Requisitos:

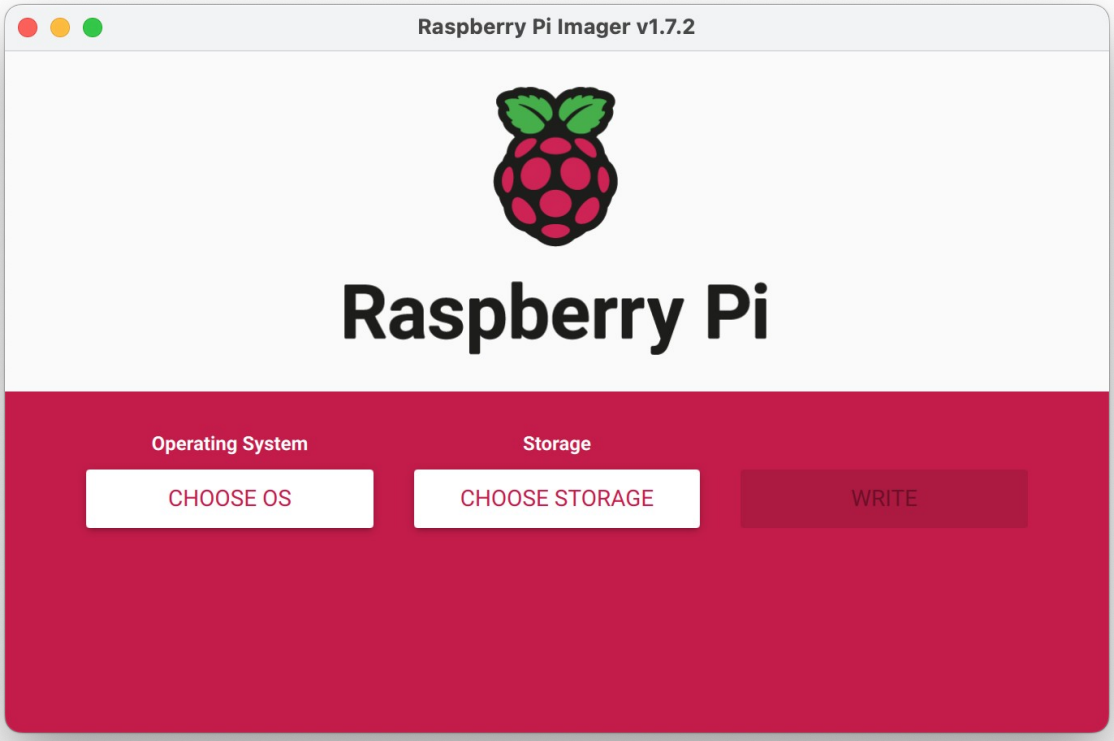
Qué necesitamos?

- **RAK2287 Pi HAT WisLink LPWAN Concentrator**
- **RAK2287 Pi HAT**
- **Raspberry Pi 3B+** o **Raspberry Pi 4**
- Tarjeta SD de 16GB + Lector de tarjetas
- Adaptador USB o USB C (Pi 4) de 5V de por lo menos 2 A
- Un ordenador Windows/Mac OS/Linux
- Raspberry Pi Imager
- El ultimo firmware de RAK2287 Pi HAT aunque en este manual no lo necesitemos

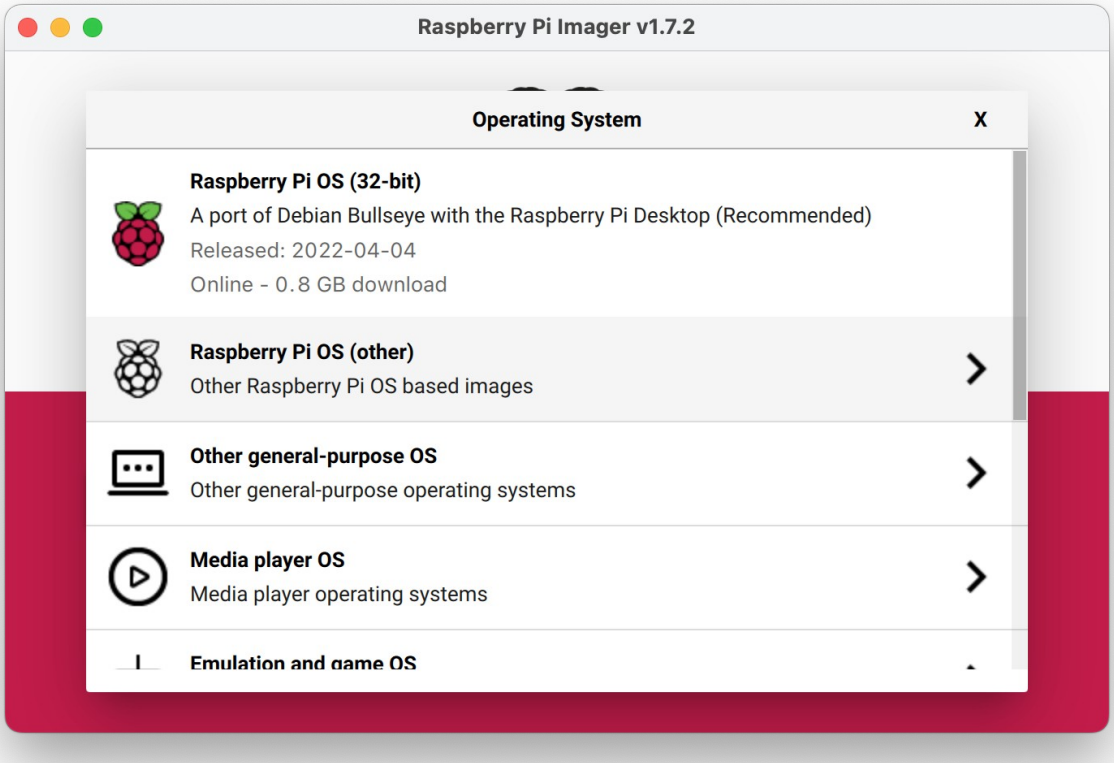
Instalacion en la raspberry:

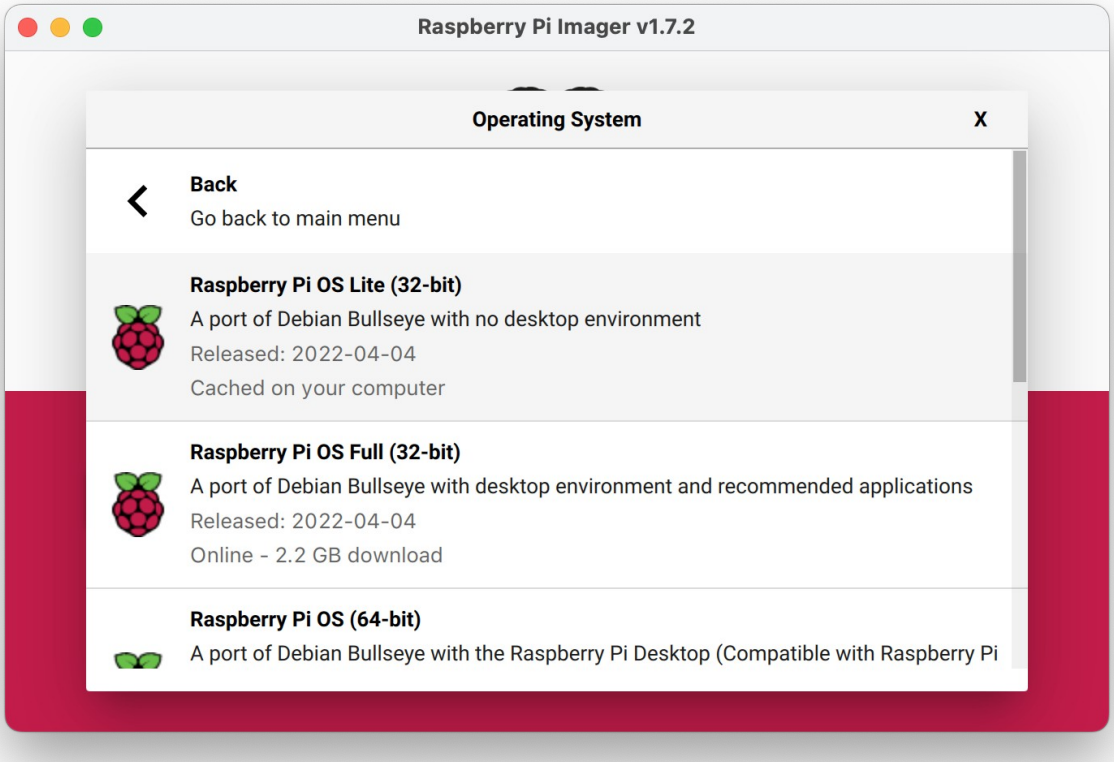
Descargar ***Raspberry Pi Imager***

Abrir ***Raspbery Pi Imager*** que se puede descargar de la pagina oficial de Raspberry.

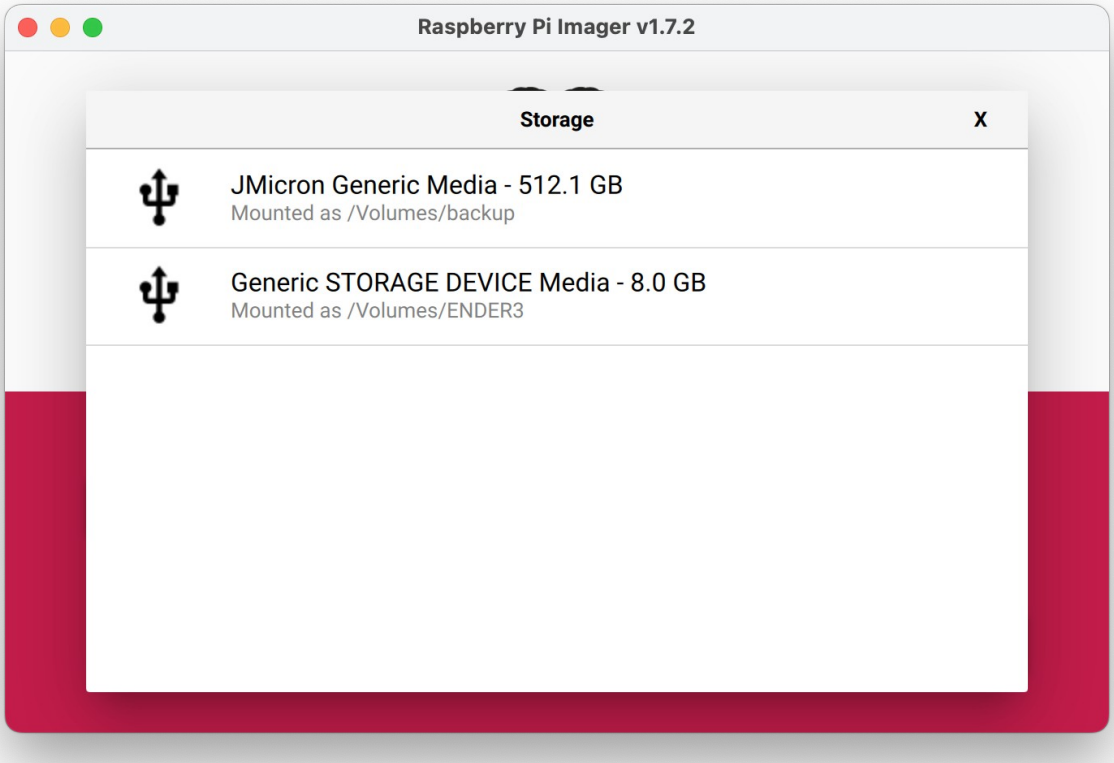


Elegimos OS Raspberry Pi OS (32-bit)



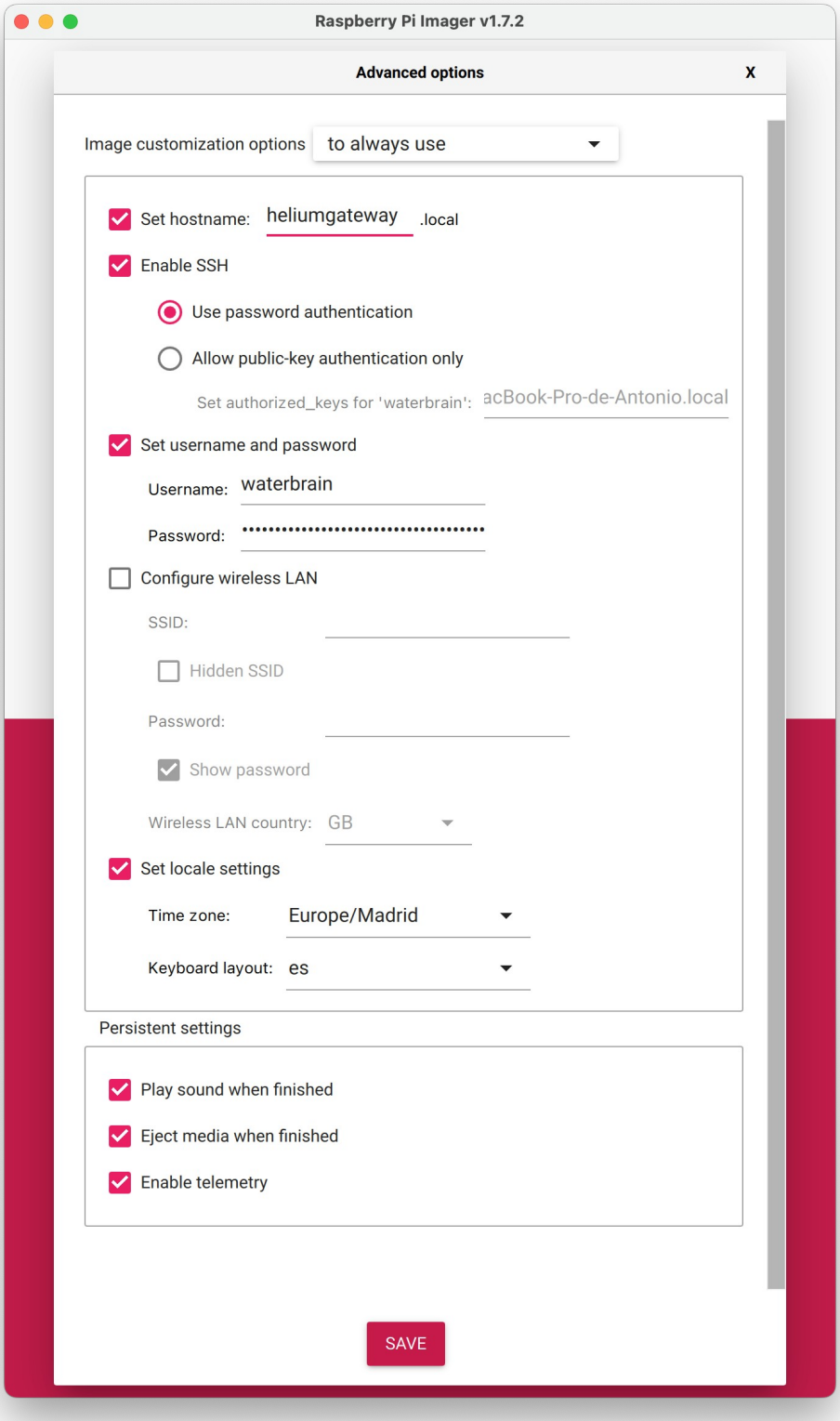


A continuación elegimos la tarjeta:


Raspberry Pi Imager v1.7.2
Storage
X
JMicron Generic Media - 512.1 GB
Mounted as /Volumes/backup
Generic STORAGE DEVICE Media - 8.0 GB
Mounted as /Volumes/ENDER3

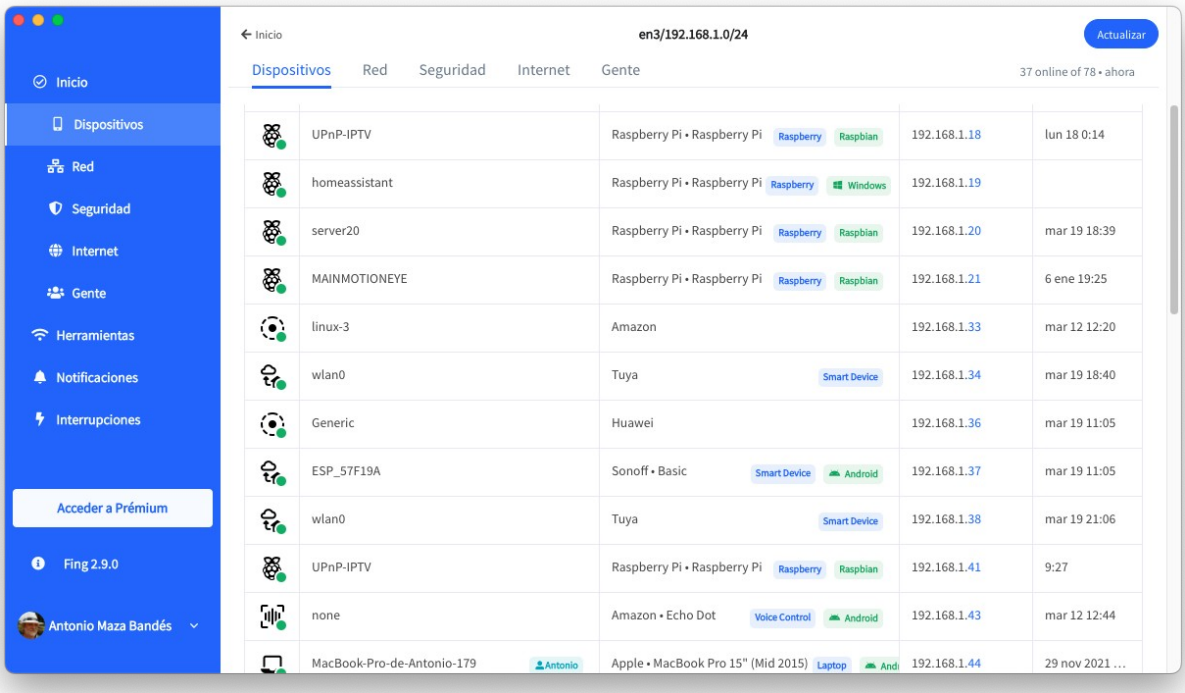

Raspberry Pi Imager v1.7.2
Raspberry Pi
Operating System
Storage
RASPBERRY PI OS LITE (32-BIT)
GENERIC STORA...
WRITE

Aquí aparece la ruedecita de configuracion la pulsamos y por razones legales y de seguridad para evitar ataques con el usuario pi yo lo dejo asi:



WRITE y ya le podemos dar a write. Esperamos a que acabe. Introducimos la imagen en la Raspberry y arracamos.

Buscamos en que IP está, yo empleo Fing que en versión libre esta para cualquier ordenador incluido iPhone o Android.

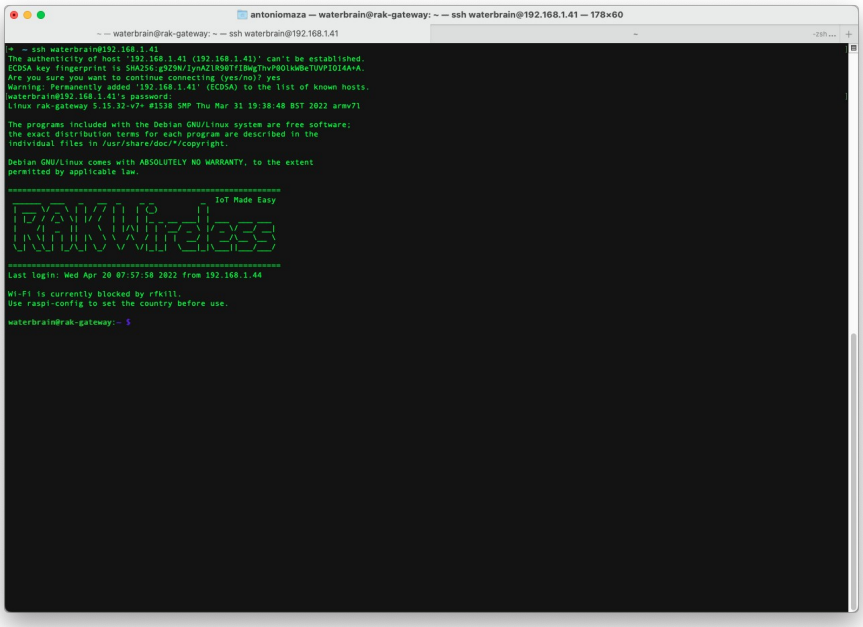


veo que a las 9:27 ha aparecido un dispositivo nuevo con la ip 192.168.1.41 por lo que se en que direccion esta la Rasberry.

Abrimos una pantalla de terminal:

- ssh waterbrain@192.168.1.41

Yo personalmente prefiero ssh, aunque se pude utilizar cualquier herramienta como **PuTTY**.Por ser la primera vez nos pide que añadamos la clave de la direccion de la Raspberry a su fichero de hosts conocidos, le decimos que si y voila:



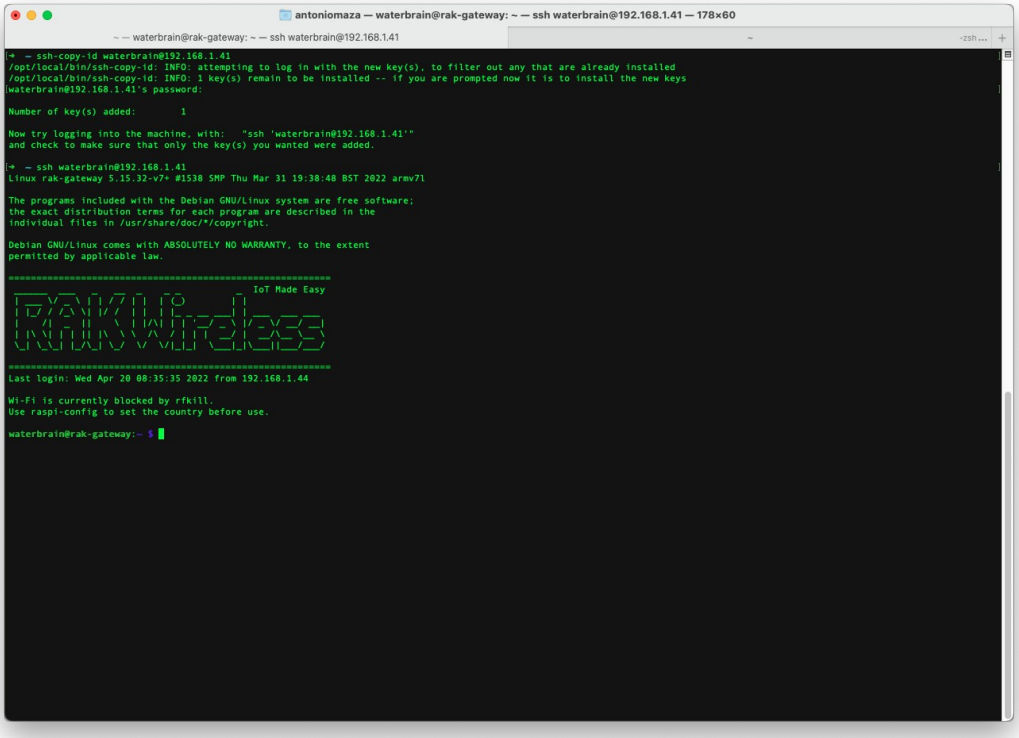
Si queremos que no nos pida la contraseña:
salimos de la Raspberry y ejecutamos:

- ssh-copy-id waterbrain@192.168.1.41

en nuestra terminal.Previo a esto debemos tener una llave o clave única generada. Yo no lo hago pero por si acaso no lo hemos hecho utilizaremos:

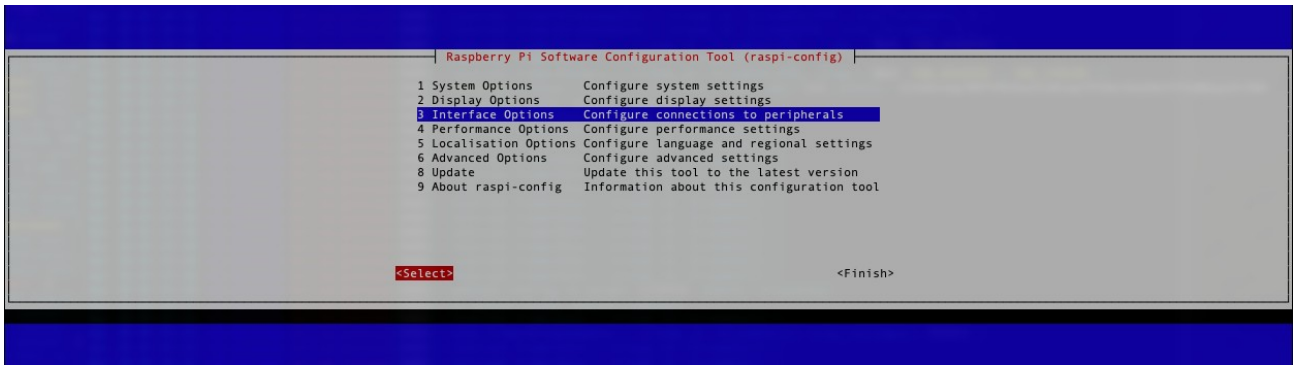
- **ssh-keygen -b 4096 -t rsa**

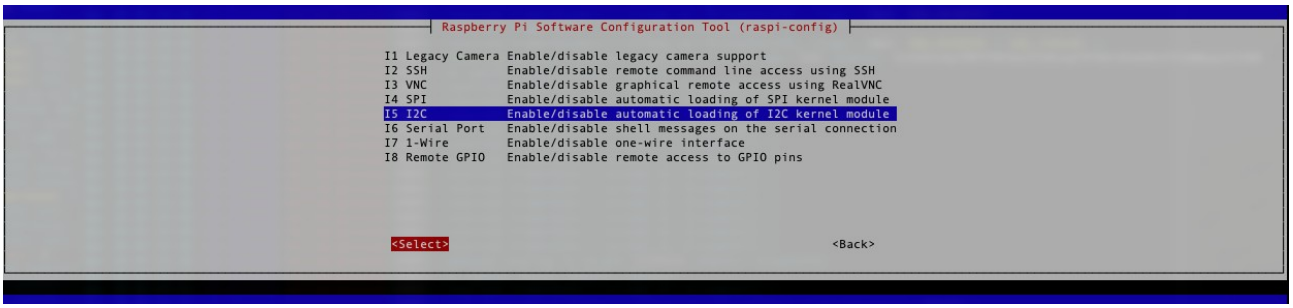
Esto generará una llave pública. Para los que tengan más interés en este tema, solo tienen que googlear un poco.

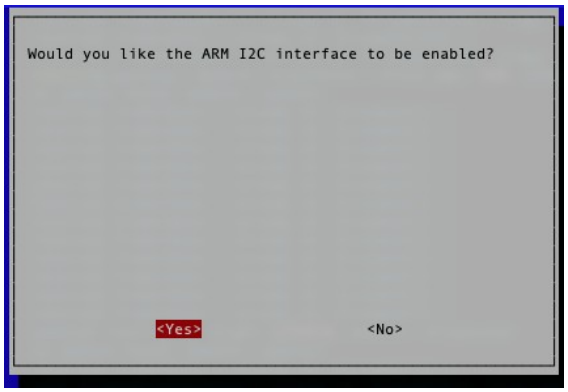


2.-:

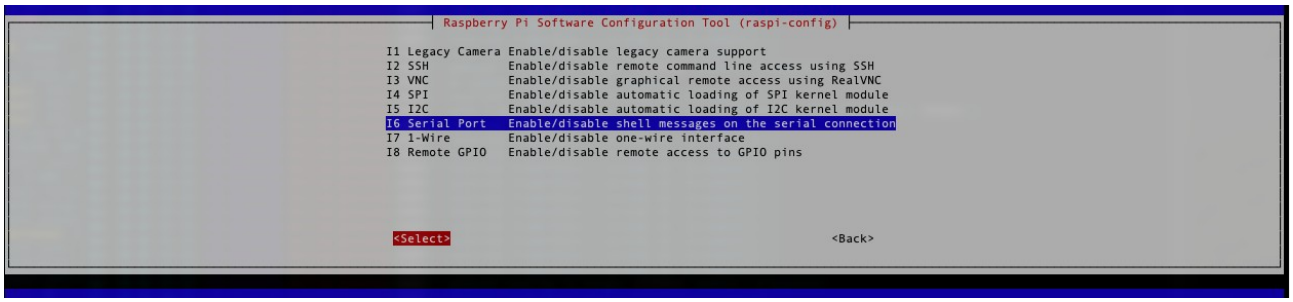
- **sudo raspi-config**

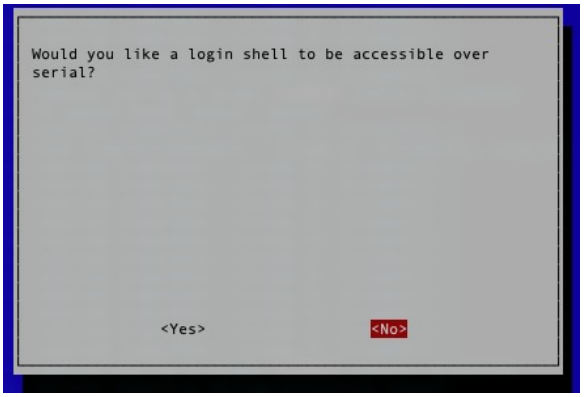


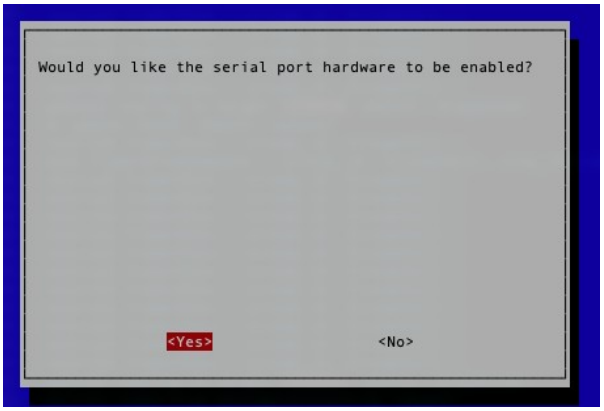


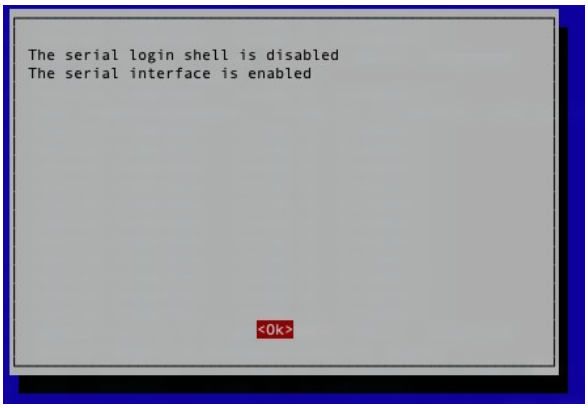

Would you like the ARM I2C interface to be enabled?
<Yes>
<No>

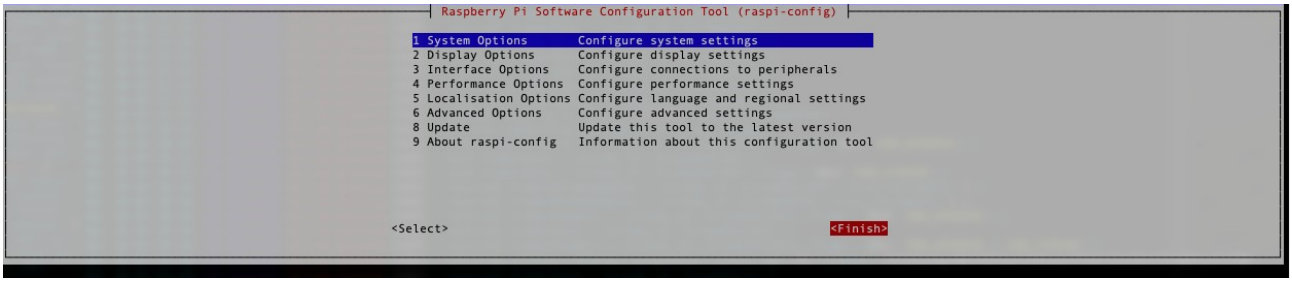

The ARM I2C interface is enabled
<Ok>

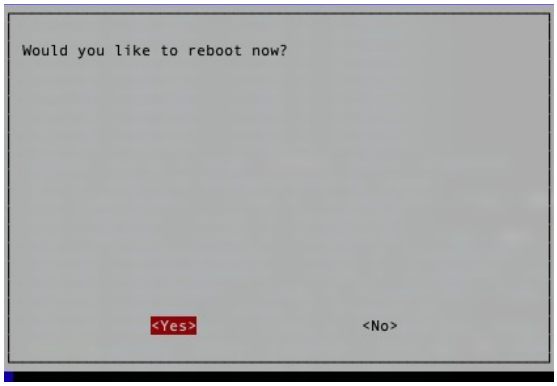

Raspberry Pi Software Configuration Tool (raspi-config)
I1 Legacy Camera Enable/disable legacy camera support
I2 SSH Enable/disable remote command line access using SSH
I3 VNC Enable/disable graphical remote access using RealVNC
I4 SPI Enable/disable automatic loading of SPI kernel module
I5 I2C Enable/disable automatic loading of I2C kernel module
I6 Serial Port Enable/disable shell messages on the serial connection
I7 1-Wire Enable/disable one-wire interface
I8 Remote GPIO Enable/disable remote access to GPIO pins
<Select>
<Back>


Would you like a login shell to be accessible over serial?
<Yes>
<No>


Would you like the serial port hardware to be enabled?
<Yes>
<No>


The serial login shell is disabled
The serial interface is enabled
<Ok>





3.- Editar .bash_aliases, yo suelo utilizar **nano**, y poner los alias siguientes:

- **nano ~/.bash_aliases**

```
alias confrak='sudo nano /opt/ttn-gateway/packet_forwarder/lora_pkt_fwd/global_conf.json'
alias confrakloger='sudo nano /opt/ttn-gateway/lora_gateway/util_pkt_logger/global_conf.json'
alias raklog='sudo tail -f /var/log/syslog | grep ttn-gateway | ccze -A'
alias rakstop='sudo systemctl stop ttn-gateway'
alias rakstart='sudo systemctl start ttn-gateway'
alias heliumlog='sudo tail -f /var/log/syslog | ccze -A | grep helium_gateway'
alias heliumstop='sudo systemctl stop helium_gateway'
alias heliumstart='sudo systemctl start helium_gateway'
alias heliumrestart='sudo systemctl restart helium_gateway'
alias rakrestart='sudo systemctl restart ttn-gateway'
alias heliuminfo='helium_gateway --stdin info | ccze -A'
alias confhelium='sudo nano /etc/helium_gateway/settings.toml'
alias heliumversions='helium_gateway update list'
alias heliumverupdate='helium_gateway update download'
alias nrlog="pm2 logs node-red | ccze -A"
alias nrstop="pm2 stop node-red"
alias nrres="pm2 reload node-red;pm2 logs node-red | ccze -A; pm2 logs node-red | ccze -A"
alias nrstart="pm2 start node-red;pm2 logs node-red | ccze -A"
alias nrrestart="pm2 restart node-red;pm2 logs node-red | ccze -A"
alias nrconf="sudo nano /lib/systemd/system/nodered.service"
alias nract="bash <(curl -sL https://raw.githubusercontent.com/node-red/raspbian-deb-package/master/resources/
update-nodejs-and-nodered)"
alias nrdir="cd .node-red/"
_dir() { [ ! -x "$(command -v ccze)" ] && sudo apt-get install ccze;  \dir "$@" | ccze -A;}; alias dir='_dir'
_ifconfig() { [ ! -x "$(command -v ccze)" ] && sudo apt-get install ccze; \ifconfig "$@" | ccze -A; }; alias
ifconfig='_ifconfig'
upgr() { [ ! -x "$(command -v ccze)" ] && sudo apt-get install ccze; echo "*** Updating"; sudo apt-get update | ccze
-A; echo "*** Upgrading"; sudo apt-get upgrade | ccze -A; }
alias ap2log="sudo journalctl -f -n 50 -u apache2 -o cat | ccze -A"
alias ap2stop="sudo service apache2 stop"
alias ap2start="sudo service apache2 start"
alias ap2restart="sudo service apache2 restart"
alias ap2reload="sudo systemctl reload apache2"
alias ngstop="sudo /etc/init.d/nginx stop"
alias ngstart="sudo /etc/init.d/nginx start"
alias ngstatus="sudo /etc/init.d/nginx status"
alias model='cat /proc/device-tree/model | ccze -A;echo'
alias cputype='cat /proc/device-tree/cpus/cpu@0/compatible | ccze -A;echo'
```

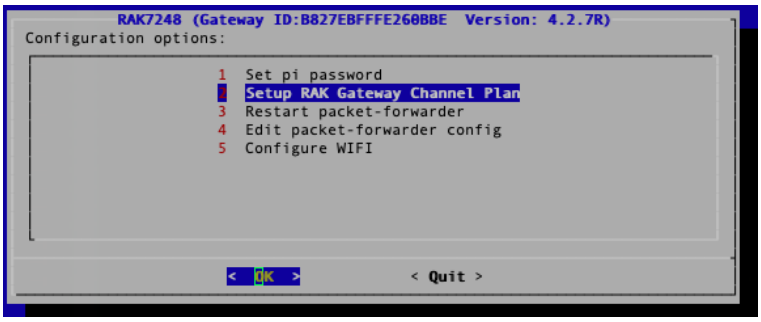
Ya sabreis que para salir de **nano** utilizamos **ctr-X** y contestar afirmativamente cuando nos pregunte si queremos guardar el archivo.

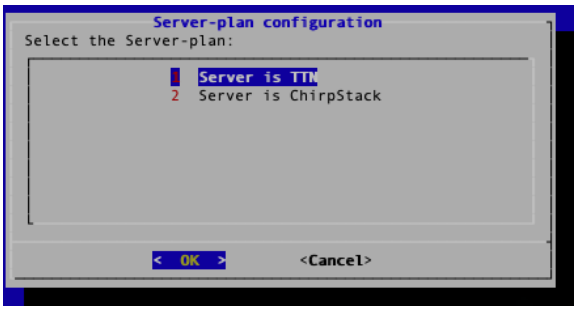
4.-

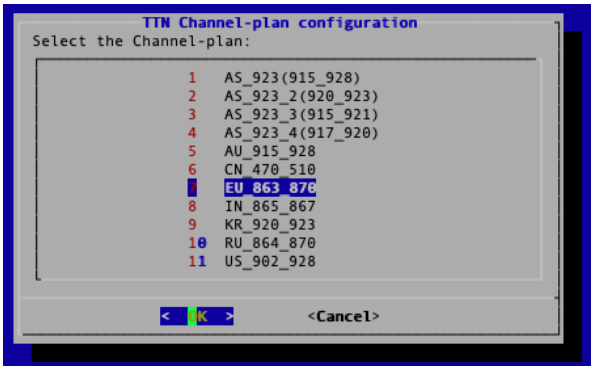
- sudo apt-get install git ccze mc macchanger
- git clone https://github.com/RAKWireless/rak_common_for_gateway.git
- cd ~/rak_common_for_gateway
- sudo ./install.sh
- sudo reboot

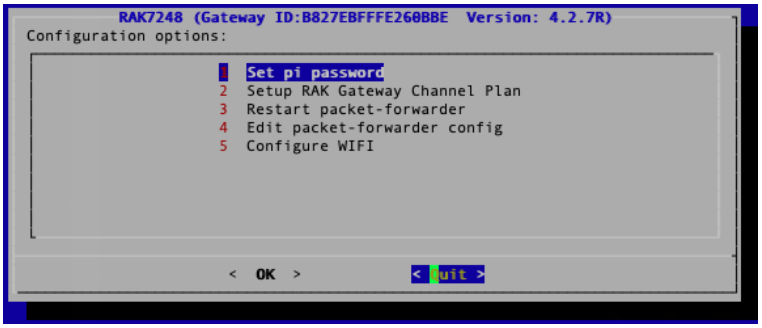
5. configurar el gateway de rak:

- **sudo gateway-config**

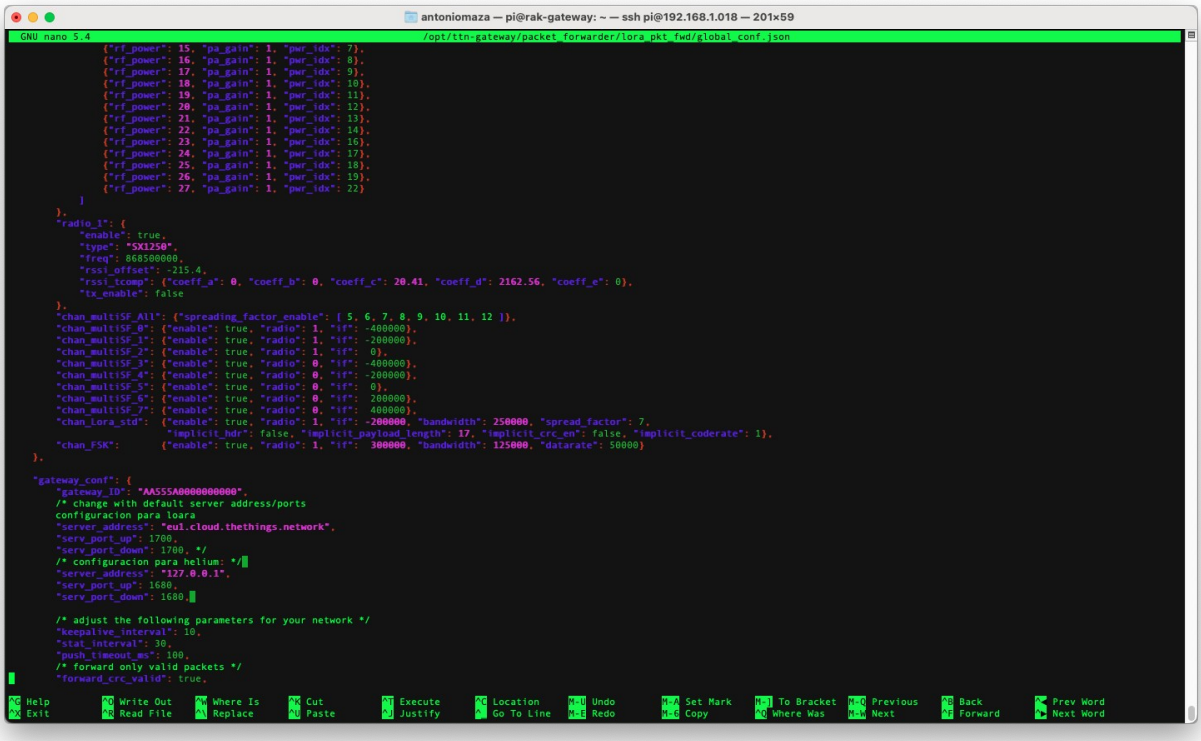








- **confrak**



Yo lo dejo así para poder jugar con lora o helium.

- **rakrestart**

6.- instalar la parte de helium:

- **cd /tmp**
- **wget https://github.com/helium/gateway-rs/releases/download/v1.0.0-alpha.22/helium-gateway-v1.0.0-alpha.22-raspi234.deb**
- **sudo dpkg -i helium-gateway-v1.0.0-alpha.22-raspi234.deb**
- **cd**
- **sudo nano /etc/helium_gateway/settings.toml**
- **sudo systemctl restart helium_gateway**

7.- ponerle una ip fija (opcional, pero muy conveniente):

Nos conectamos a la raspberry y ejecutamos:

- **sudo nano /etc/dhcpcd.conf**

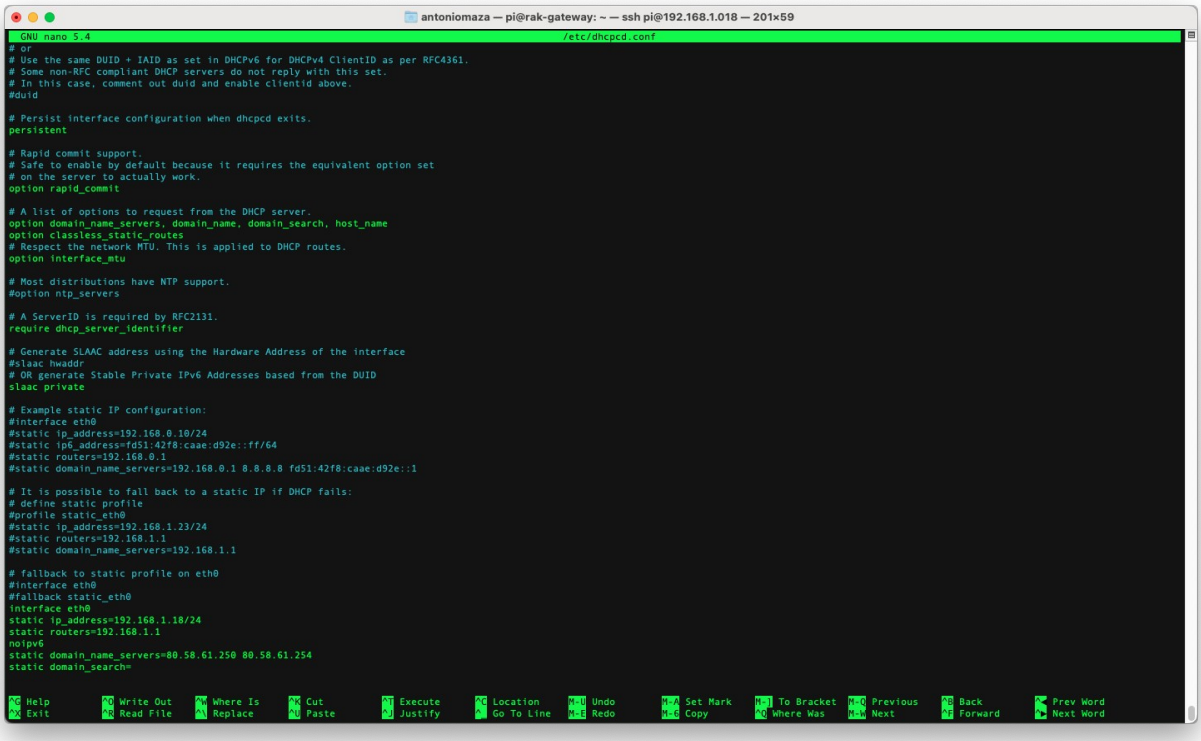
y añadimos al final lo siguiente:

interface eth0
static ip_address=192.168.1.18/24
static routers=192.168.1.1
noipv6
static domain_name_servers=80.58.61.250 80.58.61.254
static domain_search=

- **sudo nano -m /etc/network/interfaces**

auto lo
allow-hotplug wlan0
iface wlan0 inet static
address 192.168.1.105
netmask 255.255.255.0
broadcast 192.168.1.255
network 192.168.1.0
gateway 192.168.1.1
wpa-ssid "el ssid de tu router wifi”

wpa-psk "tu password"
wpa-ap-scan 1
wpa-scan-ssid 1
wpa-proto RSN
wpa-pairwise CCMP
wpa-key-mgmt WPA-PSK
dns-nameservers 80.58.61.250 80.58.61.254# interfaces(5) file used by ifup(8) and ifdown(8)



Como ya he mencionado antes, esto se realiza con nano, y ya sabéis, una vez actualizado como salimos guardando el archivo.

le damos a re-arrancar la próxima vez ya estará en esa ip.

- **sudo reboot**

ya podemos utilizar con018 y una vez hecho el login a jugar……

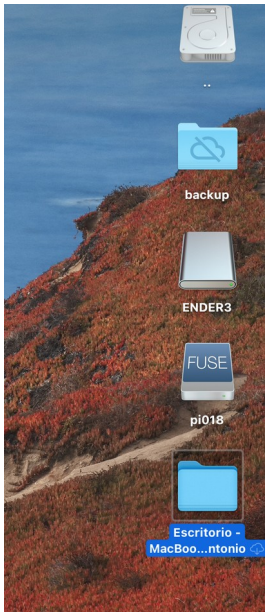
- **raklog**
- **heliumlog**
- **heliuminfo**
- **confrak**
- **confhelium**
- **heliuminfo**

o cualquiera de los alias que hemos añadido anteriormente

Corolario:
yo en mi caso ya tengo lo siguiente a mi entorno de shell
alias conecta018='ssh waterbrain@192.168.1.018'
y si tenemos alguna herramienta como **fuse** en el mac podemos añadir tambien **alias conecta018='sshfs waterbrain@192.168.1.18:/ pi018 -ovolname=pi018'**
Previamente debemos tener en nuestro home una carpeta vacía con el nombre pi018

ya podemos usar tanto un alias como otro

si le damos a conectar veremos en nuestro escritorio algo parecido a esto:



Sí clicamos en el disco pi018:

!MAGIAi Como podéis ver todo el contenido de la raspberry, algo parecido podemos hacer con **FileCilla** que esta disponible tanto para mac como para windows, pero a mi esto me parece más correcto y funcional.