload_concurrency
- Env Var: RCLONE_KDRIVE_UPLOAD_CONCURRENCY
- Type: int
- Default: 4

--kdrive-endpoint

The API endpoint to use.

Leave blank normally. There is no reason to change the endpoint except for internal use or beta testing.

Properties:

- Config: endpoint
- Env Var: RCLONE_KDRIVE_ENDPOINT
- Type: string
- Required: false

--kdrive-description

Description of the remote.

Properties:

- Config: description
- Env Var: RCLONE_KDRIVE_DESCRIPTION
- Type: string
- Required: false

PLATINUM SPONSOR



GOLD SPONSOR



GOLD SPONSOR

GOLD SPONSOR

Dropbox

CESNET

cesnet

SHARE AND ENJOY

Twitter
Facebook
Reddit
Star 60,835

LINKS

Rclone forum
GitHub project
Rclone Wiki
Sponsor
@rclone

© Nick Craig-Wood 2014-2026
Source file kdrive.md last updated 2026-10-21
Uploaded with rclone. Built with Hugo. Logo by @andy23. Served by Caddy. Hosted at Hetzner Cloud.