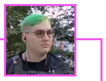
glow.li

Search

All Posts | Technology | Photography | Short | About | Feeds | Fedi Posts



Technology • 2015-11-06 Signed

Run an SSH server on your Android with Termux

With the brilliant Termux terminal emulator app you can run an SSH server on your Android.

Previously I used SSHDroid to achieve this, but with Termux is much nicer because you have access to a working package manager.

Run the service

You need to install the OpenSSH package

```
    apt install openssh
```

and use following command to start the ssh server.

```
    sshd
```

And there you go. Your ssh service is now running on port 8022.

```
    ssh localhost -p 8022
```

Adding your Public key

You can't do password authentication in Termux, therefore you need to put your OpenSSH public key into the ~/.ssh/authorized_keys file.

This file will need to be created and permissions set to 600.

```
    touch ~/.ssh/authorized_keys
    # Set Permissions to the file
    chmod 600 ~/.ssh/authorized_keys
    # Make sure the folder .ssh folder has the correct permissions
    chmod 700 ~/.ssh
```

If you do not have a OpenSSH key pair yet, you can generate one with the following command:

```
    ssh-keygen
```

You may or may not enter a passphrase and if you don't specify otherwise, your key pair will have been saved under ~/.ssh/id_rsa and ~/.ssh/id_rsa.pub. You can then add it to the ~/.ssh/authorized_keys with

```
    cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
    chmod 600 ~/.ssh/authorized_keys
```

Then you can test it by connecting to your ssh service

```
    # -i $PATH_TO_FILE/filename is only required if the id_rsa file is
not ~/.ssh/id_rsa
    ssh localhost -p 8022 -i %PATH_TO_KEY-FILE%/%NAME_OF_KEY%
```

You can now use your private key (~/.ssh/id_rsa) to login to your Termux SSH Server. Simply copy it to your computer (by copying it to internal storage first `cp`

`~/.ssh/id_rsa /sdcard`) and use it in your ssh client.

OpenSSH

If you're using OpenSSH (on Linux or Cygwin) you can use it directly:

```
    # -i $PATH_TO_FILE/filename is only required if the id_rsa file is
not ~/.ssh/id_rsa
    ssh $IP -p 8022 -i %PATH_TO_KEY-FILE%/%NAME_OF_KEY%
```

PuTTY

If you're using PuTTY you will need to convert it to the PuTTY Private Key format first.

1. Download and run PuTTYgen
2. Load the private key (id_rsa)
3. Save the private key as a *.ppk file.
4. Download and run PuTTY
5. Enter the IP address of your Android device and use port 8022
6. Under Connection>SSH>Auth you can browse for the *.pkk file
7. Click open
8. You can leave "login as:" blank

You should now be connected to your Android device via SSH.

If it still doesn't work

```
    killall sshd
    sshd -d
```

If it is still prompts you for a password you can enter sshd's debug mode with the above command and see exactly why your key has been rejected. The reason usually are bad permission on either your home directory, your .ssh folder or your authorized_keys file.

The correct permissions are:

```
    chmod 700 ~
    chmod 700 ~/.ssh
    chmod 600 ~/.ssh/*
```

I hope in the future Termux will allow us to register sshd as a proper service which would automatically start on system boot. Right now I have the 'sshd' command in my .bashrc file and I am using Tasker to launch Termux after boot. You can also use the Termux widget to quickly start sshd with a widget.

See also: Access the SSH server via USB instead of WiFi

#Technology #Android #Termux #CLI #SSH #Tutorial #2015

&lt;&lt; Next

Previous >>