

## Montando un Punto de acceso WIFI en FreeBSD



Syvic

Miércoles 10 diciembre 2003.

Una guía para gente ya experimentada en el sistema operativo libre [FreeBSD](#), para montar un punto de acceso para una red ciudadana inalámbrica.

Con ello queremos mostrar el uso de otras plataformas libres ,tan excelentes como GNU/Linux, para expandir el uso del software libre, la creación de redes ciudadanas inalámbricas y en definitiva la conquista de la red y lo que la rodea, por parte de sus usuarios.

Buen provecho.

### **0.- Propósito**

Queremos montar un punto de acceso con FreeBSD (4.8 o superior) que haga NAT con la red interna, para posteriormente poder encaminar el tráfico hacia internet. También necesitaremos monitorizar su comportamiento y que sea capaz de asignar IPs dinámicamente a los clientes wireless (DHCP). La tarjeta wireless debe llevar el chipset PRISM2 ¿Por qué NAT en lugar de bridging? Porque permite mucha más flexibilidad a la hora de tratar los paquetes, es mucho más seguro y además podremos "ver" via tcp/ip al AP ya que con bridge no podríamos. (O yo no lo conseguí) El índice es el siguiente:

- ▶ 1.- Configurando el núcleo
- ▶ 2.- Configurando la tarjeta bridge pcmcia/pci
- ▶ 3.- Configuración del firewall/NAT
- ▶ 4.- Levantar el interface wireless y definir parámetros
- ▶ 5.- Instalación de los ports necesarios
- ▶ 6.- Control de ancho de banda
- ▶ 7.- Configuración del dhcp
- ▶ 8.- Monitorización y algo de SPAM

### **1.- Configurando el núcleo**

Un AP wireless requiere que el sistema operativo reconozca los dispositivos necesarios y que pueda manejar ciertas opciones de red necesarias para permitir el acceso de una red a otra. Todo ello lo tendremos que hacer añadiendo nuevas opciones a nuestro kernel en caso de que no cuente con ellas. Como ya hay un montón de documentación entorno a este proceso (ver [El demonio\(www.eldemonio.org\)](http://www.eldemonio.org) me limitaré a indicar las opciones necesarias:

*device card*

Si tienes un kernel anterior al 4.8 deberás elegir entre una de las siguientes opciones en lugar de la anterior:

*device cardbus #(16 bits)*

## Montando un Punto de acceso WIFI en FreeBSD

device pccard #(32 bits)

device pcic #(PCI bridges)

device cbb #(Yenta)

Este es el driver que controlará la tarjeta:

*device wi*

Si quieres poder crear túneles por internet con otros nodos, necesitarás lo siguiente:

*pseudo-device tun*

pseudo-device gif

pseudo-device faith 1

Para poder habilitar el NAT, necesitaremos añadir las opciones del firewall (ipfw):

*options IPFIREWALL options IPFIREWALL\_VERBOSE*

options IPFIREWALL\_VERBOSE\_LIMIT=1

options IPDIVERT

options DUMMYNET

Otras opciones interesantes:

*options RANDOM\_IP\_ID*

options TCP\_DROP\_SYNFIN

options IPSEC

options IPSEC\_DEBUG

Si tienes duda con respecto a alguna de estas opciones, no tienes más que leer el fichero LINT en el mismo directorio bajo el cual estás configurando el kernel. Una vez nos hemos asegurado que estas opciones forman parte de nuestro kernel, compilamos, lo instalamos y reiniciamos.

Al hilo del kernel, añadiría las siguientes opciones al fichero **/etc/sysctl.conf**, que se leen durante el arranque y modifican el comportamiento del kernel en ciertos aspectos:

*net.inet.tcp.blackhole=2*

net.inet.udp.blackhole=1

net.inet.ip.forwarding=1

```
net.inet.ip.check_interface=1
```

También las podemos aplicar sin reiniciar mediante el comando sysctl. Ej: **sysctl net.inet.ip.forwarding=1**

### 2.- Configurando la tarjeta bridge pcmcia/pci

o primero que debemos hacer es que nuestro FreeBSD reconozca la tarjeta wireless pcmcia. Si el nodo lo estamos creando en un PC de sobremesa (lo más común), necesitaremos configurar el bridge pcmcia/pci.

Esto lo hacemos añadiendo a nuestro **/etc/rc.conf** las siguientes líneas:

```
pccard_enable="YES"

pccard_mem="DEFAULT"

pccard_beeper="2"

pccard_ifconfig="NO"

pccardd=/usr/sbin/pccardd

pccardc=/usr/sbin/pccardc
```

Que básicamente definen el comportamiento del programa pccard, el encargado de facilitar al S.O. el acceso a la tarjeta wireless. Si reiniciamos en este punto el ordenador, debería soltar unos cuantos pitidos al arrancar indicando que ha reconocido una tarjeta dentro del bridge. No es necesario reiniciar ahora.

En caso de que no reconozca la tarjeta al reiniciar, será necesario tocar el fichero **/etc/defaults/pccard.conf** y ajustar los parámetros necesarios dependiendo de la tarjeta.

### 3.- Configuración del firewall/NAT

Añadiremos las siguientes líneas a nuestro **/etc/rc.conf**:

```
gateway_enable="YES"

firewall_enable="YES"

firewall_type="/etc/firewall.rules"

firewall_logging="YES"

firewall_script="/etc/rc.firewall"

natd_enable="YES"

natd_interface="ed0"

natd_flags="-log_ipfw_denied yes"
```

## Montando un Punto de acceso WIFI en FreeBSD

Las cuales definen el comportamiento del firewall y del natting. Si nuestra tarjeta de red es diferente a ed0, cambiaremos este parámetro por el correspondiente. Lo podemos mirar con un simple '**ifconfig**'. El fichero **/etc/firewall.rules** lo tendremos que crear con el siguiente contenido:

```
add divert natd all from any to any via ed0
```

```
add pass all from any to any
```

Recuerda cambiar ed0 por el dispositivo de tu tarjeta de red ethernet.

### 4.– Levantar el interface wireless y definir parámetros

Por último configuraremos la interfaz wireless. Yo lo hago utilizando el fichero **/etc/rc.local**, que se ejecuta al final del arranque del servidor. Ahí deberemos añadir las siguientes líneas, cambiando los parámetros necesarios:

```
ifconfig wi0 10.0.0.1 netmask 255.255.255.0 broadcast 10.0.0.255
```

```
ifconfig wi0 inet up ssid test media DS/11Mbps mediaopt hostap stationname host
```

```
ifconfig wi0 10.0.0.1 netmask 255.255.255.0 broadcast 10.0.0.255
```

Cambia test por el essid que quieras poner y host por el nombre de tu máquina.

Si, la primera y la última línea están repetidas, es correcto. Si no te funciona correctamente, o el servidor arranca con la interface wi0 inactiva, prueba a poner esto antes de las tres líneas anteriores:

```
pccardc power 0 0
```

```
sleep 10
```

```
pccardc power 0 1
```

```
sleep 20
```

Esto reinicia la tarjeta wireless. La primera línea desactiva la tarjeta y la tercera la vuelve a activar 10 segundos después.

### 5.– Instalación de los ports necesarios

Para poder controlar nuestro nodo, necesitaremos una serie de programas adicionales, que podremos encontrar en el árbol de ports. Yo recomiendo instalar los siguientes: # **-bsd-airtools** Utilidades varias para wireless # **-trafshow** Monitoriza el tráfico de red por interface # **-isc-dhcp3** Servidor DHCP

Recomiendo la instalación mediante el programa '**portinstall**' en lugar del clásico '**make install**' sobre su árbol de ports

### 6.– Control de ancho de banda

Tenemos la posibilidad de configurar el ancho de banda que puede coger cada uno de los clientes o incluso una IP en particular, para ello deberemos añadir las siguientes líneas al fichero **/etc/firewall.rules**:

## Montando un Punto de acceso WIFI en FreeBSD

```
add pipe 1 ip from 10.0.0.0/24 to any out
```

```
add pipe 2 ip from any to 10.0.0.0/24 out
```

```
pipe 1 config bw 10Kbit/s queue 5Kbytes
```

```
pipe 2 config bw 20Kbit/s queue 7Kbytes
```

Esto le daría a cada cliente un ancho de banda de 27Kbs de bajada como máximo y 15Kbs de subida. La opción queue indica el extra de ancho de banda que asignará cuando tenga de sobra, en caso contrario se aplicaría el valor de la columna anterior. Si queremos que una ip en concreto tenga más ancho de banda pondremos algo como esto:

```
add pipe 3 ip from 10.0.0.100 to any out
```

```
add pipe 4 ip from any to 10.0.0.100 out
```

```
pipe 3 config bw 100Kbit/s queue 50Kbytes
```

```
pipe 4 config bw 100Kbit/s queue 50Kbytes
```

Lo que le daría la posibilidad de subir y bajar paquetes a 150kb/s

### 7.- Configuración del dhcp

Este es el programa encargado de repartir las ips a los clientes. El fichero de configuración está en **/usr/local/etc/dhcpd.conf** y para un ejemplo básico nos valdrá este:

```
default-lease-time 600;
```

```
max-lease-time 7200;
```

```
option subnet-mask 255.255.255.0;
```

```
option broadcast-address 10.0.0.255;
```

```
option routers 10.0.0.1; # La IP de tu AP
```

```
option domain-name-servers 172.26.0.1 # La IP de tu DNS (puede ser externo)
```

```
option domain-name "midominio.com";
```

```
# Este es el rango de ips que asignaremos. En nuestro caso de la 100 a la 200
```

```
subnet 10.0.0.0 netmask 255.255.255.0 " range 10.0.0.100 10.0.0.200; "
```

Para que se arranque este servicio junto al servidor deberemos cambiar el nombre del script de inicialización. Es decir:

```
cd /usr/local/etc/rc.d
```

```
mv isc-dhcpd.sh.sample isc-dhcpd.sh
```

### 8.- Monitorización y algo de SPAM

# Con uno de los ports que hemos instalado, el trafshow seremos capaces de ver el tráfico que se genera por interface. Las utilidades ifconfig y wicontrol nos mostrarán el estado de la conexión wireless y varios parámetros referidos a la configuración tcp y a la configuración relativa a 802.11b Y ahora, el SPAM ;-)

Si conoceis la utilidad **wavemon para linux**, echareis en falta algo similar para \*BSD.

Para medio suplir esta herramienta, he creado un script llamado **AWMU** que nos da una información similar a la de wavemon. En cuanto aprenda a programar en ncurses prometo portarla, ya que en shell script es bastante feo/lento. Lo podeis encontrar, junto con alguna captura de pantalla [aquí\(http://sindormir.net/awmu/\)](http://sindormir.net/awmu/)

---

Por **Syvic(a)sindominio.net**, publicado originalmente en su weblog: <http://blog.sindormir.net/index.php?p=15>>

---



Esta publicación esta bajo la licencia [creative commons](http://creativecommons.org/licenses/by-nc-sa/4.0/). Por tanto, se permite difundir, citar y copiar literalmente sus materiales, de forma íntegra o parcial, por cualquier medio y para cualquier propósito, siempre que se mantenga esta nota y se cite procedencia. Suburbia no asume ninguna responsabilidad por los comentarios que hagan los visitantes en este sitio. Toda la responsabilidad para verificar la veracidad y los derechos de reproducción de un envío corresponden al autor/a que lo publica. Al publicar material en este sitio, el o la autora del envío asume que puede ser redistribuido libremente.