Rsync 2 – Finishing Up

Ok...you've **read** part 1, you've set up a /share folder on a complete different drive or computer (maybe an economical Raspberry Pi hooked to an external usb). You're ready to go...now just a couple more details to polish it up.

In the following examples we use:

- a Raspberry Pi hooked to an external drive, it's address in the house network is 192.168.5.5
- it has a shared drive /share
- a subfolder on /share called LouStuff, so /share/LouStuff
- subfolders for my web stuff and my home folder, /share/LouStuff/home-lou and /share/LouStuff/var-html/

So the basic backup of /home/lou would be (*if any of this needs explained, see Part 1*)

```
rsync -avPzh -e ssh --modify-window=3601 /home/lou/
music@192.168.5.5:/share/LouStuff/home-lou
```

I don't want it to keep anything I've intentionally deleted from my system...so I'll add the –delete option

```
rsync -avPzh -e ssh --delete --modify-window=3601 /home/lou/
music@192.168.5.5:/share/LouStuff/home-lou
```

Additionally, I don't want it to back up my trash folders or my cache folders (browser cache, etc)...so I'll use the –exclude option.

```
rsync -avPzh -e ssh --modify-window=3601 --exclude '*trash*'
--exclude '*cache*' --exclude "*Trash*" --delete /home/lou/
music@192.168.5.5:/share/LouStuff/home-lou/
```

So that backs up my home folders...now for the web stuff.

```
rsync -avPzh -e ssh --modify-window=3601 --exclude '*trash*'
--exclude '*cache*'  --exclude "*Trash*" --delete /var/www/html/
music@192.168.5.5:/share/LouStuff/var-html/
```

So the entire script is

```
#!/bin/bash
LOGFILE=/home/lou/logs/backup-$(date +%Y%m%d).log

rsync -avPzh -e ssh --log-file=$LOGFILE --modify-window=3601 --exclude '*trash*'
--exclude '*cache*' --exclude "*Trash*" --delete /home/lou/
music@192.168.5.5:/share/LouStuff/home-lou/

rsync -avPzh -e ssh --log-file=$LOGFILE --modify-window=3601 --exclude '*trash*'
--exclude '*cache*'  --exclude "*Trash*" --delete /var/www/html/
music@192.168.5.5:/share/LouStuff/var-html/

cp $LOGFILE /home/lou/Desktop/
```

note that last little bit simply copies the log to my desktop so its a reminder that backups happened, etc.

Run it, test it....everything is just exactly perfect.

Except for one thing. You have to sign in to the backup server every time. We want this to run without that requirement so we can put it in cron and have it automatically happen every night while I'm sleeping.

So, somehow, we have to start connect through ssh without requiring a signon.

<u>Note:</u> DO test the script before proceeding to ensure everything is actually configured properly and working! If something isn't right, correct it before generating keys for password-less access.

So...first, make a key for yourself on the machine you're saving FROM

```
>ssh-keygen
>Enter passphrase (empty for no passphrase):
>Enter same passphrase again:
```

When it asks for a passphrase (twice), just hit enter.

Now copy it to the remote machine...the machine you're backing up TO

```
>ssh-copy-id -i ~/.ssh/id_rsa.pub music@192.168.5.5
```

notice that I sent it to music@192.168.5.5 because 'music' is the user name that I sign into that server with.

It should ask for your password ...go ahead and enter it.

Now try to ssh to that machine without entering a password.

```
>sh music@192.168.5.5
```

It should accept your connection without requesting a password.

Now try your script again. You should be able to run it and rsync your files to the backup computer without having to sign on.

Now you're ready to add your script to cron so it happens every day without you having to do a thing.

You're welcome.

–---

Additional stuff:

Rsync Part 1 in .pdf: http://louwilkinson.us/documents/rsync-demo.pdf

Rsync Part 1 video: https://www.youtube.com/watch?v=4M8P6fNg884

Setting up a RaspberryPi external server in .pdf (*this is about using it as a streaming audio server, but if you don't install minidlna, then its simply an external server for backups or whatever*):
http://louwilkinson.us/documents/home%20streaming%20audio%20server.pdf

Setting up a RaspberryPi external server video:
https://www.youtube.com/watch?v=ip2Va8cxO_I

And don't kid yourself...those RaspberryPis are outstanding for this...for $40 and the price of a usb drive, you have an entire stand alone backup server.