Dropbox

Paths are specified as `remote:path`

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

The initial setup for dropbox involves getting a token from Dropbox which you need to do in your browser. `rclone config` walks you through it.

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:

```
n) New remote
d) Delete remote
q) Quit config
e/n/d/q> n
name> remote
Type of storage to configure.
Choose a number from below, or type in your own value
[snip]
XX / Dropbox
   \ "dropbox"
[snip]
Storage> dropbox
Dropbox App Key - leave blank normally.
app_key>
Dropbox App Secret - leave blank normally.
app_secret>
Remote config
Please visit:
https://www.dropbox.com/1/oauth2/authorize?client_id=XXXXXXXXXXXXXXX&response_type=code
Enter the code: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX_XXXXXXXXXX
--------------------
[remote]
app_key =
app_secret =
token = XXXXXXXXXXXXXXXXXXXXXXXXXXXXX_XXXX_XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
--------------------
y) Yes this is OK
e) Edit this remote
d) Delete this remote
y/e/d> y
```

You can then use it like this,

List directories in top level of your dropbox

```
rclone lsd remote:
```

List all the files in your dropbox

```
rclone ls remote:
```

To copy a local directory to a dropbox directory called backup

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

Dropbox for business

Rclone supports Dropbox for business and Team Folders.

When using Dropbox for business `remote:` and `remote:path/to/file` will refer to your personal folder.

If you wish to see Team Folders you must use a leading `/` in the path, so `rclone lsd remote:/` will refer to the root and show you all Team Folders and your User Folder.

You can then use team folders like this `remote:/TeamFolder` and `remote:/TeamFolder/path/to/file`.

A leading `/` for a Dropbox personal account will do nothing, but it will take an extra HTTP transaction so it should be avoided.

Modified time and Hashes

Dropbox supports modified times, but the only way to set a modification time is to re-upload the file.

This means that if you uploaded your data with an older version of rclone which didn't support the v2 API and modified times, rclone will decide to upload all your old data to fix the modification times. If you don't want this to happen use `--size-only` or `--checksum` flag to stop it.

Dropbox supports [its own hash type]() which is checked for all transfers.

Restricted filename characters

Character	Value	Replacement
NUL	0x00	␀
/	0x2F	／
DEL	0x7F	␡
\	0x5C	＼

File names can also not end with the following characters. These only get replaced if they are the last character in the name:

Character	Value	Replacement

Character	Value	Replacement
SP	0x20	␠

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

Standard Options

Here are the standard options specific to dropbox (Dropbox).

--dropbox-client-id

OAuth Client Id Leave blank normally.

- Config: client_id
- Env Var: RCLONE_DROPBOX_CLIENT_ID
- Type: string
- Default: ""

--dropbox-client-secret

OAuth Client Secret Leave blank normally.

- Config: client_secret
- Env Var: RCLONE_DROPBOX_CLIENT_SECRET
- Type: string
- Default: ""

Advanced Options

Here are the advanced options specific to dropbox (Dropbox).

--dropbox-token

OAuth Access Token as a JSON blob.

- Config: token
- Env Var: RCLONE_DROPBOX_TOKEN
- Type: string
- Default: ""

--dropbox-auth-url

Auth server URL. Leave blank to use the provider defaults.

- Config: auth_url
- Env Var: RCLONE_DROPBOX_AUTH_URL
- Type: string
- Default: ""

--dropbox-token-url

Token server url. Leave blank to use the provider defaults.

- Config: token_url
- Env Var: RCLONE_DROPBOX_TOKEN_URL
- Type: string
- Default: ""

--dropbox-chunk-size

Upload chunk size. (< 150M).

Any files larger than this will be uploaded in chunks of this size.

Note that chunks are buffered in memory (one at a time) so rclone can deal with retries. Setting this larger will increase the speed slightly (at most 10% for 128MB in tests) at the cost of using more memory. It can be set smaller if you are tight on memory.

- Config: chunk_size
- Env Var: RCLONE_DROPBOX_CHUNK_SIZE
- Type: SizeSuffix
- Default: 48M

--dropbox-impersonate

Impersonate this user when using a business account.

- Config: impersonate
- Env Var: RCLONE_DROPBOX_IMPERSONATE
- Type: string
- Default: ""

--dropbox-shared-files

Instructs rclone to work on individual shared files.

In this mode rclone's features are extremely limited - only list (ls, lsl, etc.) operations and read operations (e.g. downloading) are supported in this mode. All other operations will be disabled.

- Config: shared_files
- Env Var: RCLONE_DROPBOX_SHARED_FILES
- Type: bool
- Default: false

--dropbox-shared-folders

Instructs rclone to work on shared folders.

When this flag is used with no path only the List operation is supported and all available shared folders will be listed. If you specify a path the first part will be interpreted as the name of shared folder. Rclone will then try to mount this shared to the root namespace. On success shared folder rclone proceeds normally. The shared folder is now pretty much a normal folder and all normal operations are supported.

Note that we don't unmount the shared folder afterwards so the --dropbox-shared-folders can be omitted after the first use of a particular shared folder.

- Config: shared_folders
- Env Var: RCLONE_DROPBOX_SHARED_FOLDERS
- Type: bool
- Default: false

--dropbox-encoding

This sets the encoding for the backend.

See: the [encoding section in the overview](#) for more info.

- Config: encoding
- Env Var: RCLONE_DROPBOX_ENCODING
- Type: MultiEncoder
- Default: Slash,BackSlash,Del,RightSpace,InvalidUtf8,Dot

Limitations

Note that Dropbox is case insensitive so you can't have a file called "Hello.doc" and one called "hello.doc".

There are some file names such as `thumbs.db` which Dropbox can't store. There is a full list of them in the ["Ignored Files" section of this document](#). Rclone will issue an error message `File name disallowed - not uploading` if it attempts to upload one of those file names, but the sync won't fail.

Some errors may occur if you try to sync copyright-protected files because Dropbox has its own [copyright detector](#) that prevents this sort of file being downloaded. This will return the error `ERROR : /path/to/your/file: Failed to copy: failed to open source object: path/restricted_content/.`

If you have more than 10,000 files in a directory then `rclone purge dropbox:dir` will return the error `Failed to purge: There are too many files involved in this operation`. As a work-around do an `rclone delete dropbox:dir` followed by an `rclone rmdir dropbox:dir`.

Get your own Dropbox App ID

When you use rclone with Dropbox in its default configuration you are using rclone's App ID. This is shared between all the rclone users.

Here is how to create your own Dropbox App ID for rclone:

1. Log into the [Dropbox App console](#) with your Dropbox Account (It need not to be the same account as the Dropbox you want to access)

2. Choose an API => Usually this should be `Dropbox API`
3. Choose the type of access you want to use => `Full Dropbox` or `App Folder`
4. Name your App. The app name is global, so you can't use `rclone` for example
5. Click the button `Create App`
6. Fill `Redirect URIs` as `http://localhost:53682/`
7. Find the `App key` and `App secret` Use these values in rclone config to add a new remote or edit an existing remote.