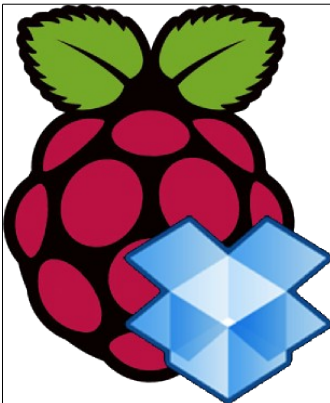


Uso de Dropbox desde RaspberryPi



... recuperamos un Super Poder!

Buenas tardes, nuestro amigo Rubén nos consulta vía Facebook:



Primero comentar que al próximo que me hable de usted ... bueno ... El caso es que estaba trabajando en un script que hiciera esto mismo, ya que Dropbox no tiene cliente nativo para arquitecturas ARM (nuestro querido Raspberry Pi), y justo en este momento Rubén nos informa de este script, veámoslo.

Si os pasáis por la lista de request features de Dropbox, veréis que hay muchas peticiones de que desarrollen un cliente para arquitecturas ARM, entre ellas la mía. Pero mientras que lo desarrollan, vamos a probar el script de [esa web](#).

Lo primero antes de nada es tener una cuenta en Dropbox, desde [aquí](#) puedes darte de alta.



Leemos un poco las características de dropbox_uploader y parece que las cumple todas, vayamos a nuestro RaspberryPi:

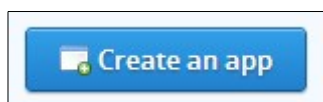
```
1 cd /usr/local/bin/
2 wget https://raw.githubusercontent.com/andreafabrizi/Dropbox-Uploader/master/dropbox_uploader.sh
3 chmod +x dropbox_uploader.sh
```

Bien, fácil y sencillo, como a mi me gusta, continuemos:

```
1 ./dropbox_uploader.sh
```

Nos pide la app key, esto forma parte de un sistema de autenticación llamado OAUTH, simplemente seguimos las instrucciones que nos dice y vamos a la URL <https://www2.dropbox.com/developers/apps> y hacemos clic

en:



Y el siguiente formulario lo dejamos tal que así:

A form titled "Create a new app" with a plus icon. It contains fields for "App type" (set to "Core API"), "App name" (set to "MyUploader2563715548"), and "Description" (set to "dropbox_uploader para Raspberry Pi"). Below these is an "Access" section with two radio buttons: "App folder" (selected) and "Full Dropbox". At the bottom are "Create" and "Cancel" buttons.

Create a new app

App type [?] Core API

App name

Description

Access ☒ **App folder** - Your app only needs access to a single folder within the user's Dropbox (recommended).
☐ **Full Dropbox** - Your app needs access to the user's entire Dropbox.

Create Cancel

El nombre hay que dejarlo tal cual, es el que nos indica el script, evidentemente en la descripción ponemos lo que queramos. En mi caso he marcado Full Dropbox, porque pretendo darle un uso intensivo a varios directorios, si sólo vamos a usar un directorio con este script, marcaríamos App folder. Le damos a Create y nos tiene que salir la siguiente información:

A form titled "MyUploader2563715548" showing app details. It includes fields for "App name" (MyUploader2563715548), "App status" (Development), "App key" (redacted), "App secret" (redacted), and "Access type" (Full Dropbox).

MyUploader2563715548

App name

App status Development (Apply for production status)




App key

App secret

Access type Full Dropbox

La App key y el App secret son tuyos exclusivos, no se te ocurra ir por ahí publicándolos. Si volvemos al script, este nos pide primero la App key, luego el App secret y por último nos vuelve a pedir si vamos a darle acceso full o app, yo elijo 'f'. Para continuar con el proceso de autenticación OAUTH, nos indica una URL, que tendremos que visitar para dar acceso al script.

An OAuth authorization screen for the application "MyUploader2563715548". It features icons for a question mark, a refresh button, and the Dropbox logo. The text asks for permission to connect to Dropbox, listing three points: full access, a warning to trust the app, and the current user's email. At the bottom are "Permitir" and "Rechazar" buttons.

La aplicación MyUploader2563715548 quiere conectar con tu Dropbox.

- Esta aplicación obtendrá acceso a tu Dropbox completo.
- Asegúrate de que confías en esta aplicación antes de continuar.
- Has iniciado sesión como REDACTED@gmail.com. Si deseabas conectarte desde otra cuenta, puedes [cerrar sesión](#).

Permitir Rechazar

Le damos a permitir, volvemos al script y pulsamos intro.

Realmente el proceso es sencillísimo y seguramente te sonará de haber usado algo parecido alguna vez, como decía antes, este procedimiento se llama OAUTH, y lo he puesto pasito a pasito porque (con suerte) lo usaremos más de una vez. Podemos comprobar que funciona ejecutando por ejemplo:

```

1 # ./dropbox_uploader.sh list
2 > Listing "/"... DONE
3 [D] blogger
4 [D] botijos
5 [D] Camera Uploads
6 [D] chorradas
7 [D] compartido
8 [D] curro
9 [D] paraAndroid
10 [F] parapapa.txt
11 [D] Photos
12 [D] Public
13 [F] todo.kml
14 [D] torrents
15 [D] Videos

```

Hacer una copia de seguridad periódica de un directorio de Dropbox

El proceso anterior no tiene ningún mérito por mi parte, así que ahora vamos a darle la utilidad que buscaba desde el principio. Alguna vez te habrá pasado que has perdido las fotos del móvil al tocar donde no debías, o no te acordaste de hacer una copia al cambiarlo por otro. Para solucionar esto, los de Dropbox se pusieron las pilas y añadieron la opción de subida instantánea para la cámara de tu móvil, si no lo tienes ¿a que esperas?. Pero aún así, quiero que si Dropbox algún día le da por ser de pago, o por ser comprado por Microsoft, tener de antemano una copia del directorio "Camera Uploads" en mi NAS.

El NAS lo tengo montado en el RaspberryPi en /mnt/remoto , así que empezamos por crear un directorio:

```
1 mkdir -p /mnt/remoto/copia-fotos-juanmol
```

y ahora el script /usr/local/bin/backup_dropbox.sh :

```

1 #!/bin/bash
2 # Programado por Juanmol para http://rsppi.blogspot.com
3 destino=/mnt/remoto/copia-fotos-juanmol
4 origen=Camera\ Uploads
5 listado=/tmp/listado.txt
6 log=/var/log/backup_dropbox.log
7
8 cd $destino
9 dropbox_uploader.sh list "Camera Uploads" > $listado
10
11 IFS=$'\n' ; for n in $(cat $listado | grep -v -e Listing -e DONE | sed 's/\ \
12 [F]\ //g');
13 do
14     if [ -e $n ];
15     then
16         echo $n ya existe, no se hace nada ;
17     else
18         echo se copia $n a las $(date) >> $log ;
19         dropbox_uploader.sh download "$origen/$n" ;
20     fi
21 done
22 rm -r $listado

```

esta feo que yo lo diga, pero el script es precioso :) La primera vez que lo ejecutes tardará un siglo y medio en sincronizar (al menos en mi caso). Así que antes de ponerlo en el CRON, te recomiendo que lo ejecutes a mano, una vez. Cuando haya terminado, puedes añadir la siguiente línea al archivo `/etc/crontab`:

```
1 */5 * * * *    root    /bin/bash
2 /usr/local/bin/backup_dropbox.sh
```

y cada 5 minutos se descargará las fotos y vídeos nuevos que hagas con el móvil.

Todo el mérito es para los desarrolladores del script `dropbox_uploader.sh` yo sólo he añadido una sincronización.