

Setting up Google Drive for Allstar Backups and General Storage

If you currently have a Google account or you create a Google account you have 15GB of combined storage in the cloud. This includes the space you would use for email if you are using Google mail. Users seldom use even a fraction of the allotted space so saving Allstar backups or other files to your Google drive should have little impact.

The first thing you need to do is setup a Google account on the web if you do not already have one. Once you have created an account you will have a Google drive which will be empty. You can use this drive to store items from your desktop or share files with others.

The gdrive program in HAMVOIP Allstar was made available in an update as of March 25, 2019 and is in your search path. Once you have a valid Google account all you need to do is run a gdrive command in your Allstar server to start the authentication process. At the Linux prompt type -

gdrive about

You will get a message back with a URL. Carefully copy the URL and paste it into your Internet browser. The response will be an authentication code that you in turn copy and paste back into the code response question at your server. It should then authenticate and you are ready to go.

The authorization information is stored in a json file in /root/.gdrive. I have found that this authorization information can be copied intact to other servers which then authorizes gdrive on that server. I have not tried this outside of a LAN on a single public IP. This saves the trouble of having to authorize all of your servers but if they are located at different locations you may have to go through the auth process at those locations also.

The gdrive system has some bugs but there are work arounds. You can test the system by uploading a file. Any file will do. Lets upload the rpt.conf file -

gdrive upload /etc/asterisk/rpt.conf

You likely might get the following message -

“Failed to upload file: googleapi: Error 403: Rate Limit Exceeded, rateLimitExceeded”

If you do just up arrow and repeat the upload string. It may take several tries but it will eventually upload. You can then see the file in your Google drive using your browser. The error messages are random and annoying but seem to be part of the process as they have been there and complained about for years. The HAMVOIP Google backup and restore scripts have a retry mechanism that eventually processes the command but you may have to wait a few tens of seconds for this to happen. Be patient.

There are many commands that are documented here -

<https://github.com/gdrive-org/gdrive#downloads>

One drawback is that uploaded files are given a file ID masking the real name. Here is an example -

Uploading /root/rpt.conf.back

Uploaded 1aDBklMr1DXPIwXMIs4XeEZSaOr89ZcGn at 24.1 KB/s, total 24.1 KB

The file has the real name at the Google drive browser window but the file ID must be use in gdrive. So to download the file from your Google drive you would have to use the file ID **NOT** the real name.

gdrive download 1aDBklMr1DXPIwXMIs4XeEZSaOr89ZcGn

This would return it to the original filename on your server.

I would suggest that instead of remembering file ID's that if you do download it back to a server you do it through the browser in which case you would drag the file from the browser to your computer and then using scp copy it to your Allstar server.

You can however use the gdrive list command to show the file name /ID relationship. You would then select the proper file ID to restore.

gdrive list

Id	Name	Type	Size	Created
1m4j-lhZKrAq2ot0EJvNwHgdVzBparN1z	alarm.files	bin	1.2 MB	2019-03-24 02:35:15
17j0OxJ4tt4FzUa5LIMa5fyi_FAK9F_7S	community.files	bin	79.4 MB	2019-03-24 02:00:26
1bgHDRtNAG2sBxzS3hyJSjFjHRHsrqMSW	SayTides.sh	bin	2.0 KB	2019-03-24 01:54:15

Automatically backing up at predefined intervals

google-backup.sh

The script **google-backup.sh** will backup selected files to your Google drive. It can be run manually or in the background with a cron job. The files that are selected are determined by the /usr/local/etc/

To back once a week you would do the following:

```
15 3 * * 0 /usr/local/sbin/google-backup.sh
```

This example will do a weekly backup at 3:15 AM on Sunday morning. You can change this to any time or interval you desire. Doing this once a week is probably the most often you would want to do it. You can always run it manually after you make changes.

Output is logged to **/var/log/asterisk/backup_log**

A manual run looks like this:

google-backup.sh

Using include definitions from /usr/local/etc/backup.sh

Backup Started...

Compressing files -

```
tar zcf /tmp/Pi3-test_backup_2019-03-24-2123.tgz --exclude=/root/restore /root /etc/asterisk /usr/local/etc /var/spool/cron /srv/http/allmon2/allmon.ini.php /srv/http/supermon/allmon.ini /srv/http/supermon/global.inc /tmp/.backup_info
```

Attempting Upload of /tmp/Pi3-test_backup_2019-03-24-2123.tgz
Success - Uploading /tmp/Pi3-test_backup_2019-03-24-2123.tgz
Uploaded 1kYf8b6d2uLQ9CLJip41Q2-HsbC0KU1m1 at 676.7 KB/s, total 25.3 MB - Retries = 0

Restoring Google Files

google-restore.sh

The script **google-restore.sh** allows you to restore selected files to your server without having to deal with the FILE ID string. It asks if you want to have a wide or narrow display listing. Narrow would be useful in situations where you had a limited screen width. It also asks you if you want to display all of your Google drive files or just those saved as backup files.

Here is an example listing of all files in narrow mode:

Id	Name
1) 1GfRkV2CpPjvFpJHxRC54jz69j3VxNZeX	Pi3-test_backup_2019-03-25-1200.tgz
2) 1v4RxGW7tU62P_ZNr478zNZg5F5JrJc8L	Pi3-test_backup_2019-03-25-1155.tgz
3) 1z3Gnc-dnWyb9xn4MlnPQChI0Qnm-8Ec8	Pi3-test_backup_2019-03-24-2145.tgz
4) 1m4j-lhZKxAq2ot0SJvNwHgdJzBparN1z	alarm.files
5) 17j0OxJ4tx4KzUa5LIMa5fyi_FCE9F_7S	community.files
6) 1bgHDWtNXg2sBxzS3hyJSjFjHRMxqrqMSW	SayTides.sh
7) QUIT	

==== Please select your number choice ==== >

and a display showing just backup files:

Id	Name
1) 1GfRkV2CRpJvcpJHxRC54jx69j3VjNZeX	Pi3-test_backup_2019-03-25-1200.tgz
2) 1v4RmGW7tUx2P_ZNr478zNZg5W5MrJc8L	Pi3-test_backup_2019-03-25-1155.tgz
3) 1z3Gxi-dnWyb9xn4MlnPQChI0Enm-8Ec8	Pi3-test_backup_2019-03-24-2145.tgz
4) QUIT	

==== Please select your number choice ==== >

You select by the left column number. If the file you are trying to download already exists in the /tmp directory you will be asked if you want to overwrite the file.

Google-drive.sh

The **google-drive.sh** script is a general utility to easily upload, download, or list files at your Google drive. It offers the ability to retry until the command is accepted instead of having to reissue the command many times which is often necessary when using the gdrive command alone. It also does a file ID lookup to make downloading files as easy as uploading. Here are the available commands:

google_drive.sh upload <filename> - upload a file to Google drive
google_drive.sh download <filename> - Download a file from Google drive
google_drive.sh list - List all files at Google drive

The Filename can contain paths. The download path and filename are checked for existence and the user is given an overwrite option. Uploads use Google revision ID's so same name files can be uploaded multiple times.