

Troy's place



learning

Backing up Android to rsync.net

Ages ago I decided to get an account on rsync.net for backups. It worked well for Linux but I couldn't get it working for Android. Which is funny because I thought naively that rsync, being open and, well, venerable, would just work anywhere. It turns out though that most backup apps don't support rsync but instead support various closed protocols like dropbox or whatnot. Those that would work with rsync (like FolderSync) use some weird Android library that isn't compatible with with most SFTP protocols. I dunno. [Go look it up](#).

Anyway this drove me crazy for a few years but I finally found a workaround: [termux](#). It's like a little actual Linux on your Android device. It's fun, it's open, and it works. Below is how to do it.

- Make sure you have developer options enabled on the phone.
- Through developer options set the phone to stay on when plugged in.
- Plug in phone.
- Install f-droid
- Install termux from f-droid
- Install termux-api from f-droid
- Make the termux can read the phone's ssd:
 - `termux-setup-storage`
 - Click the "allow access to storage" popup on phone
- Update and install packages for termux:
 - `pkg upgrade`

- `pkg install openssh tmux vim rsync termux-api termux-job-scheduler`
 - (vim optional but nice)
- Find your username is:
 - `whoami`
- set a password
 - `passwd`
 - follow prompts
- start the ssh server:
 - `sshd`
- Find your phone's IP address
 - `ifconfig`
- ssh to phone to make life easier
 - `ssh -p 8022 user@phone_ip`
 - where `phone_ip` is the ip address of your phone
- `ssh-keygen -t rsa -b 4096`
- Copy ssh key to rsync server (making sure to not overwrite any existing keys)
 - `cat ~/.ssh/id_rsa.pub | ssh 123@tv-s009.rsync.net 'dd of=\.ssh/authorized_keys oflag=append conv=notrunc'`
- Enter a tmux **shell** if you're worried about being disconnected:
 - `tmux`
- Copy what you want to back up:
 - `rsync -avH`
`/data/data/com.termux/files/home/storage/dcim/Camera/9988@usw-s009.rsync.net:photos`
- Disconnect from tmux if you want:
 - `<ctrl> B D`
- If you want to automate it's easier to use `termux-job-scheduler` than `cron` because `termux-job-scheduler` uses Android's native scheduler and doesn't fall prey to Android **job killing issues** like `cron` would.
 - Command to run the script every hour only when plugged in and on wifi, making sure the script persists through reboots:

- `termux-job-scheduler --network unmetered --charging true --persisted true --period-ms 3600000 -s /data/data/com.termux/files/usr/bin/photos.sh`
- where `photos.sh` is an executable bash script that holds your `rsync` command.

That's basically it. All I really care about is backing up my photos and that does the trick.

[Previous: My first Ruby project!](#)

[Next: SaaS but without all the sass](#)

Search

Recent Posts

[Setting up Vaultwarden on a Digital Ocean droplet](#)

[When my dog fell down a mineshaft](#)

[Configuring Uptime Kuma with Telegram notifications using Podman as its container manager](#)

[SaaS but without all the sass](#)

[Backing up Android to rsync.net](#)

Archives

[February 2025](#)

[October 2024](#)

[January 2024](#)

[January 2023](#)

[December 2022](#)

[November 2022](#)

Categories

Camping

learning

meta

open source

ruby

saas