Test Bed Installation Guide

Paul Patras

27 December, 2018

Test Bed Installation Guide

Software Package Pre-requisites

Given that you will work with manual configurations throughout, you will need several packages to offer IP addresses to clients connecting to the Wi-Fi Access Point (AP) that you will create and respectively to be able to resolve names. To achieve the first objective, this example works with isc-dhcp-server. The second will be handled by dnsmasq. The following commands should be issued (note these are run as super user, whilst in other cases elevating to root first will be mandatory).

```
\$ sudo apt-get install isc-dhcp-server
\$ sudo apt-get install dnsmasq
```

Configuration Files

(Three files are provided: `dhcpd.conf`, `wpa_supplicant.conf`, and 'hostapd.conf'. I will explain how these have to be modified/used.)

My advice is to use a public DNS server for name resolution. For this edit `dnsmasq.conf` file

```
\$ sudo vim /etc/dnsmasq.conf
```

Add the following line there

```
server=8.8.8.8
```

Check the identifiers of your wireless interfaces. It is likely that your AlfaNetwork card will show up with an identifier of the form `wlx000...` and the wireless adapter native to your laptop as `wlp1s0`. You want the AlfaNetwork interface to work as AP.

```
\$ iwconfig
```

Configure a static IP address for the AP interface. Your machine's WiFi card will be configured as client to the Wi-Fi network where you usually connect to (at homw), with address obtained via DHCP. To this end, edit /etc/network/interfaces

```
\$ sudo vim /etc/network/interfaces
```

and have entries similar to the ones shown below (adapt interface names as required):

```
auto wlx00c0ca8284d8
iface wlx00c0ca8284d8 inet static
        address 172.25.1.1
        netmask 255.255.255.0
        network 172.25.1.0
 auto wlp1s0
 iface wlp1s0 inet dhcp
```

The Wi-Fi Client (Test bed WAN)

As discussed, the wireless adapter native to your machine will act as client and provide the link to the Internet, whereas the AlfaNetwork device will serve your test mobile phone, with which you will experiment with various heart rate monitors.

The first step is thus to prepare the WAN link with wpa_supplicant. Begin by generating a configuration entry for your network

```
\$ wpa_passphrase networkname passphrase
```

In the above `networkname` is the ESSID of the Wi-Fi network to which you routinely connect to (e.g. at home) and `passphrase` should be replaced with your actual password that you use to connect to this network. The above command should produce an output of the following form:

```
network={
    ssid="networkname"
    #psk="passphrase"
    psk=bf3a0ba59951c1a440c41a6b39d3beb70b05adaf6d48fd87c78a3f83c1b3fef1
}
```

(Note that if you will end up using eduroam, the configuration will be different.)

Append the above block at the end of the `wpa_supplicant.conf` file provided

```
\$ vim wpa_supplicant.conf
```

The Wi-Fi AP (Test bed LAN)

Configure the AP by modifyting the `hostapd.conf` file provided. Presuming you are in the same folder where that file is located, run:

```
\$ vim hostapd.conf
```

Modify `line 8` to point to `wlx00....` The current example has the configuration of the machine I used for testing, i.e.

```
interface=wlx00c0ca8284d8
```

You need to put here the identifier that `iwconfig` returned to you for the AlfaNetwork card.

Then at `line 88` change the SSID name if you want your network to be called something else. Currently this is left as

```
ssid=test
```

Note that I presumed your home network uses WAP2 with PSK (see `line 1253`). If this is not the case, you will need to change this.

Uncomment `line 1262` and set a passphrase in ASCII. In the example provided, this is already uncommented and the you will see a line that reads

```
wpa_passphrase=goodburgersdeservefries
```

Presuming you have installed the isc-dhcp-server package, modify the attached `dhcpd.conf` file (if you are not happy with the address range set there) and then copy this file to `/etc/dhcp/dhcpd.conf`

```
\$ sudo cp dhcpd.conf /etc/dhcp/dhcpd.conf
```

Modify /etc/default/isc-dhcp-server

```
\$ sudo vim /etc/default/isc-dhcp-server
```

At `line 21` this should read (modify if your interface name is different):

```
INTERFACES="wlx00c0ca8284d8"
```

Launching All Services

(Note you may need to keep multiple terminals open to be able to track that everything work well.)

Stop network manager. (If at any stage you want to go back to using it, you need to comment out the changes you made to `/etc/network/interfaces` first.)

```
\$ sudo service network-manager stop
```

Start `dnsmasq`.

```
\$ sudo service dnsmasq restart
```

Restart client (Test bed WAN) interface

```
\$ sudo ifdown wlp1s0; sudo ifup wlp1s0
```

Then start the client. Note that for this you need to become root (`sudo` will not work).

```
\$ sudo su
\# wpa_supplicant -i wlp1s0 -d -c wpa_supplicant.conf
```

Get an address for this interface (replace interface name if needed).

```
\$ sudo dhclient wlp1s0
```

Restart the AP interface.

```
\$ sudo ifdown wlx00c0ca8284d8; sudo ifup wlx00c0ca8284d8
```

Start the `hostapd` daemon.

```
\$ sudo hostapd hostapd.conf
```

Start the DHCP server.

```
\$ sudo service isc-dhcp-server restart
```

Enable port forwarding (you need to become root first) and forwarding rules (change interface names if needed).

```
\$ sudo su
\# echo 1 > /proc/sys/net/ipv4/ip_forward

\# iptables -A FORWARD -i wlx00c0ca8284d8 -o wlp1s0 -j ACCEPT
\# iptables -A FORWARD -i wlp1s0 -o wlx00c0ca8284d8 -m state --state ESTABLISHED,RELATED -j ACCEPT
\# iptables -t nat -A POSTROUTING -o wlp1s0 -j MASQUERADE
```

Your test phone should see the newly created newtowrk and be able to connect to it. To test that this worked, disable the mobile data on the phone and browse to any web page.

This should be it. You can now launch `wireshark` or `mitmproxy` on the AP interface, i.e.`wlx00...` to start capturing traffic.