

RaspberryPi_Travel_Router

Introduction

In this document we'll show you how to edit the ansible playbooks to configure a Raspberry Pi as a travel router that auto connects to an OpenVPN Access Server. This guide was written with the assumption of living in the USA. Playbooks were developed and tested from an ansible control node running ansible [core 2.12.2] at the time of writing.

What's covered

- How to create a bootable microSD card with Raspberry Pi OS.
- How to configure and run these playbooks with your settings.

What's Not covered

- How to setup an OpenVPN Server's
- How to setup your ansible control server (It just needs ansible & git installed)

Before you begin

You'll need:

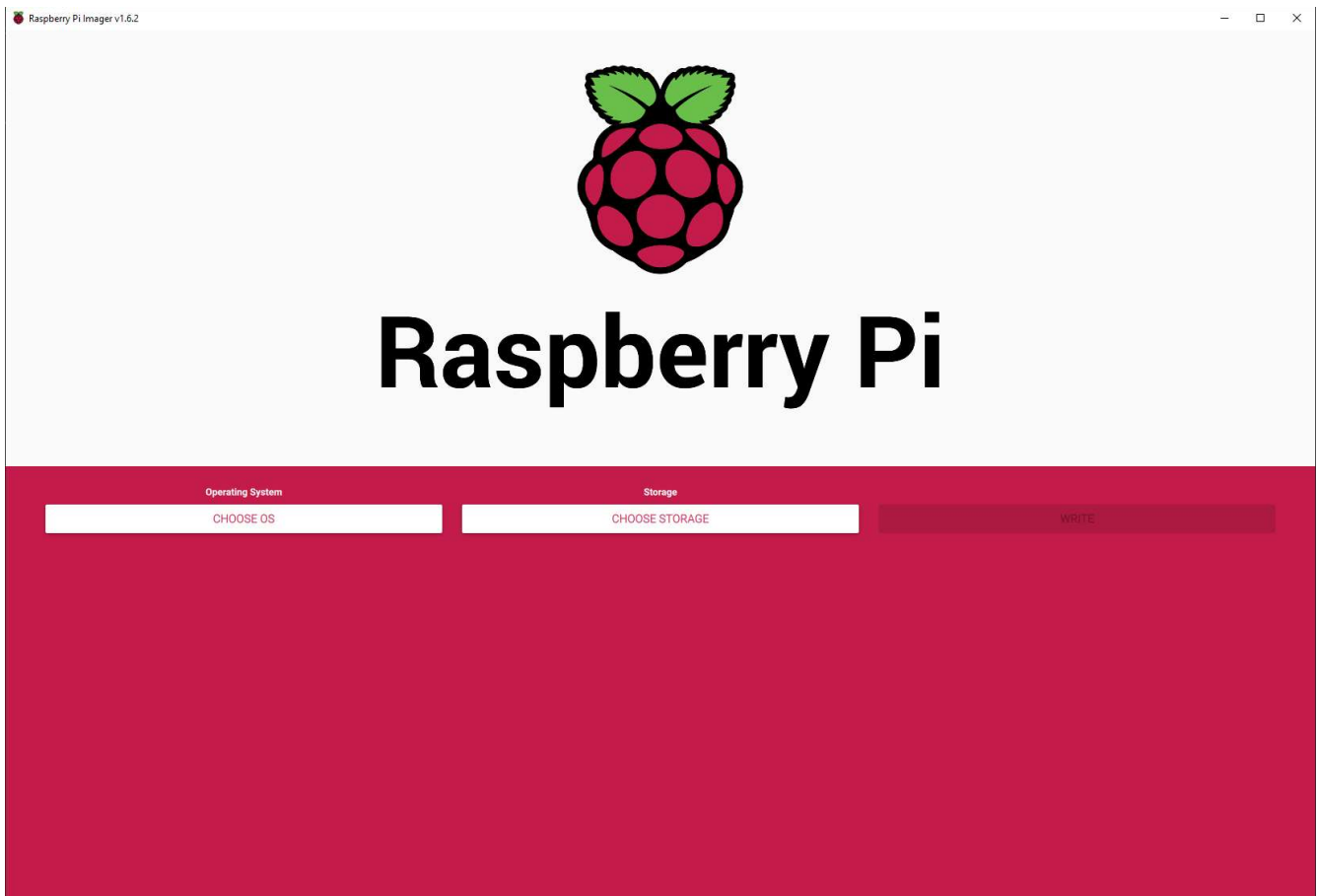
- A Raspberry Pi (I used a Pi 3B+ and a 4b for testing)
- A microSD card (8 GB or more recommended).
- A USB dongle for your Pi (I've used this one for a few years
<https://www.amazon.com/dp/B003MTTJOY>)
- A computer with a microSD card drive, or an SD card drive and a microSD card adapter. This computer will also need Raspberry Pi Imager installed.
- Access to a server/service running OpenVPN & a downloaded copy of your client.ovpn file.
This doc assumes you use username/password to auth to your VPN server.
- An ansible "control node" (the machine that runs Ansible). To avoid issues following along with this, I would make sure you are running ansible core 2.12.2 or higher.
- The Ansible community module installed on your control node.
ansible-galaxy collection install community.general

Install Raspberry Pi OS Lite SD card

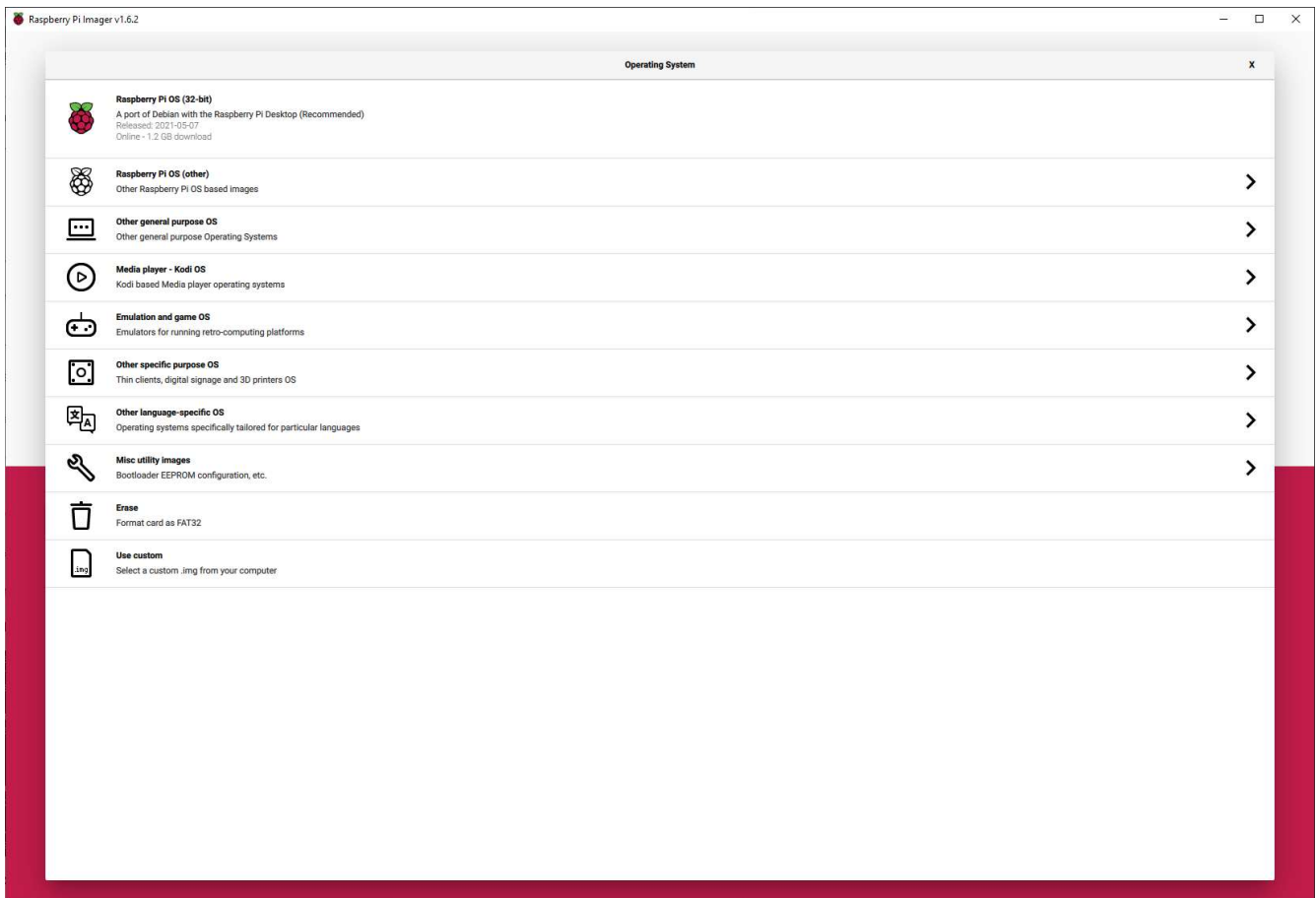
The first step is to install Raspberry Pi OS Lite on your Raspberry Pi and connect it to the network. Since I don't like to disable host_key_checking in my ansible.conf file I add my ansible control server's public key to my Pi OS when I image the SD card with pi imager.

Note: The steps given in this section will erase all existing content on your microSD card. If you already have Pi OS running on your Raspberry Pi you can skip the next section

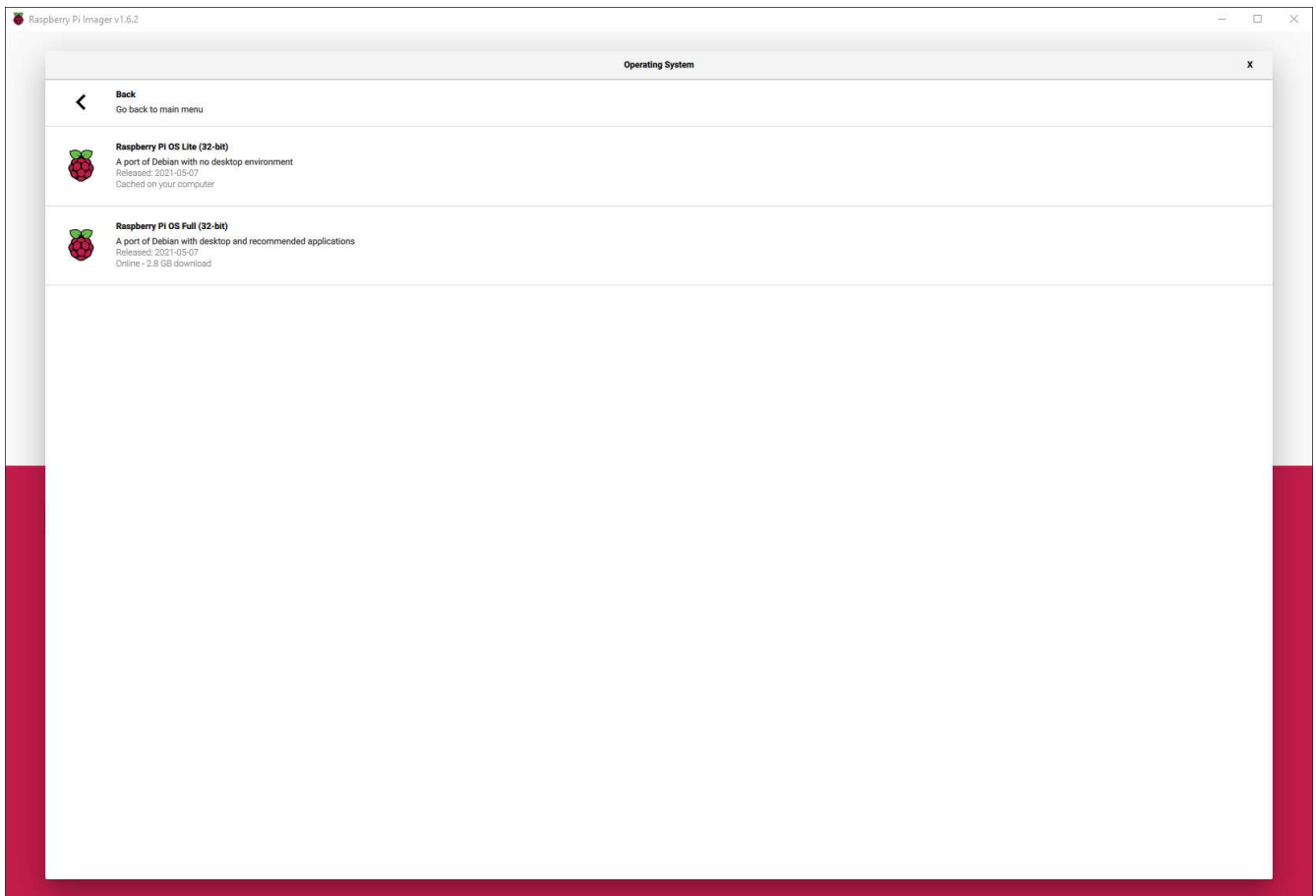
1. Open Raspberry Pi imager
2. Select “**Choose OS**”



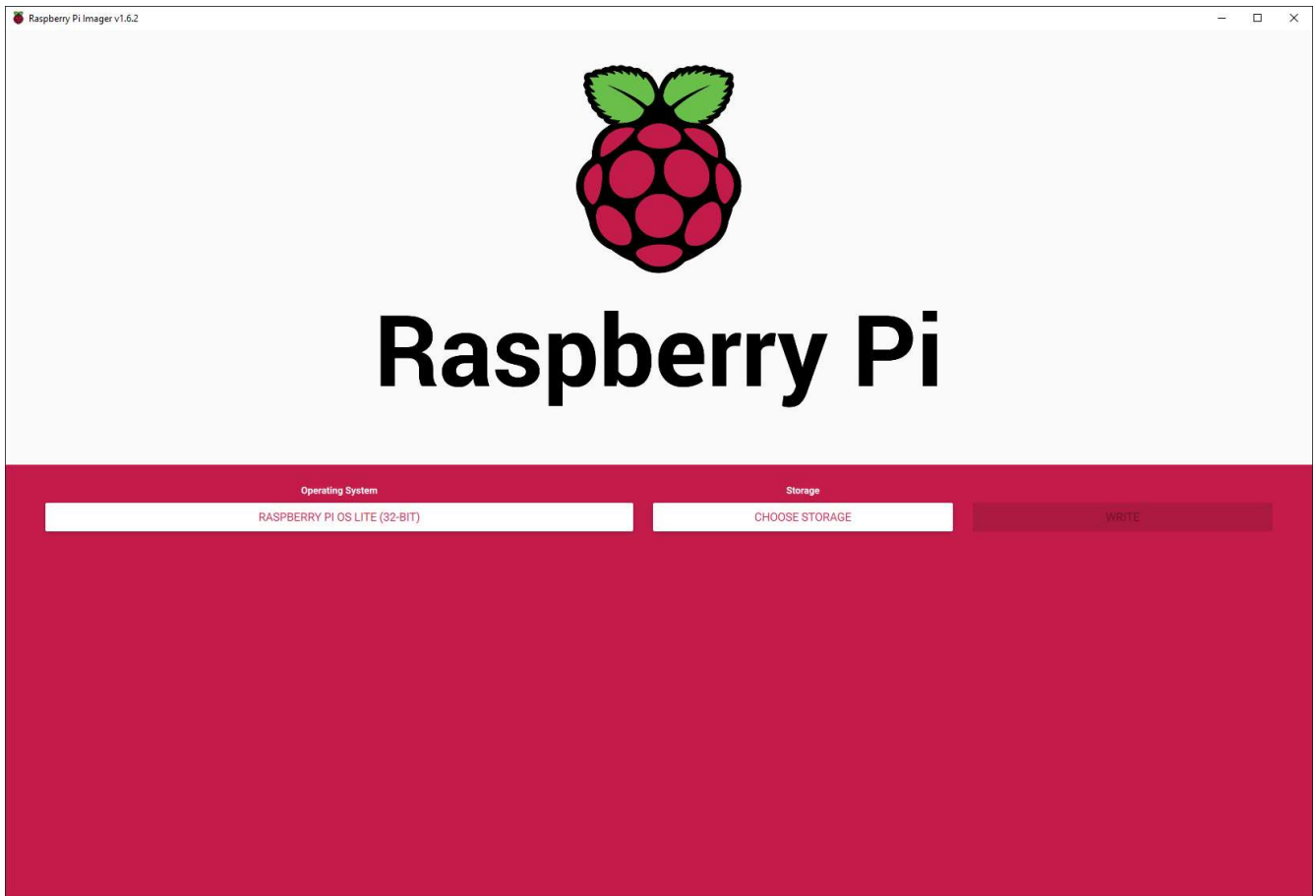
3. Select **Raspberry Pi OS (Other)**



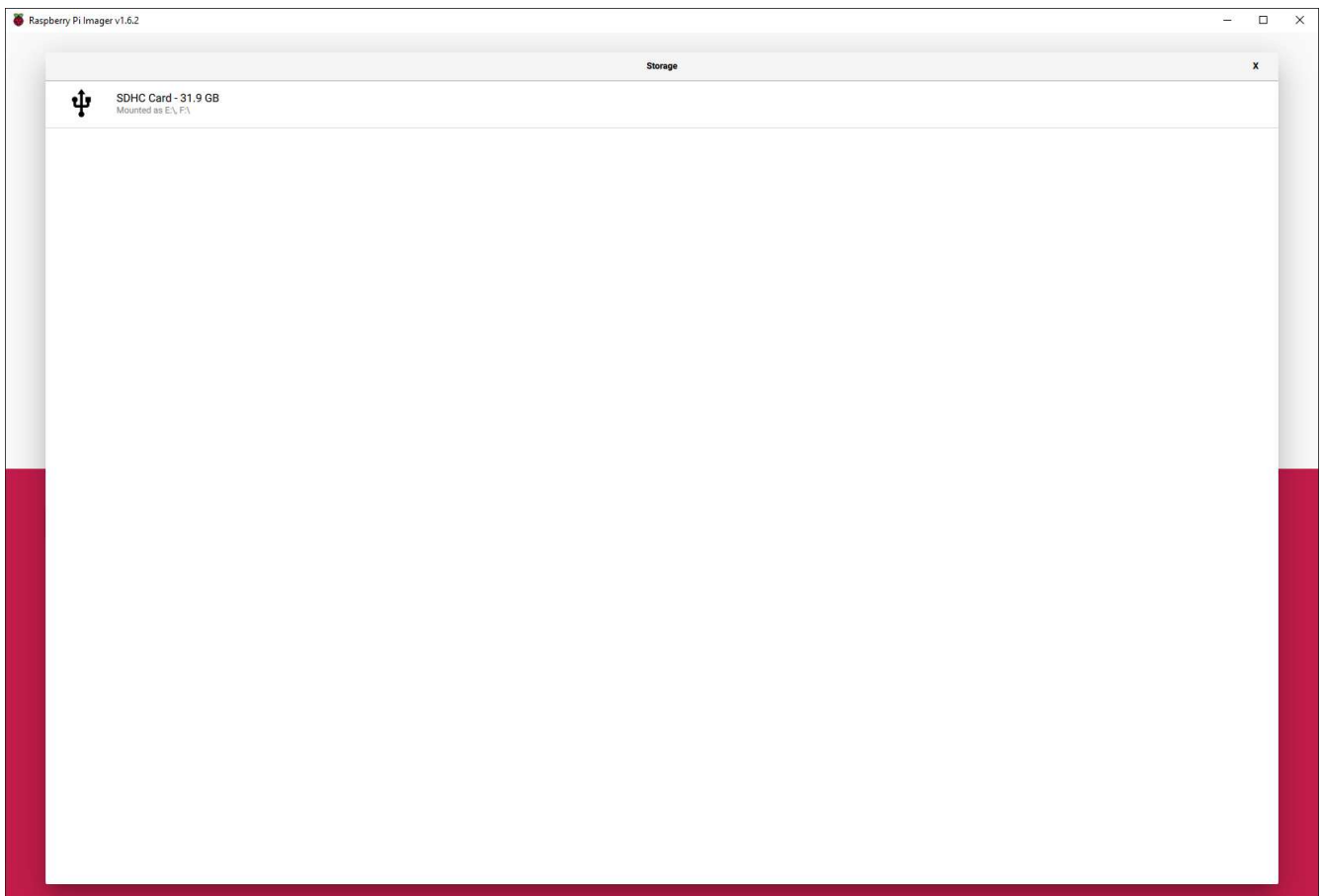
4. Select “Raspberry Pi OS Lite”



5. Now Select “**Choose Storage**”

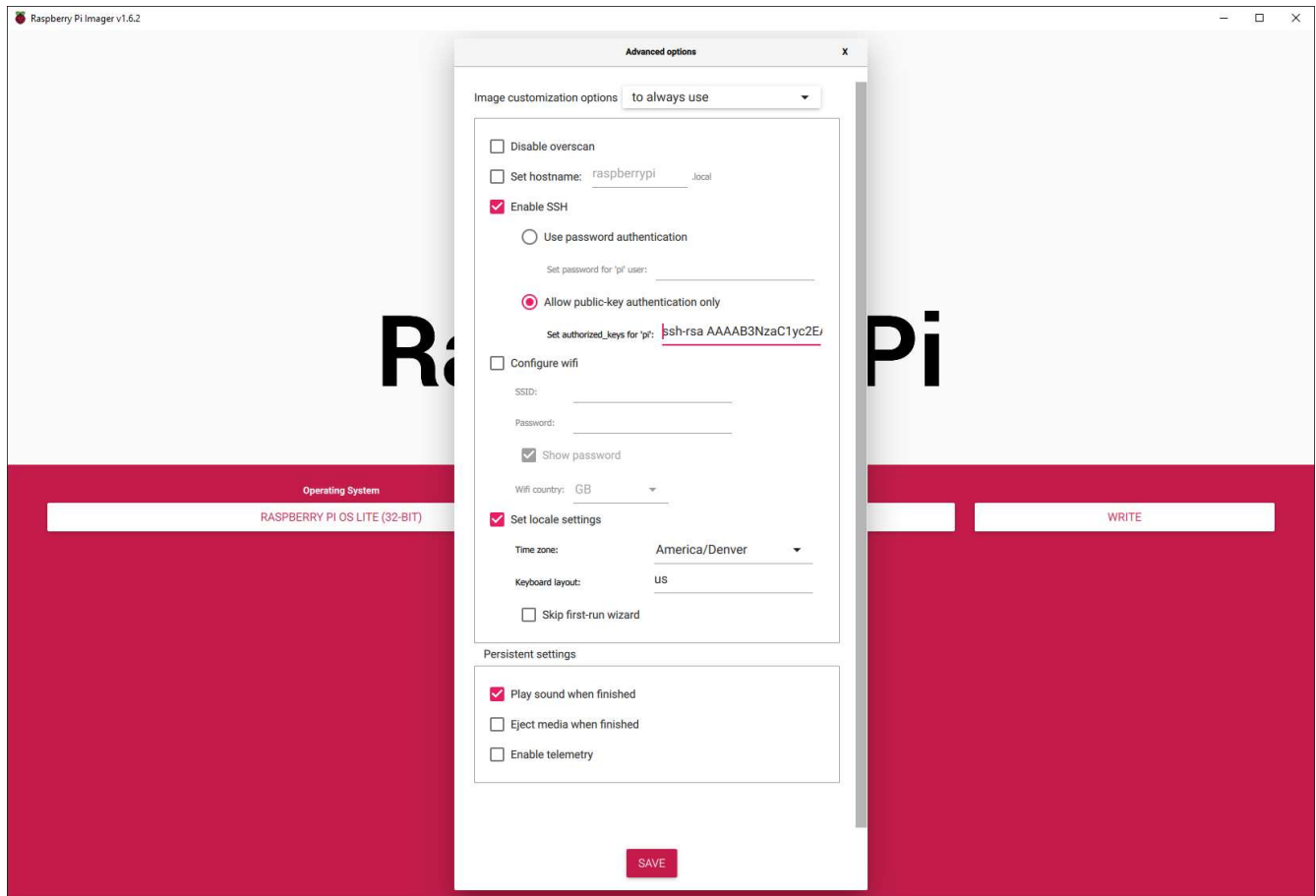


6. Select your SD card.

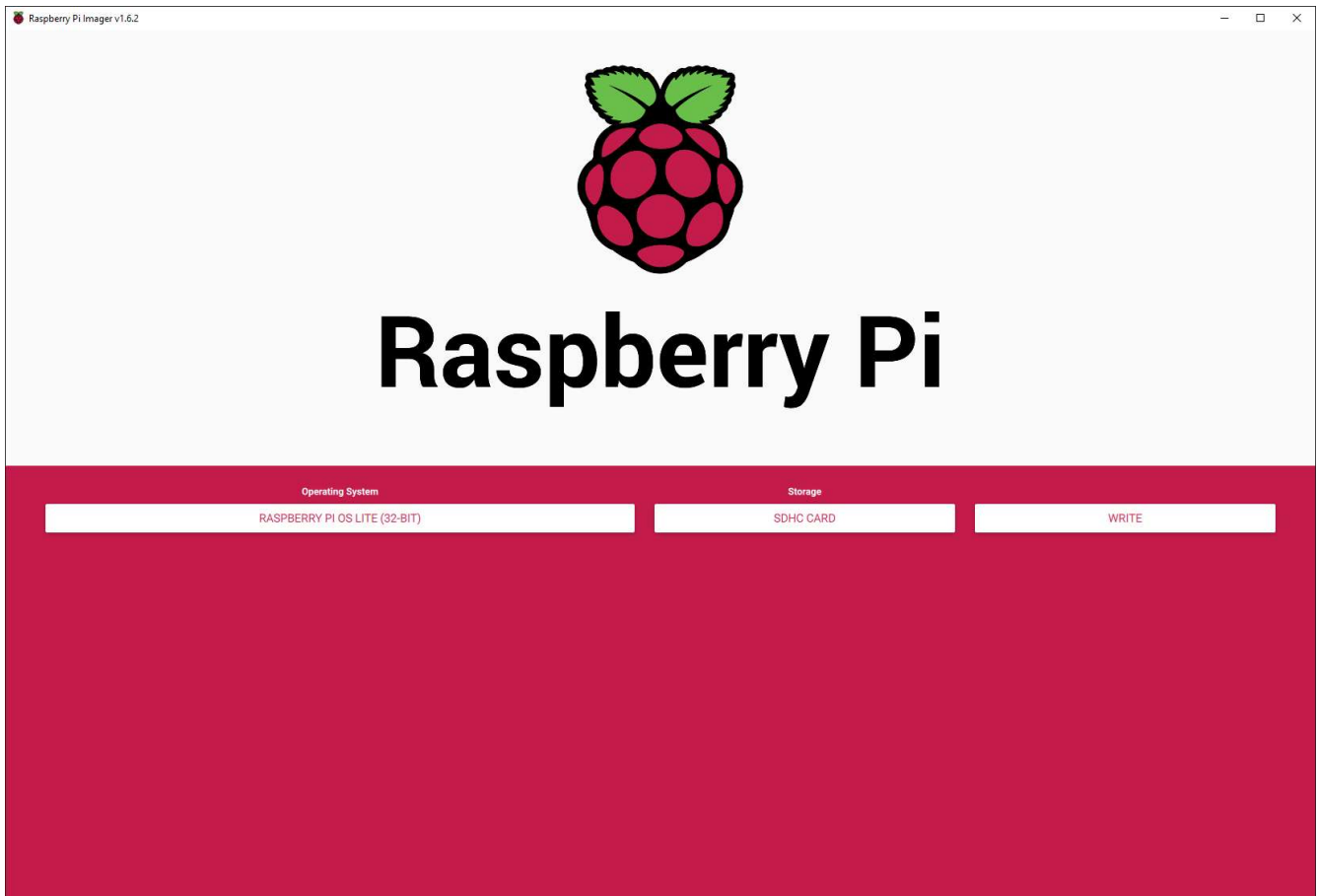


7. Now back at the main imager screen. Press “**CTRL+SHIFT+X**” to open the advanced menu

8. Enable SSH, Add the public key from your ansible control server and set your location. And click Save.



9. Select “**Write**” to write Pi OS to your SD card.




10. Wait for image to complete
11. Once complete, un-mount the SD card and put in it your Raspberry Pi.

Boot the Raspberry Pi

1. Hard wire your pi into your router.
2. Get its IP address from your DHCP server(or you can connect a monitor to it and see it at the login screen)
3. From your ansible control server try and ssh into the Pi (ssh [pi@IP_ADDRESS](#)). This will ensure you can connect to the pi from your control server as well as add its public key to your ansible server key store.
4. Close the ssh connect to the pi (exit)

Clone down this repository and edit the playbooks.

1. Clone down this repository into a directory on your control server.
2. Update the inventory file with your raspberry pi's IP to it. ("sudo vim /etc/ansible/hosts")



```
[raspberrypi]
192.168.1.115
```

3. Next copy your client.ovpn file to the "files" folder

rename this file to client.ovpn if its not called client.ovpn

I highly recommend you add your client.ovpn to your .gitignore file before you ever push this repo to your own repo

4. Update the tasks in roles/initial_setup/tasks to match your country & timezone.

4a. Verify you locale. `vim roles/initial_setup/tasks/set_locale.yml`.

4b. Verify your wifi country. `vim roles/initial_setup/tasks/set_wifi_country.yml`

4c. Set/Verify your timezone(optional). `vim roles/initial_setup/tasks/set_timezone`

Running Ansible

1. From your control node and within the root folder of the cloned repo run the following command.

`ansible-playbook install_vpn -K --user pi -i inventory`

2. Enter the default password for the pi user (raspberrypi)
3. Enter the desired username and password for your new user (this will replace the pi user)
4. Enter the username and password of the account you use to log into your VPN (this is to setup auto vpn login on boot)
5. Enter the ssid and passphrase for the wifi network you want your external devices connect to when you travel.
6. Enter the ssid and passphrase of the wifi network you want your Pi to connect to for internet. I'm using my old iphone7 hot spot in this example. You would just need to update `/etc/wpa_supplicant/wpa_supplicant-wlan0.conf` file when ever you need to connect to new wifi networks when you travel. I usually just do this via ssh since its quick.

1. `wpa_passphrase {{ wan_ssid }} {{ wan_pass_phrase }} >>`
`/etc/wpa_supplicant/wpa_supplicant-wlan0.conf`


```
BECOME password:
Enter new user's Username (this will replace the default 'pi' user): chris
Enter new user's Password:
confirm Enter new user's Password:
Enter your VPN's username (the account you use to connect to your VPN Server): chris
Enter your VPN's Password:
confirm Enter your VPN's Password:
Enter ssid of lan wifi (the ssid the pi hosts for devices to connect to): PizzaTime
Enter the passphrase of the lan wifi (Passphrase you will use to connect TO THE raspberry pi):
confirm Enter the passphrase of the lan wifi (Passphrase you will use to connect TO THE raspberry pi):
Enter ssid of wan wifi (the ssid the pi connects to for internet): iPhone7
Enter the passphrase of the wan wifi (the passphrase of the wifi your pi connects to for internet):
confirm Enter the passphrase of the wan wifi (the passphrase of the wifi your pi connects to for internet):

PLAY [Setup raspberry pi travel router] *****
```

7. The playbooks take a 10-15 minutes to run though depending on how many OS updates there are.

Helpful resources

<https://docs.ansible.com/ansible/latest/index.html>

<https://www.raspberrypi.org/>