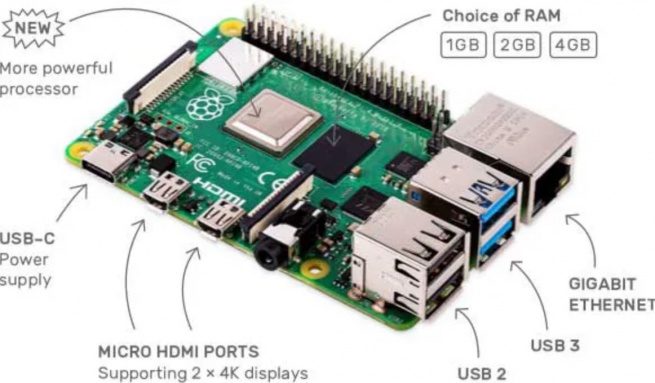
**Hugo Alejandro Hernández Cárdenas**

**Daniel Felipe Cadavid Duque**

**Documentación Raspberry Pi 4**

1. **Especificaciones Raspberry Pi 4 Model B**

* CPU: Broadcom BCM2711, Quad-Core Cortex-A72 64 bits a 1.5GHz de velocidad
* Memoria: 1GB, 2GB o 4GB de RAM LPDDR4.
* Conectividad: Wi-Fi b/g/n/ac (2.4GHz y 5GHz), Bluetooth 5.0 con BLE, puerto Gigabit Ethernet, 2 puertos USB 3.0 y 2 puertos USB 2.0
* GPIO: 40-pin compatible con las otras Raspberry Pi.
* Video y sonido: 2 puertos micro HDMI (hasta 4K a 60 FPS), 2-lane MIPI DSI display, 2-lane MIPI CSI camera, 4-pole stereo audio y video
* Multimedia: H.265 hasta 4K a 60 FPS, H.264 a 1080p a 60 y 30 FPS.
* Soporte tarjetas microSD: soporta microSD para el sistema operativo y también para almacenamiento de datos.
* Alimentación: vía USB tipo C con 5V y 3A, vía PoE con HAT (se vende por separado), vía GPIO con mínimo 3A.



1. **Análisis Raspberry Pi 4**
   1. **Puntos Fuertes**

* Excelente rendimiento del hardware en términos de CPU y memoria RAM.
* Posibilidad de comprarla con 1GB, 2GB o 4GB de memoria RAM.
* Rendimiento excelente del puerto Gigabit Ethernet.
* Rendimiento bueno en la conectividad Wi-Fi, debemos tener en cuenta que no tenemos antenas externas, y hemos conseguido en la habitación contigua 100Mbps reales, tanto en 2.4GHz como en 5GHz.
* Vamos a poder transferir datos en red local a más de 900Mbps sin ningún problema.
* Rendimiento y experiencia de usuario notable con el Bluetooth 5.0.
* Rendimiento excelente en lectura y escritura de los puertos USB 3.0 con formato EXT4.
* Rendimiento excelente en lectura de los puertos USB 3.0 con formato NTFS, y notable en escritura.
* Experiencia de usuario multimedia a la hora de reproducir películas en resolución 4K y Full HD.
* Infinitas posibilidades de personalización, instalando diferentes sistemas operativos compatibles, usándola de servidor NAS etc.
* Incorporación de Wi-Fi y Bluetooth, además de conservar el GPIO y el resto de las conexiones típicas.

**2.2 Puntos Débiles**

* El puerto USB tipo C de carga tiene un fallo de diseño y provoca que algunos transformadores de corriente no alimenten el dispositivo. Es recomendable adquirir la fuente de alimentación oficial.
* El Wi-Fi no sincroniza en modo Wi-Fi AC con 80MHz de ancho de canal, solamente en N (5GHz) a 40MHz.
* Rendimiento insuficiente de los puertos USB 2.0.
* Se calienta en exceso la CPU cuando estamos haciendo un uso intensivo, llegando hasta los 85ºC y sufriendo CPU throttling para controlarla

1. **Direccionamiento IPv4**

En vez de utilizar el rango que por defecto nuestro router lo cambiaremos, por ejemplo, por 192.168.0.0/16, la cual es una red clase C privada.

Reservaremos un espacio de direcciones comprendido entre la 192.168.0.100 y 192.168.0.119 donde ubicaremos el **router**(192.168.0.100), la Raspberry Pi (192.168.0.101) y **computador fijo** (192.168.0.110).

El resto del espacio, 192.168.0.1-192.168.0.99 y 192.168.0.120-192.168.0.254, será utilizado como **rango de asignación dinámica,** un rango innecesariamente grande para una red doméstica, pero que resulta interesante de configurar.

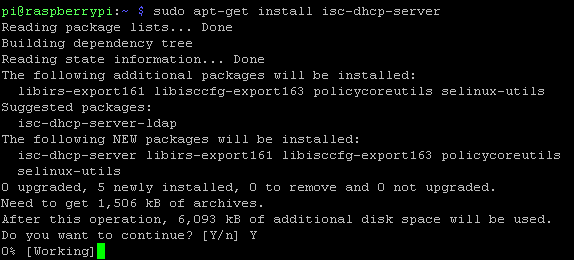
|  |  |
| --- | --- |
| Red | 192.168.0.0 |
| Mascara de subred | 255.255.255.0 / 24 |
| Rango DHCP | 192.168.0.1-192.168.0.99,  192.168.0.120-192.168.0.254 |
| Gateway | 192.168.0.100 |
| Raspberry | 192.168.0.101 |
| Computador Fijo | 192.168.0.110 |

1. **Servidor DHCP**

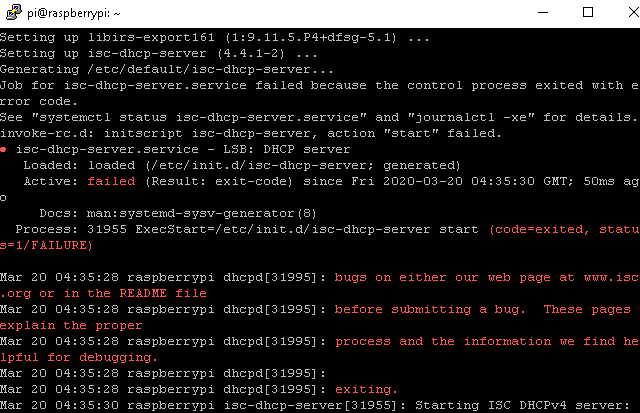
**4.1 Instalación del Servidor DHCP**

Para instalar el servidor DHCP primero tenemos que conectarnos vía SSH al Raspberry e instalar isc-dhcp-server con el gestor de paquetes de Raspian:

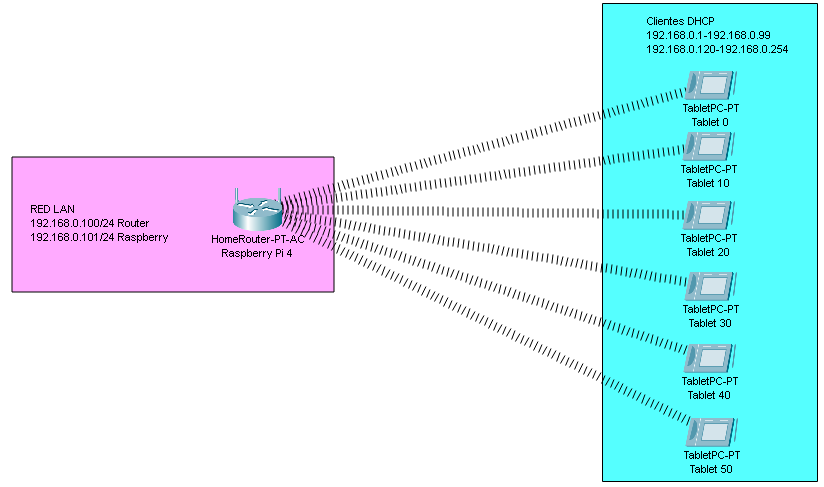
*sudo apt-get install isc-dhcp-server*

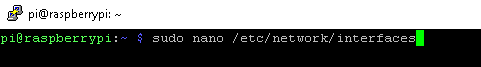


**NOTA**: Al instalarlo saldrá un mensaje de aviso, pero este sale porque actualmente no existe una buena configuración establecida



* 1. **Configuración del Servidor DHCP**





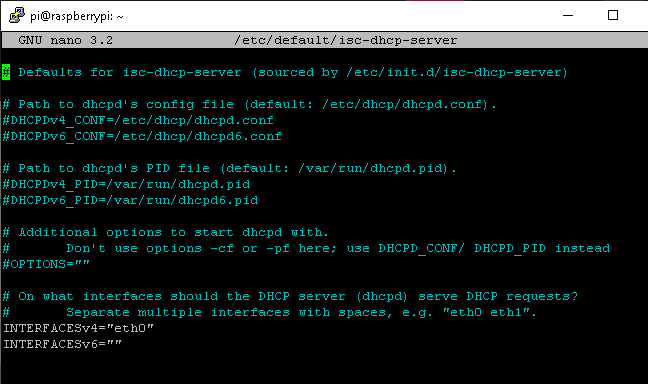
Hay que configurar principalmente dos archivos a la hora de instalar nuestro servidor DHCP:

* + - 1. **/etc/default/isc-dhcp-server**

En este archivo podremos configurar la interfaz de red en la que ejecutaremos el servicio. Así que deberemos abrir el archivo

sudo nano /etc/default/isc-dhcp-server y buscar la línea **“INTERFACES”** para dejarla tal que así:

INTERFACES = “eth0”



* + - 1. **/etc/dhcp/dhcpd.conf**

Algo opcional pero muy recomendable es que antes de modificar dicho archivo o cualquier otro archivo de configuración es la de hacer una copia de seguridad de este:

*sudo cp /etc/dhcp/dhcp.config /etc/dhcp/dhcpd.bakcup*

Luego se modifica dicho archivo

*sudo nano /etc/dhcp/dhcpd.conf*

Nos encontraremos con un archivo muy extenso, aunque la mayoría está comentada con # al principio de la línea. Podemos borrar todo su contenido o modificar y añadir las líneas que nos interesan:

#Opciones globales

ddns-update-style none; #Puede actualizar los registros de un DNS si la IP de un servidor de la LAN cambia, no nos interesa.

option domain-name "home.local"; #Establecemos un nombre de dominio.

option domain-name-servers 8.8.8.8, 8.8.4.4; #Configuramos los servidores DNS.

default-lease-time 259200; #Tiempo por defecto de la concesión en segundos.

max-lease-time 604800; #Tiempo máximo de la concesión en segundos.

authoritative; #Establece el servidor como principal.

log-facility local7;

#Opciones de subred

subnet 192.168.0.0 netmask 255.255.255.0 { #Declaración de la subred.

range 192.168.0.1 192.168.0.99; #Primer rango de direcciones.

range 192.168.0.120 192.168.0.254; #Segundo rango de direcciones.

option routers 192.168.0.100; #Gateway (nuestro router).

option subnet-mask 255.255.255.0; #Máscara de subred.

option broadcast-address 192.168.0.255; #Dirección de broadcast

}

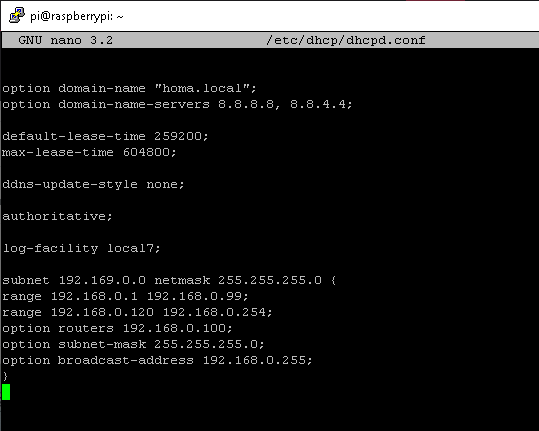
#Reserva de direcciones

host timbleck-pc1 { #Declaramos la reserva

hardware ethernet 00:11:22:33:44:55; #Dirección MAC del dispositivo.

fixed-address 192.168.0.110; #Dirección IP reservada.

}



* + - 1. **Opciones globales**: La primera parte del archivo, aquí se declaran las opciones que serán comunes para todas las subredes que configuremos en el servidor.
      2. **Opciones de subred:** Son opciones que sólo son válidas para dicha subred.
      3. **Reserva de direcciones:** Asociamos direcciones MAC a direcciones IP, cuando estas se conectan en vez de darles una IP de alguno de los rangos configurados les dará la que hayamos configurado manualmente.
  1. **Pasos finales**

El primer paso va a ser cambiar la IP interna del Router y asignarle la que hemos establecido anteriormente, así nos conectamos a su servidor web y le otorgamos la IP 192.168.0.100.

Lo siguiente será cambiar la IP de nuestro Raspberry:

*sudo nano /etc/network/interfaces*

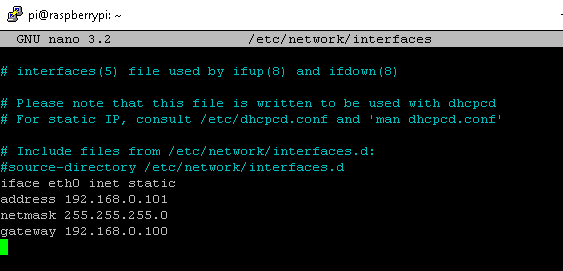
Debemos realizar la siguiente configuración en dicho archivo:

iface eth0 inet static

address 192.168.0.101

netmask 255.255.255.0

gateway 192.168.0.100



Con todo lo configurado anteriormente se procede a activar el servidor DHCP por medio del comando:

*sudo systemctl start isc-dhcp-server.service*

También se puede reiniciar(restart) el servidor DHCP o detener(stop) con los siguientes comandos:

*sudo systemctl restart isc-dhcp-server.service*

*sudo systemctl stop isc-dhcp-server.service*

Para terminar, se debe reiniciar el dispositivo Raspberry, el cual una vez reiniciado intentará levantar el servicio y así otorgará dicho rango de direcciones IP a los distintos dispositivos de la red.

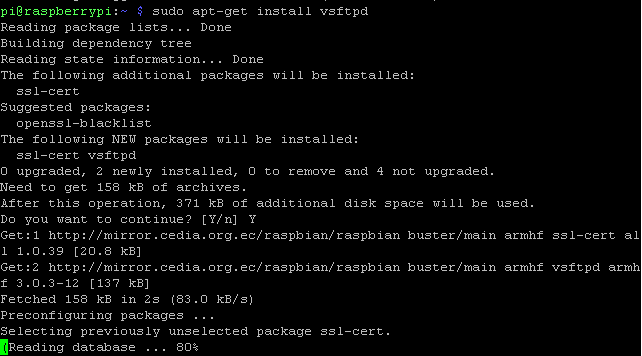
Para conectarse a la red se puede hacer físicamente con el cable ethernet o bien:

* Linux: sudo dhclient-r
* Windows: ipconfig /release y luego ipconfig /renew

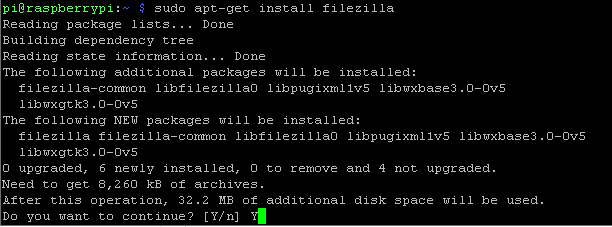
**Instalación y Configuración del Servidor FTP**

Se recomienda actualizar todo usando *sudo apt update && sudo apt full-upgrade*; una vez que haya terminado, reinícielo y comenzaremos con la instalación.

Lo primero que debemos hacer es instalar vsftpd de la siguiente forma:



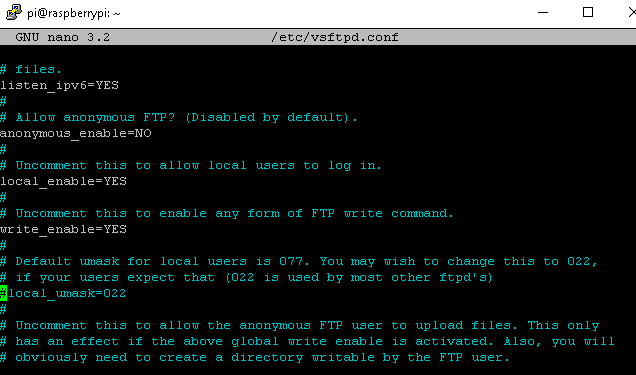
Luego se instala el Servicio de Cliente Filezilla



Luego debes configurar vsftpd para utilizar permitir la conexión local y el permiso de escritura, con el siguiente comando ingresados al archivo de configuración.

local\_enable=YES

write\_enable=YES

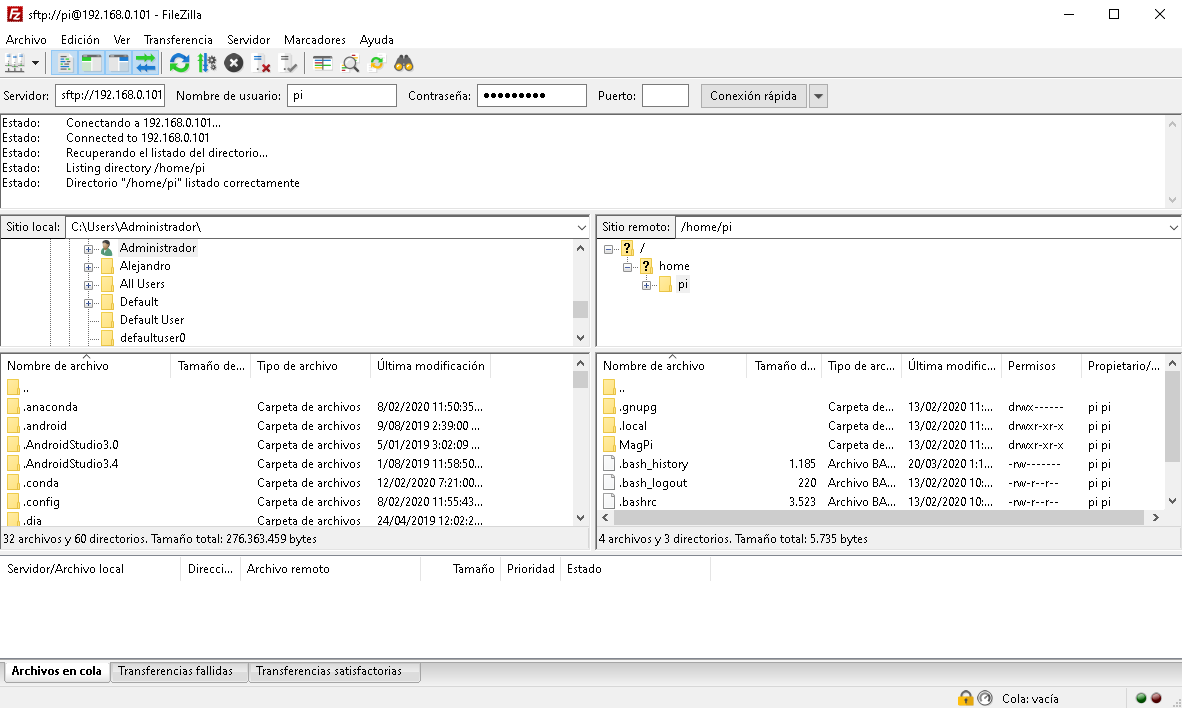


Ahora solo nos queda reiniciar el servicio de nuestro servidor FTP con el siguiente comando.

*sudo service vsftpd restart*

Con esto estamos listos para realizar nuestra primera conexión desde nuestro cliente FTP, para esto puedes usar Filezilla o el que estimes conveniente.

Los datos para conectarte son los mismos que usamos para conectarnos por SSH a nuestra Raspberry Pi, usuario "pi" y clave "raspberry" a no ser que la cambiáramos en algún momento, la única salvedad es que tenemos que colocar nuestra ip de la siguiente forma sftp://192.168.0.101, recordar que es mediante el puerto 22



**Raspberry como Router**

**Instalación**

**Preparación**

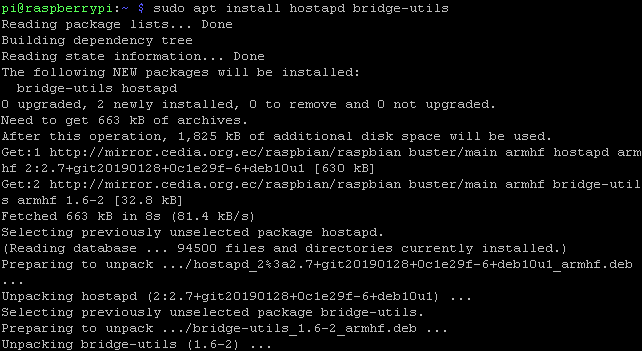
Use sudo raspi-config para configurar el Pi a su gusto, en este caso se recomienda cambiar la opción numero 2, la cual es el Wi-Fi de nuestra Raspberry.

Ahora actualícelo usando *sudo apt update && sudo apt full-upgrade*; una vez que haya terminado, reinícielo y comenzaremos con la instalación.

**Configuración hostapd**

Primero, instale hostapd y bridge-utils con el siguiente comando.

*sudo apt install hostapd bridge-utils*



Ahora viene la parte importante. El siguiente paso es editar el hostapd.conf.

*sudo nano /etc/hostapd/hostapd.conf*

Si el archivo no está vacío, simplemente elimine su contenido. Usaremos una configuración personalizada de todos modos. La configuración mínima que debe usar se ve así:

# Bridge mode

bridge=br0

# Networking interface

interface=wlan1

# WiFi configuration

ssid=RouterEDUAPP

channel=1

hw\_mode=g

country\_code=CO

ieee80211n=1

ieee80211d=1

wmm\_enabled=1

# WiFi security

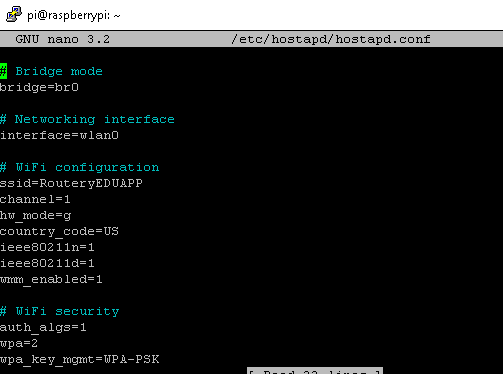
auth\_algs=1

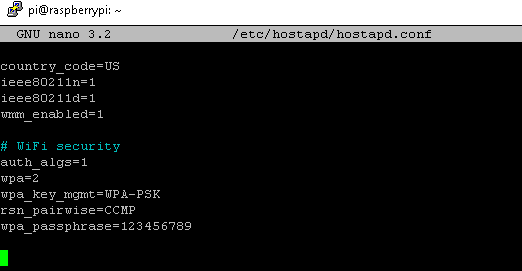
wpa=2

wpa\_key\_mgmt=WPA-PSK

rsn\_pairwise=CCMP

wpa\_passphrase=123456789





Simplemente pegue el texto en el archivo de configuración de hostapd, edite country\_code = CO en su país, guárdelo y listo. Luego debe actualizar wpa\_passphrase a su contraseña WiFi-personalizada y, opcionalmente, canalizar a otro canal WiFi.

Para crear y usar el puente de red solo tenemos que editar un archivo más. Ahora edite el archivo de interfaces de red así:

*sudo nano /etc/network/interfaces*

Se elimina todo lo que esté en este archive y se copia lo siguiente:

auto lo

iface lo inet loopback

# Ethernet

auto eth0

allow-hotplug eth0

iface eth0 inet manual

# WiFi

auto wlan0

allow-hotplug wlan0

iface wlan0 inet manual

wireless-power off

# Bridge

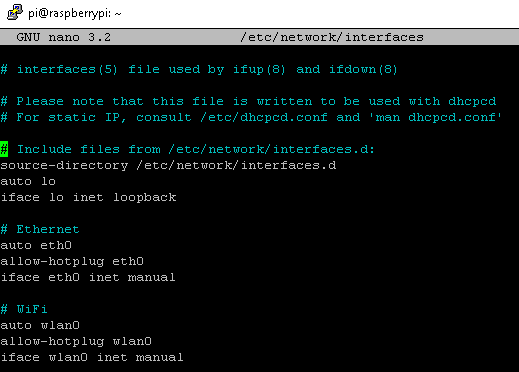
auto br0

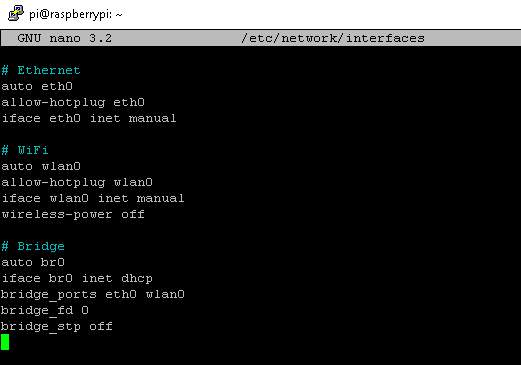
iface br0 inet dhcp

bridge\_ports eth0 wlan0

bridge\_fd 0

bridge\_stp off





**Pasos Finales**

Después de reiniciar rápidamente con sudo reboot, puede comenzar a probar todo. Para probar la configuración de hostapd, use este comando:

*sudo hostapd /etc/hostapd/hostapd.conf.*

Nota: Si sale sin que usted haga algo, entonces hay algo mal con eso. Para ingresar al modo de depuración, use:

*sudo hostapd -dd /etc/hostapd/hostapd.conf.*

Mientras prueba la configuración de hostapd, también puede intentar conectarse como cliente. Si ha hecho todo correctamente, debería poder hacerlo sin ningún problema.

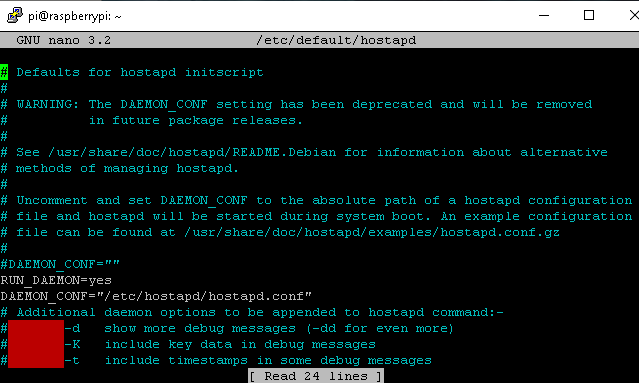
Para permitir que hostapd se ejecute al arrancar, debe editar un último archivo.

*sudo nano /etc/default/hostapd*

Ahora pegue este texto en el archivo y guárdelo:

RUN\_DAEMON=yes

DAEMON\_CONF="/etc/hostapd/hostapd.conf"



A partir de ahora, hostapd comenzará cada vez que se inicie su Pi.

**Problema de inicio automático**

Algunos usuarios informaron problemas con el método mencionado anteriormente de inicio automático de hostapd. Desde el lado de hostapd, todo esto debería funcionar, pero por alguna razón parece que a Raspbian ya no le gusta eso. La última solución es pegar los siguientes comandos para sobrescribir el enmascaramiento predeterminado de hostapd de Raspbian.

*sudo systemctl unmask hostapd*

*sudo systemctl enable hostapd*

*sudo systemctl start hostapd*

