

Install

Rclone is a Go program and comes as a single binary file.

Quickstart

- [Download](#) the relevant binary.
- Extract the `rclone` or `rclone.exe` binary from the archive
- Run `rclone config` to setup. See [rclone config docs](#) for more details.

See below for some expanded Linux / macOS instructions.

See the [Usage section](#) of the docs for how to use rclone, or run `rclone -h`.

Script installation

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

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

For beta installation, run:

```
curl https://rclone.org/install.sh | sudo bash -s beta
```

Note that this script checks the version of rclone installed first and won't re-download if not needed.

Linux installation from precompiled binary

Fetch and unpack

```
curl -O https://downloads.rclone.org/rclone-current-linux-amd64.zip
unzip rclone-current-linux-amd64.zip
cd rclone-*-linux-amd64
```

Copy binary file

```
sudo cp rclone /usr/bin/
sudo chown root:root /usr/bin/rclone
sudo chmod 755 /usr/bin/rclone
```

Install manpage

```
sudo mkdir -p /usr/local/share/man/man1
sudo cp rclone.1 /usr/local/share/man/man1/
sudo mandb
```

Run `rclone config` to setup. See [rclone config docs](#) for more details.

```
rclone config
```

macOS installation with brew

```
brew install rclone
```

macOS installation from precompiled binary, using curl

To avoid problems with macOS gatekeeper enforcing the binary to be signed and notarized it is enough to download with `curl`.

Download the latest version of rclone.

```
cd && curl -O https://downloads.rclone.org/rclone-current-osx-amd64.zip
```

Unzip the download and cd to the extracted folder.

```
unzip -a rclone-current-osx-amd64.zip && cd rclone-*-osx-amd64
```

Move rclone to your \$PATH. You will be prompted for your password.

```
sudo mkdir -p /usr/local/bin  
sudo mv rclone /usr/local/bin/
```

(the `mkdir` command is safe to run, even if the directory already exists).

Remove the leftover files.

```
cd .. && rm -rf rclone-*-osx-amd64 rclone-current-osx-amd64
```

Run `rclone config` to setup. See [rclone config docs](#) for more details.

```
rclone config
```

macOS installation from precompiled binary, using a web browser

When downloading a binary with a web browser, the browser will set the macOS gatekeeper quarantine attribute. Starting from Catalina, when attempting to run `rcclone`, a pop-up will appear saying:

```
"rcclone" cannot be opened because the developer cannot be verified.
macOS cannot verify that this app is free from malware.
```

The simplest fix is to run

```
xattr -d com.apple.quarantine rclone
```

Install with docker

The rclone maintains a [docker image for rclone](#). These images are autobuilt by docker hub from the rclone source based on a minimal Alpine linux image.

The `:latest` tag will always point to the latest stable release. You can use the `:beta` tag to get the latest build from master. You can also use version tags, e.g. `:1.49.1`, `:1.49` or `:1`.

```
$ docker pull rclone/rclone:latest
latest: Pulling from rclone/rclone
Digest: sha256:0e0ced72671989bb837fea8e88578b3fc48371aa45d20
...
$ docker run --rm rclone/rclone:latest version
rclone v1.49.1
- os/arch: linux/amd64
- go version: go1.12.9
```

There are a few command line options to consider when starting an rclone Docker container from the rclone image.

- You need to mount the host rclone config dir at `/config/rclone` into the Docker container. Due to the fact that rclone updates tokens inside its config file, and that the update process involves a file rename, you need to mount the whole host rclone config dir, not just the single host rclone config file.
- You need to mount a host data dir at `/data` into the Docker container.
- By default, the rclone binary inside a Docker container runs with UID=0 (root). As a result, all files created in a run will have UID=0. If your config and data files reside on the host with a non-root UID:GID, you need to pass these on the container start command line.
- If you want to access the RC interface (either via the API or the Web UI), it is required to set the `--rc-addr` to `:5572` in order to connect to it from outside the container. An explanation about why this is necessary is present [here](#).
 - NOTE: Users running this container with the docker network set to `host` should probably set it to listen to localhost only, with `127.0.0.1:5572` as the value for `--rc-addr`
- It is possible to use `rclone mount` inside a userspace Docker container, and expose the resulting fuse mount to the host. The exact `docker run` options to do that might vary slightly between hosts. See, e.g. the discussion in this [thread](#).

You also need to mount the host `/etc/passwd` and `/etc/group` for fuse to work inside the container.

Here are some commands tested on an Ubuntu 18.04.3 host:

```
# config on host at ~/.config/rclone/rclone.conf
# data on host at ~/data

# make sure the config is ok by listing the remotes
docker run --rm \
  --volume ~/.config/rclone:/config/rclone \
  --volume ~/data:/data:shared \
  --user $(id -u):$(id -g) \
  rclone/rclone \
  listremotes

# perform mount inside Docker container, expose result to host
mkdir -p ~/data/mount
docker run --rm \
  --volume ~/.config/rclone:/config/rclone \
  --volume ~/data:/data:shared \
  --user $(id -u):$(id -g) \
  --volume /etc/passwd:/etc/passwd:ro --volume /etc/group:/etc/group:ro \
  --device /dev/fuse --cap-add SYS_ADMIN --security-opt apparmor=unconfined \
  rclone/rclone \
  mount dropbox:Photos /data/mount &
ls ~/data/mount
kill %1
```

Install from source

Make sure you have at least [Go](#) 1.12 installed. [Download go](#) if necessary. The latest release is recommended. Then

```
git clone https://github.com/rclone/rclone.git
cd rclone
go build
./rclone version
```

This will leave you a checked out version of rclone you can modify and send pull requests with. If you use `make` instead of `go build` then the rclone build will have the correct version information in it.

You can also build the latest stable rclone with:

```
go get github.com/rclone/rclone
```

or the latest version (equivalent to the beta) with

```
go get github.com/rclone/rclone@master
```

These will build the binary in `$(go env GOPATH)/bin` (`~/go/bin/rclone` by default) after downloading the source to the go module cache. Note - do **not** use the `-u` flag here. This causes go to try to update the dependencies that rclone uses and sometimes these don't work with the current version of rclone.

Installation with Ansible

This can be done with [Stefan Weichinger's ansible role](#).

Instructions

1. `git clone https://github.com/stefangweichinger/ansible-rclone.git` into your local roles-directory
2. add the role to the hosts you want rclone installed to:

```
- hosts: rclone-hosts
  roles:
    - rclone
```