

# Installation, Configuration & Usage of rclone

Ultra.cc support is unable to provide support for rclone, this is due to the large volume of variables and different configurations possible with rclone. The guides found here on the knowledge-base should be able to guide you through using rclone, and any further questions can easily be answered with a quick Google search. You may also be able to find community support for rclone through our community Discord server or the Rclone forums.

Please make yourself aware of the Ultra.cc [Fair Usage Policy](#). It is very important *not* to mount your Cloud storage to any of the premade folders. Do *not* download directly to a rclone mount from a torrent or nzbget client. Both will create massive instability for both you and everyone else on your server. *Always follow the documentation and create a new folder for mounting. It is your responsibility to ensure usage is within acceptable limits.*

Please do not mount to any of the default directories such as:

downloads  
files  
media  
bin  
.apps  
.config  
www  
/homexx/username/  
or any pre-created directory found on your Ultra.cc Slot

In this guide, we will be going over the installation of rclone. We'll also cover the basic usages such as setting up a remote and how to copy files between your slot and a remote file host.

## Installation on your Ultra.cc Service

To begin, make sure you know how to SSH into your slot. All rclone commands are performed via SSH. You can find a guide on SSH [here](#).

To install rclone, run the command below; this command will automatically install rclone to your slot for you.

**rclone stable**

```
curl https://scripts.usbx.me/main/MergerFS-Rclone/Installer%20Scripts/rclone-install-stable.sh | bash
```

## rclone beta

```
curl https://scripts.usbx.me/main/MergerFS-Rclone/Installer%20Scripts/rclone-install-beta.sh | bash
```

# Installation on your local PC

## Windows

- Download the latest [rclone version](#).
- Extract the compressed zip file to any location of your liking using software like 7zip or WinRAR.
- Move `rclone.exe` to the following folder `C:\Windows\System32`.

## MacOS

- Open a terminal window.
- Install brew with the command given below:

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install.sh)"
```

- Install sudo and macfuse with brew using the commands below:

```
brew install macfuse  
brew install sudo
```

- Switch to root and finally install rclone using the commands below:

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

## Linux

- Open a terminal window.
- Run the command given below:

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

# Configuration

## OAuth (Google Drive)

- Navigate to [Google Developer Console](#) and log into your Google account.
- Click Create on the right-hand side of the screen. On the screen that pops up, enter a Project name and then click **Create**.
  - I'd recommend against using rclone in the project name; even just the default will do.

### New Project

 You have 12 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

[MANAGE QUOTAS](#)

**Project name \***  

Project ID: psychic-linker-243816. It cannot be changed later. [EDIT](#)

**Location \***  [BROWSE](#)

Parent organization or folder

**CREATE** **CANCEL**

- Click **Enable APIs and Services**, search for Drive and select **Google Drive**.
- Click **Enable** on the window that pops up.



## Google Drive API

Google

The Google Drive API allows clients to access resources from Google Drive

[ENABLE](#)[TRY THIS API ↗](#)**Type**[APIs & services](#)**Last updated**

1/10/19, 7:04 AM

**Category**

Storage

G Suite

**Service name**[drive.googleapis.com](#)

### Overview

The Google Drive API allows clients to access resources from Google Drive.

### About Google

Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.

### Tutorials and documentation

[Learn more ↗](#)

### Terms of service

By using this product you agree to the terms and conditions of the following license(s): [Google APIs Terms of Service](#) ↗

- Click **Create Credentials** on the top right side of the screen.
- Select **User data** and click **Next**.
- Fill out App name and email addresses as you see fit. Then **Save and Continue**.
- Click **Save and Continue** again. No need to set Scopes.
- Set Application type to **Desktop app** and then click on **Create**.
  - Then you will be given the option to Download your credentials and also a link to your credentials page.
  - You can download them to save them locally, but they will always be visible in your Google account's cloud console.
- Click on the **credentials page** link to go to your Credentials page. Here you will find your credentials under OAuth2.0 Client IDs.

**API APIs & Services**

To use this API, you may need credentials. Click 'CREATE CREDENTIALS' to get started.

**Enabled APIs & services**

**API/Service Details**

**DISABLE API**

**Google Drive API**

The Google Drive API allows clients to access resources from Google Drive.

By Google Enterprise API

**Service name**: drive.googleapis.com | **Type**: Public API | **Status**: Enabled

**LEARN MORE** **TRY IN API EXP**

**METRICS** **QUOTAS** **CREDENTIALS** **DRIVE UI INTEGRATION**

Select Graphs: **Graphs** 30 days

Filters: Versions: v1, v2, v2beta, v2internal, ... Credentials: Unspecified, Anonymous, ...

Methods:

## Create credentials

Credential Type

OAuth Consent Screen

Scopes (optional)

OAuth Client ID

**5 Your Credentials**

### Download your credentials

Download this credential information in JSON format. This is always available for you on the [credentials page](#).

Client ID —  

 DOWNLOAD

Your OAuth consent screen has been configured and is ready for use by others in your Google Workspace organization.

If you need users outside of your organization to have access, you can change the app type to "External" on the [OAuth consent screen page](#). Most external apps also require [verification](#), which can take up to 4-6 weeks (depending on which OAuth scopes your app uses).

**DONE**

CANCEL

## rclone

Now we need to configure a remote to use with rclone. For this guide, we will be configuring Google Drive. This is the most common remote people tend to use as it offers ample storage capacities for a reasonable price. Should you wish to use a different cloud host, feel free to modify the steps, you take.

- Run the command `rclone config`

```
kbguides@lw914: ~$ rclone config
```

```
2019/06/15 18:16:33 NOTICE: Config file "/home27/kbguides/.config/rclone/rclone.conf" not
```

```
found - using defaults
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q>
```

- Press **n** and then **Enter**. Type the name you wish to use for your remote and then press **Enter** once more. Here, we'll use `gdrive` as our remote name.
- Scroll through the list of supported remotes and pick the one you wish. For this example, we will be using Google Drive, so we will search through the listed remote providers and select the number corresponding with Google Drive.

```
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
...
14 / Google Cloud Storage (this is not Google Drive)
  \ "google cloud storage"
15 / Google Drive
  \ "drive"
16 / Hubic
  \ "hubic"
...
```

- You will be prompted to enter your `client_id` that we created earlier in the guide. Please head over to your credentials page in your google developer console.
- Once you have followed the steps in the guide, copy your client ID and paste it into the terminal, next press **Enter**. Now copy your client secret, paste it in, and again press **Enter**.

```
Google Application Client Id

Setting your own is recommended.

See https://rclone.org/drive/#making-your-own-client-id for how to create your own.

If you leave this blank, it will use an internal key which is low performance.

Enter a string value. Press Enter for the default ("").

client_id> example12345

Enter a string value. Press Enter for the default ("").

client_secret> example12345
```

- Choose the scope you wish to give to rclone. Full Access is safe and is likely the most useful one to you, so in this case we will type **1** then press **Enter**.

```

Scope that rclone should use when requesting Access from Drive.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value
1 / Full access to all files, excluding Application Data Folder.
  \ "drive"
2 / Read-only access to file metadata and file contents.
  \ "Drive.readonly"
  / Access to files created by rclone only.
3 | These are visible on the drive website.
  | File authorization is revoked when the user deauthorizes the app.
  \ "Drive.file"
  / Allows read and write Access to the Application Data folder.
4 | This is not visible in the drive website.
  \ "Drive.appfolder"
  / Allows read-only Access to file metadata but
5 | does not allow any access to read or download file content.
  \ "drive.metadata.readonly"

scope> 1

```

- Unless you know what you are doing, leave the `root_folder` blank and press **Enter**. Leave Service Account Credentials JSON file path blank also, again press **Enter**. Then type `n` to choose to not edit advanced config and press **Enter**.

The ID of the root folder  
Leave blank normally.  
Fill in to access "Computers" folders. (see docs).  
Enter a string value. Press Enter for the default ("").  
`root_folder_id>`

Service Account Credentials JSON file path  
Leave blank normally.  
Needed only if you want use SA instead of interactive login.  
Enter a string value. Press \*\*Enter\*\* for the default ("").  
`service_account_file>`

Edit advanced config? (y/n)  
y) Yes  
n) No  
`y/n> n`

- Type `n` to choose to not use auto-config and press **Enter**.

- Next, you will be provided with an rclone command that you must execute on your local PC.
- Copy the `rclone authorize "drive" "your_token"` command from your Ultra.cc terminal and paste it on your local PC's terminal. This will open your browser automatically.
  - Refer to the earlier section of installing rclone on your local PC if required.
- Choose the Google Drive account you wish to use and click **Allow**, giving rclone permission to use it.
- Go back to the terminal of your local PC. Rclone will have generated a code, which you must copy.
- Paste the code in your Ultra.cc terminal and press **Enter**.
- Finally, type `n` to choose to not configure as a team drive and press **Enter**.

Use auto config?

\* Say Y if not sure  
 \* Say N if you are working on a remote or headless machine

y) Yes (default)

n) No

y/n> n

Option config\_token.

For this to work, you will need rclone available on a machine that has a web browser available.

For more help and alternate methods see: [https://rclone.org/remote\\_setup/](https://rclone.org/remote_setup/)

Execute the following on the machine with the web browser (same rclone version recommended):

`rclone authorize "drive"`

"eyJjbGllbnRfaWQiOiI50TIyMjQwOTkzNTYtMWtsZmJrYnFjamE5YWtqdmJiZTZsNGgwdHFhc25pZXEuYXBw"

Then paste the result.

Enter a value.

`config_token>`

- You will be shown a confirmation screen. If all is OK, type `y` and then press **Enter** to save your configuration.
- If you notice any issues, you can edit them from here by typing `e`, or delete them using `d`. Finally, press `q` and then **Enter** to quit the rclone config wizard.

```
-----
[gdrive]
type = drive
client_id = 78311[REDACTED].apps.googleusercontent.com
client_secret = 034[REDACTED]
scope = drive
token = {"access_token": [REDACTED]}
team_drive =
-----
```

y) Yes this is OK (default)  
 e) Edit this remote

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

# Usage

rclone is interacted with purely through SSH. Please ensure you are familiar with the Linux terminal and using SSH prior to using rclone.

## General Commands

These commands are helpful to remember. They allow you to interact with rclone and move files around between your local and remote storage or even between two remote destinations.

config - Execute this command to add, modify or remove remote file hosts.

Usage: `rclone config`

copy - Used to copy files between two locations, remote -> remote, remote -> local, local -> remote

Usage: `rclone copy [-P] {origin} {destination}`

move - Same as copy however, it does not leave the files at the source

Usage: `rclone move [-P] {origin} {destination}`

sync - Will make the destination directory identical to the origin. If files exist on the destination that does not exist on origin, they will be deleted. Be careful with the sync command, as it can cause data loss.

Usage: `rclone sync [-P] {origin} {destination}`

When dealing with remote filesystems, use:

{remote}: {path}

For example, if you wished to copy a file named `movie.mkv` from your current working directory to a path named `Movies` in a remote name `Drive` you'd use this command:

`rclone copy movie.mkv Drive:Movies`

Revision #47

Created 6 September 2019 13:41:39

Updated 11 December 2023 20:43:37 by iStanCFC