RPI-Wireless-Hotspot

From eLinux.org

Contents

- 1 What does it do?
- 2 What do you need?
- 3 What skill level is required?
- 4 How does it work?
- 5 Instructions

What does it do?

This project configures your Raspberry Pi to connect to the Internet through ethernet, and share that connection over WiFi.

What do you need?

- Any Raspberry Pi, model B with power supply
- A boot SD card for the Raspberry Pi.
- A USB WiFi device that supports "Access Point" mode; the Raspberry Pi 3 has a built-in AP Wi-Fi module.
- An Ethernet cable to connect to the local network.

Please make sure you Wifi dongle supports Access Point or Master Mode

- Edimax does **NOT** support Access Point (**UPDATE 8/22/15**: Edimax DOES support Access point. Updated project has rtl drivers in hostapd 2.4: Raspberry Hotspot with Edimax USB WiFi Adapter (https://github.com/lostincynicism/hostapd-rtl8188))
- AirLink 101 / AWL5088 does **NOT** support Access Point
- Panda Wireless Ultra, Mid-Range, 300Mbps and Dual Band Wireless N adapters support Access Point
- Ralink RT5370 and RT5372 **DO** support Access Point

What skill level is required?

This project does not require any coding or compilation. Very basic Linux and networking knowledge would be useful, but not essential.

To edit a configuration file (for example /etc/udhcpd.conf) use the following command

```
sudo nano /etc/udhcpd.conf
```

You will find yourself in a simple editor. Move around using the arrow keys. To save the file press Ctrl-o. To exit press Ctrl-x.

How does it work?

The Raspberry Pi is configured as a WiFi Hotspot, just like you would see in an internet cafe. It allows you to connect to the internet over WiFi using the Raspberry Pi as the bridge to the internet. The basic steps are

- Enable a WiFi Access Point and broadcast on the channel of your choice
- Assign dynamic IP addresses to any device that connects to WiFi network
- Join the WiFi and Ethernet networks together by using Network Address Translation

Instructions

The following steps were performed on Raspbian but should be much the same on any Debian-based distro.

1. Install the necessary software.

```
sudo apt-get install hostapd udhcpd
```

2. Configure DHCP. Edit the file /etc/udhcpd.conf and configure it like this:

```
start 192.168.42.2 # This is the range of IPs that the hostspot will give to client devices.
end 192.168.42.20
interface wlan0 # The device uDHCP listens on.
remaining yes
opt dns 8.8.8.8 4.2.2.2 # The DNS servers client devices will use.
opt subnet 255.255.255.0
opt router 192.168.42.1 # The Pi's IP address on wlan0 which we will set up shortly.
opt lease 864000 # 10 day DHCP lease time in seconds
```

Edit the file /etc/default/udhcpd and change the line:

```
DHCPD_ENABLED="no"
```

to

```
#DHCPD_ENABLED="no"
```

You will need to give the Pi a static IP address with the following command:

```
sudo ifconfig wlan0 192.168.42.1
```

To set this up automatically on boot, edit the file /etc/network/interfaces and replace the line "iface wlan0 inet dhcp" to:

```
iface wlan0 inet static
  address 192.168.42.1
  netmask 255.255.255.0
```

If the line "iface wlan0 inet dhcp" is not present, add the above lines to the bottom of the file.

Change the lines (they probably won't all be next to each other):

```
allow-hotplug wlan0
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet manual
```
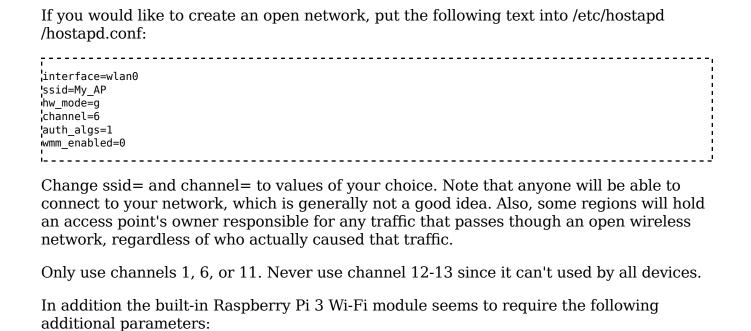
to:

```
#allow-hotplug wlan0
#wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
#iface default inet dhcp
```

3. Configure HostAPD. You can create an open network, or a WPA-secured network. A secure network is recommended to prevent unauthorized use and tampering, but you can also create an open network. To create a WPA-secured network, edit the file /etc/hostapd /hostapd.conf (create it if it doesn't exist) and add the following lines:

```
interface=wlan0
driver=nl80211
ssid=My_AP
hw_mode=g
channel=6
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=My_Passphrase
wpa_key_mgmt=WPA-PSK
#wpa_pairwise=TKIP      # You better do not use this weak encryption (only used by old client devices)
rsn_pairwise=CCMP
```

Change ssid=, channel=, and wpa_passphrase= to values of your choice. *SSID* is the hotspot's name which is broadcast to other devices, *channel* is what frequency the hotspot will run on, *wpa_passphrase* is the password for the wireless network. For many, many more options see the file /usr/share/doc/hostapd/examples/hostapd.conf.gz

If you would like to create an open network, put the following text into /etc/hostapd /hostapd.conf:

```
interface=wlan0
ssid=My_AP
hw_mode=g
channel=6
auth_algs=1
wmm_enabled=0
```

Change ssid= and channel= to values of your choice. Note that anyone will be able to connect to your network, which is generally not a good idea. Also, some regions will hold an access point's owner responsible for any traffic that passes though an open wireless network, regardless of who actually caused that traffic.

Only use channels 1, 6, or 11. Never use channel 12-13 since it can't used by all devices.

In addition the built-in Raspberry Pi 3 Wi-Fi module seems to require the following additional parameters:

```
ieee80211n=1          # 802.11n support
wmm_enabled=1         # QoS support
ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]
```

Edit the file /etc/default/hostapd and change the line:

```
#DAEMON_CONF=""
```

to:

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

4. Configure NAT (Network Address Translation). NAT is a technique that allows several devices to use a single connection to the internet. Linux supports NAT using Netfilter (also known as iptables) and is fairly easy to set up. First, enable IP forwarding in the kernel:

```
sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
```

To set this up automatically on boot, edit the file /etc/sysctl.conf and add the following line to the bottom of the file:

```
net.ipv4.ip_forward=1
```

Second, to enable NAT in the kernel, run the following commands:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

These instructions don't give a good solution for rerouting https and for URLs referring to a page inside a domain, like www.nu.nl/38274.htm. The user will see a 404 error. Your Pi is now NAT-ing. To make this permanent so you don't have to run the commands after each reboot, run the following command:

```
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

Now edit the file /etc/network/interfaces and add the following line to the bottom of the file:

```
up iptables-restore < /etc/iptables.ipv4.nat
```

5. Fire it up! Run the following commands to start the access point:

```
sudo service hostapd start
sudo service udhcpd start
```

Your Pi should now be hosting a wireless hotspot. To get the hotspot to start on boot, run these additional commands:

```
sudo update-rc.d hostapd enable
sudo update-rc.d udhcpd enable
```

At the completion of these instructions, your Pi should be providing a wireless network and allowing other devices to connect to the Internet. From my experience, the Pi makes a decent access point, although with cheaper WiFi dongles range will be fairly limited. I haven't stress tested this setup, but it seems to work fairly well and is handy when a "real" access point isn't available. I wrote most of the instructions from memory, if you find any errors/typos I'll correct them.

This tutorial originally was a post on the Raspberry Pi forum here (http://www.raspberrypi.org/phpBB3/viewtopic.php?f=36&t=19120), you can reply to that topic if you have issues. Thanks go to all the people who tested my tutorial on the forum, and to poing who contributed the WPA HostAPD config.

Please make sure you Wifi dongle supports Access Point or Master Mode

- Edimax doesn't support Access Point (**UPDATE 8/22/15**: Edimax DOES support Access point, hostapd 2.4 with rtl driver: Hostapd-rtl8188 (https://github.com /lostincynicism/hostapd-rtl8188))
- AirLink 101 / AWL5088 doesn't support Access Point

- Panda Wireless PAU03, PAU05, PAU06, PAU07, PAU09 Wireless N USB Adapters support Access Point
- Ralink RT5370 supports Access Point

Retrieved from "http://elinux.org/index.php?title=RPI-Wireless-Hotspot&oldid=417201"

- This page was last modified on 21 August 2016, at 23:58.
- This page has been accessed 409,712 times.
- Content is available under a Creative Commons Attribution-ShareAlike 3.0 Unported License unless otherwise noted.