Print Files on Linux via Dropbox

If you would to enable cloud printing in Linux, you can easily do that using Dropbox. (see similar solutions for Windows and Mac)

Assume that you have a folder called PrintQueue in your Dropbox Folder in Linux. Create a shell script as shown below and setup a cron job against this script that maybe runs after every 'n' seconds (or minutes).

```
#!/bin/sh

DIR = ~/DropBox/PrintQueue

for i in $(ls -l $DIR | grep ^- | awk '{print $9}')

do

 lpr $DIR/$i
 rm $DIR/$i

done
```

From your remote computer or your mobile phone, add any file to the PrintQueue Folder in Dropbox and, within minutes, it should start printing on your printer attached to your Linux machine.

If you have multiple printers attached to Linux computer, use the lp â€“p command to specify the printer name.

Once the file is printed, it is removed from the queue.

Reader Comments

```
#!/bin/bash
FILES=/home/arjanw/Dropbox/PrintQueue/*
shopt -s nullglob
for f in $FILES
do
echo “Processing $f file…”
lpr “$f”
rm “$f”
done
```

Written by Arjan Waardenburg on 10.01.10

Hey, love this solution, but am running into trouble with the directories in both the original solution, and the suggested solution in the above comment. In Amit’s solution, I get a ‘DIR: not found’ even after double-checking the address and trying variations. In Arjan’s solution, I get

```
"Processing /home/david/Dropbox/PrintQueue/test.txt
fileâ€¦"
lpr: Error – unable to access
“"/home/david/Dropbox/PrintQueue/test.txt"” – No such file
or directory
rm: cannot remove
`"/home/david/Dropbox/PrintQueue/test.txt"’: No such file
or directory
```

which is weird: it finds the directory and the enclosed file, but then reports that there is no such file or directory?

Any ideas? (I do not have experience writing shell scripts, so would not recognize even simple problems … though I know enough to check the paths). I’m running this on Ubuntu Linux, 10.04

dave

Written by David Maddalena on 10.01.10

This sounds like exactly what I’m looking for!

But I also have no experience in creating or running shell scripts. And I have no idea what a “cron job” is.

I’d really love for this to work, but as of now I have no idea where to start, besides copying all the script above into a text document.

It would be great if this was laid out in a step-by-step solution the way the original article (link) is written.
I’m running Linux Mint 9.

Written by Adam on 10.01.10

@ David,

try removing the quotes from lpr “$f” and rm “$f”

I haven’t tried it myself, but the errors you’re getting show the file in double double-quotes which leads me to believe that the $f variable might already include the quotes.

also, as long as there are no spaces in the filename the quotes are not needed so for testing purposes you should be able to remove them in order to simply debugging.

Written by Lucas Short on 10.01.10

```
#!/bin/sh

DIR = ~/DropBox/PrintQueue

for i in $(ls -l $DIR | grep ^- | awk ‘{print $9}’)

do
```

```
lpr $DIR/$i
rm $DIR/$i

done
```

this original script didn’t work for me either, but I can explain why.

“ls -l $DIR | grep ^- | awk ‘{print $9}’”
this command basically turns every line of the ls command into a space-delimited array and them assumes that the filename is stored in element 9… not always the case i guess, i changed the 9 to an 8 and everything works now.

run the above command a few times, changing the number at the end until you end up with a list of filenames.

Written by Lucas Short on 10.01.10

Ok… Someone needs to create a simple App that does this… preferably for Ubuntu…

Written by Bruce Wagner on 10.01.10

The original script does fail on filenames with spaces, my script as seen above works when the smartquotes (inserted by the comment system) are converted to normal quotes.

Written by Arjan Waardenburg on 10.01.10

And for the cron job (scheduling jobs to run at certain times), there is a somewhat easier way via a gui app gnome-schedule to create a cron job
On ubuntu :
sudo apt-get install gnome-schedule
After this it is available from the menu in System Tools

Written by Arjan Waardenburg on 10.01.10

You should probably leave out the echo statement by the way.

Written by Arjan Waardenburg on 10.01.10

Hi Again!
So I merged Lucas’ fixes- it turned out to be 8 for me too. (Also the extra quotation marks) with Arjan’s. This is what I came up with that works for me:

```
#!/bin/bash
FILES=/home/adam/Dropbox/PrintQueue/*
shopt -s nullglob
for f in $FILES
do
echo "Processing $f fileâ€¦"
lpr $f
rm $f
done
```

Now, my issue is to make this run every 30 seconds, or n seconds I suppose. So I don’t have to run this script everytime I send something to the Dropbox, as it would defeat the purpose of having it anyway. (I could just find it and print it manually.)

So could someone please explain the “cron job”?

Written by Adam on 10.01.10

@Lucas,

thanks! Very good: removing the quotes where indicated liberated Arjan’s script. Works. Now to decide whether to set up a constantly running chron job for the infrequent iPhone print, or to run it from issh. Thanks agian,

dave

Written by David Maddalena on 10.01.10

@Arjan, thanks for clarifying the comment-system’s crudding of your script: the straight quotes worked to make spaces in filenames acceptable again. Great work.

Written by David Maddalena on 10.02.10

Please use the correct quotes as indicated above, it is needed to print files where the filename contains spaces.

Written by Arjan Waardenburg on 10.02.10

I am very new to Ubuntu, but I have not been able to get this to work at all. I’ve tried using Arjan above code, with the appropriate name change, but don’t know what else I’m doing wrong.