Setting up your Raspberry Pi Zero W (Wireless)

Instructions below are

- specific to the Raspberry Pi Zero W target board
- valid as of August 2022.

Table of Contents

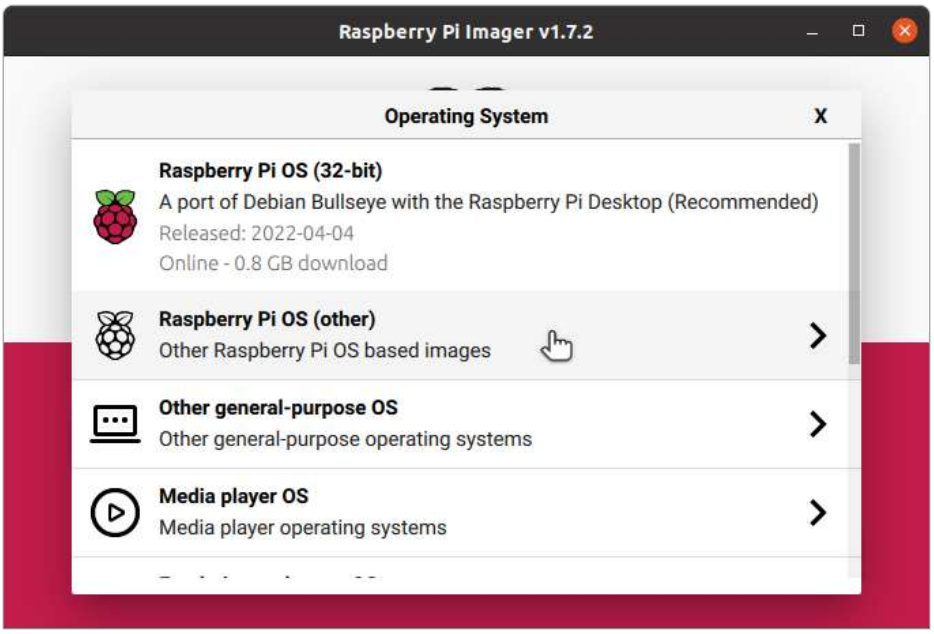
I Installing the Raspberry Pi OS on an SD card

1. Burn the microSD card with a recent and stable Raspberry Pi OS image; simultaneously have the Raspberry Pi *Imager app* setup the network, username, and so on...

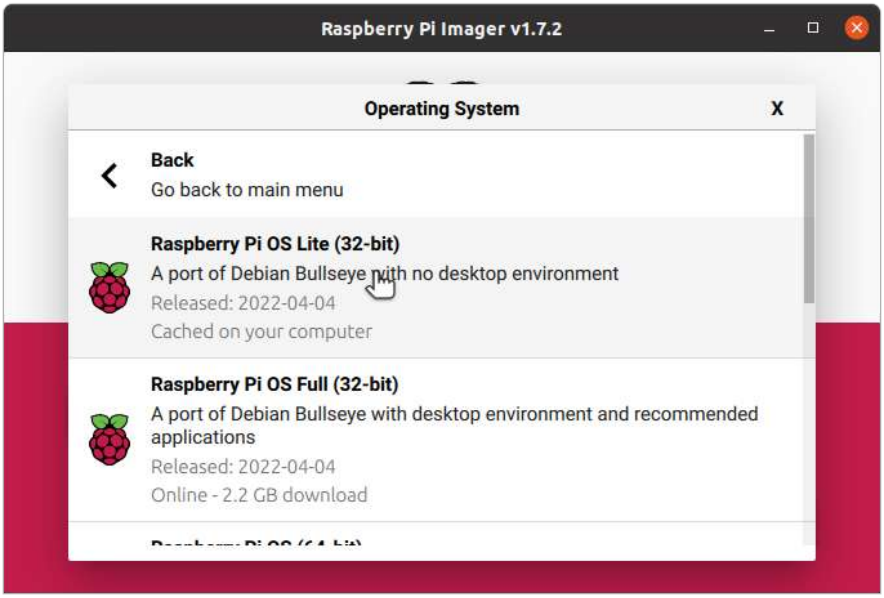
   i. Download the Raspberry Pi Imager app
   Navigate to https://www.raspberrypi.com/software/ , download and install it on your host system (you can even install it onto a Linux guest (that supports graphics) and run it from there)

   ii. Run it, thus installing a recent stable Raspberry Pi OS image onto your microSD card!
      a) A 45 second video demonstrating how: https://youtu.be/ntaXWS8Lk34 ; take a look!

   iii. Select the OS:
      a) click the 'CHOOSE OS' button
      b) To optimize, and for the Raspberry Pi Zero (W) target, click on the second choice: Raspberry Pi OS (other) [below]

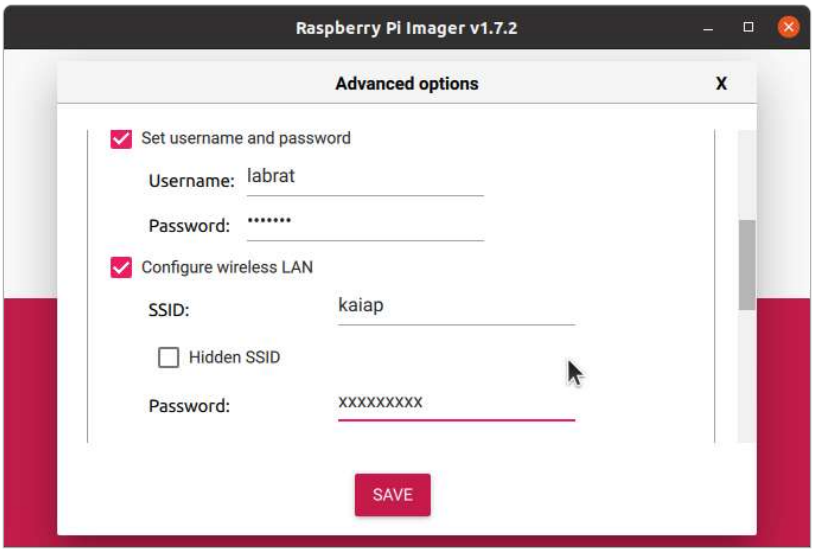


      c) then click on the first one:
      Raspberry Pi OS Lite (32-bit) – no desktop environment



iv. Then select the microSD card via the STORAGE button

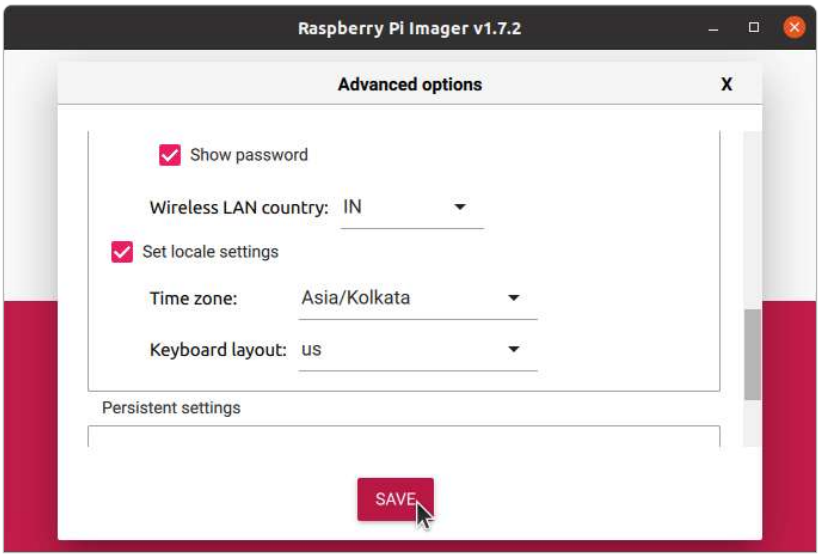
v. *NOTE: Very IMPORTANT -*
Next, before writing the image, DO make use of the Imager app's **Settings gearwheel** button!It's *very useful* to use the Imager app to pre-configure the R Pi OS – you can select:
   a) the board hostname ('rpi0w_<myname>' perhaps)
   b) do Enable SSH
      i. to keep it simple, you can initially use password authentication, but, as you learn more, DO switch to authentication via SSH keys only – its far more secure!
   c) Security: be sure to choose a username other than the default one (pi); f.e., `labrat` ; supply a password
   d) *Very Important:* Configure the wireless LAN - the WiFi AP; specify the SSID and password. We're going to use the Raspberry Pi Zero W in 'headless' mode – no HDMI monitor and all that jazz! You'll need to configure the WiFi AP in order to login over SSH.



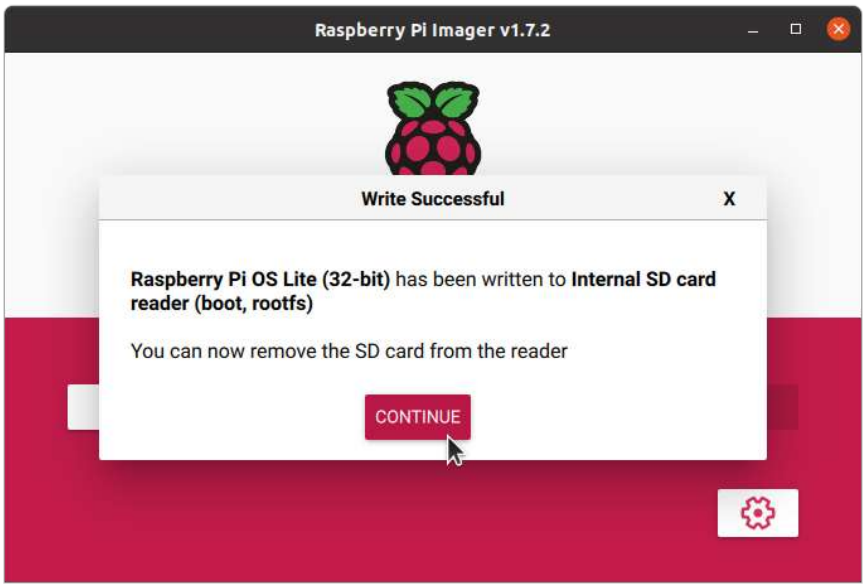
      i. Don't forget to scroll down, set the:

1. Country for Wireless to India (IN)
2. Locale (Time zone as Asia/Kolkata, keyboard as ‘us’)

e) Click on the **SAVE** button when all set...

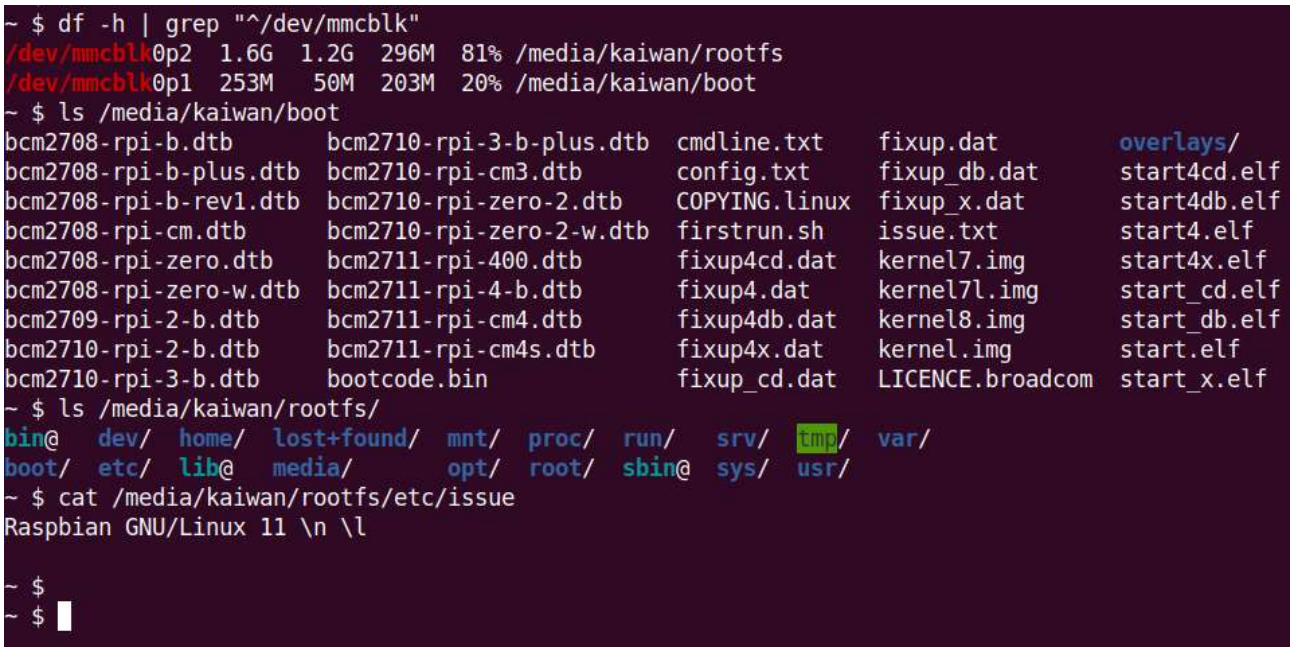


vi. Now click on the **WRITE** button; once you confirm, the image gets written and verified.



It can take a while, be patient... When done, eject the SD card, click on the **CONTINUE** button.

2. On a Linux (or MacOS) host (*or guest!*), I reinsert the SD card and do this to verify the OS image is written:



On a Windows host you can always lookup the SD card content with the File explorer app.

Don’t forget to unmount (eject) the SD card before physically removing it from it’s slot.

II Login to the board over SSH

To login over SSH, we obviously **require the system’s (DHCP-assigned) IP address**. We’re assuming you’ve installed the Raspberry Pi OS and configured the wireless network – as the previous section described.

Obtain the device’s IP address

- Power off the Raspberry Pi board
- Insert the microSD card containing the installed Raspberry Pi OS into your board, and apply power
- It should boot up ... give it a minute or two and then proceed to the next step...
- Ensure you’re *on the same local network* as the device (f.e. 192.168.1.x or 10.20.1.x). To find it’s IP address:
  - On a Windows host, via a browser:
    - navigate to the router gateway (f.e. 192.168.1.1 or 10.20.1.1)
    - login if required
    - Lookup the DHCP client list in the web browser app; the Raspberry Pi should show up; look for the hostname you gave it when installing the image. Once located, note it’s allocated IP address.
  - On a Linux/Mac host:
    - Here too you can of course use a browser as in the previous paragraph (assuming you’re on a Linux system running a graphical desktop)

- Via the Linux CLI, there are several ways to figure the device's IP address; among them are:
  - Install the nmap and arp-scan utils:
    `sudo apt install nmap arp-scan -y`
  - `sudo arp-scan --localnet |egrep -i "Raspberry"`
  - `sudo nmap -sn -PR <domain-to-scan>/24 # domain f.e. 10.20.1.0`
  - *The 'nmap' way tends to be best; use my localnet_discov.sh helper script to make it easy!*

An example of using it to discover the board's IP address:

```
~ $ localnet_discov.sh 10.20.1.0 | grep -i rpi0wlabrat
B8:27:EB:[illegible],rpi0wlabrat.wlan,10.20.1.63
~ $
```

You can see the board hostname (highlighted in red colour) clearly.

Advanced: how to setup the Wireless network on the CLI

Just in case the Raspberry Pi *Imager* app does not succeed in setting up the wireless network...

1. On your Linux host, run *wpa_passphrase* and obtain the required snippet for a given SSID:

```
~ $ wpa_passphrase
usage: wpa_passphrase <ssid> [passphrase]

If passphrase is left out, it will be read from stdin
~ $
~ $ wpa_passphrase OfficeNet01 WelCome2WFO@Last
network={
        ssid="OfficeNet01"
        #psk="WelCome2WFO@Last"
        psk=b03af193ebd2ad742ea6f397f9da0357a005c644bcc91329b7f61b17d0046c3d
}
~ $
```

2. Copy-paste the output into the *wpa_supplicant.conf* file on the SD card:

```
sudo nano <rootfs_mountpoint>/etc/wpa_supplicant/wpa_supplicant.conf
<copy-paste>
<save & exit>
```

Obviously, delete the commented-out cleartext password line for security.

Headless login over SSH

Once you obtain the board IP address, login over SSH:

```
ssh <username>@<IPaddr>
```

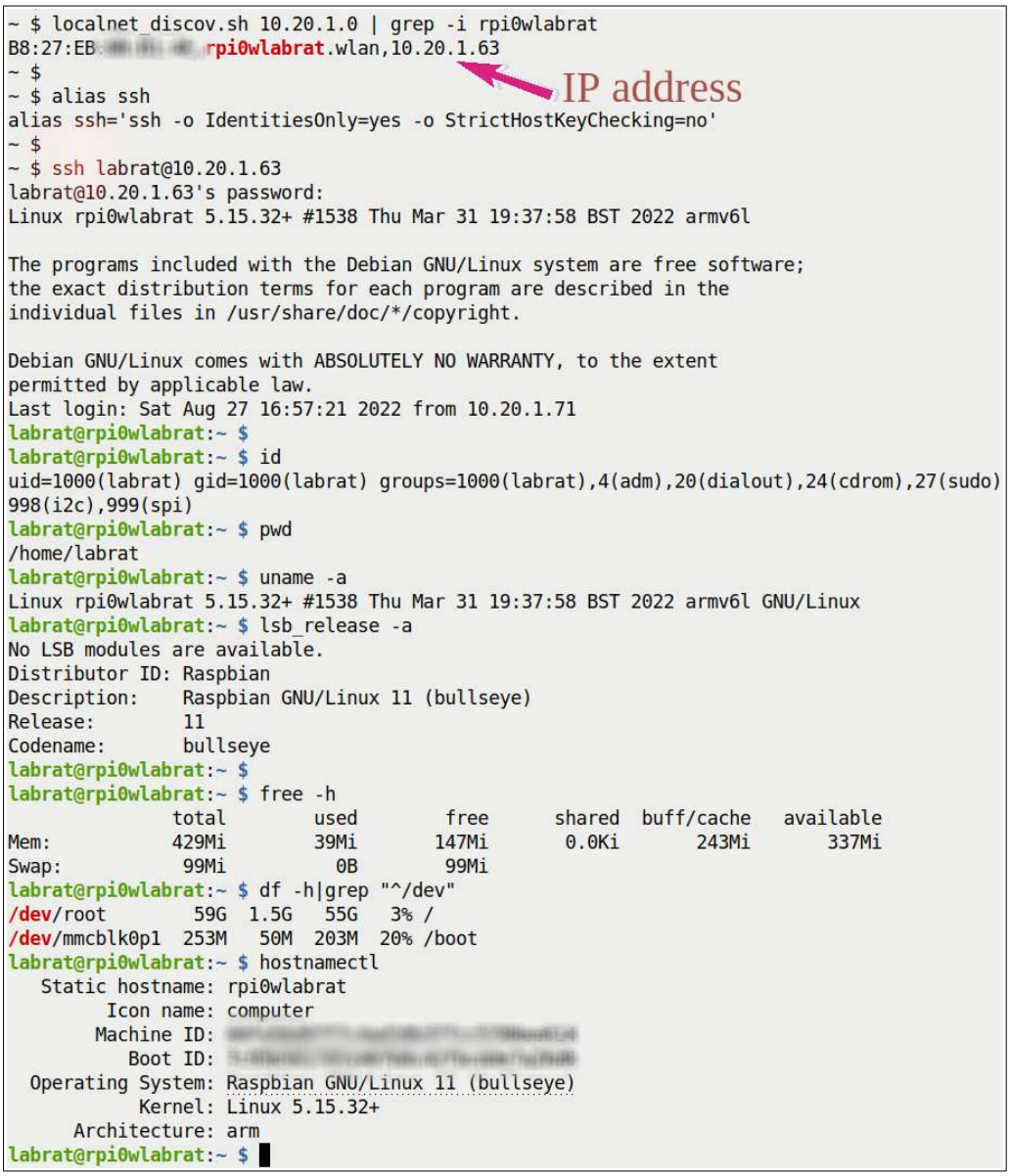
F.e.
```
ssh myname@192.168.1.108
```

If it fails with a message like:
"... Too many authentication failures ..."

try this:

```
alias ssh='ssh -o IdentitiesOnly=yes -o StrictHostKeyChecking=no'
alias scp='scp -o IdentitiesOnly=yes -o StrictHostKeyChecking=no'
```

and then rerun the *ssh* command shown above! It should succeed.
A sample screenshot:



Tips:

- On a Windows host:
  - if required, setup the SSH client (eg. Putty) to use these parameters:
    `-o IdentitiesOnly=yes -o StrictHostKeyChecking=no`
    to ssh and scp

- On a Linux/Mac host:
  - put the ssh and scp aliases into a startup script

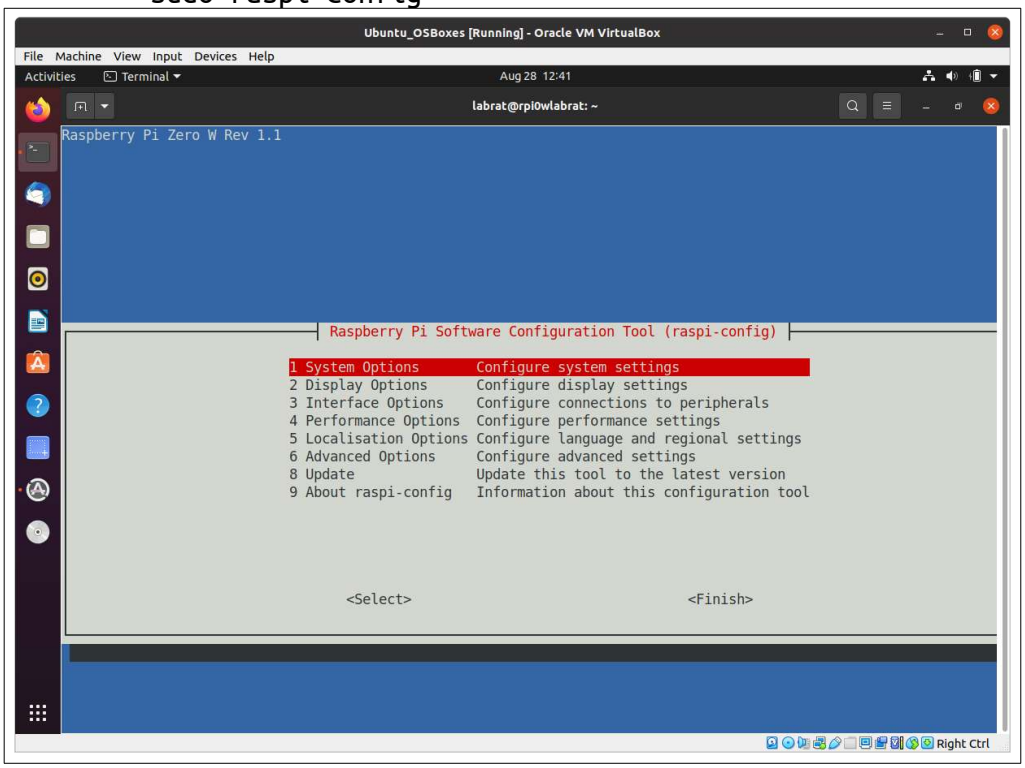
- *Security: use password-less logins by setting up SSH keys!*
  Ref: https://www.raspberrypi-spy.co.uk/2019/02/setting-up-ssh-keys-on-the-raspberry-pi/

- **If all else fails when trying to ssh in**, power off the R Pi, attach a USB-to-serial dongle, setup a terminal emulator (Hyperterminal / Putty on Windows, minicom on Linux), login over it and get the IP address (`ip a` or `ifconfig`).
  Ref: *WORKING ON THE CONSOLE WITH THE RASPBERRY PI, kaiwanTECH, Dec 2018*.

What to do once logged in to the Raspberry Pi

Once logged in to the R Pi board:

- First run
  `sudo raspi-config`



and set things up as required...

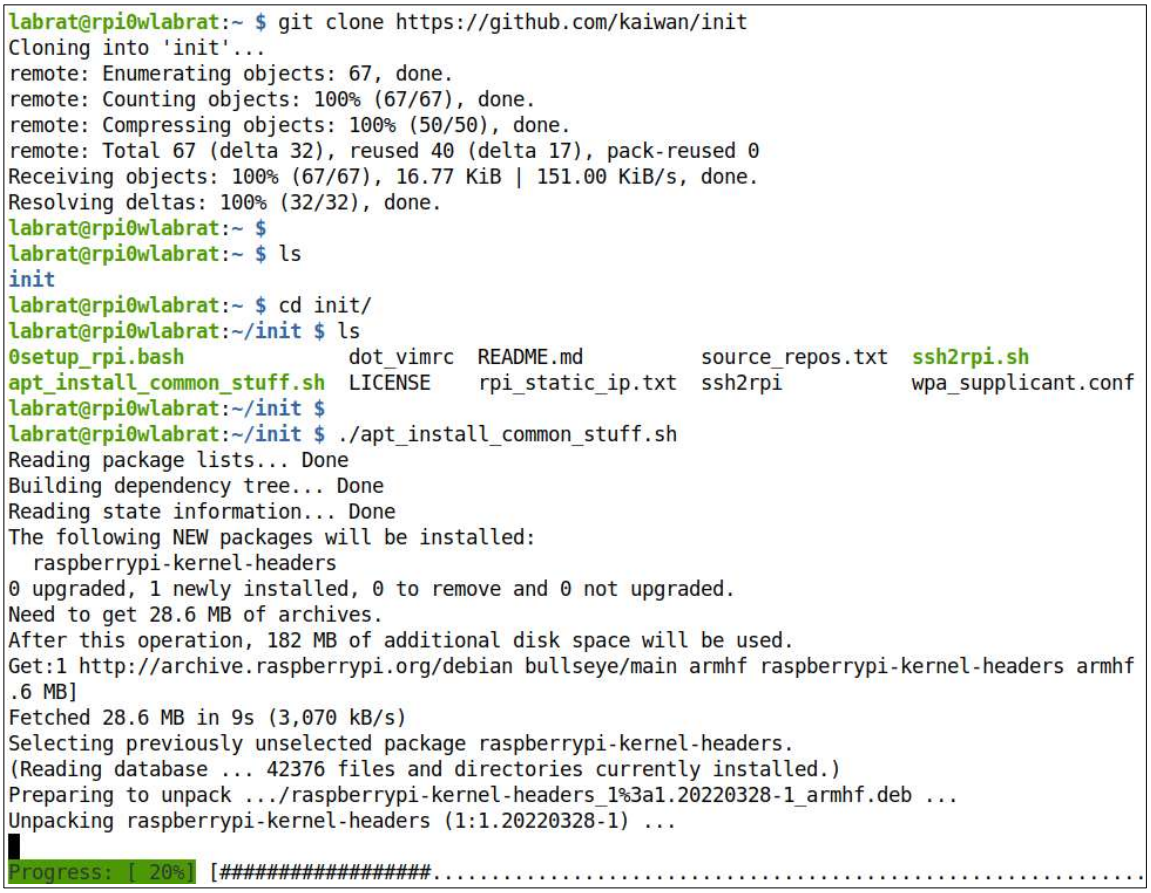
- Next, install minimally required packages first:

  `sudo apt install -y git perl`

- Then - optional, useful – to install common development + other tooling and utils:

  `git clone https://github.com/kaiwan/init`

  The partially truncated screenshot shows cloning the 'init' repo and running the *apt_install_common_stuff.sh* to install common tooling...



(This script even sets up a useful startup script that will auto-execute whenever you spawn a shell!).

There, you're all set!
