

Getting Started with Raspberry Pi

Kyle McCarty and Marc Los Huertos*

October 20, 2020

*Acknowledgments: Summer research students, Anna Burns and Alision Joseph contributed to these documents.

Contents

1	Introduction	3
1.1	What is a Raspberry Pi?	3
1.2	Why use the Raspberry Pi?	3
2	Unpacking and Connecting the Pi	3
2.1	Packaging List	3
2.1.1	“Vilros RP Zero W Basics Kit”	3
2.1.2	Other items	3
2.2	Install Raspberry Pi OS on SD card	4
2.3	Create Secure Shell Connectivity Files	4
2.4	Assemble and Connecting the Pi	10
2.4.1	Putting the Pi together	10
3	Accessing and updating Raspberry Pi OS	11
3.1	Remote connection via SSH	11
3.1.1	Installing Raspberry Pi Finder and finding Pi’s IP address	11
3.1.2	SSH connection	12
3.2	Update and Upgrading Raspberry Pi OS	18
4	Configuring the Pi	19
4.1	Changing the Pi User Password	20
4.2	Network Options	21
4.3	Boot Options	24
4.4	Localization Options	25
4.5	Interfacing Options (enable SSH)	27
4.6	Virtual Network Computing (VNC) Option	28
5	Finishing Up	31
6	Putting the Pi in the case	33

1 Introduction

1.1 What is a Raspberry Pi?

The Raspberry Pi is an tiny computer, that includes a microprocessor, a bit of memory, a slot for an SD card, input/output (I/O) ports, e.g. HDMI, USB, headphone, camera, and some general purpose input/output (GPIO) pins for various types of electrical connectors.

1.2 Why use the Raspberry Pi?

Generally, Raspberry Pis draw considerably less power than regular computers, are a lot smaller, and are relatively cost-effective. In addition, the GPIO pins allow for connecting and controlling various types of electrical components, such as LEDs and sensors. Raspberry Pis are very flexible devices. They can be used for personal computers, home surveillance systems, weather stations, adblockers for your home network, retro gaming machines, as an AI assistant, and so much more! In this class, we'll be using it as an environmental monitoring device.

2 Unpacking and Connecting the Pi

2.1 Packaging List

2.1.1 “Vilros RP Zero W Basics Kit”

1. Raspberry Pi Zero W board
2. Vilros Case, with 3 covers
3. 2.5A power supply
4. Heatsink
5. HDMI to mini-HDMI adapter
6. USB to micro-USB adapter
7. Header pin diagram
8. Camera module cable (not used)

2.1.2 Other items

1. SD card
2. SD card to USB adapter
3. USB multiport adapter
4. Breadboard
5. Wires
6. Line level converter (LLC)
7. MCP3008
8. various MQ-sensors

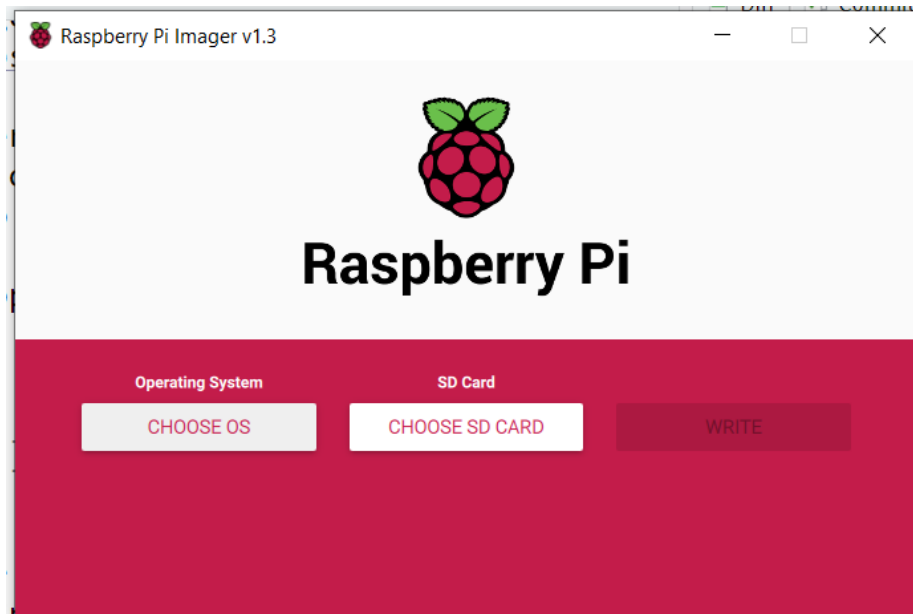


Figure 1: Raspberry Pi Image Software. Use the program to select Raspberry Pi OS (Operating System) 32-bit. Then select the SD card location. Finally, click on “Write”.

2.2 Install Raspberry Pi OS on SD card

1. Download Raspberry Pi Imager for your main computer’s operating system (OS) at <https://www.raspberrypi.org/downloads/>
2. Install Raspberry Pi Imager
3. Connect the Pi’s SD card to your main computer.
4. Use Raspberry Pi Imager to install/write Raspberry Pi OS to SD card (Figure).
5. Alternatively, manually copy Raspberry Pi OS and NOOBS to SD card, using the link above.

2.3 Create Secure Shell Connectivity Files

1. Add a “ssh” file to your SD card.¹ When the Raspberry Pi boots up, it will look for a file named “ssh” to determine if one can remotely login to

¹What is SSH protocol? SSH, also referred to as Secure Shell, is a method for secure remote login from one computer to another. It provides several alternative options for strong authentication, and it protects the communications security and integrity with strong encryption.

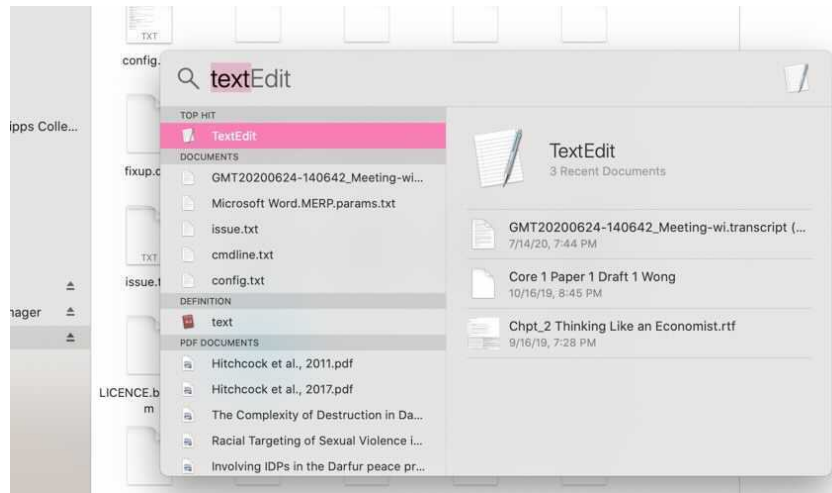


Figure 2: Finding the TextEdit application

the Pi. If the file is there, then remote login is enabled. If the file is not present, then remote login is disabled.

- Do this by creating an empty text file named “ssh”. Doing this on PCs is really easy – but as on Apple computers it frustratingly complex.

For Mac Users:

- Find and start the textedit application (Figure 2).
- Save file (Figure 3).
- Open a new document (Figure 4).
- Save file without richtext (Figure 5).
- Remove extension. (Figure 6).

For PC Users:

- Start a text editor and create an empty file named “ssh”. Make sure the file you create has no file extension. You will have to “Save As” with your text editor to ensure the file is saved the correct way; meaning the file shouldn’t end in “.txt”, “.doc”, etc. Sometimes there is the option from a dropdown to change the file extension.
- Copy “ssh” file to the SD card.
- You may be prompted and asked if you are sure you want to save the file without an extension, select “Yes.”

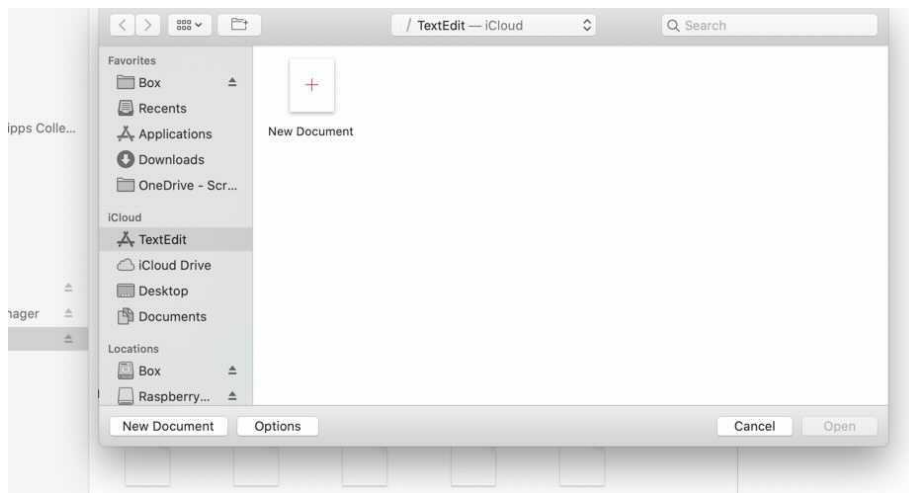


Figure 3: Save file with out richtext formatting.

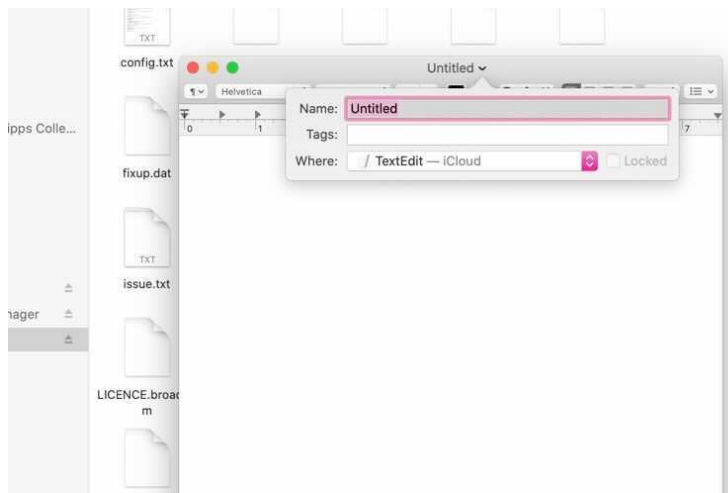


Figure 4: Save empty file

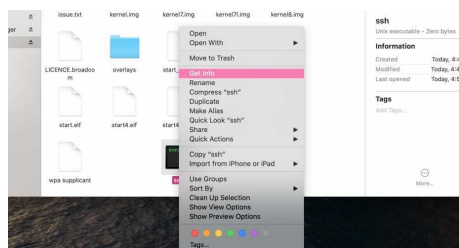


Figure 5: Get information on the file

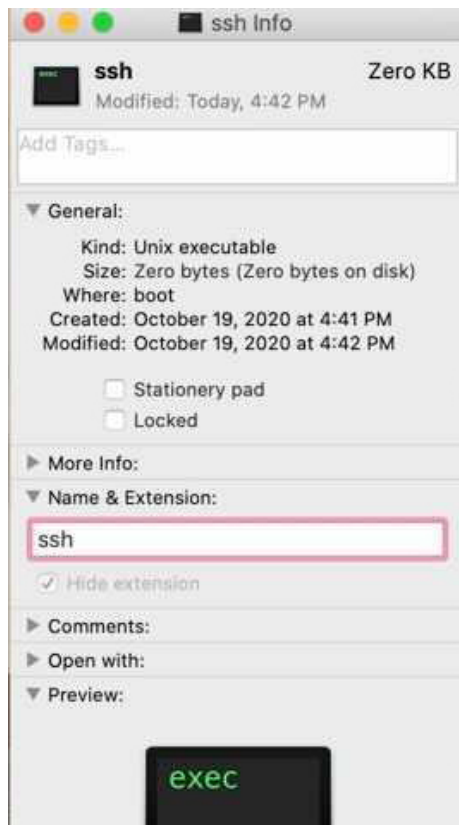


Figure 6: Remove file name extension.

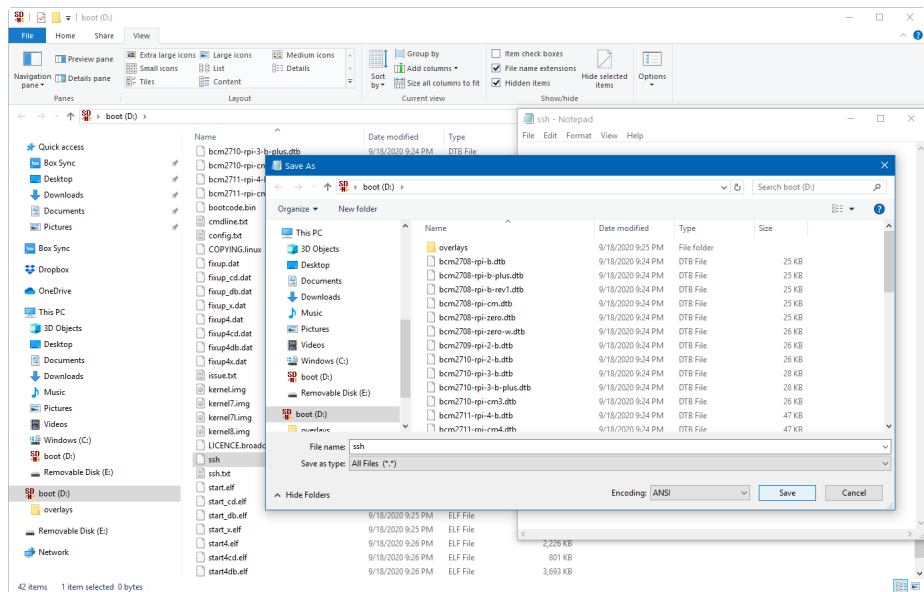
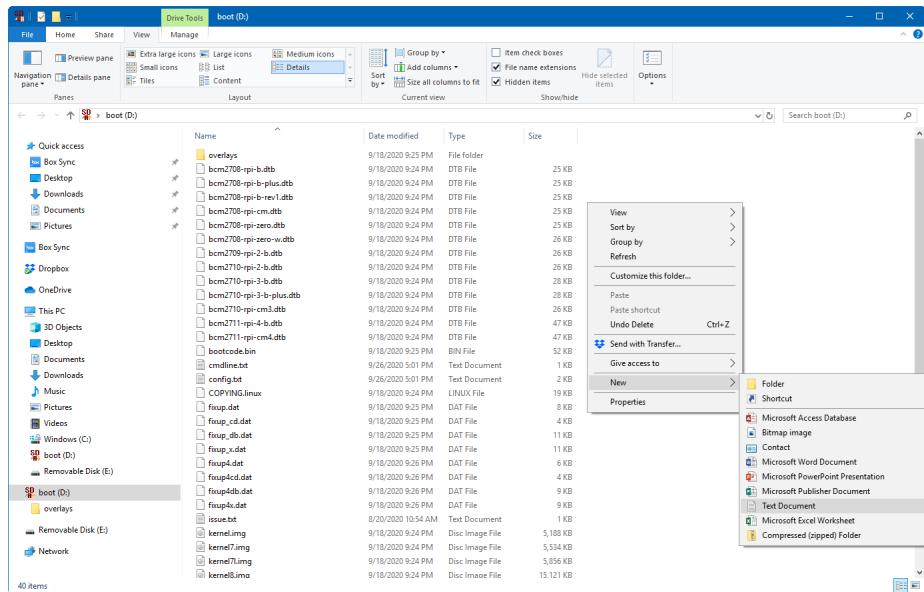


Figure 7: .

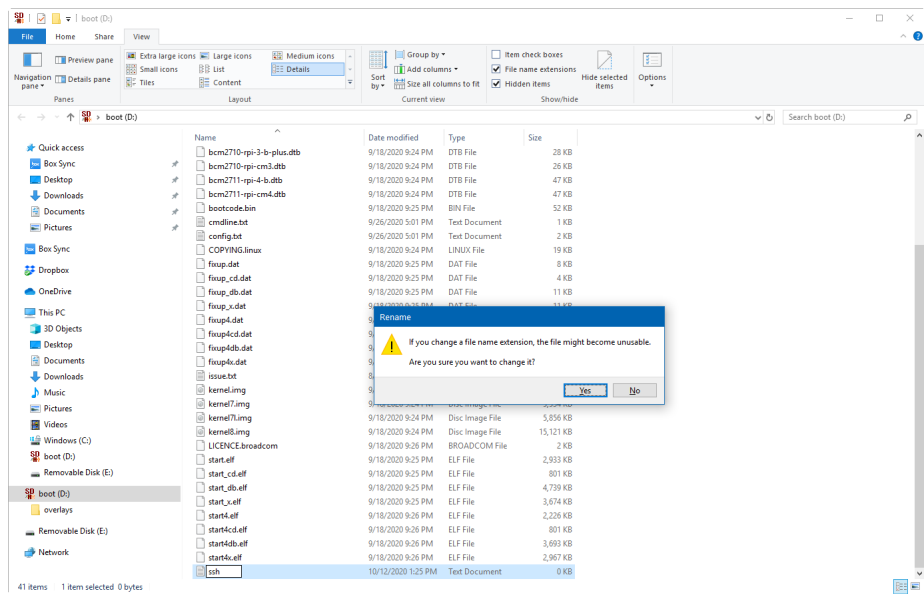


Figure 8: Saving shh file on a PC.

Create wpa_supplicant.conf file (everone!):

2. Using simpler methods as the "ssh" file, create and add a file called **"wpa_supplicant.conf"** to the boot partition² on the SD card. This can be created with any text editor.
 - Make sure the file you created has the **".conf"** file extension in the name.
 - The **"wpa_supplicant.conf"** needs to have the WiFi network information in it for the Raspberry Pi to connect on boot up.
 - Modify this file with a text editor and type the lines below. NOTE: DO NOT COPY AND PASTE. When we copy and paste from other programs hidden text can be inserted that cannot be read by the Pi's operating system correctly.

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=US

network={
    ssid="WIFI NETWORK NAME"
    psk="WIFI PASSWORD"
}
```

- If you are not in the United States, input your country's ISO code instead of **"US"** on the **"country=US"** line.
- Make sure to change **"WIFI NETWORK NAME"** to your WiFi network name, remember this name and password are case sensitive.
- Make sure to change **"WIFI PASSWORD"** to your WiFi network's password.

2.4 Assemble and Connecting the Pi

2.4.1 Putting the Pi together

1. Safely eject the SD card from your main computer.
2. Place the SD card in the Raspberry Pi Zero W's SD card slot.

NOTE: If do not have peripheral devices to be connected to the Pi, i.e. monitor, mouse and keyboard, you will be making a "headless" connection and should skip the next few steps.

 - Attach your peripheral devices, *if you have them*. This includes a monitor, mouse, and keyboard. It is okay to not have them.

²where is this?

- You will need to use the HDMI to micro-HDMI adapter to hook a monitor up to the Raspberry Pi Zero W.
 - Also, you'll need to use the Anker multiport USB adapter to use a keyboard **and** a mouse. The Raspberry Pi Zero W only has one micro-usb port.
 - You can get away with using just a monitor and keyboard, but if you are not comfortable navigating with a keyboard only, it'll be difficult.
3. Lastly, making sure the the power adapter for the Pi is plugged in to a power source, and verifying the switch on the adapter is in the “off” position, connect the power cable into the Pi.
 4. Now, turn the power switch on.

NOTE! Make sure not to turn the device off, or on/off/on, otherwise the “ssh” and “wpa_supplicant.conf” files will not be in your boot directory on the SD card anymore. If, for whatever reason, the Pi loses power after putting those file in the boot directory, you will have to go back and add them, as in step 2.3. This is **especially** important for *headless* users (users with no monitor). Once we complete the set up, via the section ?? we won't need to worry about this issue.

3 Accessing and updating Raspberry Pi OS

Now, there are different ways to remotely connect with the Raspberry Pi if you are a *headless* user. Unfortunately though, you will have to connect to the Pi using **SSH** the **first time**. After that, you can install a remote server on the Pi so you can connect with it using a program with a GUI. I recommend using SSH, though, as it is a *much* quicker connection.

3.1 Remote connection via SSH

3.1.1 Installing Raspberry Pi Finder and finding Pi's IP address

The goal with this step is to find the **local IP address** of the Pi. There are a lot of different ways to do this. If you are computer savvy, go ahead and find the IP address of your Pi and ignore this step.

1. The easiest way to find the IP address of your Pi, if you don't know networking or computers that well, is to use the **Raspberry Pi Finder** by Adafruit.
2. With your main computer, go to this website to download the application for your OS:
<https://github.com/adafruit/Adafruit-Pi-Finder/>. However, this doesn't work for everyone. For Mac users I suggest try this

<http://ivanx.com/ivanx/raspberrypi/>. If this program works, follow the instructions and skip to Section 3.2.

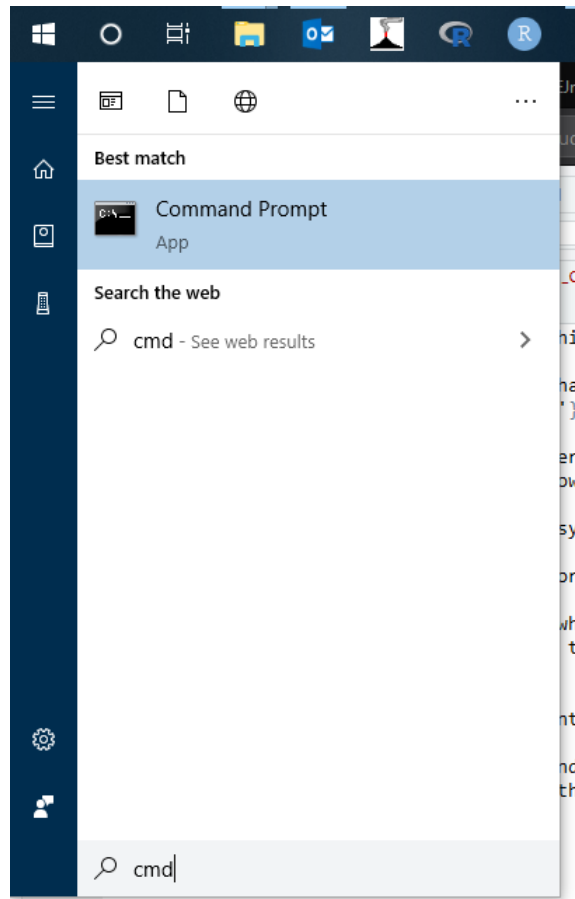
3. Scroll down and click the link that says “**Download the latest release**”
4. Scroll down to “**Assets**” and there you will see the .zip files of the program for the different OSs.
 - “**osx**” is for Mac users.
 - “**win32**” is for Windows user.
5. Download the .zip file for your system and then unzip it when finished downloading.
6. Run Raspberry Pi finder.³
7. Click “**Find my Pi!**” for the program to locate your Pi. Wait a few minutes if it doesn’t find it immediately. Sometimes it can take quite a while!⁴
8. The IP address should be listed when finished. It should look something like “**192.168.1.XXX**”.
9. Use the IP address to connect to the Pi.

3.1.2 SSH connection

- For Windows users, you will want to use the **Command Prompt**.
 1. Press the **Windows** key on your keyboard, type “**CMD**”, and **Enter**.

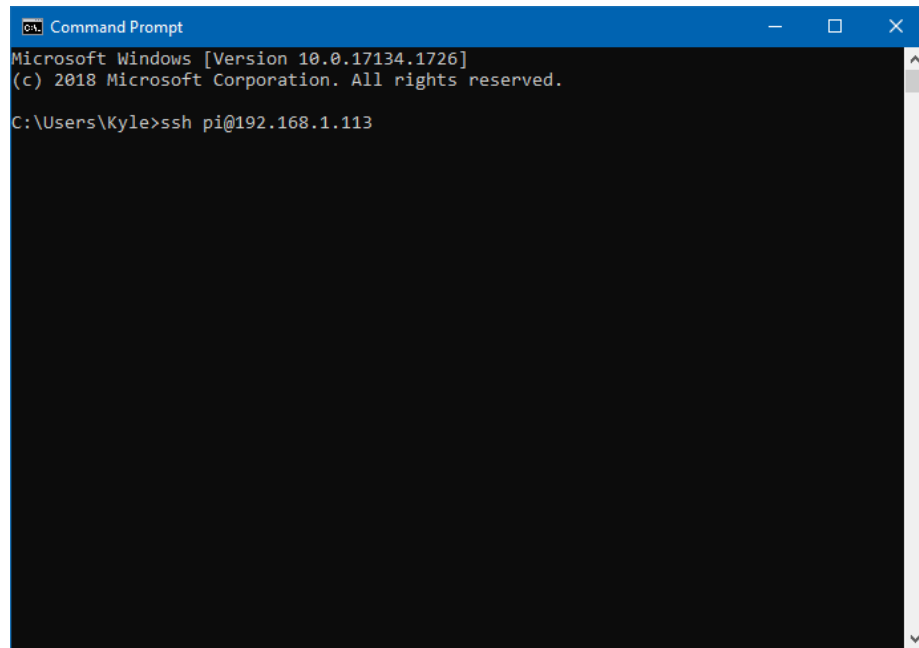
³where do we the the exe file?– doesn’t ren all by itself.

⁴need a workaround, this doesn’t work...



2. In the Command Prompt, type the command below making sure to substitute “192.168.0.XXX” with the IP address found in step 8

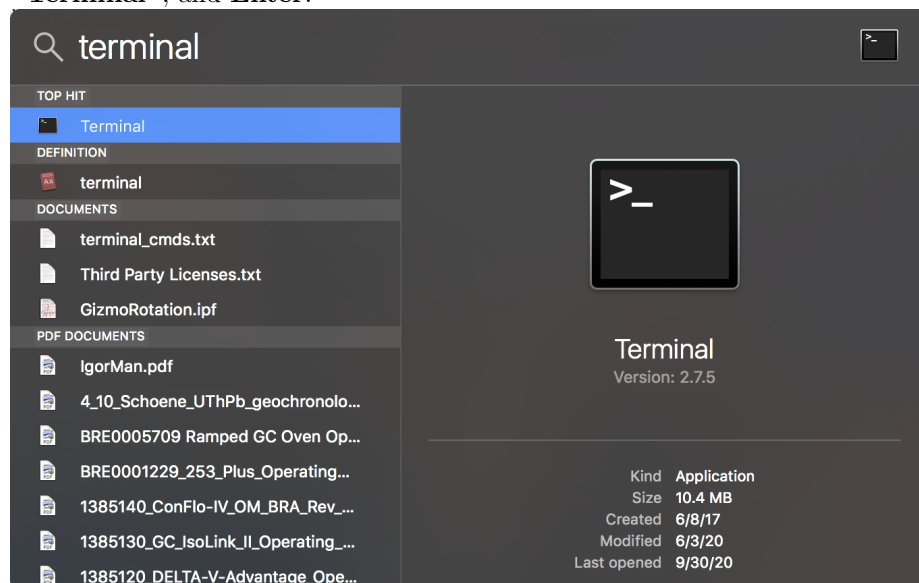
```
ssh pi@192.168.0.XXX
```



```
Command Prompt
Microsoft Windows [Version 10.0.17134.1726]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Kyle>ssh pi@192.168.1.113
```

- For Mac users, you will want to use the *Terminal*.
 1. Press the *Command + Spacebar* keys on your keyboard, type “**Terminal**”, and **Enter**.



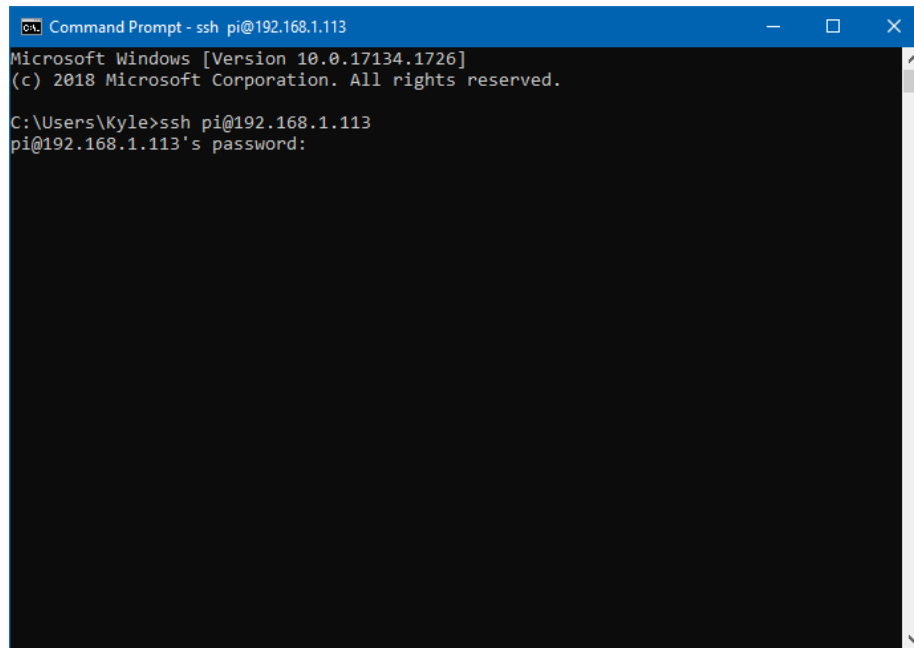
2. In Terminal, type the command below making sure to substitute “192.168.1.XXX” with the IP address found in step 8

```
ssh pi@192.168.1.XXX
```



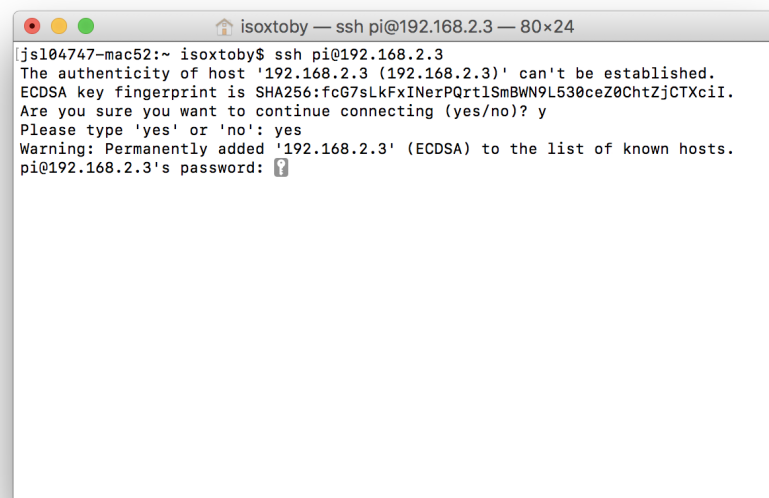
NOTE: Sometimes we get a warning that the host key needs to be changed. If you get a yes/no prompt, type yes. If you get a warning that the key needs to be changed or that someone is snooping then contact the instructor for help with this.

1. For both Mac and Windows users, you should now see a prompt in the command line asking for the Pi's password.



```
Command Prompt - ssh pi@192.168.1.113
Microsoft Windows [Version 10.0.17134.1726]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Kyle>ssh pi@192.168.1.113
pi@192.168.1.113's password:
```



```
isoxtoby — ssh pi@192.168.2.3 — 80x24
[jsl04747-mac52:~ isoxtoby$ ssh pi@192.168.2.3
The authenticity of host '192.168.2.3 (192.168.2.3)' can't be established.
ECDSA key fingerprint is SHA256:fcG7sLkFxINerPQrtlSmBWN9L530ceZ0ChtZjCTXciI.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '192.168.2.3' (ECDSA) to the list of known hosts.
pi@192.168.2.3's password: ?
```

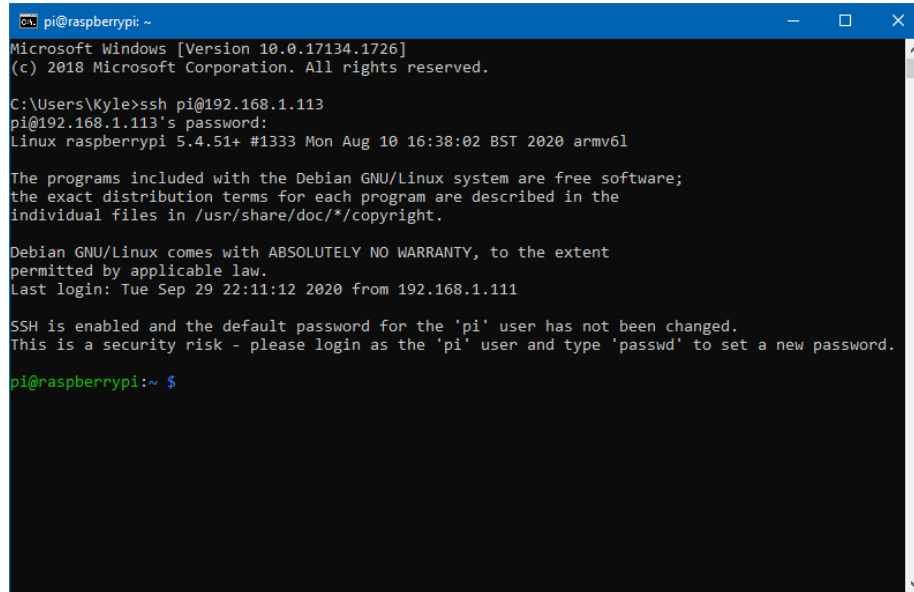
By default, the password is “**raspberry**”.

NOTE The command line won’t show the password while you are typing it, nor will it show placeholder characters. You will need to just type it

and **Enter**.

2. If all went well, you should see that you've connected to your Pi via the command line interface (CLI). You should see the text:

```
pi@raspberrypi/:~ $ _
```



```
pi@raspberrypi: ~
Microsoft Windows [Version 10.0.17134.1726]
(c) 2018 Microsoft Corporation. All rights reserved.

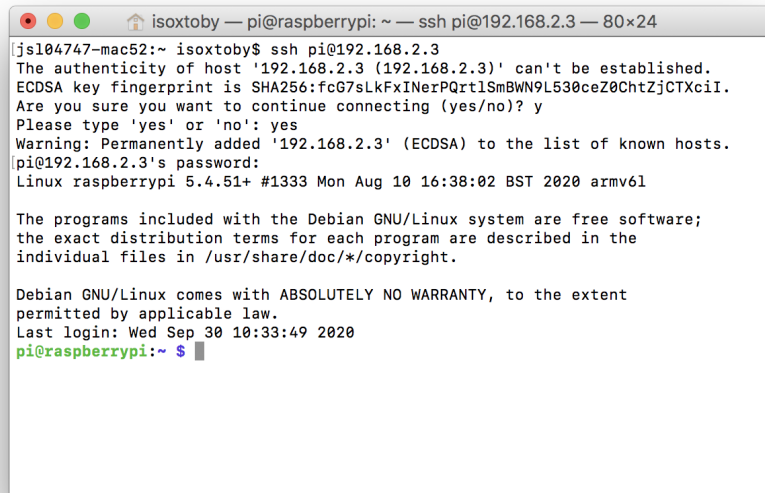
C:\Users\Kyle>ssh pi@192.168.1.113
pi@192.168.1.113's password:
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 29 22:11:12 2020 from 192.168.1.111

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $
```



```
isoxtoby — pi@raspberrypi: ~ — ssh pi@192.168.2.3 — 80x24
jsl04747-mac52:~ isoxtoby$ ssh pi@192.168.2.3
The authenticity of host '192.168.2.3 (192.168.2.3)' can't be established.
ECDSA key fingerprint is SHA256:fcG7sLkFxiNerPQrtlSmBWN9L530ceZ0ChtZjCTXciI.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '192.168.2.3' (ECDSA) to the list of known hosts.
pi@192.168.2.3's password:
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 30 10:33:49 2020

pi@raspberrypi:~ $ █
```

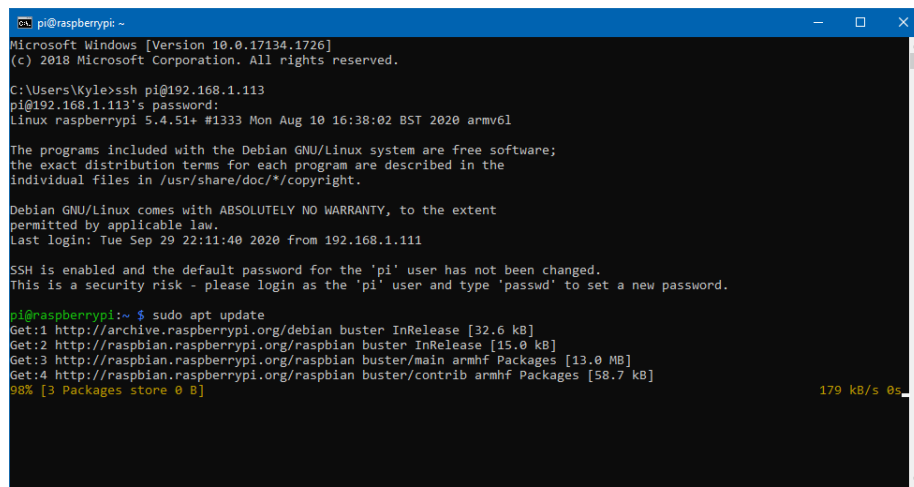
3. There will also be a warning that the Pi's password hasn't been changed. We will want to change that, but we are not going to do that right now.

3.2 Update and Upgrading Raspberry Pi OS

Here we will be updating the Raspberry Pi. This is the first thing we are going to want to do to make sure there are now errors later on. We will need to implement a few commands into the Pi's command line. Don't be afraid! Its easy!

1. First we are going to want the Pi to talk to the official Raspberry Pi servers to get the latest version list of our programs on the Pi. In your SSH session, type the following command:

```
sudo apt update
```



```
pi@raspberrypi ~  
Microsoft Windows [Version 10.0.17134.1726]  
(c) 2018 Microsoft Corporation. All rights reserved.  
  
C:\Users\Kyle>ssh pi@192.168.1.113  
pi@192.168.1.113's password:  
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Tue Sep 29 22:11:40 2020 from 192.168.1.111  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.  
  
pi@raspberrypi:~$ sudo apt update  
Get:1 http://archive.raspberrypi.org/debian buster InRelease [32.6 kB]  
Get:2 http://raspbian.raspberrypi.org/raspbian buster InRelease [15.0 kB]  
Get:3 http://raspbian.raspberrypi.org/raspbian buster/main armhf Packages [13.0 MB]  
Get:4 http://raspbian.raspberrypi.org/raspbian buster/contrib armhf Packages [58.7 kB]  
98% [3 Packages store 0 B] 179 kB/s 0s
```

2. Secondly, we want the Pi to compare the version list with its current packages and programs and update where needed. In your SSH session, type:

```
sudo apt full-upgrade
```

3. You will be asked if you are sure you want to upgrade. Type “y” and **Enter**.

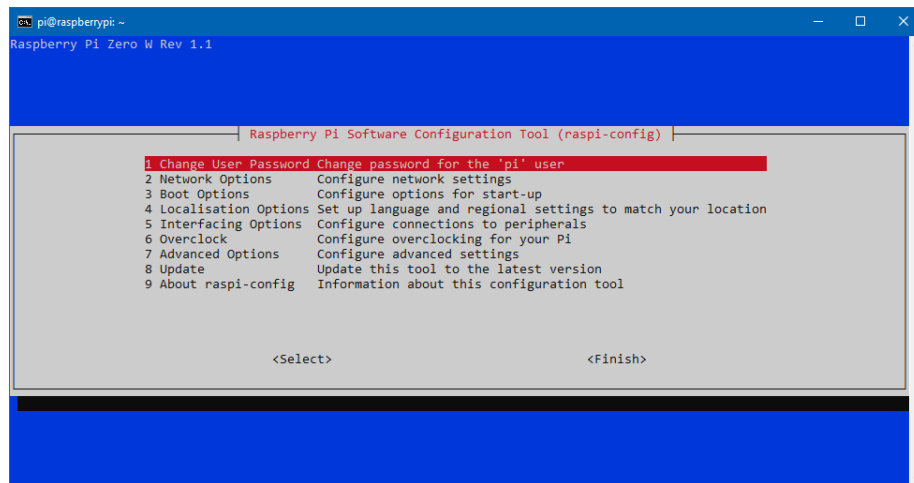
- If **power** was severed from the Pi, you will need to repeat section 2.3 over again. The Pi discards these files after it boots up, so if you shut off the Pi, or lose power, you'll have to re-add them to the boot directory.

4.1 Changing the Pi User Password

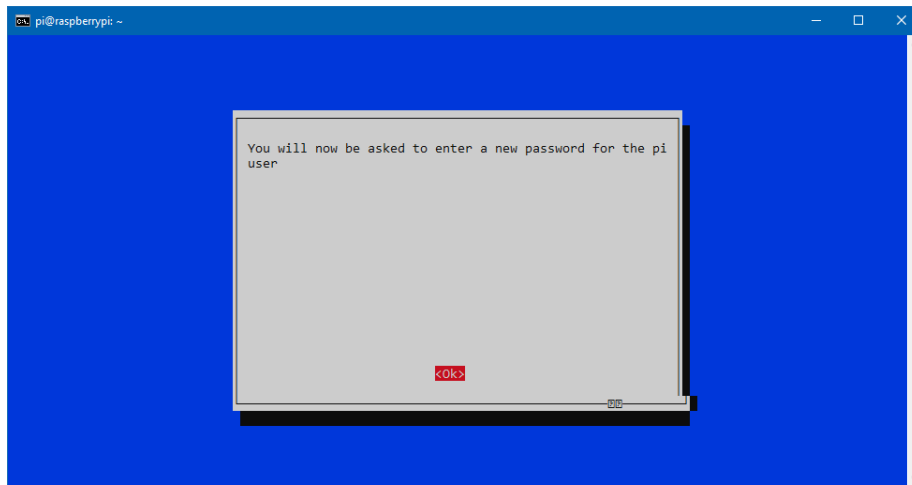
As a security measure, since your Pi is in your WiFi network, you'll want to change the password for the Pi.

1. Run the Raspberry Pi configuration utility by using this command in the CLI:

```
sudo raspi-config
```



2. Navigate to “Change User password for the 'pi' user” and hit **Enter**.
3. It will prompt you that it is going to ask for the new password. Press **Enter**, and type your new password followed by **Enter** again.



4. Verify the password by typing it again.

```

pi@raspberrypi ~
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Kyle>ssh pi@192.168.1.113
pi@192.168.1.113's password:
Permission denied, please try again.
pi@192.168.1.113's password:
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 30 00:03:22 2020 from 192.168.1.111

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ sudo raspi-config

New password:
Retype new password:

```

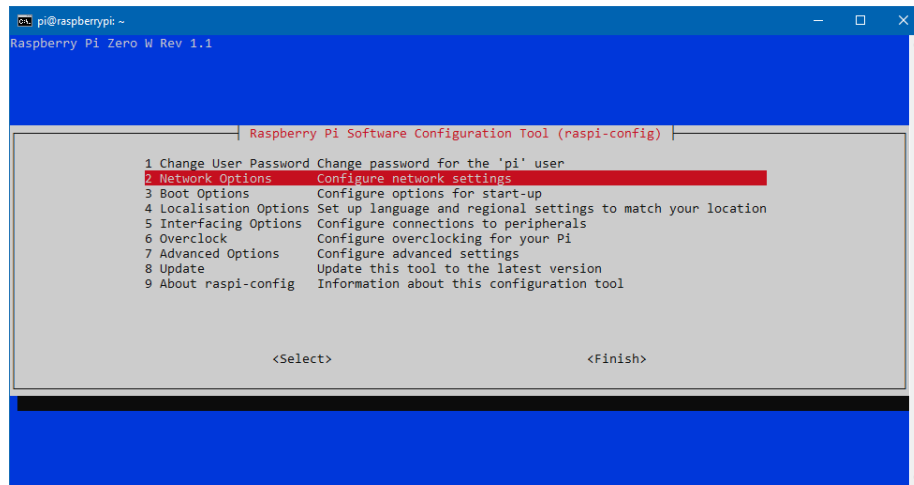
5. Your Pi now has your new password. Don't forget it!

4.2 Network Options

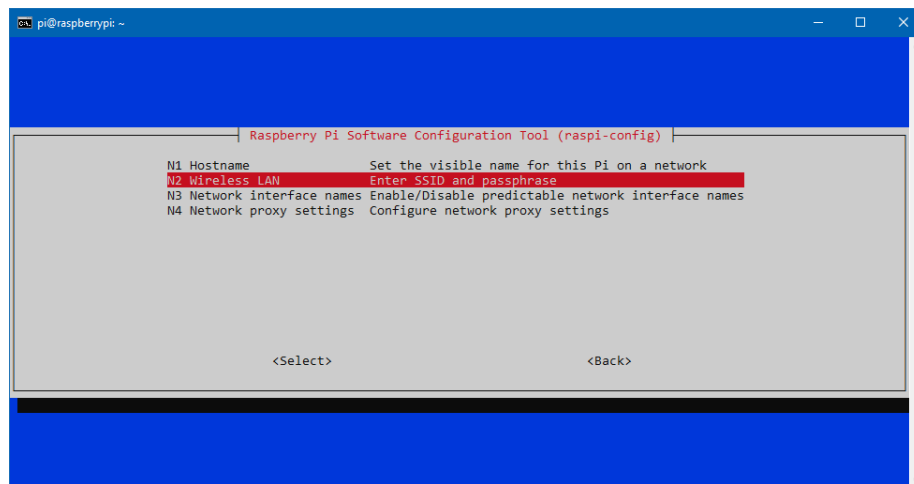
We want to make sure the Pi's network options are configured properly so we don't have to add a **wpa_supplicant.conf** file to the boot directory when booting anymore.

Note: This section has caused problems. If you experience issues at this point where the Pi stops communicating, skip this step.

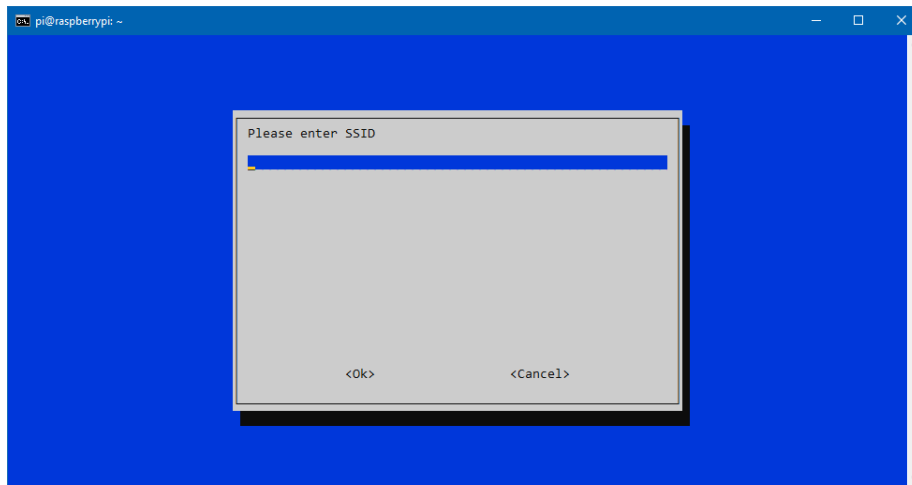
1. While still in the raspi-config utility, navigate to “**Network Options**”.



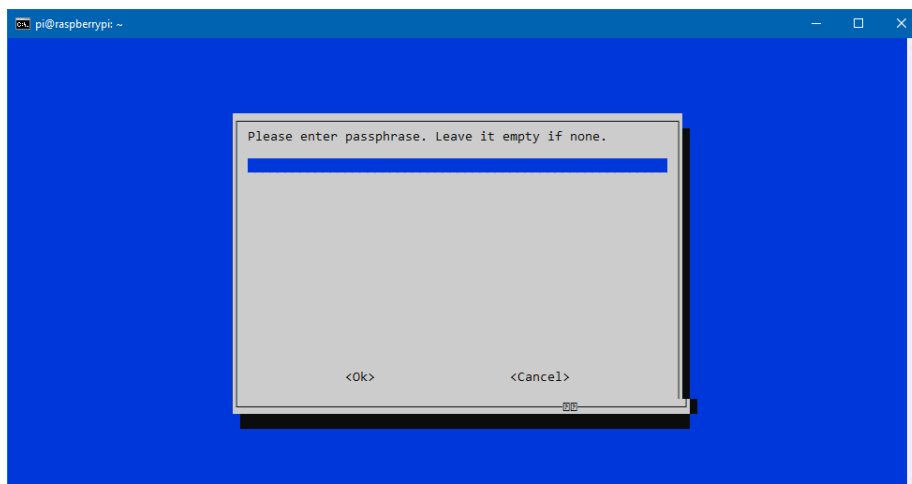
2. Navigate to “Wireless LAN” and hit **Enter**.



3. It will prompt you to enter your WiFi **SSID**. This is the name of the WiFi network. Enter it and press **Enter**.



4. Next it will ask for the WiFi password/passphrase. Type that in and hit **Enter**.



5. If done correctly, it will notify you it was successfully done via the CLI.

```
pi@raspberrypi ~  
C:\Users\Kyle>ssh pi@192.168.1.113  
pi@192.168.1.113's password:  
Permission denied, please try again.  
pi@192.168.1.113's password:  
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Wed Sep 30 00:03:22 2020 from 192.168.1.111  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.  
  
pi@raspberrypi:~$ sudo raspi-config  
  
New password:  
Retype new password:  
passwd: password updated successfully  
pi@raspberrypi:~$
```

6. Your Pi now has the wireless network configured properly.

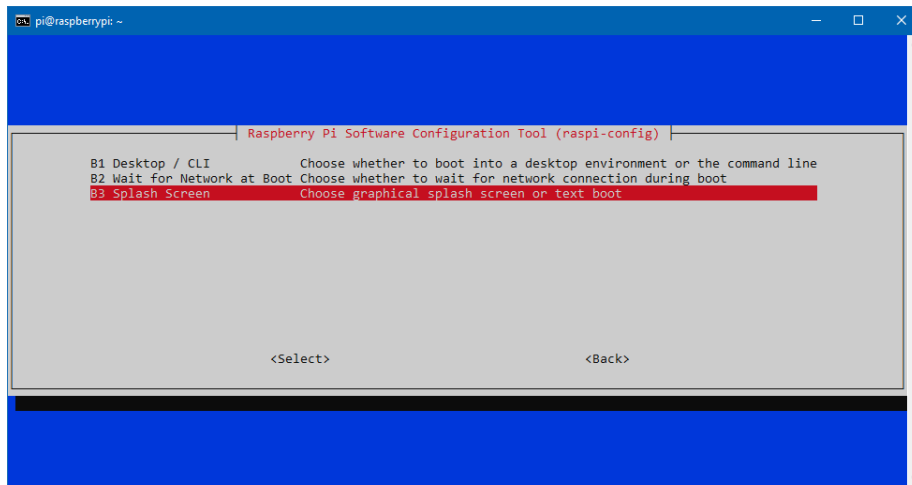
4.3 Boot Options

This next configuration is more for fun and is **OPTIONAL**. We are going to make the Raspberry Pi “text boot” instead of just showing a splash screen. This is fun because it shows you the boot process via the CLI so you can see *exactly* what is starting during boot up!

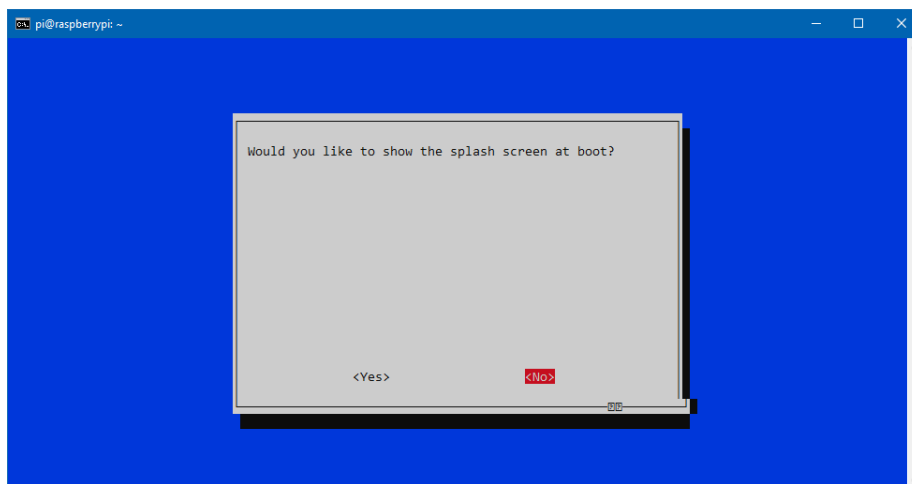
1. While still in the raspi-config utility, navigate to “Boot Options”.

```
pi@raspberrypi ~  
Raspberry Pi Zero W Rev 1.1  
  
Raspberry Pi Software Configuration Tool (raspi-config)  
  
1 Change User Password Change password for the 'pi' user  
2 Network Options Configure network settings  
3 Boot Options Configure options for start-up  
4 Localisation Options Set up language and regional settings to match your location  
5 Interfacing Options Configure connections to peripherals  
6 Overclock Configure overclocking for your Pi  
7 Advanced Options Configure advanced settings  
8 Update Update this tool to the latest version  
9 About raspi-config Information about this configuration tool  
  
<Select> <Finish>
```

2. Navigate to “Splash Screen”.



3. It will prompt you if you want to see the **splash screen** at boot. Select “NO”

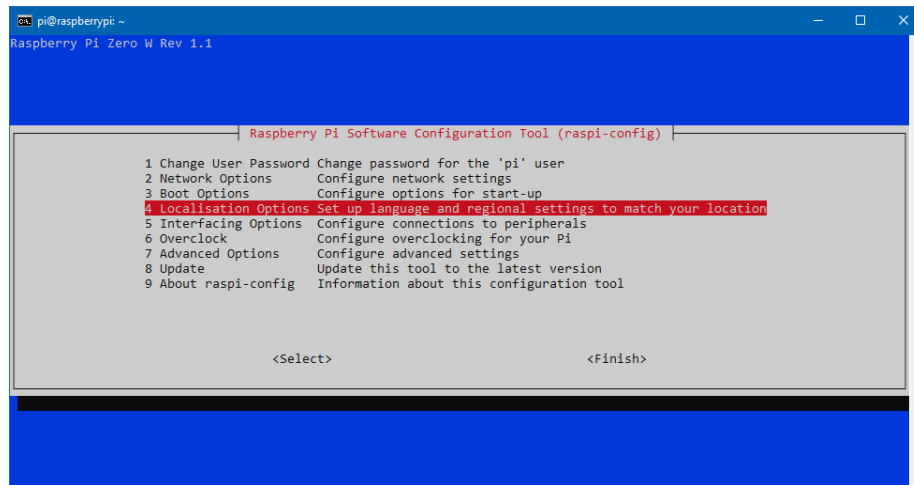


4. It will then tell you that the splash screen at boot is disabled. Woohoo!

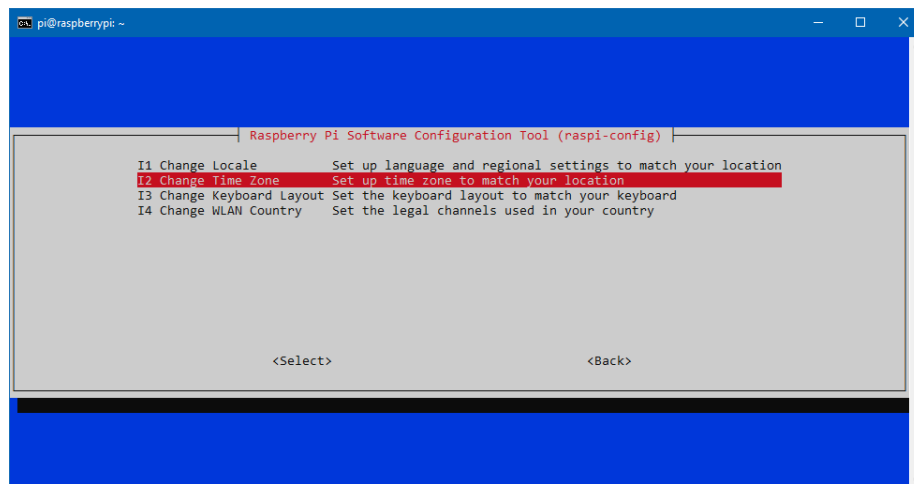
4.4 Localization Options

This may or may not be a problem for some people. I had to make sure my time zone was set correctly. I believe by default the Pi's time zone is set for Europe.

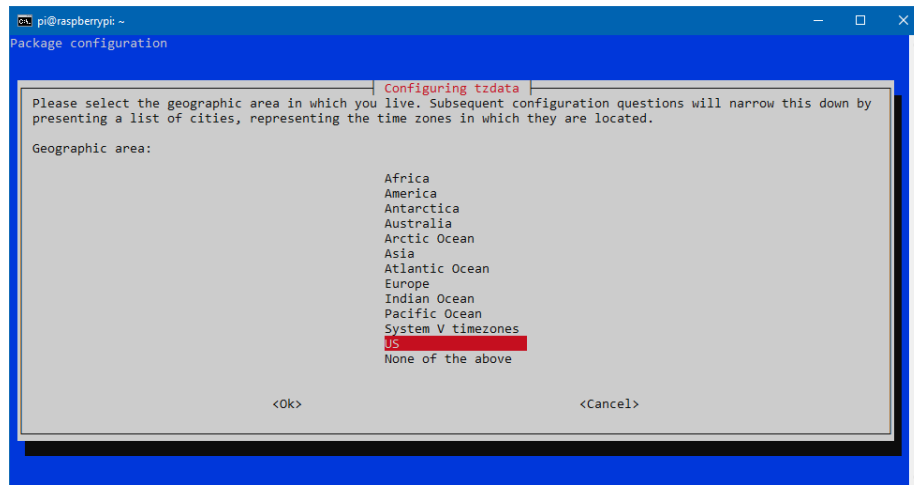
1. While still in the raspi-config utility, navigate to “**Localisation Options**”.



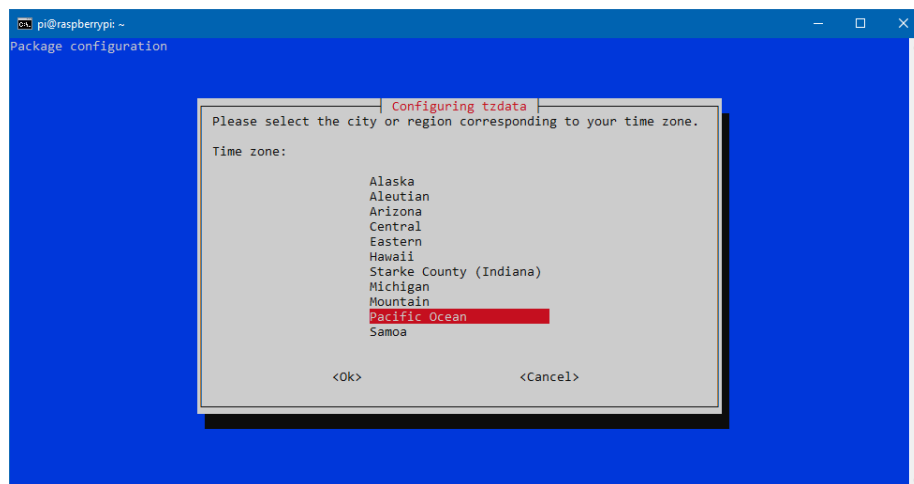
2. Navigate to “Change Time Zone”.



3. In a few seconds it will prompt you to choose your **geographic area**.
Select whichever corresponds to yours. In my case it was the US.



4. It will then tell ask you what your **time zone** is. Select what applies. In my case is was Pacific Ocean.



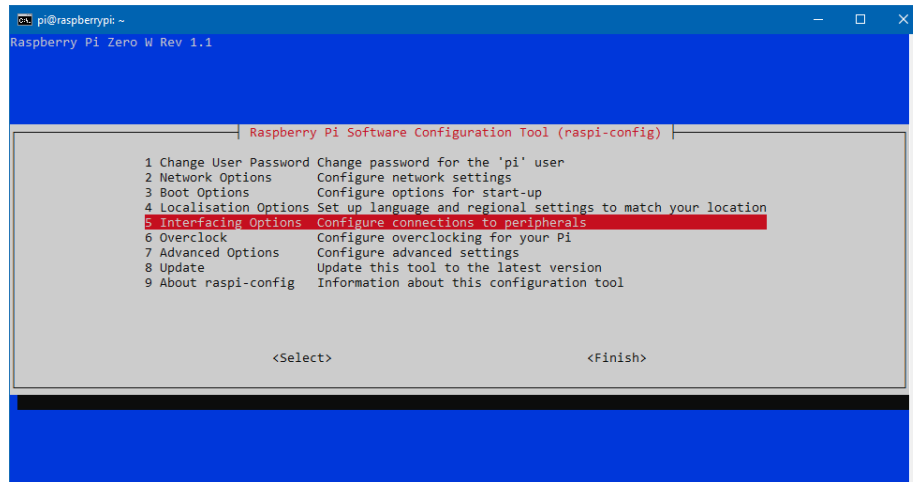
5. Then, it will flash the CLI with the Coordinated Universal Time (UTC) and your time zone's time and go back to the configuration utility.

4.5 Interfacing Options (enable SSH)

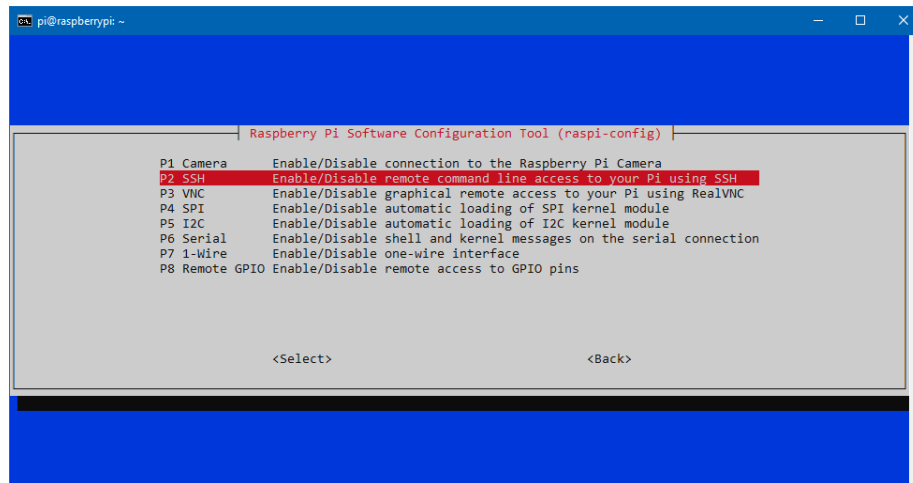
IMPORTANT We need to make sure we enable SSH in the Pi's configuration. This is extremely important and allows us to connect to the Pi via SSH in all future instances without having to add a ssh file to the boot directory.

1. While still in the raspi-config utility, navigate to **“Interfacing Options”**. Remember where this is located because we will need to access this later

on.



2. Navigate to “SSH”.

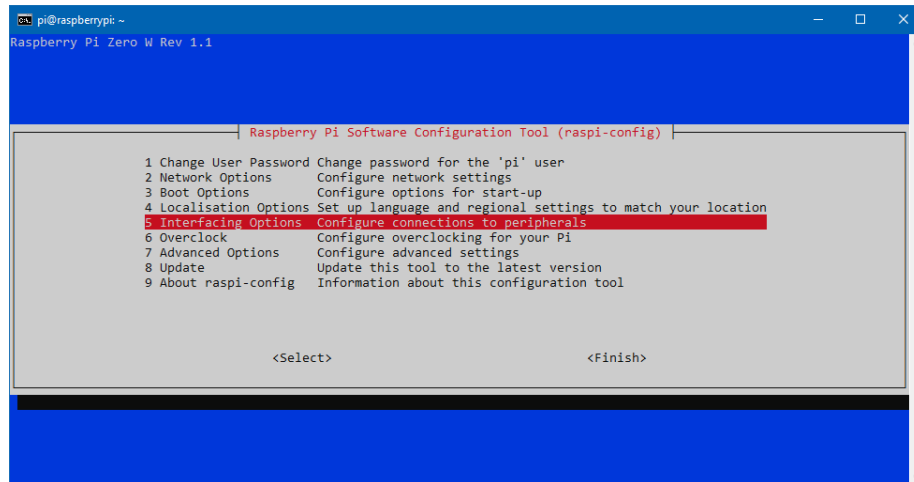


3. The Pi will then ask if you want the **SSH server** enabled. Select **Yes**.
4. Wait a few seconds and the Pi should notify you the SSH server is enabled.

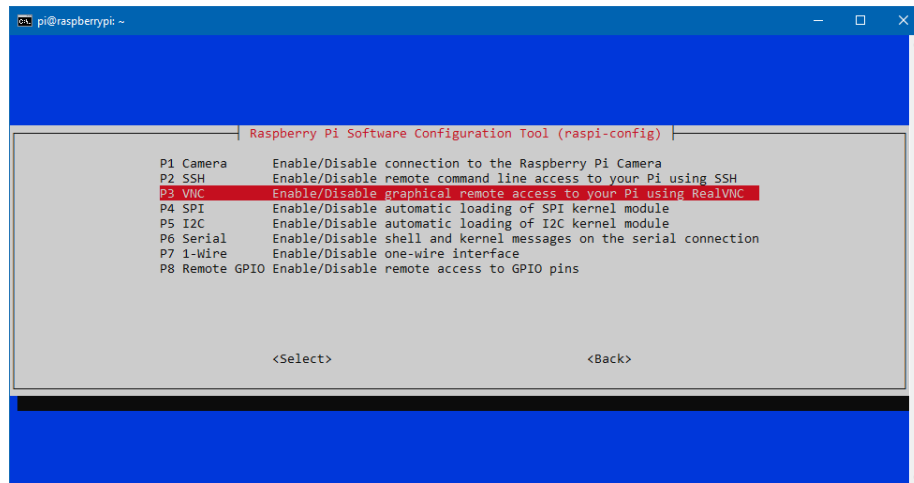
4.6 Virtual Network Computing (VNC) Option

NOTE For those who find the CLI a bit tricky, or just don't like it, you can enable a virtual network computing software called RealVNC before shutting down the Pi. This will let you control the Pi via a graphical user interface (GUI), like what people are normally used to on MacOS or Windows.

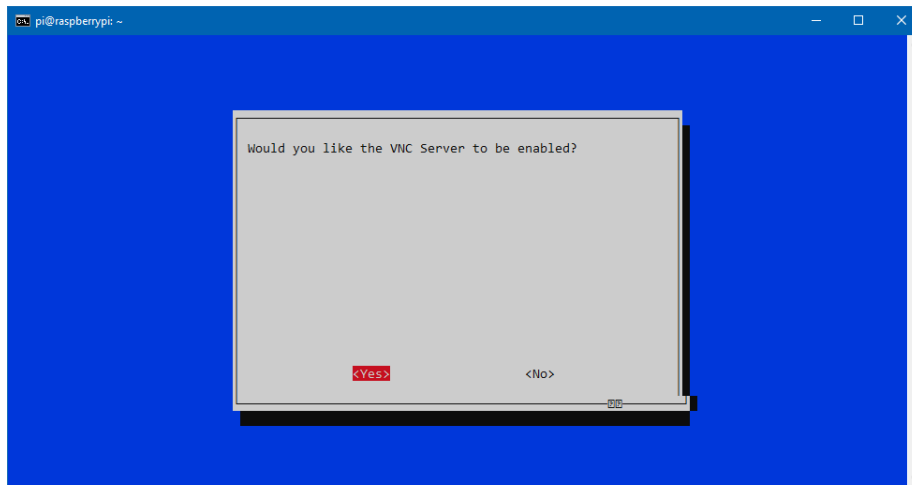
1. While still in the raspi-config utility, navigate to “Interfacing Options”.



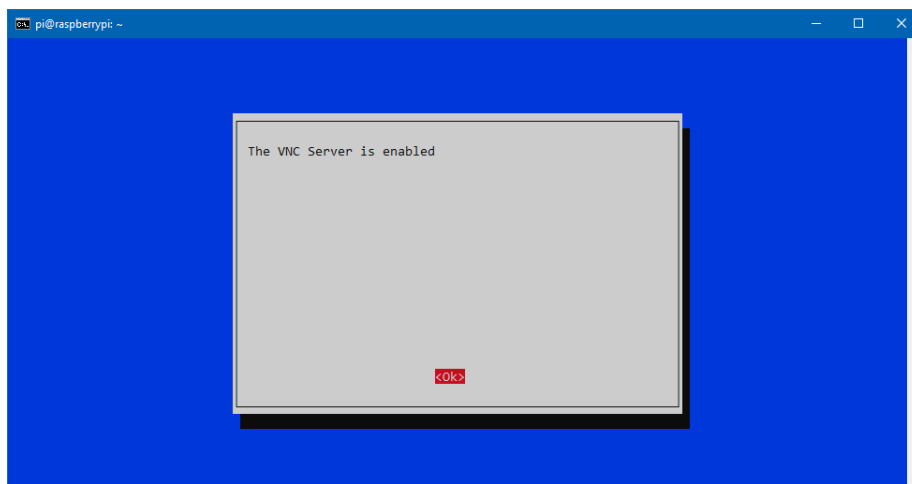
2. Navigate to “VNC” and select it.



3. The Pi will prompt you if you want the **VNC Server** enabled. Select “Yes”.



4. Wait a second and the Pi should let you know that the **VNC Server** is now enabled. This server will start automatically when the Pi boots up.

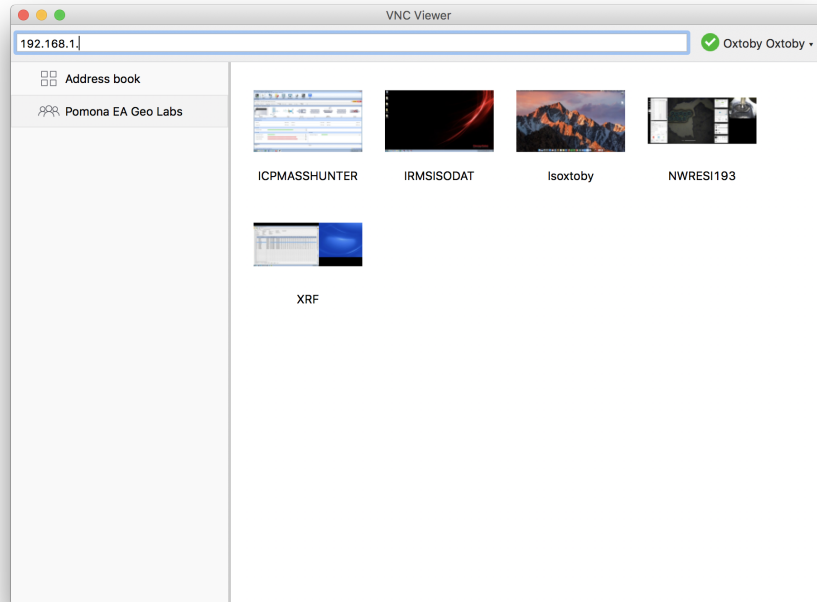


5. You still need to download **RealVNC Viewer** on the computer you want to remotely access the Pi with.
6. **RealVNC Viewer** can be downloaded at:
<https://www.realvnc.com/en/connect/download/viewer/>^{5 6}

⁵I don't think this will work – I prefer using remote desktop connection... why did you pick this one?

⁶Raspberry Pi OS comes with RealVNC Server already installed, you just have to enable the option (covered in the SOP). The other requirement is that you download RealVNC Viewer on the computer you want to VNC in with. I chose this route because it already is sort of equipped with RasPi OS, and the Viewer is cross-platform so I wouldn't need to write two sections of the SOP, on OSX and Windows.

7. Download the client that is for your OS and then install it.
8. When you open VNC Viewer, you should see something like the image below. In the top toolbar, input the IP address of the Pi and hit “**Enter**”.

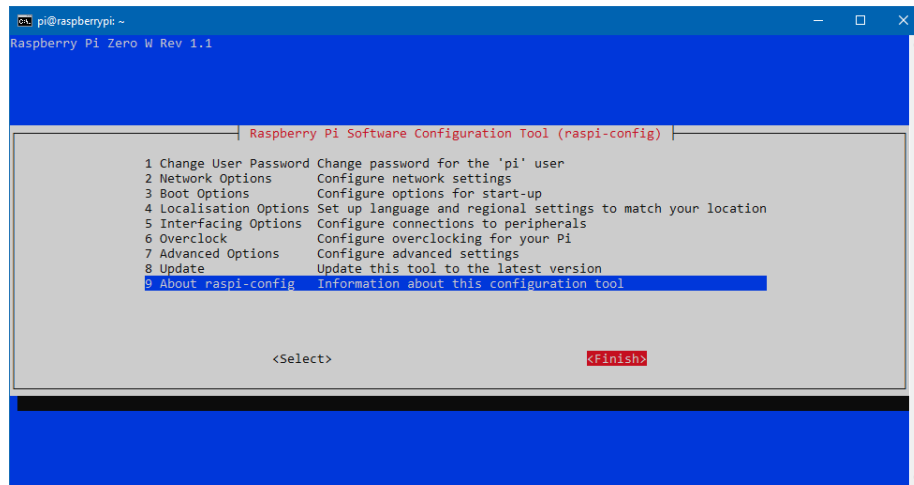


9. You will be prompts for the username (pi) and the password – These are your pi’s username and password. Remember changing your password will be a good idea a some point.
10. Wait a minute and you should eventually see the desktop of the Pi. You can can browse and manipulate with your mouse etc.

5 Finishing Up

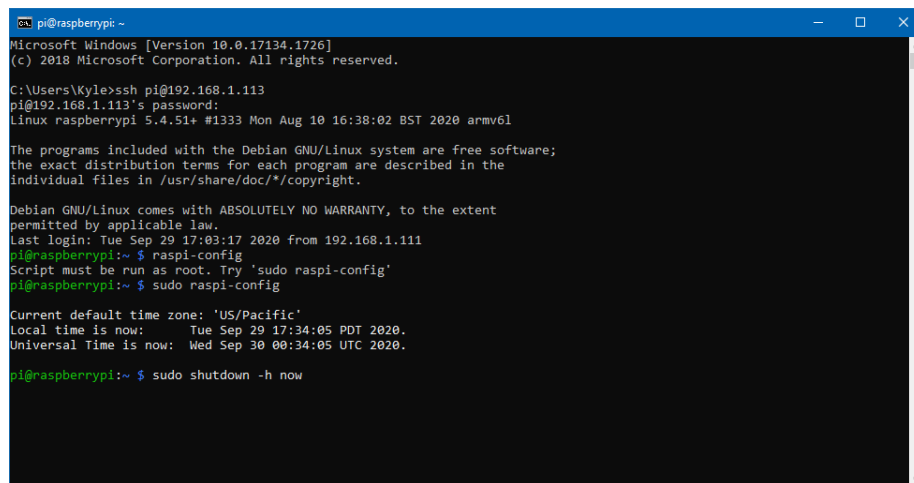
You Pi should be configured now! When the Pi does get turned off and turned back on, it should boot and connect to your WiFi and allow you to SSH into it. Now, we still need to close the Pi’s configuration utility and shut down the Pi.

1. While still in the raspi-config utility, navigate down as far as it lets you (**About raspi-config**).



2. You will have to press the right arrow key to get down to “**FINISH**”. Tricky and a little unintuitive, I know...
3. Select **FINISH** and it should exit the utility back to the CLI.
4. Next, run this last command in the CLI to shutdown the Pi:

```
sudo shutdown -h now
```



5. Soon the Pi will shutdown and you'll get a message in your Command Prompt/Terminal something along the lines that the connection was closed or lost.


```
Command Prompt
Microsoft Windows [Version 10.0.17134.1726]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Kyle>ssh pi@192.168.1.113
pi@192.168.1.113's password:
Linux raspberrypi 5.4.51+ #1333 Mon Aug 10 16:38:02 BST 2020 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 29 17:03:17 2020 from 192.168.1.111
pi@raspberrypi:~$ raspi-config
Script must be run as root. Try 'sudo raspi-config'
pi@raspberrypi:~$ sudo raspi-config

Current default time zone: 'US/Pacific'
Local time is now: Tue Sep 29 17:34:05 PDT 2020.
Universal Time is now: Wed Sep 30 00:34:05 UTC 2020.

pi@raspberrypi:~$ sudo shutdown -h now
Connection to 192.168.1.113 closed by remote host.
Connection to 192.168.1.113 closed.

C:\Users\Kyle>
```

6. Flip the switch on the power supply to the Pi to make sure power is not going to the Pi.

CONGRATULATIONS! You done configuring the Pi. *Whew...*

6 Putting the Pi in the case

- Attach the Pi to the bottom Vilros case making sure to line the dowels within the case with the mounting holes on the Pi.
- **Do not** attached the top part of the case yet. Being able to see the Pi's board will help you with pin determination.