

Software Access Point

From ArchWiki

A software access point is used when you want your computer to act as an wifi access point for the local wireless network. It saves you the trouble of getting a separate wireless router.

Related articles

Network Configuration

Wireless Setup

Ad-hoc networking

Internet Sharing

Contents

- 1 Requirements
- 2 Overview
- 3 Wifi Link Layer
- 4 Network configuration
 - 4.1 Bridge Setup
 - 4.2 NAT Setup
- 5 Scripts
- 6 Troubleshooting
 - 6.1 WLAN is very slow
 - 6.2 NetworkManager is interfering
- 7 See also

Requirements

- A nl80211 compatible wireless device (e.g. ath9k)

Overview

Setting up an access point comprises two main parts:

- Setting up the **wifi link layer**, so that wireless clients can associate to your computer's "software access point" and send/receive IP packets from/to your computer; this is what the `hostapd` package will do for you.
- Setting up the **network configuration** on you computer, so that your computer will properly relay IP packets from/to its own Internet connection from/to wireless clients.

Wifi Link Layer

The actual Wifi link is established via the `hostapd` (<https://www.archlinux.org/packages/?name=hostapd>) package (available in the official repositories). The package has WPA2 support.

Adjust the options in `hostapd` configuration file if necessary. Especially, change the `ssid` and the

`wpa_passphrase` . See `hostapd` Linux documentation page (<http://wireless.kernel.org/en/users/Documentation/hostapd>) for more information.

```
/etc/hostapd/hostapd.conf
-----
ssid=YourWifiName
wpa_passphrase=Somepassphrase
interface=wlan0
bridge=br0
auth_algs=3
channel=7
driver=nl80211
hw_mode=g
logger_stdout=-1
logger_stdout_level=2
max_num_sta=5
rsn_pairwise=CCMP
wpa=2
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP CCMP
-----
```

For automatically starting `hostapd`, enable the `hostapd.service` .

Note: If you have a card based on RTL8192CU chipset, you will have to build this (<http://anarsoul.blogspot.com.es/2013/08/access-point-with-raspberry-pi-and.html>) patched version of *hostapd* and replace `driver=nl80211` with `driver=rtl871xdrv` in the `hostapd.conf` file.

Network configuration

There are two basic ways for implementing this:

1. **bridge:** create a network *bridge* on your computer (wireless clients will appear to access the same network interface and the same subnet that's used by your computer)
2. **NAT:** with IP forwarding/masquerading and DHCP service (wireless clients will use a dedicated subnet, data from/to that subnet is NAT-ted -- similar to a normal WiFi router that's connected to your DSL or cable modem)

The bridge approach is simpler, but it requires that any service that's needed by your wireless clients (like, DHCP) is available on your computers external interface. That means it will not work if you have a dialup connection (e.g., via PPPoE or a 3G modem) or if you're using a cable modem that will supply exactly one IP address to you via DHCP.

The NAT aproach is more versatile, as it clearly separates wifi clients from your computer and it's completely transparent to the outside world. It will work with any kind of network connection, and (if needed) you can introduce traffic policies using the usual iptables approach.

Of course, it is possible to *combine both things*. For that, studying both articles would be necessary. Example: Like having a bridge that contains both an ethernet device and the wireless device with an static ip, offering DHCP and setting NAT configured to relay the traffic to an additional network device - that can be ppp or eth.

Bridge Setup

You need to create a network *bridge* and add your network interface (e.g. `eth0`) to it. You **should not** add the

wireless device (e.g. `wlan0`) to the bridge; `hostapd` will add it on its own.

If you use `netctl`, see [Bridge with netctl](#) for details (just do not add `tap0` used in that example).

(If you have an existing bridge, used by a virtual machine for example, then you should reuse that one.)

NAT Setup

See [Internet Sharing](#) for details.

On that article, the device connected to the lan is `net0`. That device would be in this case your wireless device (e.g. `wlan0`).

Scripts

`create_ap` (<https://bbs.archlinux.org/viewtopic.php?pid=1269258>) script combines `hostapd` (<https://www.archlinux.org/packages/?name=hostapd>), `dnsmasq` and `iptables` to create a NATed Access Point.

Troubleshooting

WLAN is very slow

This could be caused by low entropy. Consider installing `haveged`.

NetworkManager is interfering

`hostapd` may not work, if the device is managed by `NetworkManager`. You can mask the device:

<code>/etc/NetworkManager/NetworkManager.conf</code>
<pre>[keyfile] unmanaged-devices=mac:<hwaddr></pre>

See also

- Router
- Hostapd : The Linux Way to create Virtual Wifi Access Point (<http://nims11.wordpress.com/2012/04/27/hostapd-the-linux-way-to-create-virtual-wifi-access-point/>)
- tutorial and script for configuring a subnet with DHCP and DNS (http://xyne.archlinux.ca/notes/network/dhcp_with_dns.html)

Retrieved from "https://wiki.archlinux.org/index.php?title=Software_Access_Point&oldid=291755"

Category: Wireless Networking

-
- This page was last modified on 5 January 2014, at 23:39.

- Content is available under GNU Free Documentation License 1.3 or later unless otherwise noted.