**Digital Signatures and Work Instruction Report**

**Computer Security & Cryptography**

**Louise Findlay (1905807)**

**Digital Signature**

A Digital Signature, also known as a message authentication code (MAC), is a signature made from an algorithm. It is created by taking a piece of the message (plaintext) and hashing it (message digest) and encrypting it with a private key. (McDermott 2020) It’s a cryptographic technique used to authenticate and encrypt messages.

Once the message has been sent and the signature attached, the receiver will open the message. Using the same message digest and the public key (the counterpart to the sender’s private key), the sender will decrypt the digital signature. If the decrypted signature is the same as the message digest, then the message was definitely sent by the alleged sender and not from a third-party pretending to be them. (McDermott 2020)

The advantage of using digital signatures is that the signature cannot be stolen, forged or the message altered by a malicious attacker. (McDermott 2020)

OpenPGP is an open-source version of PGP (Pretty Good Privacy) which is an application used to encrypt data using hashing, compression and cryptographic techniques such as public and symmetric key combined with digital signatures. (Cyber Defence Magazine 2017) OpenPGP provides five features: authentication, confidentiality, compression, email-compatibility and segmentation and reassembly. (McDermott 2020)

Confidentiality is achieved through a combination of symmetric-key and public-key cryptography. First the message in encrypted by symmetric encryption using a session key unique to the message sent. Then the session key is encrypted with the public key using public-key cryptography which forms the start of the message. The rest of the message contains the plaintext encrypted by the session key. Finally, the message is compressed (if needed) and sent. (Callas et al. 1998)

On receipt of the encrypted message, it’s decrypted using the private key and then the session key. Finally, it’s decompressed whether it’s required. (Callas et al. 1998)

Compression is used on most emails sent using OpenPGP and has the handy side-effect of protecting against altered data because decompression will fail if the data has been altered. (Callas et al. 1998)

A variety of algorithms are used to implement OpenPGP’s features. Asymmetric types include RSA and ElGamal, symmetric types include AES and CAST5, hash types include MD5 and SHA and zip and Radix-64 are used for compression and compatibility respectively. (McDermott 2020)

One of the advantages of using OpenPGP is that it requires little computing power to accomplish and isn’t complicated to setup between the sender and receiver every time they want to send an email to each other. (McDermott 2020)

One of the disadvantages of OpenPGP is that it is susceptible to malware because the encryption involved protects the email against anti-malware programs. (Jeong and Kang 2013)

To conclude, digital signatures are a method to ensure authenticity and encrypt messages which is secure because the message can’t be stolen or forged. OpenPGP is an application which provides a suite of cryptography tools to encrypt and decrypt emails which includes digital signatures, public-key and symmetric cryptography.

**References**

McDermott, R., 2020. Message Authentication, Digital Signatures and Hybrid Cryptosystems, CM3104. [PowerPoint presentation]. Message Authentication, Digital Signatures and Hybrid Cryptosystems. Computer Security and Cryptography. Robert Gordon University, School of Computer Science and Digital Media, 18th March 2020 <http://campusmoodle.rgu.ac.uk/pluginfile.php/4899423/mod_resource/content/2/L15%20-%20Digital%20Signatures%20and%20Hybrid%20Cryptosystems.pdf> [Accessed 18th April 2020].

Cyber Defence Magazine, 2017. PGP, OpenPGP, and GPG: What They Are and When to Use Them. [online]. New York: Gary S. Miliefsky. Available from: <https://www.cyberdefensemagazine.com/pgp-openpgp-and-gpg-what-they-are-and-when-to-use-them> [Accessed 18th March 2020].

Callas, J., Donnerhacke, L., Finney, H. and Thayer, R., 1998. OpenPGP message format. RFC 2440, November.

Jeong, Y.S. and Kang, S.G., 2013, January. E-mail encryption methods and lawful interception methods of it. In 2013 15th International Conference on Advanced Communications Technology (ICACT) (pp. 29-32). IEEE.

**Work Instruction – Set up Rclone**

**Applies to:**

End users using a Linux operating system and Google Drive cloud storage

**Objective:**

To configure the Rclone package in order to access Google Drive storage on Linux machines

**Pre-Requisites**

The end user must already have a Google account connected

The end user must have files already stored in Google Drive

**Date**

10th April 2020

**Version History**

1.0 – 10th April 2020

**Author**

Louise Findlay

**Introduction**

Rclone is a command-line tool used on Linux operating systems in order to connect remotes (cloud storage providers) to the local filesystem. This tool is vital for remotes where Linux isn’t natively supported, and it also provides advanced functionality such as syncing which Google Drive doesn’t support for Linux. This work instruction will detail how to setup Rclone and configure a Google Drive remote.

**Step 1 – Open the Terminal**

**Explanation**

Since Rclone is a command-line application, you will need to open a terminal window.

**Instruction**

On GUI systems, often a terminal shortcut will be present on the desktop. For systems without a GUI, this step is unnecessary as the terminal will be opened on boot.

**Expected Result**

A terminal window is open

**Step 2 – Download Rclone**

**Explanation**

Next, you will need to download Rclone. Using the command below, will automatically install the latest version of Rclone and the one that is correct for your distribution and architecture of your machine. (Craig-Wood, 2014)1

There are security implications of installing Rclone this way because the script could be malicious but it’s much easier to do so because you don’t have figure out the architecture your computer is running on, download the correct version, unzip the file and move it to the correct location.

**Instruction**

Type in curl https://rclone.org/install.sh | sudo bash and press enter

**Expected Result**

Rclone is installed

**Step 3 – Configure Rclone**

**Explanation**

Now you need to configure rclone. There are two ways to do this. You can create a config file with all the relevant parameters or use the interactive mode where it will provide default options and prompt you for the relevant information.

Since the interactive mode is easier, that is the way this guide will follow.

**Instruction**

Type in rclone config to start the interactive configurator.

**Expected Result**

Text appears which says press n to make a new remote

**Step 4 – Create New Remote**

**Explanation**

Since you’ve never installed Rclone before, there won’t be anything configured so you need to make a new remote.

**Instruction**

Press n to create a new remote

**Illustration**



**Expected Result**

A prompt for the name of the remote appears

**Step 5 – Name New Remote**

**Explanation**

Next, you need to name the remote. This is just for the purposes of differentiating it from other remotes you may create in the future so it can be anything you want.

**Instruction**

Type in the new of the remote and press enter

**Illustration**



**Expected Result**

A list of storage providers appears

**Step 6 – Choose Storage Provider**

**Explanation**

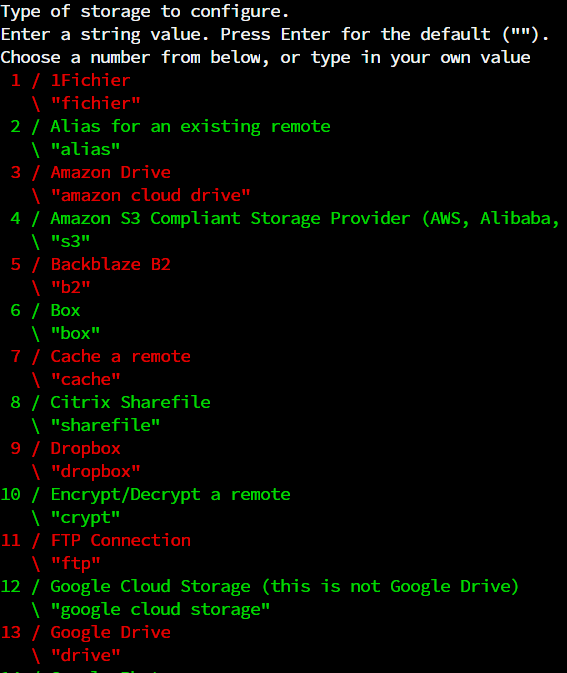
Rclone supports many storage providers so you need to specifically choose Google Drive.

In the example below, 13 is the correct number, though this can change when Rclone adds support for more remotes

**Instruction**

Type in the number corresponding to Google Drive and press enter

**Illustration**



**Expected Result**

A prompt to enter your Google Application Client ID appears.

**Step 7 – Create Google Drive Client ID**

**Explanation**

Rclone provides a default Client ID for Google Drive but it is susceptible to rate limiting because so many users use it instead of creating their own. Having your own Client ID is best practise so this guide will show you how to do so. (Craig-Wood, 2014)2

**Instruction**

Open up a web browser such as Mozilla Firefox and enter this URL, <https://console.developers.google.com/> (Craig-Wood, 2014)3

**Illustration**



**Expected Result**

The Google API Console webpage is open in your web browser

**Step 8 – Create Google API Project**

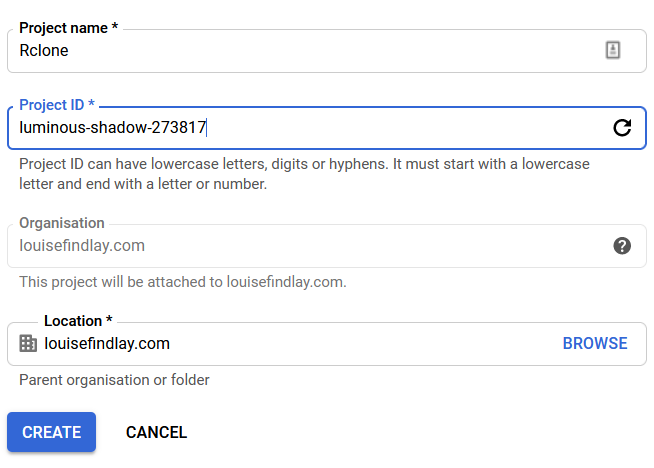
**Explanation**

In order to get a Client ID, you need to create a project to attach it to.

**Instruction**

Click Create A New Project, fill in the project name with a title such as Rclone and click the blue Create button.

**Illustration**



**Expected Result**

A new Google API project has been created

**Step 9 – Enable Google Drive API**

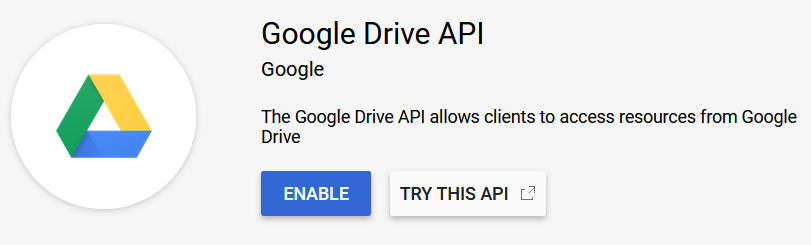
**Explanation**

Once the Google API project has been created, you now need to enable the Google Drive API because that’s what you’re trying to connect to.

**Instruction**

Click Enable APIs & Services, search for Google Drive, click Google Drive API and then click the blue Enable button.

**Illustration**



**Expected Result**

The Google Drive API has been enabled for your API project and the Google Drive API overview page appears.

**Step 10 – Create Google Drive API Credentials**

**Explanation**

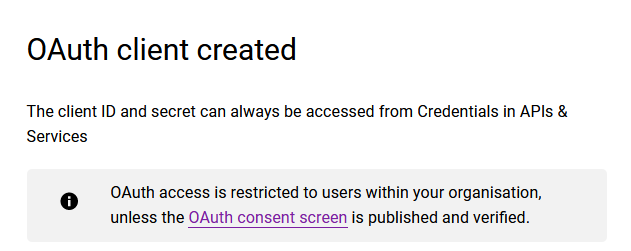
Now that the Google API project has been connected to Google Drive, the credentials (including the Client ID) can be created.

**Instruction**

Click Credentials in the sidebar, then OAuth Consent Screen. Select Internal as the Application type, enter anything you prefer for the Application name and click the blue Save button.

Go back to the Credentials page and click Create Credentials and OAuth Client ID. Set the Application type as Other, name it however you like and click the blue Create button.

**Illustration**



**Expected Result**

An OAuth client created popup should appear with your Client ID and Client Secret.

**Step 11 – Continuing Rclone Configuration**

**Explanation**

Now you have your Google Drive API Credentials, you can continue to configure your Google Drive remote in Rclone.

The scope of Rclone defines the permissions that Rclone has to read and write your files on Google Drive.

The root folder id and service account file are advanced options that you can leave as default and you don’t need to edit the advanced config.

**Instruction**

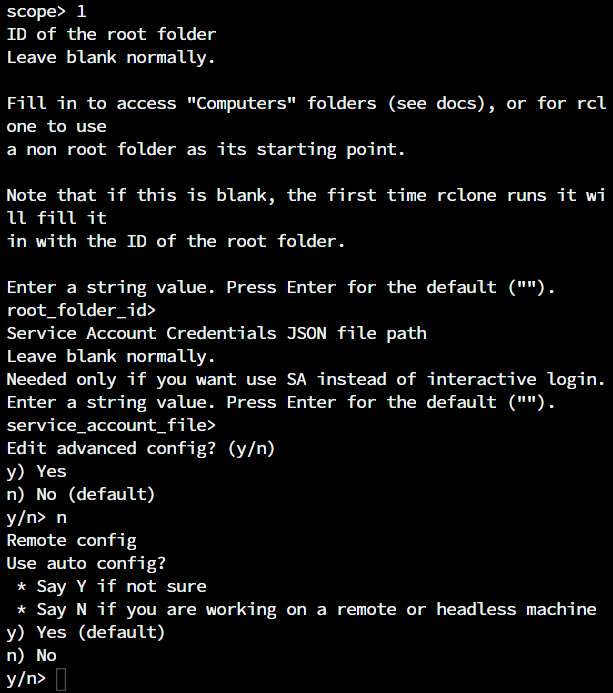
Copy and paste your Client ID into the terminal windows. Press enter and do the same for your Client Secret.

When you are prompted for the scope, type 1 and then press enter.

For the next two options, press enter.

When asked if you want to edit advanced config, type n and press enter.

**Illustration**



**Expected Result**

You are prompted whether to use auto config or not

**Step 12 – Remote Configuration**

**Explanation**

Now you are prompted where to use auto config or not. If you are using a GUI system (had to open the terminal at the start of this guide), then you should type y and press enter. If you didn’t or are connecting remotely to a computer, then you should type n and press enter. This is because the following steps require an internet browser.

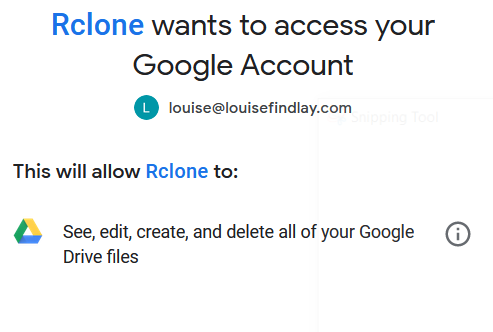
Since the test machine for this guide is a remote computer, the following instructions will follow a non-auto config.

**Instruction**

Type n and press enter.

Copy and paste the link into your web browser. If you used a keyboard shortcut (CTRL + C) and as a result exited the process, repeat steps 1-6 and 11.

**Illustration**



**Expected Result**

You should be presented with a Rclone wants to access your Google Account screen.

**Step 13 – Allow Rclone Access to Google Drive**

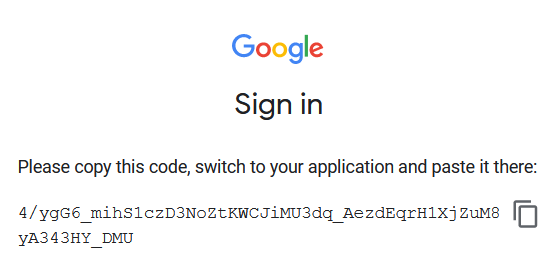
**Explanation**

Next, you need to give Rclone permission to access your Google Drive and copy and paste the verification code back in the terminal window.

**Instruction**

Click the blue Allow button, copy and paste the verification code into the terminal and press enter.

**Illustration**



**Expected Result**

You should be presented with an option of whether to configure a team drive or not.

**Step 14 – Finish Configuring Google Drive Remote**

**Explanation**

Is the Google Drive a team drive (shared drive) or your own personal drive? If someone set it up for you, then it’s a team drive (now called shared drive), otherwise it’s not. (Google, n.d.)4

Once you’ve done that then your Google Drive Remote has been setup.

To test this, you can list the files and folders you have in Google Drive.

**Instruction**

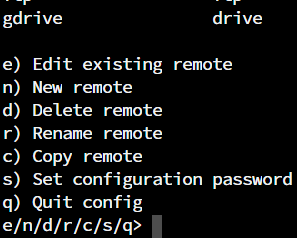
If you’re setting up a team drive, type y and press enter.

If not, type n and press enter.

Then, type y and then enter.

To test, type q and then enter to exit the configuration editor. Finally, type rclone lsd (name of remote):

**Illustration**



**Expected Result**

A remote of type drive should be listed and you should see the root level of your Google Drive storage.

**Work Instruction – Upload/Sync Files to Google Drive using Rclone**

**Applies to:**

End users using a Linux operating system and Google Drive cloud storage

**Objective:**

To use Rclone to upload files to Google Drive

**Pre-Requisites**

The end user must already have a Google account connected

The end user must have files already stored in Google Drive

Rclone must be installed on the system (see Work Instruction – Setting up Rclone)

A Google Drive remote must be set up (see Work Instruction – Setting up Rclone)

**Date**

13th April 2020

**Version History**

1.0 – 13th April 2020

**Author**

Louise Findlay

**Introduction**

Rclone is a command-line tool used on Linux operating systems in order to connect remotes (cloud storage providers) to the local filesystem. This tool is vital for remotes where Linux isn’t natively supported, and it also provides advanced functionality such as syncing which Google Drive doesn’t support for Linux. Syncing is a vital feature because it allows data to be accessed offline which is necessary if your data needs to be available at all times. This work instruction will show how to upload files to Google Drive using Rclone. It will also cover syncing since the commands are similar.

**Step 1 – Open the Terminal**

**Explanation**

Since Rclone is a command-line application, you will need to open a terminal window.

**Instruction**

On GUI systems, often a terminal shortcut will be present on the desktop. For systems without a GUI, this step is unnecessary as the terminal will be opened on boot.

**Expected Result**

A terminal window is open.

**Step 2 – Copy/Move/Sync the Files**

**Explanation**

Now you just need to upload the files. Depending on whether you want to retain the files in the local filesystem, you can copy or move them using rclone copy (Craig-Wood, 2014)5 or rclone move. (Craig-Wood, 2014)6

If you want to sync files from the cloud to your local filesystem, then use rclone sync (Craig-Wood, 2014)7 instead.

The -v flag sets the logging level to verbose, so you know what’s going on. Optionally, you can move logging to a file using the –log-file flag with the parameter of a file location such as /opt/rclone.log.

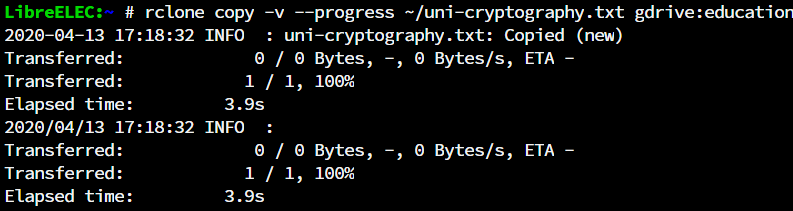
The –progress flag lets you see the progress of the file transfer.

The process is the exact same for uploading folders.

**Instruction**

Type rclone copy/move/sync (depending of which you want) -v –progress (location of the file you want to upload) (remote name):(folder you want the file to be uploaded to).

**Illustration**



**Expected Result**

Rclone starts to upload the files/folders you have chosen, and you can see the progress as shown in the image above or a log file has been created if you have specified so.

**Step 3 – Check the Files Have Transferred**

**Explanation**

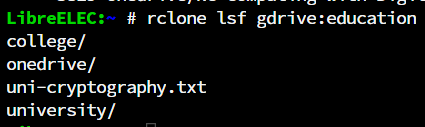
You can check that the files have transfers either through viewing the log or listing the files in the remote.

**Instruction**

If you have set the logging to be output to a file. Type cat and the path to the log file e.g. cat /opt/rclone.log.

To check that the files have uploaded in the remote. Type rclone lsf remotename:folder and press enter. (Craig-Wood, 2014)8

**Illustration**



**Expected Result**

The uploaded file is present in the list.

**Work Instruction – Set up Encryption with Rclone**

**Applies to:**

End users using a Linux operating system and have Rclone installed

**Objective:**

To set up an encrypted remote using Rclone

**Pre-Requisites**

Rclone must be installed on the system (see Work Instruction – Setting up Rclone)

**Date**

14th April 2020

**Version History**

1.0 – 14th April 2020

**Author**

Louise Findlay

**Introduction**

Rclone is a command-line tool used on Linux operating systems in order to connect remotes (cloud storage providers) to the local filesystem. This tool is vital for remotes where Linux isn’t natively supported, and it also provides advanced functionality such as syncing which Google Drive doesn’t support for Linux. This work instruction will show how to encrypt your files using Rclone by setting up a folder with everything’s encrypted to securely store the files that need it while still being able to access your other files freely through the web interface.

**Step 1 – Open the Terminal**

**Explanation**

Since Rclone is a command-line application, you will need to open a terminal window.

**Instruction**

On GUI systems, often a terminal shortcut will be present on the desktop. For systems without a GUI, this step is unnecessary as the terminal will be opened on boot.

**Expected Result**

A terminal window is open.

**Step 2 – Create Crypt Remote**

**Explanation**

To encrypt your files, you need to create another remote which will do the encryption process. There are different encryption settings you can choose from and this guide will implement the strongest available. (Craig-Wood, 2014)9

The remote to encrypt/decrypt should be the name of the remote you created in Work Instruction – Setting up Rclone and the path should be the name of the folder you want to store the encrypted files e.g. encrypted or secret. All your encrypted files will be stored there and can only be decrypted if you setup Rclone on the device with the same credentials.

**Instruction**

Type rclone config and press enter.

Type n and press enter.

Type a name for the remote and press enter. Common names are crypt or secret.

Type the number for encrypt/decrypt a remote and press enter.

The location should be the name of the remote you created in the Work Instruction – Setting up Rclone, followed by a colon and the name of the encrypted folder you want to create and then press enter.

Type 1 and press enter.

Type 1 and press enter.

Type g and press enter.

Type 1024 and press enter.

Make sure to copy the password to a secure location otherwise you won’t be able to unencrypt your files on another device or if you delete Rclone.

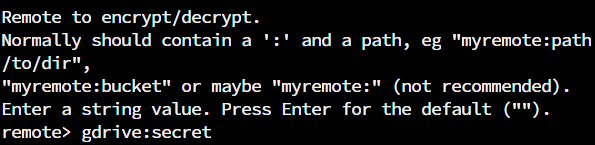
Type y and press enter.

Similar to the process above, repeat the last four instructions.

Type y and press enter.

Finally, type q and press enter to exit the configurator.

**Image**



**Expected Result**

A crypt remote has been created

**Step 3 – Upload Files to the Crypt Remote**

**Explanation**

Now that the crypt remote has been created, you need to upload some files to it.

This is a similar process as Work Instruction – Upload/Move/Sync Files to Google Drive using Rclone.

**Instruction**

Type rclone copy (file-name) (encrypted-remote name): and press enter.

**Image**



**Expected Result**

The specified file has been moved to the encrypted remote

**Step 4 – Check Files Have Been Uploaded to Encrypted Remote**

**Explanation**

There are two ways to check if files have been uploaded to the encrypted remote.

You can use the normal gdrive remote and look into the encrypted folder which will show the encrypted version of the files.

If you want to view the unencrypted files, then you would need to search the encrypted remote.

This again is a similar process as Work Instruction – Upload/Move/Sync Files to Google Drive using Rclone.

**Instruction**

To view the encrypted files, type rclone lsf (remote-name):(remote-path) and press enter.

To view the unencrypted files, type rclone lsf (encrypted remote-name): and press enter.

**Image**



**Expected Result**

The files you copied to the encrypted remote are displayed.

**Work Instruction – Mount a Google Drive Folder using Rclone**

**Applies to:**

End users using a Linux operating system and Google Drive cloud storage

**Objective:**

To use Rclone to mount a Google Drive folder to the local filesystem.

**Pre-Requisites**

The end user must already have a Google account connected

The end user must have files already stored in Google Drive

Rclone must be installed on the system (see Work Instruction – Setting up Rclone)

A Google Drive remote must be set up (see Work Instruction – Setting up Rclone)

**Date**

15th April 2020

**Version History**

1.0 – 15th April 2020

**Author**

Louise Findlay

**Introduction**

Rclone is a command-line tool used on Linux operating systems in order to connect remotes (cloud storage providers) to the local filesystem. This tool is vital for remotes where Linux isn’t natively supported, and it also provides advanced functionality such as mounting which Google Drive doesn’t support for Linux. Mounting is a vital feature because it allows you to access your Google Drive files as if they were local ones. This is similar to Windows’ mapped network drives (Microsoft, n.d.)10 but superior because Linux doesn’t classify it as a network drive and prohibit as Windows does.

**Step 1 – Open the Terminal**

**Explanation**

Since Rclone is a command-line application, you will need to open a terminal window.

**Instruction**

On GUI systems, often a terminal shortcut will be present on the desktop. For systems without a GUI, this step is unnecessary as the terminal will be opened on boot.

**Expected Result**

A terminal window is open.

**Step 2 – Create the Mounted Folder**

**Explanation**

You need to create a folder on the local filesystem to store your cloud files in.

Make sure to give the appropriate permissions so you can access the folder.

**Instruction**

Type mkdir (folder-name) and press enter.

Then type chmod 755 (folder-name) and press enter.

**Image**



**Expected Result**

A new folder has been created.

**Step 3 – Mount the Google Drive Remote in the Mount folder**

**Explanation**

Now the mounted folder has been created, you can now start the mount command.

The parameters used in the mount command are ones that work best for this particular setup. It’s best to experiment and try out different ones. For example, if you have gigabit upload speed, you might need to limit the maximum data transferred to prevent rate limiting (Google stops you transferring data if you do too much in one day.) (Craig-Wood, 2014)11

**Instruction**

Type rclone mount --dir-cache-time 96h --cache-info-age 100h --vfs-cache-mode writes --allow-other --log-level DEBUG --log-file /(folder-name)/rclone.log (remote-name): / /(local mount folder) and press enter. (Findlay, 2019)12

**Image**



**Expected Result**

The Google Drive mount has been mounted to the mounted folder.

**Step 4 – Open A New Terminal Window**

**Explanation**

Since the mount command is running, you need to open a new terminal window.

**Instruction**

Repeat the process you took in Step 1.

**Expected Result**

A new terminal window is open

**Step 5 – Check the Google Drive Files are Present in the Mounted Folder**

**Explanation**

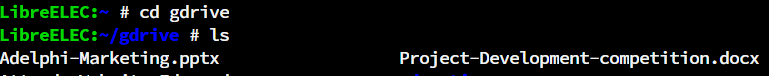
Finally, the last step is to check it has worked. The process is the exact same as you would list files in a local folder.

**Instruction**

Type cd (path to the mounted folder) and press enter.

Type ls and press enter.

**Image**



**Expected Result**

The Google Drive files are listed.

**Work Instruction – Uninstall Rclone**

**Applies to:**

End users using a Linux operating system and have Rclone installed

**Objective:**

To uninstall Rclone

**Pre-Requisites**

Rclone must be installed on the system (see Work Instruction – Setting up Rclone)

**Date**

13th April 2020

**Version History**

1.0 – 13th April 2020

**Author**

Louise Findlay

**Introduction**

Rclone is a command-line tool used on Linux operating systems in order to connect remotes (cloud storage providers) to the local filesystem. This tool is vital for remotes where Linux isn’t natively supported, and it also provides advanced functionality such as syncing which Google Drive doesn’t support for Linux. This work instruction will show how to uninstall Rclone if it is no longer needed.

**Step 1 – Open the Terminal**

**Explanation**

Since Rclone is a command-line application, you will need to open a terminal window.

**Instruction**

On GUI systems, often a terminal shortcut will be present on the desktop. For systems without a GUI, this step is unnecessary as the terminal will be opened on boot.

**Expected Result**

A terminal window is open.

**Step 2 – Remove Rclone Files**

**Explanation**

Uninstalling Rclone is just as easy as installing it. The process only takes two commands. (Craig-Wood 2018)13

**Instruction**

Type sudo rm /usr/bin/rclone and press enter

Type sudo rm /usr/local/share/man/man1/rclone.1 and press enter

**Expected Result**

Rclone is uninstalled

**References**

1. Craig-Wood, N., 2014. Rclone downloads. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/downloads> [Accessed 10th April 2020].

1. Craig-Wood, N., 2014. Google drive. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/drive/#making-your-own-client-id> [Accessed 10th April 2020].
2. Google, n.d. APIs & Services – Google API Console. [online]. Mountain View, California: Google. Available from: [https://console.developers.google.com](https://console.developers.google.com/) [Accessed 10th April 2020].
3. Google, n.d. What are shared drives? – Google Suite Admin Help. [online]. Mountain View, California: Google. Available from: <https://support.google.com/a/answer/7212025?hl=en> [Accessed 10th April 2020].
4. Craig-Wood, N., 2014.Rclone copy. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/commands/rclone_copy> [Accessed 13th April 2020].
5. Craig-Wood, N., 2014.Rclone move. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/commands/rclone_move> [Accessed 13th April 2020].
6. Craig-Wood, N., 2014.Rclone sync. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/commands/rclone_sync> [Accessed 13th April 2020].
7. Craig-Wood, N., 2014.Rclone ls. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/commands/rclone_ls> [Accessed 13th April 2020].
8. Craig-Wood, N., 2014. Crypt. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/crypt> [Accessed 13th April 2020].
9. Microsoft, n.d. Map a network drive. [online]. Redmond, Washington: Microsoft. Available from: <https://support.microsoft.com/en-gb/help/4026635/windows-map-a-network-drive> [Accessed 13th April 2020].
10. Craig-Wood, N., 2014.Rclone mount. [online]. Unknown: Nick Craig-Wood. Available from: <https://rclone.org/commands/rclone_mount> [Accessed 13th April 2020].
11. Findlay, L., 2019.Rclone Service Failed – question – rclone Forum. [online]. 19th July 2019. Rclone Forum. Available from: <https://forum.rclone.org/t/rclone-service-failed/10700> [Accessed 15th April 2020].
12. Craig-Wood, N., 2018. How do I completely uninstall/remove rclone on my Mac?– question – rclone Forum. [online]. 16th March 2018. Rclone Forum. Available from: <https://forum.rclone.org/t/how-do-i-completely-uninstall-remove-rclone-on-my-mac/5083> [Accessed 13th April 2020].