Mail.ru Cloud

Mail.ru Cloud is a cloud storage provided by a Russian internet company Mail.Ru Group. The official desktop client is Disk-O:, available only on Windows. (Please note that official sites are in Russian)

Currently it is recommended to disable 2FA on Mail.ru accounts intended for rclone until it gets eventually implemented.

Features highlights

- Paths may be as deep as required, e.g. `remote:directory/subdirectory`
- Files have a `last modified time` property, directories don't
- Deleted files are by default moved to the trash
- Files and directories can be shared via public links
- Partial uploads or streaming are not supported, file size must be known before upload
- Maximum file size is limited to 2G for a free account, unlimited for paid accounts
- Storage keeps hash for all files and performs transparent deduplication, the hash algorithm is a modified SHA1
- If a particular file is already present in storage, one can quickly submit file hash instead of long file upload (this optimization is supported by rclone)

Configuration

Here is an example of making a mailru configuration. First create a Mail.ru Cloud account and choose a tariff, then 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.
Type of storage to configure.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value
[snip]
XX / Mail.ru Cloud
   \ "mailru"
[snip]
Storage> mailru
User name (usually email)
Enter a string value. Press Enter for the default ("").
user> username@mail.ru
Password
y) Yes type in my own password
g) Generate random password
y/g> y
Enter the password:
password:
Confirm the password:
password:
Skip full upload if there is another file with same data hash.
This feature is called "speedup" or "put by hash". It is especially efficient
in case of generally available files like popular books, video or audio clips
[snip]
Enter a boolean value (true or false). Press Enter for the default ("true").
Choose a number from below, or type in your own value
 1 / Enable
   \ "true"
 2 / Disable
   \ "false"
speedup_enable> 1
Edit advanced config? (y/n)
y) Yes
n) No
y/n> n
Remote config
--------------------
[remote]
type = mailru
user = username@mail.ru
pass = *** ENCRYPTED ***
speedup_enable = true
--------------------
y) Yes this is OK
e) Edit this remote
```

```
d) Delete this remote
y/e/d> y
```

Configuration of this backend does not require a local web browser. You can use the configured backend as shown below:

See top level directories

```
rclone lsd remote:
```

Make a new directory

```
rclone mkdir remote:directory
```

List the contents of a directory

```
rclone ls remote:directory
```

Sync `/home/local/directory` to the remote path, deleting any excess files in the path.

```
rclone sync -i /home/local/directory remote:directory
```

Modified time

Files support a modification time attribute with up to 1 second precision. Directories do not have a modification time, which is shown as "Jan 1 1970".

Hash checksums

Hash sums use a custom Mail.ru algorithm based on SHA1. If file size is less than or equal to the SHA1 block size (20 bytes), its hash is simply its data right-padded with zero bytes. Hash sum of a larger file is computed as a SHA1 sum of the file data bytes concatenated with a decimal representation of the data length.

Emptying Trash

Removing a file or directory actually moves it to the trash, which is not visible to rclone but can be seen in a web browser. The trashed file still occupies part of total quota. If you wish to empty your trash and free some quota, you can use the `rclone cleanup remote:`command, which will permanently delete all your trashed files. This command does not take any path arguments.

Quota information

To view your current quota you can use the `rclone about remote:` command which will display your usage limit (quota) and the current usage.

Restricted filename characters

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

Character	Value	Replacement
"	0x22	＂
*	0x2A	＊
:	0x3A	：
<	0x3C	＜
>	0x3E	＞
?	0x3F	？
\	0x5C	＼
|	0x7C	｜

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

Limitations

File size limits depend on your account. A single file size is limited by 2G for a free account and unlimited for paid tariffs. Please refer to the Mail.ru site for the total uploaded size limits.

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

Standard Options

Here are the standard options specific to mailru (Mail.ru Cloud).

--mailru-user

User name (usually email)

- Config: user
- Env Var: RCLONE_MAILRU_USER
- Type: string
- Default: ""

--mailru-pass

Password

NB Input to this must be obscured - see rclone obscure.

- Config: pass
- Env Var: RCLONE_MAILRU_PASS
- Type: string

- Default: ""

--mailru-speedup-enable

Skip full upload if there is another file with same data hash. This feature is called "speedup" or "put by hash". It is especially efficient in case of generally available files like popular books, video or audio clips, because files are searched by hash in all accounts of all mailru users. It is meaningless and ineffective if source file is unique or encrypted. Please note that rclone may need local memory and disk space to calculate content hash in advance and decide whether full upload is required. Also, if rclone does not know file size in advance (e.g. in case of streaming or partial uploads), it will not even try this optimization.

- Config: speedup_enable
- Env Var: RCLONE_MAILRU_SPEEDUP_ENABLE
- Type: bool
- Default: true
- Examples:
  - "true"
    - Enable
  - "false"
    - Disable

Advanced Options

Here are the advanced options specific to mailru (Mail.ru Cloud).

--mailru-speedup-file-patterns

Comma separated list of file name patterns eligible for speedup (put by hash). Patterns are case insensitive and can contain '*' or '?' meta characters.

- Config: speedup_file_patterns
- Env Var: RCLONE_MAILRU_SPEEDUP_FILE_PATTERNS
- Type: string
- Default: "*.mkv,*.avi,*.mp4,*.mp3,*.zip,*.gz,*.rar,*.pdf"
- Examples:
  - ""
    - Empty list completely disables speedup (put by hash).
  - "*"
    - All files will be attempted for speedup.
  - "*.mkv,*.avi,*.mp4,*.mp3"
    - Only common audio/video files will be tried for put by hash.
  - "*.zip,*.gz,*.rar,*.pdf"
    - Only common archives or PDF books will be tried for speedup.

--mailru-speedup-max-disk

This option allows you to disable speedup (put by hash) for large files (because preliminary hashing can exhaust you RAM or disk space)

- Config: speedup_max_disk
- Env Var: RCLONE_MAILRU_SPEEDUP_MAX_DISK
- Type: SizeSuffix
- Default: 3G
- Examples:
    - "0"
        - Completely disable speedup (put by hash).
    - "1G"
        - Files larger than 1Gb will be uploaded directly.
    - "3G"
        - Choose this option if you have less than 3Gb free on local disk.

--mailru-speedup-max-memory

Files larger than the size given below will always be hashed on disk.

- Config: speedup_max_memory
- Env Var: RCLONE_MAILRU_SPEEDUP_MAX_MEMORY
- Type: SizeSuffix
- Default: 32M
- Examples:
    - "0"
        - Preliminary hashing will always be done in a temporary disk location.
    - "32M"
        - Do not dedicate more than 32Mb RAM for preliminary hashing.
    - "256M"
        - You have at most 256Mb RAM free for hash calculations.

--mailru-check-hash

What should copy do if file checksum is mismatched or invalid

- Config: check_hash
- Env Var: RCLONE_MAILRU_CHECK_HASH
- Type: bool
- Default: true
- Examples:
    - "true"
        - Fail with error.
    - "false"
        - Ignore and continue.

--mailru-user-agent

HTTP user agent used internally by client. Defaults to "rclone/VERSION" or "--user-agent" provided on command line.

- Config: user_agent
- Env Var: RCLONE_MAILRU_USER_AGENT
- Type: string
- Default: ""

--mailru-quirks

Comma separated list of internal maintenance flags. This option must not be used by an ordinary user. It is intended only to facilitate remote troubleshooting of backend issues. Strict meaning of flags is not documented and not guaranteed to persist between releases. Quirks will be removed when the backend grows stable. Supported quirks: atomicmkdir binlist unknowndirs

- Config: quirks
- Env Var: RCLONE_MAILRU_QUIRKS
- Type: string
- Default: ""

--mailru-encoding

This sets the encoding for the backend.

See: the encoding section in the overview for more info.

- Config: encoding
- Env Var: RCLONE_MAILRU_ENCODING
- Type: MultiEncoder
- Default: Slash,LtGt,DoubleQuote,Colon,Question,Asterisk,Pipe,BackSlash,Del,Ctl,InvalidUtf8,Dot