Raspberry Pi - Support #813

Setup a Raspberry Pi as a Wireless Access Point/Router on Arch

05/23/2016 08:45 PM - Daniel Curtis

Status:	Rejected	**Start date:**	05/23/2016
Priority:	Normal	**Due date:**	
Assignee:	Daniel Curtis	**% Done:**	40%
Category:	Wireless Support	**Estimated time:**	1.50 hour
Target version:	Arch Linux	**Spent time:**	4.50 hours

Description

Here is a procedure on how I setup a Raspberry Pi as a wireless access point or wireless router on Arch Linux.

NOTE: The original project used Raspbian as the base Operating System, so I needed to adapt the existing code to work on Arch Linux.

WARNING: This is currently (6/4/16) a work in progress. As such some features may not work properly.

Prepare the Environment

- Before installation of the components, make sure everything is up to date using the following command:

```
pacman -Syu
```

- Install a few dependencies:

```
pacman -S sudo nginx git curl ntp dnsmasq hostapd wireless_tools
```

- Edit the sudoers file:

```
visudo
```

  - Add the following to the end of the file:

```
http ALL=(ALL) NOPASSWD:/usr/bin/ifconfig,/usr/bin/sudo,/usr/bin/ip link set wlan0 down,/u
sr/bin/ip link set wlan0 up,/bin/cat /etc/wpa_supplicant/wpa_supplicant-wlan0.conf,/bin/cp
 /tmp/wifidata /etc/wpa_supplicant/wpa_supplicant-wlan0.conf,/usr/bin/wpa_cli scan_results
, /usr/bin/wpa_cli scan,/usr/bin/systemctl restart wpa_supplicant@wlan0,/usr/bin/systemctl
 enable wpa_supplicant@wlan0,/usr/bin/systemctl disable wpa_supplicant@wlan0,/usr/bin/syst
emctl is-enabled,/bin/cp /tmp/hostapddata /etc/hostapd/hostapd.conf, /bin/systemctl start
hostapd,/bin/systemctl stop hostapd,/bin/systemctl stop dnsmasq, /bin/systemctl stop dnsma
sq,/bin/cp /tmp/dhcpddata /etc/dnsmasq.conf
nf
```

- Start and enable ntpd:

```
systemctl enable ntpd
systemctl start ntpd
```

- Edit the wpa_supplicant config file:

```
nano /etc/wpa_supplicant/wpa_supplicant-wlan0.conf
```

- And add the following:

```
ctrl_interface=/run/wpa_supplicant
update_config=1
```

- Start and enable wpa_supplicant for wlan0 at boot:

```
systemctl enable wpa_supplicant@wlan0
systemctl start wpa_supplicant@wlan0
```

Install PHP

- Install fcgiwrap

```
pacman -S fcgiwrap php-fpm
```

- Edit the php-fpm config:

```
vi /etc/php/php-fpm.conf
```

  - Make the following changes:

```
listen = /run/php-fpm/php-fpm.sock
listen.owner = http
listen.group = http
listen.mode = 0660
```

- Create a configuration directory to make managing individual server blocks easier

```
mkdir /etc/nginx/conf.d
```

- Edit the main nginx config file:

```
vi /etc/nginx/nginx.conf
```

  - And strip down the config file and add the include statement at the end to make it easier to handle various server blocks:

```
worker_processes  1;
error_log  /var/log/nginx-error.log;

events {
    worker_connections  1024;
}

http {
    include       mime.types;
    default_type  application/octet-stream;
    sendfile        on;
    keepalive_timeout  65;

    include /etc/nginx/conf.d/*.conf;
}
```

- Start and enable nginx, fcgiwrap, and php-fpm at boot:

```
systemctl enable nginx
systemctl start nginx
systemctl enable fcgiwrap
systemctl start fcgiwrap
systemctl enable php-fpm
systemctl start php-fpm
```

Install WiFiPi AP

- Clone the WiFiPi AP code:

```
sudo git clone https://git.gnetsolutions.net/TokinRing/wifipi-ap.git /srv/http/wifipi.example.com
```

- Set the files ownership to the http user:

```
sudo chown -R http:http /srv/http/wifipi.example.com
```

- Add a **wifipi.example.com server block**:

```
vi /etc/nginx/conf.d/wifipi.example.com.conf
```

  - Add the following:

```
proxy_buffer_size   128k;
proxy_buffers   4 256k;
proxy_busy_buffers_size   256k;

server {
    listen       80;
    server_name  wifipi.example.com;
    root         /srv/http/wifipi.example.com;
    access_log   /var/log/wifipi.example.com-access.log;
    error_log    /var/log/wifipi.example.com-error.log;

    location / {
        index  index.php index.html index.htm;
    }

    location ~ \.php$ {
        fastcgi_pass unix:/run/php-fpm/php-fpm.sock;
        fastcgi_buffer_size 128k;
        fastcgi_buffers 4 256k;
        fastcgi_busy_buffers_size 256k;
        fastcgi_param SCRIPT_FILENAME /srv/http/wifipi.example.com$fastcgi_script_name;
        fastcgi_param PATH_INFO $fastcgi_script_name;
        include fastcgi_params;
    }

    location ~ \.cgi$ {
        root           /srv/http/wifipi.example.com;
        fastcgi_buffer_size 128k;
        fastcgi_buffers 4 256k;
        fastcgi_busy_buffers_size 256k;
        fastcgi_pass   unix:/run/fcgiwrap.sock;
```

```
        include         fastcgi.conf;
    }
}
```

- Add the PHP files for the site and change the ownership to the http user:

```
chown -R http:http /srv/http/wifipi.example.com
```

- Restart nginx:

```
systemctl restart nginx
```

Resources

- https://github.com/billz/raspap-webgui
- http://sirlagz.net/2013/02/06/script-web-configuration-page-for-raspberry-pi/
- http://raspberrypihq.com/how-to-turn-a-raspberry-pi-into-a-wifi-router/
- http://sirlagz.net/2012/08/09/how-to-use-the-raspberry-pi-as-a-wireless-access-pointrouter-part-1/
- https://wiki.archlinux.org/index.php/Software_Access_Point

History

#1 - 05/23/2016 10:25 PM - Daniel Curtis

- Description updated

- Status changed from New to In Progress

- % Done changed from 0 to 20

#2 - 06/04/2016 02:16 PM - Daniel Curtis

- Description updated

- % Done changed from 20 to 40

#3 - 06/04/2016 03:02 PM - Daniel Curtis

- Description updated

#4 - 02/09/2022 09:36 AM - Daniel Curtis

- Status changed from In Progress to Rejected