RouterPi – AP - WirelessBridge

Done on a kali pi

By Ben Halpern

Goal:

The mission here is to create a raspberry pi Access Point first and for most. From there we will proceed to create a travel router, an access point that is able to piggyback public wifi and host a more secure network in public spaces.

First we must understand what Dual Band wifi is, since the Raspberry pi has dual band capabilities, and we should understand it before we proceed.

<https://www.lifewire.com/dual-band-wireless-networking-explained-818279>

Modern Wifi Adapters have dual-band capability, meaning that they are able to broadcast both 2.5ghz and 5ghz. However most if not all wireless adapters are only able capable of receiving/broadcasting one signal at a time. So in order to set up our Wireless Access Point/ Router we need another wireless dongle.

Raspberry Pi Access Point

Source and Articles:

<https://pimylifeup.com/raspberry-pi-wifi-extender/>

<https://www.shellvoide.com/wifi/setup-wireless-access-point-hostapd-dnsmasq-linux/>

<https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md>

<http://www.intellamech.com/RaspberryPi-projects/rpi3_simple_wifi_ap.html>

<https://seravo.fi/2014/create-wireless-access-point-hostapd>

So I think I’m going to combined the Wifi extender article and the access-point article to be able to create a router than both connect to the wifi as well as run as a simple private network AP.

First things first we need to install hostapd and dnsmasq to allow us to set up the AP and then stop the services since the configuration files aren’t set up yet.

```

sudo apt-get install dnsmasq hostapd -y

sudo systemctl stop dnsmasq

sudo systemctl stop hostapd

```

------------------------------ TMP until tested ------------------------------------

We need to configure wlan0, inform dhcpd to ignore wlan0 by editing sudo nano /etc/dhcpcd.conf

* To the end add ``` denyinterfaces wlan0 ```
* Configure the wlan IP sudo nano /etc/network/interfaces

```

allow-hotplug wlan0

iface wlan0 inet static

address 192.168.8.1

netmask 255.255.255.0

broadcast 192.168.8.255

```

Restart dhcpcd

**Now we need to configure a static IP to configure a standalone network to act as a server.**

sudo nano /etc/dhcpcd.conf

to set up a static IP address, edit the configuration file. Add the following to the end of the file.

```

Interface wlan0

static ip\_address= 192.168.8.1/24

nohook wpa\_supplicant

```

**-----TMP until tested ------**

We switch the interface into monitor mode to allow it to broadcast.

```

ifconfig

ifconfig wlan0 down

iwconfig wlan0 mode monitor

ifconfig wlan0 up

```

**Now we create the hostapd.conf file**

```

nano hostapd.conf

```

In the file write the following

```

interface=wlan0

driver=nl80211

ssid=[AP NAME]

hw\_mode=g

chanel=[AP Channe:6]

macaddr\_acl=0

ignore\_broadcast\_ssid=0

auth\_algs=1

wpa=2

wpa\_key\_mgmt=WPA-PSK

rsn\_pairwise=TKIP # or ccmp

wpa\_passphrase=pass

```

Now save the file

And edit the dnsmasq.conf

```  
nano /etc/dnsmasq.conf

```

The in the dnsmasq.conf add the following

```

Interface=wlan0

dhcp-range=192.168.8.2,192.168.8.30,255.255.255.0,12h

dhcp-option=3,192.168.8.1

dhcp-option=6,192.168.8.1

server=8.8.8.8

log-queries

log-dhcp

listen-address=127.0.0.1

```

-----traffic forwarding----------------------------------------------------------------------------------------------------

Set up traffic forwarding, edit the sysctl.conf file

```

sudo nano /etc/sysctl.conf

```

Add the following/uncomment the line

```

net.ipv4.ip\_forward=1

```

-------------------------------------------------------------------------------------------------------------------------------

Now we have our AP configured

Now we create the hostapd script to activate the AP

```

#!/bin/bash

#echo 1 > /proc/sys/net/ipv4/ip\_forward

#iptables –table nat –append POSTROUTING –out-interface wlan0 -j MASQUERADE

#iptables -A FORWARD -i wlan0 -o wlan1 -m state --state RELATED,ESTABLISHED -j ACCEPT

#iptables -A FORWARD -i wlan1 -o wlan0 -j ACCEPT

#iptables –table nat –append POSTROUTING –out-interface eth0 -j MASQUERADE

#iptables –append FORWARD –in-interface wlan0 -j ACCEPT

#Uncomment the lines above if you’d like to set up Ip forwarding to forward internet traffic through the router from one interface to another

service hostapd start

service dnsmasq start

echo run the dnsmasq start file in another terminal

hostapd /etc/hostapd/hostapd.conf

#EoF

```

Dnsmasq start

```

#!/bin/bash

#you can customize this script so it the user can set their custom interface ip address

#To configure the interface type the following in the terminal

ifconfig wlan0 up 192.168.8.1 netmask 255.255.255.0

route add -net 192.168.8.0 netmask 255.255.255.0 gw 192.168.8.1

dnsmasq -C dnsmasq.conf -d

#Eof

```

----Troubleshooting-------------------------------------------------------------------------------------------------------

When I run the my “startHostapd” script that I wrote it I get an error:

Line 2: invalid/unknown driver 'nl80211'

My issue turned out that instead of n1 it needs to be nl as in nL and I was able to start my AP

Getting the response

Wlan0: AP-ENABLED

Wireless Bridge

Sources and Articles

<https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md>