

Tiered Storage Benchmarking

Software Installation Instructions

Sara Willis

January 31, 2020

Contents

0.1	Globus Online Interface	1
0.2	Globus CLI - Permanent Endpoint	2
0.3	Cyberduck CLI	5
0.3.1	Usage	5
0.4	Cyberduck GUI	6
0.5	Gdrive	11
0.5.1	Usage	11
0.6	RClone	12
0.7	iRODS	18

0.1 Globus Online Interface

There are instructions on how to set up Globus and link it to endpoints in our [Online Documentation](#). Once you've gotten your endpoint(s) set up (either to the HPC filexfer node, your personal computer, or both), you can add your UofA Google Drive as an endpoint via the following steps:

1. Search for the endpoint **arizona#hpc-dtn1** and click the resulting endpoint

ENDPOINT	SIMPLIFIED	STRICT	STATUS	ROLE
arizona#hpc-dtn1	Managed Public Endpoint		GCSv5 Connector	GM

2. Under the **Collections** tab, select **Add a Collection**

COLLECTION NAME

3. Select Google Drive

Create a Guest Collection

- UA HPC Storage Gateway (POSIX)
- Google Drive Storage Gateway (Google Drive)
- UA S3 Gateway (S3)

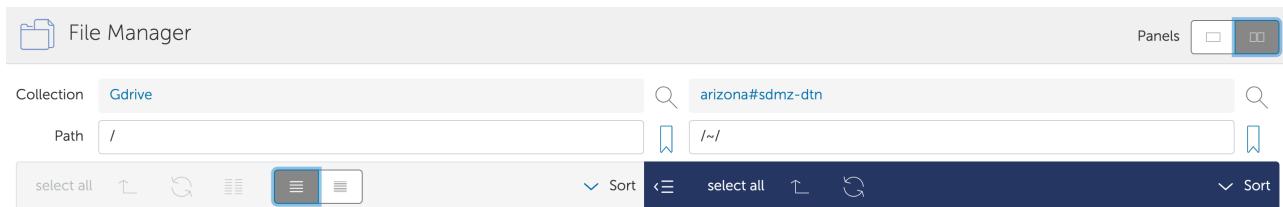
© 2019 University of Chicago [legal](#)

4. Name your **Collection Display Name** something descriptive and identifying for future use

Globus collection information:

Collection Display Name	Gdrive
Description	Shared data Project ABC
Keywords	genomics, Higgs boson, climate change

5. You can now make transfers to/from Google Drive! Set up transfers in the **File Manager**, accessible from the side menu



0.2 Globus CLI - Permanent Endpoint

On your personal computer, you can download and install globus using pip. You can initiate transfers between HPC and Google Drive/HPC and your PC from your own machine.

```
pip install --upgrade --user globus-cli
globus --help # Check that globus CLI has been installed
globus update # to update your version of the CLI to the latest
pip uninstall globus-cli # to remove CLI
```

```
globus transfer --no-verify-checksum <source_path+filename> <destination_path+filename>
```

To search for UofA's endpoint, use:

```
dhcp-10-132-178-181:~ sarawillis$ globus endpoint search arizona#sdmz-dtn
ID | Owner | Display Name
--- | --- | ---
27cf226c-5402-11e6-824b-22000b97daec | tmerritt@arizona.edu | arizona#sdmz-dtn
8c3f744e-baa8-11e9-9396-02ff96a5aa76 | 0df23eb0-3c61-4dd0-86ed-cec97c0f0aaf@clients.auth.globus.org | sdmz-dtn

dhcp-10-132-178-181:~ sarawillis$UAGlobus=27cf226c-5402-11e6-824b-22000b97daec
(base) dhcp-10-132-178-181:~ sarawillis$ globus endpoint show $UAGlobus
\Display Name: arizona#sdmz-dtn
ID: 27cf226c-5402-11e6-824b-22000b97daec
Owner: tmerritt@arizona.edu
Activated: False
Shareable: True
Department: None
Keywords: None
Endpoint Info Link: None
Contact E-mail: uits-hpc-team@list.arizona.edu
Organization: University of Arizona
Department: None
Other Contact Info: None
Visibility: True
Default Directory: /~/
Force Encryption: False
Managed Endpoint: True
Subscription ID: 50762b6c-44e0-11e9-a618-0a54e005f950
Legacy Name: u_ynojplx3nmi6llt4wmzmpoaqcq#sdmz-dtn
Local User Info Available: True
(base) dhcp-10-132-178-181:~ sarawillis$ globus ls $UAGlobus:\~/CyberduckProfilingTest/
GdriveProfilingTest/
GlobusProfilingTest/
Intro_to_HPC/
ProgrammingSandbox/
PythonTests/
R/
RcloneProfilingTest/
Slurm_Hello_World/
ZippedExecutables/
bin/
include/
lib/
lib64/
miniconda3/
mpi_hello_world/
ondemand/
share/
InstallFSL.sh
Transfer.pbs
TransferToGoogleDocs.py
UploadTests_gdrive_2019-08-19 19:31:03.581258.csv
UploadTests_gdrive_2019-08-29 13:41:10.123468.csv
fslinstaller.py
osacc
restats
testInstall.r
```

There are instructions available that document how to use a virtual environment to install Globus: <https://docs.globus.org/cli/installation/virtualenv/>.

0.3 Cyberduck CLI

These instructions are for users who do not have root privileges and who want to set up their own personal copy of duck.

1. Go to https://repo.cyberduck.io/stable/x86_64/
2. Download the latest version
3. sftp to filexfer node and put file into bin. In my case, I made a directory called Duck in bin where I stored the zipped file. Unzipping duck generates a lot of files and keeping them partitioned isn't a bad idea.
4. Unpack the rpm file and add the executable to your PATH variable. For details, see Installing Software on HPC

0.3.1 Usage

Once the program has been unpacked and the system knows where to find the program, you can set it up to connect with your personal Google Drive account.

You should only need to connect it to your personal Google Drive account the first time you use it. An example is shown below:

1. In the terminal, specify the direction of your file transfer and any additional options

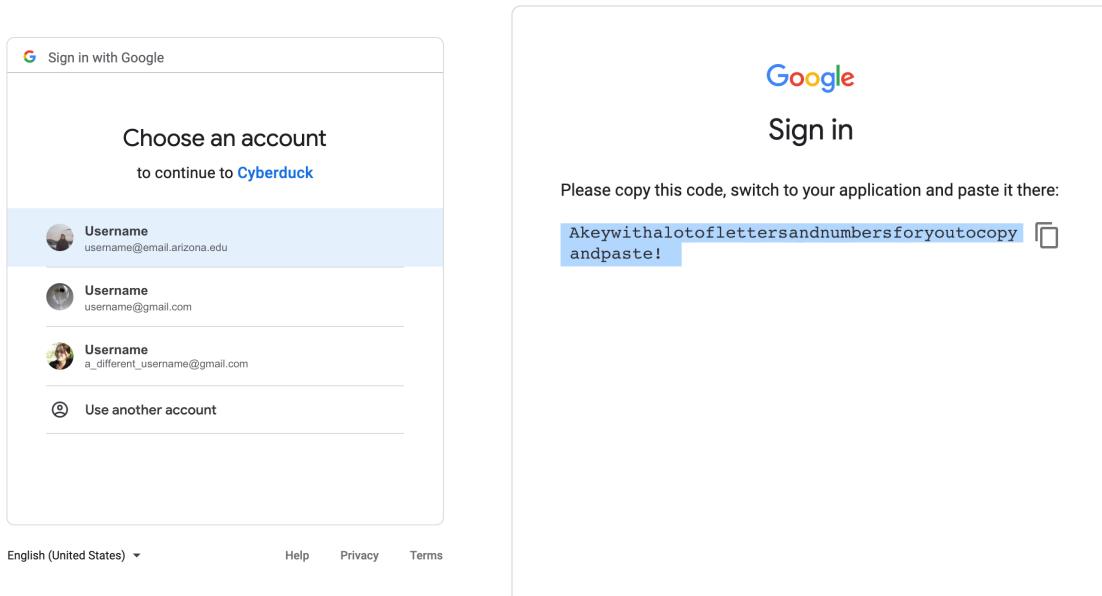
```
duck --username NetID@email.arizona.edu --download "googledrive:My Drive/<remote_filename>" \
<local_filename>
```

2. Copy/paste the url the program gives you into your browser

```
Resolving www.googleapis.com...GLib-GIO-Message: 00:00:00.000: Using the 'memory' GSettings
backend. Your settings will not be saved or shared with other applications.
Google Drive connection opened...

https://accounts.google.com/o/oauth2/auth?client_id=0000000000000000.apps.googleusercontent.com&
redirect_uri=urn:ietf:wg:oauth:2.0:oob&response_type=code&scope=https://www.googleapis.com
/auth/drive&state=00000000
```

3. Select the relevant gdrive account and copy the key that appears in the browser



4. Paste the copied key into the terminal. Note: you will not see any characters when you paste the key in. This is normal behavior and everything is working.

```
OAuth2 Authentication. Paste the authentication code from your web browser.
Authentication Code:
```

5. If you don't want to authenticate again in the future, save the password with "y" and always use the option --username NetID@email.arizona.edu.

```
WARNING! Passwords are stored in plain text in ~/.duck/credentials.
Save password (y/n): y
```

For more information on Duck usage:

```
duck --help
```

Warning: When downloading files, if you only specify the filename without including a path, it will save to the app folder under opt/duck/ which already has a large number of files in it. Specifying the absolute path with the filename will avoid this problem. You can also specify the relative path, but you will need to remember that the path is relative to the opt/duck directory and not your working directory.

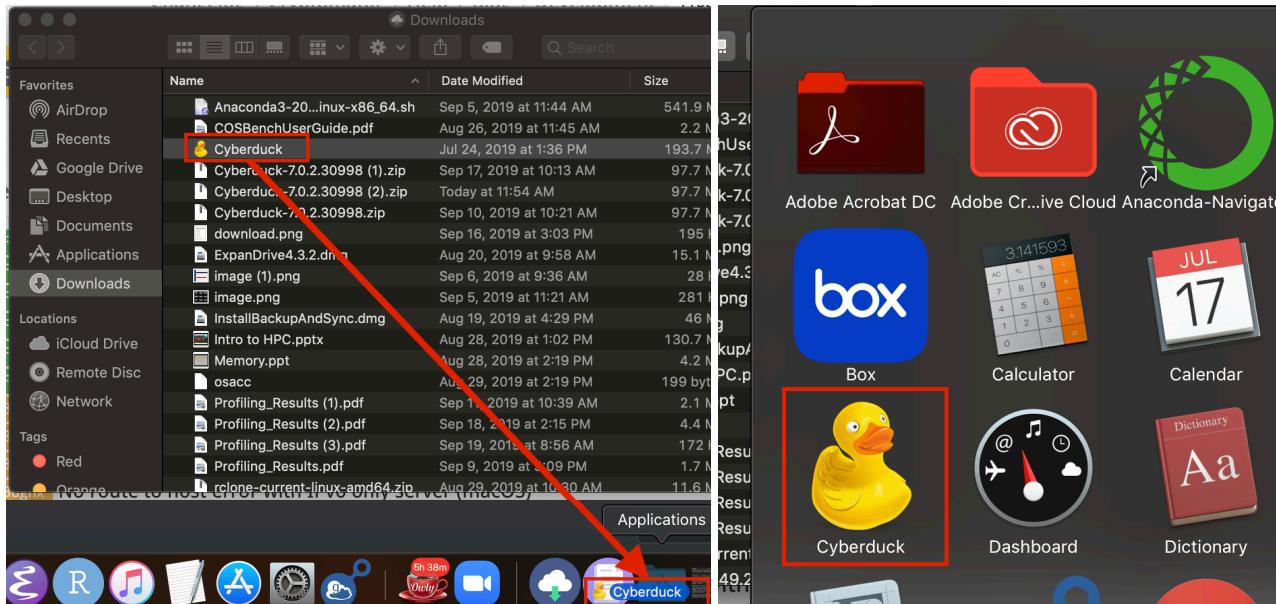
Warning: When downloading, even if you include the --existing overwrite option, the file that you're going to overwrite isn't immediately overwritten. Cyberduck creates a folder where it downloads your file in chunks and once all the chunks have been downloaded, they then get concatenated into the file that will overwrite the previously-existing one. This means that if you're low on space where you're saving your file, you may run exceed your storage quota.

0.4 Cyberduck GUI

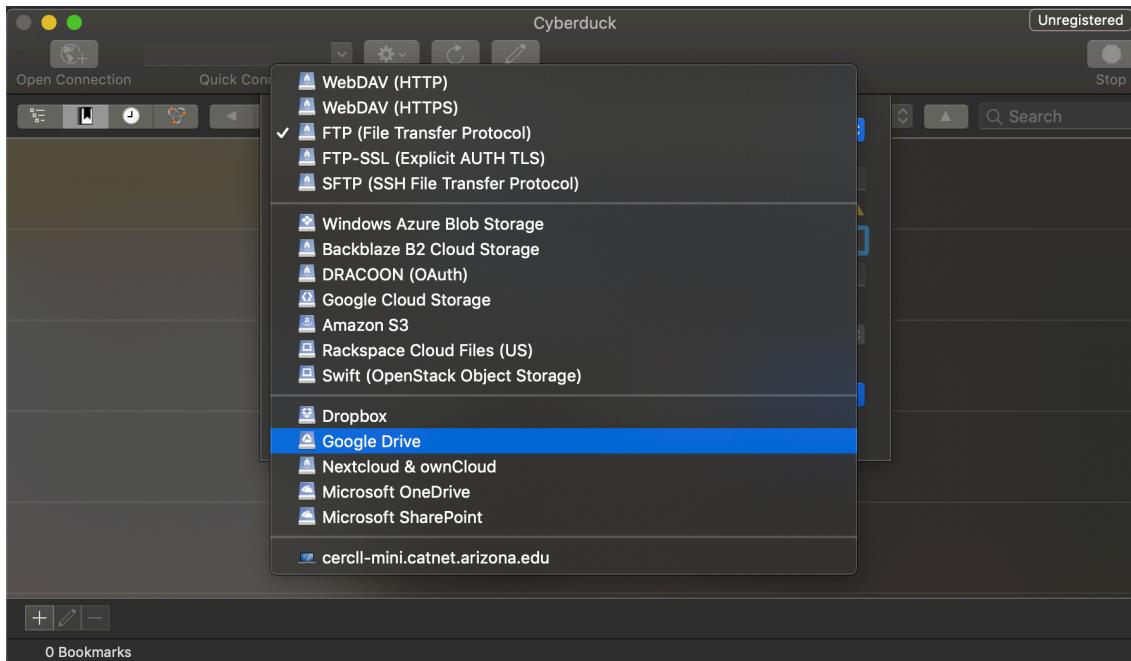
Mac

Installation is fairly simple. Various versions of Cyberduck are [available for download](#). Until the current bug is fixed that successfully allows for Duo Mobile authentication, I recommend using the previous version 7.0.2.

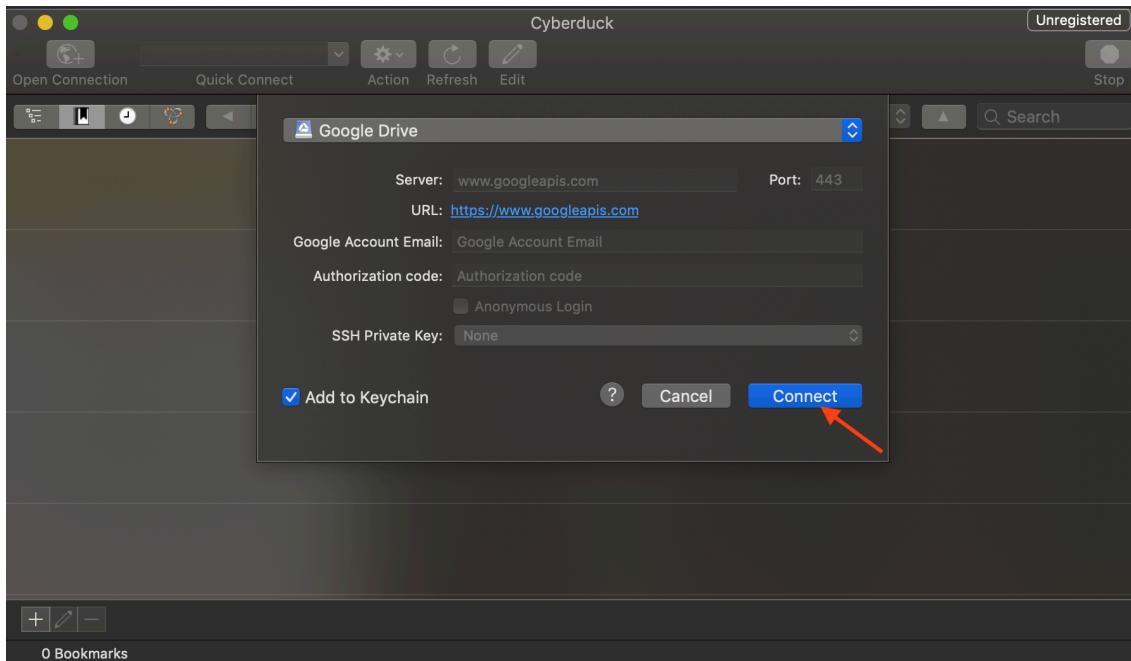
1. Download the zipped file, unpack, drag/drop the program into your applications folder, and double-click the icon to start Cyberduck.



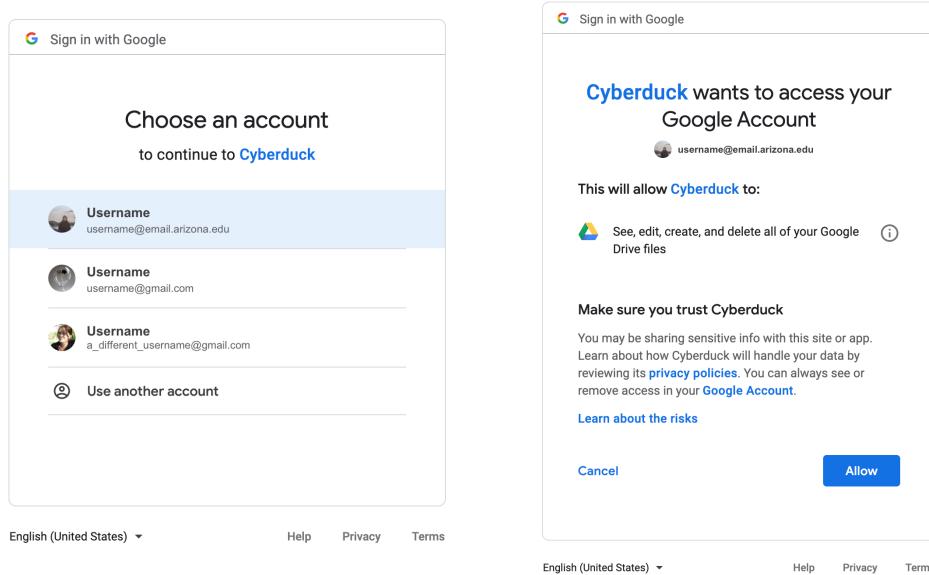
2. To establish a connection with Google Drive, use the drop-down menu by clicking “Open Connection”.



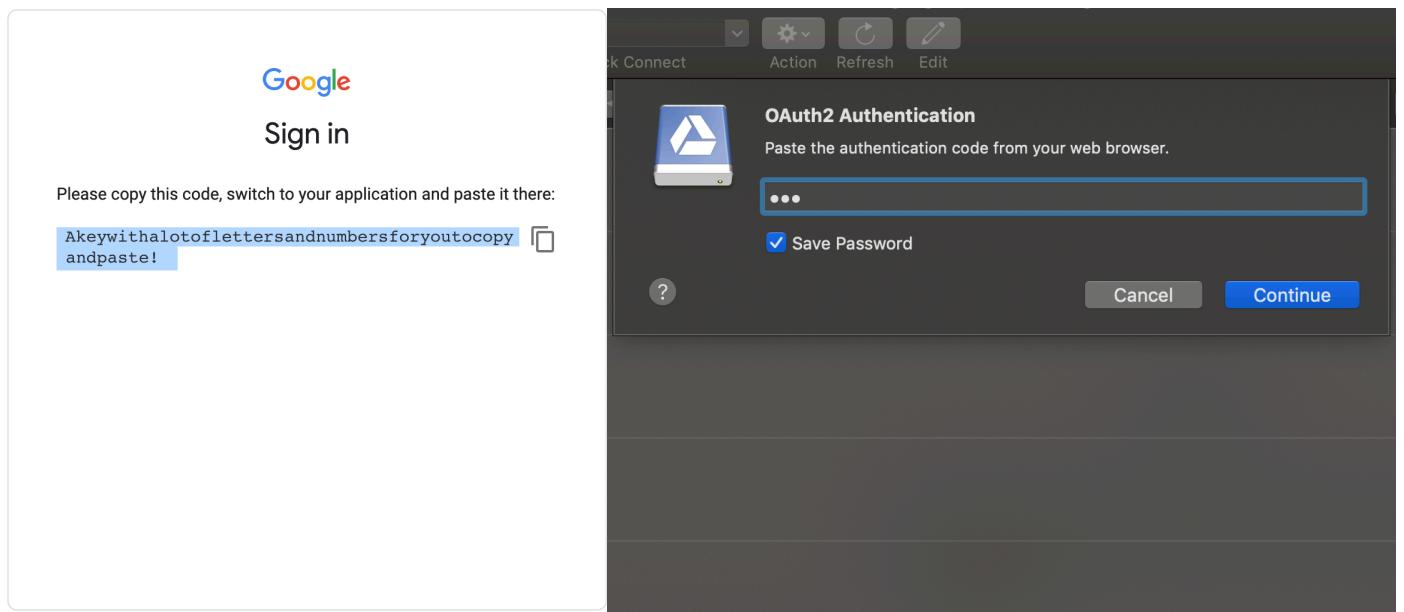
3. The default settings do not need to be edited; click Connect to proceed.



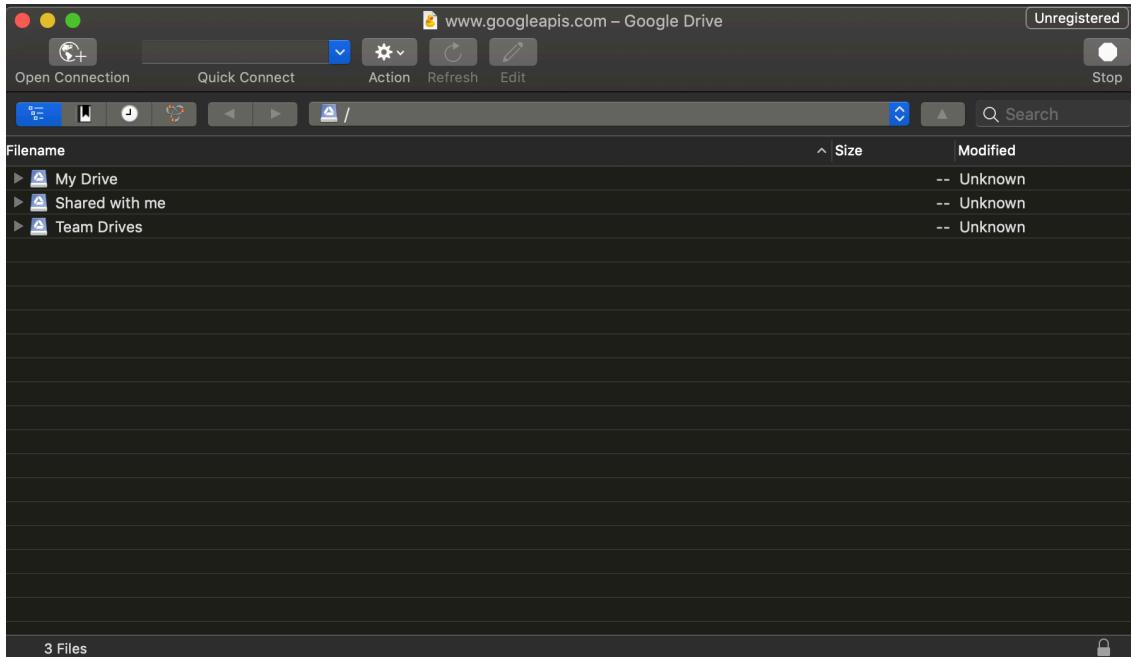
4. Select the account you'd like to connect with and grant Cyberduck access



5. Copy and paste the code that appears in your browser

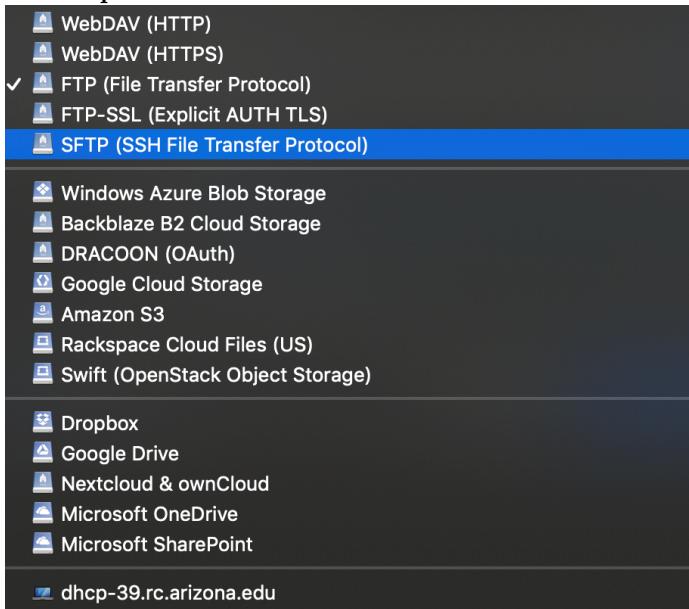


6. You are now connected to Google Drive!



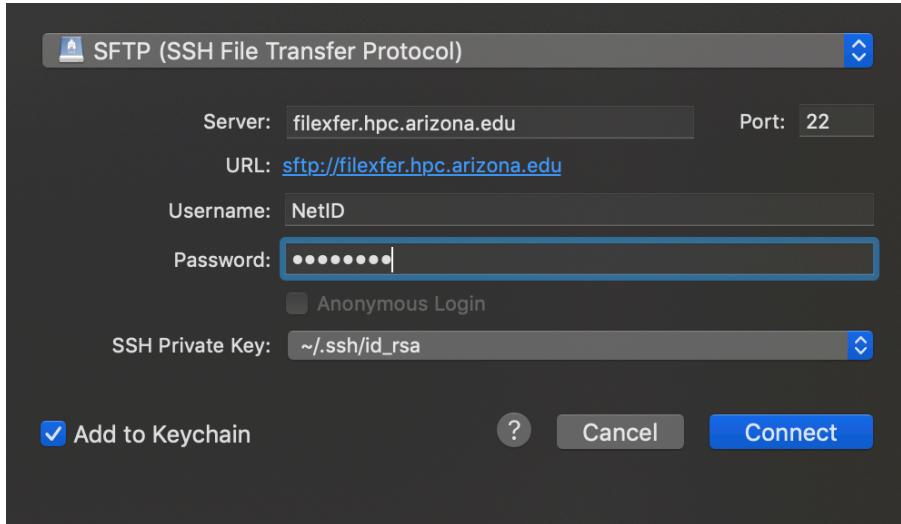
To make transfers between your personal computer and your Google Drive account, you can simply drag-and-drop files. To transfer files between Google Drive and HPC, you will need to open a second window to connect to HPC's filexfer node:

1. Go to file and select **New Browser**
2. Under **Open Connection** select **SFTP**

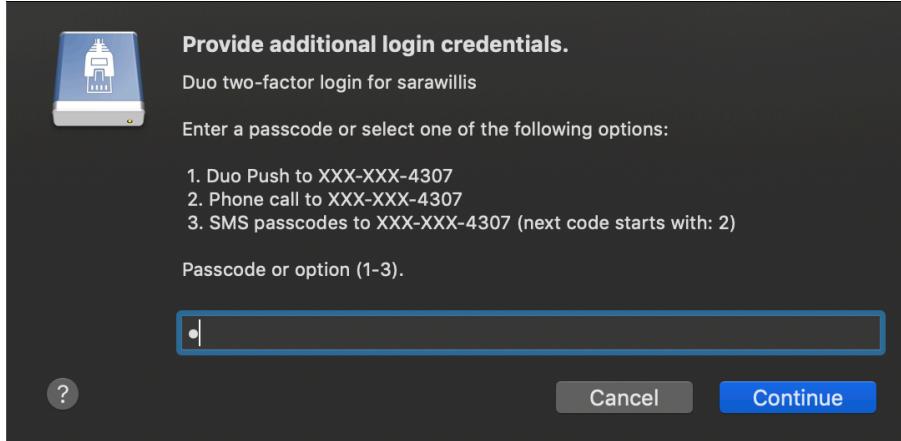


3. Enter your credentials to connect with the filexfer node:

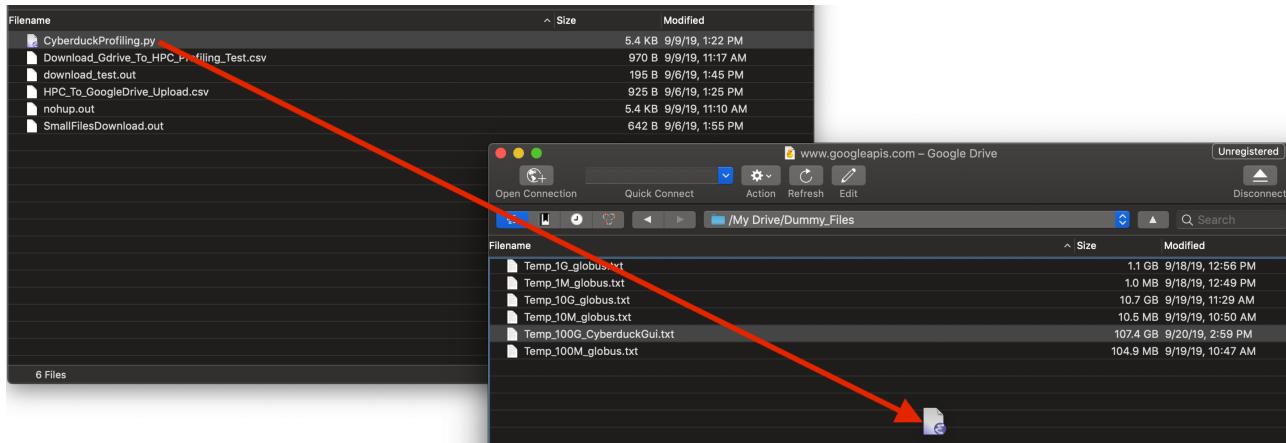
- Server: filexfer.hpc.arizona.edu
- Port: 22 (default)
- Username: <NetID>
- Password: <Your password>



4. You will be prompted to enter an option for two-factor authentication



5. You are now connected! To initiate a transfer, drag-and-drop files between the Google Drive and filexfer windows



0.5 Gdrive

There currently may be technical issues for new installations of Gdrive: <https://github.com/gdrive-org/gdrive/issues/506>. If you experience this issue and have an older installation of gdrive (e.g. on your personal computer), you can copy over the working .gdrive directory (in your home directory). This hidden folder has a file in it that provides authentication.

Gdrive's Github gives a good overview of the installation process for various operating systems here: <https://github.com/gdrive-org/gdrive>

To run Gdrive on HPC, download the Linux 64-bit file, transfer it to HPC, and add executable permissions:

```
chmod +x gdrive
```

You may also want to change the filename to something like `Gdrive`.

0.5.1 Usage

The command

```
gdrive help
```

gives a list of relevant commands. When used in conjunction with a command, gdrive prints specific instructions for a command, e.g.:

```
(base) cc-ea-lafrese:Desktop sarawillis$ gdrive help download
Download file or directory
gdrive [global] download [options] <fileId>

global:
  -c, --config <configDir>           Application path, default: /Users/sarawillis/.gdrive
  ...
```

0.6 RClone

Rclone setup requires some additional steps if you want to use your own key. I'll include instructions on that below. Fortunately, Rclone has good documentation.

1. Download the .rpm file from <https://rclone.org/downloads/>
2. Use sftp to transfer rpm file to filexfer node
3. Rpm2cpio <filename>.rpm | cpio -idv # To unpack the rpm file without installation which requires root privileges
4. cd usr/bin # usr is located in the unpacked file
5. ./rclone config # and follow prompts:

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

name> <something descriptive>

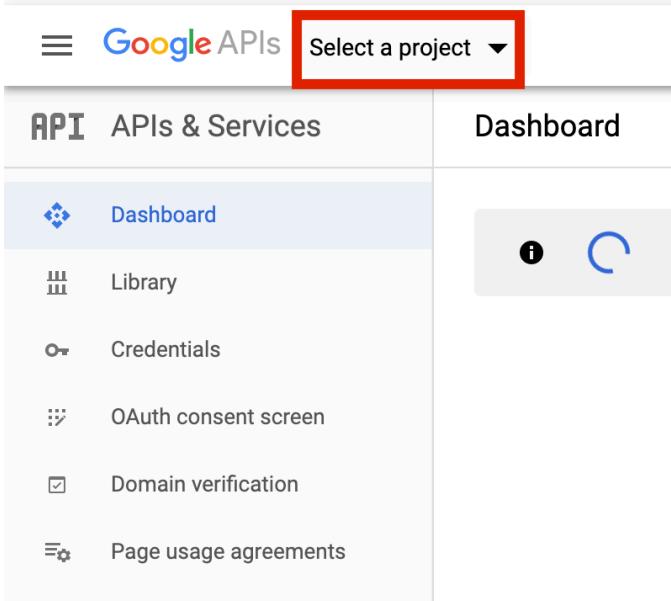
Type of storage to configure.
Enter a string value. Press Enter for the default ("").
Choose a number from below, or type in your own value
1 / 1Fichier
  \ "fichier"
2 / Alias for an existing remote
  \ "alias"
...
Storage> 12

** See help for drive backend at: https://rclone.org/drive/ **

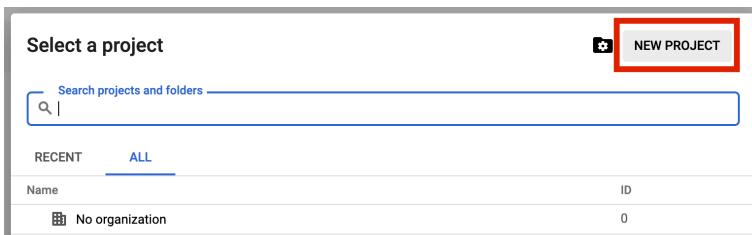
Google Application Client Id
Setting your own is recommended.
See https://rclone.org/drive/#making-your-own-client-id for how to create your own.
If you leave this blank, it will use an internal key which is low performance.
Enter a string value. Press Enter for the default ("").
```

As rclone states, you can either create your own client ID or you can use the default. I haven't done any testing to determine the speeds using the public key vs. creating your own. Note: Thus far, I have not been successful in creating a client ID with my UofA profile and have only been successful when using a personal account. To create your own key, follow the [instructions on the rclone site](#) which I'll replicate below:

1. Go to [Google's API console](#)
2. click **Select a project**



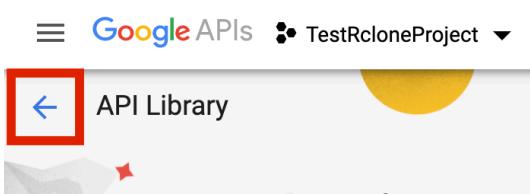
3. Select "New Project"



4. Give your project a descriptive name and then click **Create**

This screenshot shows the "New Project" creation form. At the top, it says "New Project". Below that is a message: "⚠ You have 12 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)". Underneath is a "MANAGE QUOTAS" link. The main form has a "Project name *" field where "TestCloneProject" is typed in. To the right of the field is a help icon (a question mark inside a circle). Below the project name is a note: "Project ID: testrcloneproject. It cannot be changed later. [EDIT](#)". The next section is "Location *", which shows "No organization" and a "BROWSE" button. A note below says "Parent organization or folder". At the bottom are two buttons: a blue "CREATE" button and a white "CANCEL" button.

5. This will take you to an API Library page. To continue with this process, click the back button



6. Select **Credentials** from the side menu and select OAuth client ID

A screenshot of the Google APIs & Services Credentials page. On the left, there's a sidebar with "APIs & Services" at the top, followed by "Dashboard", "Library", "Credentials" (which is highlighted with a red box), "OAuth consent screen", "Domain verification", and "Page usage agreements". On the right, under "APIs" and "Credentials", there's a message about needing credentials to access APIs. Below that is a "Create credentials" button with a dropdown. Underneath are four options: "API key", "OAuth client ID" (which is highlighted with a red box), "Service account key", and "Help me choose".

7. Click **Configure consent screen** and name your application something descriptive. The Application name is what will pop up when Google Drive asks for your consent when you try to connect to it using Rclone.

 A screenshot of the "Configure consent screen" page. It has a warning message: "⚠ To create an OAuth client ID, you must first set a product name on the consent screen" and a "Configure consent screen" button. Below is a form field labeled "Application name" with a question mark icon, containing the text "The name of the app asking for consent". The input field contains "Rclone".

8. Give your OAuth client ID a name and click **Create**. Your Client ID and Secret will appear. You'll need to hold onto these for the next step in the Rclone setup

[← Create OAuth client ID](#)

For applications that use the OAuth 2.0 protocol to call Google APIs, you can use an OAuth 2.0 client ID to generate an access token. The token contains a unique identifier. See [Setting up OAuth 2.0](#) for more information.

- Application type**
- Web application
 - Android [Learn more](#)
 - Chrome App [Learn more](#)
 - iOS [Learn more](#)
 - Other

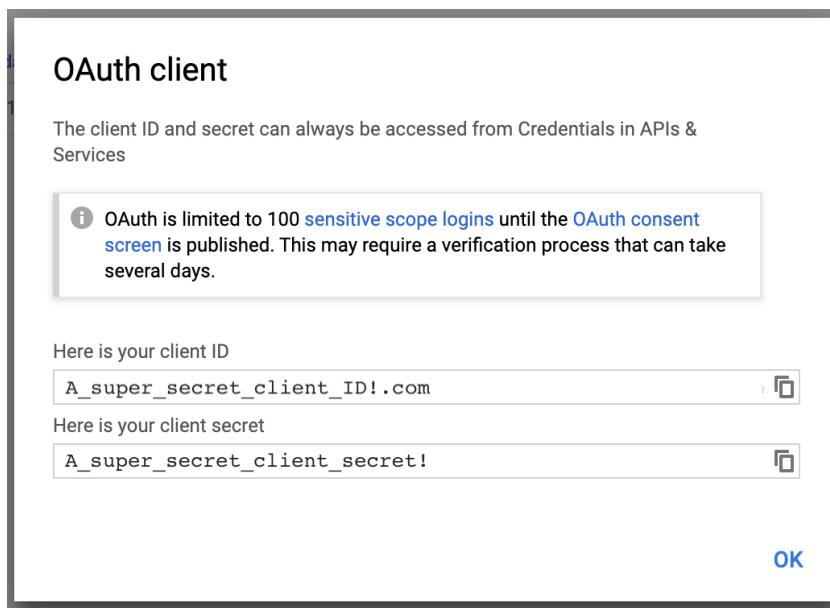
Name [?](#)

SarasTest

[Create](#)

[Cancel](#)

9. Your Client ID and Secret will appear. You'll need to hold onto these for the next step in the Rclone setup



If you decided to get your own Client ID, once you have it and your Client Secret, enter them to continue. If you've decided to use the public key, leave the entry blank and hit enter.

```
client_id> <Your Client ID>

Google Application Client Secret
Setting your own is recommended.
Enter a string value. Press Enter for the default ("").
client_secret> <Your Client Secret>

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 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 in 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> <1-5>  
  
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  
  
Use auto config?  
 * Say Y if not sure  
 * Say N if you are working on a remote or headless machine  
y) Yes  
n) No  
y/n> N # Use N if working on HPC, Y is fine for your PC  
If your browser doesn't open automatically go to the following link: <longurl>  
Log in and authorize rclone for access  
Enter verification code> <verification code>  
  
Configure this as a team drive?  
y) Yes  
n) No  
y/n> n  
  
-----  
[GoogleDriveTest]  
type = drive  
client_id = <Your Client ID>  
client_secret = <Your Client Secret>  
scope = drive  
token = {Info}  
-----  
y) Yes this is OK  
e) Edit this remote  
d) Delete this remote  
y/e/d> y
```

```
Current remotes:
```

Name	Type
=====	=====
GoogleDriveTest	drive
MyGoogleDrive	drive

```
e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q>
```

Multiple Google Drive connections can be established so you can connect to as many drives as you wish. You just need to go through the installation process for each new connection you create, though it isn't necessary to create multiple client IDs.

0.7 iRODS

iRODS is configured on HPC. To transfer files to CyVerse, you will need an account: www.cyverse.org

Information on setting up iRODS can be found here: <https://public.confluence.arizona.edu/display/UAHPC/Transferring+Files#TransferringFiles-iRODS>

Information on using i commands can be found here: <https://wiki.cyverse.org/wiki/display/DS/Using+iCommands>