

Controlador Host AP para integrados Prism2 de Intersil

Jouni Malinen

Traducción al castellano: Paul Salazar Mora

Tabla de contenidos

1. [Introducción](#)
2. [Descarga](#)
3. [Comentarios, correo, lista de distribución](#)
4. [Equipo necesario](#)
 - 4.1. [Hardware](#)
 - 4.2. [Software](#)
5. [Instalación y configuración](#)
 - 5.1. [Sistemas con módulos PCMCIA incluidos en el núcleo](#)
 - 5.2. [Sistemas con módulos pcmcia-cs externos](#)
6. [Bridging entre redes guiadas \(con cables\) e inalámbricas.](#)
7. [Monitorización IEEE 802.11](#)

1. Introducción

Este es un controlador para tarjetas inalámbricas basadas en el integrado Prism2 de Intersil. El controlador activa un modo de funcionamiento de la tarjeta denominado Host AP, lo que significa que implementa las funciones de gestión IEEE 802.11, haciendo funcionar al ordenador que aloja la tarjeta como un Punto de Acceso (AP). Este modo no requiere ningún firmware especial en la tarjeta inalámbrica. Además del modo Host AP, el controlador implementa las operaciones normales en modo BSS e IBSS.

El modo Host AP que el firmware de Intersil implementa en los integrados Prism2 se ocupa de tareas críticas en tiempo, como el envío de balizas y el acuse de recibo de tramas (*frame acknowledging*), pero deja otras tareas de gestión al controlador, el cuál implementa la funcionalidad básica necesaria para inicializar y configurar tarjetas basadas en Prism2, envío y recepción de tramas, y para recoger estadísticas. También incluye las siguientes funciones IEEE 802.11: autenticación (y desautenticación), asociación (reasociación y disociación), transmisión de datos entre dos estaciones inalámbricas, modo de señalización de ahorro energético (PS), y almacenado de tramas (*frame buffering*) para estaciones en modo PS. El controlador también tiene varias funciones para depuración de desarrollos y para investigación de entornos IEEE 802.11, como acceso a los registros de configuración *hardware*, registros de E/S, y tramas con cabeceras 802.11.

2. Descarga

Copyright (c) 2001-2002, SSH Communications Security Corp

Autor: Jouni Malinen, <jkmaline@cc.hut.fi> o <jkmaline@ssh.com>

Este programa es software libre; puede redistribuirlo y/o modificarlo de acuerdo con los términos

de la licencia [GNU General Public License](#) versión 2, publicada por la fundación Free Software Foundation. Lea el archivo COPYING para más detalles.

Nota importante: esta es una versión de desarrollo destinada principalmente a investigación. No se ha probado exhaustivamente, y puede presentar algunos errores. No debe utilizarse en sistemas de producción. La instalación de este controlador puede requerir conocimientos más o menos extensos de IEEE 802.11, pcmcia-cs, Linux, etc.

Distribución: <http://people.ssh.com/jkm/Prism2/Prism2-2002-03-27.tar.gz>

Cambios: <http://people.ssh.com/jkm/Prism2/ChangeLog>

Problemas conocidos en esta versión:

- cuando la tarjeta inalámbrica está aún activa, cerrar pcmcia-cs puede dejar el módulo cargado en memoria; retirar la tarjeta después de esto puede colgar el sistema (esto se puede evitar desactivando siempre el dispositivo antes de sacarlo, esto es, hacer `'ifconfig wlan0 down'` antes de cerrar pcmcia-cs). Esto parece funcionar correctamente, al menos con las instalaciones basadas en controladores pcmcia del núcleo.

Versiones antiguas:

- [Prism2-2002-02-13.tar.gz](#)
- [Prism2-2002-01-03.tar.gz](#)
- [Prism2-2001-12-18.tar.gz](#)
- [Prism2-2001-12-10.tar.gz](#)
- [Prism2-2001-11-10.tar.gz](#)
- [Prism2-2001-11-05.tar.gz](#)
- [Prism2-2001-09-27.tar.gz](#)
- [Prism2-2001-07-11.tar.gz](#)

3. Comentarios, correo, lista de distribución

Cualquier comentario, notificación de éxito o fallo, idea para mejoras, petición de funcionalidad nueva, etc., se agradecerá mucho, y deberá enviarse a [<jkm@ssh.com>](mailto:jkm@ssh.com) y/o [<jkmaline@cc.hut.fi>](mailto:jkmaline@cc.hut.fi).

Por favor, entended que no siempre tengo tiempo para responder mensajes de una forma rápida, aunque trataré de leerlos todos cuando el tiempo me lo permita. Existe también una lista de distribución para mensajes relacionados con el controlador Host AP. Como esta lista tienen una audiencia mayor, es posible que aquí reciba una respuesta más rápida. De cualquier manera, la lista está recomendada para cuestiones generales sobre el controlador y su desarrollo. Además, enviaré a esta lista los avisos cuando una versión nueva se encuentre disponible.

La dirección de la lista es [<hostap@lists.ssh.com>](mailto:hostap@lists.ssh.com) (es decir, que los mensajes enviados a esta dirección serán enviados a todos los suscriptores). Puede suscribirse enviando un mensaje a [<hostap-subscribe@lists.ssh.com>](mailto:hostap-subscribe@lists.ssh.com). La lista está gestionada con Ezmlm, y sus funciones como la consulta de mensajes pasados pueden utilizarse a través de correo electrónico. También es posible leer los mensajes de la lista a través [de la web](#).

4. Equipo necesario

4.1. Hardware

El controlador debería funcionar con tarjetas inalámbricas basadas en el integrado Prism2 de Intersil, especialmente con su procesador MAC (HFA3841 o HFA3842). Esta versión se probó con éxito en unos cuantos modelos diferentes, a saber, la Compaq WL100 y WL200, D-Link DWL-650 y Linsys WPC11, pero puede que con otros modelos el controlador no se comporte igual.

Si tienes una tarjeta inalámbrica basada en el Prism2 y no consigues hacerla funcionar con este controlador, hazme saber de qué modelo se trata, y quizá pueda encontrar una solución al problema. Esto sería más sencillo si tuviese acceso al modelo específico de tarjeta, pero quizá pueda resolver algún caso basándome en registros (*logs*) de depuración y otra información fácilmente recolectable.

Además de con el Prism2, este controlador puede funcionar parcialmente con otros integrados. Por lo que sé, otros circuitos integrados no tienen modo Host AP, así que en ellos no se contempla esta funcionalidad, pero el modo *management* (por ejemplo, ser una estación en un BSS) parece funcionar en el Prism1 (tarjetas Intel, 3Com y Symbol), y con las tarjetas Lucent WaveLAN. Me gustaría añadir funcionalidad para modo AP en estas tarjetas, pero no estoy seguro de si se puede hacer sin firmware especial. Si tienes información a este respecto, por favor ponte en contacto conmigo.

Este controlador debería poder portarse a otras arquitecturas. Se ha probado en plataformas ix86 y PowerPC; para otras plataformas, probablemente se necesiten algunos cambios, al menos en lo referente a macros de E/S.

4.2. Software

Este controlador ha sido desarrollado para Linux 2.4.x, pero puede que funcione con los núcleos 2.2.x (lo he probado unas cuantas veces en núcleos 2.2.x con éxito, pero ya me he desacostumbrado a estas versiones del núcleo, así que quizá pueda darse algún problema de compatibilidad).

5. Instalación y configuración

5.1. Sistemas con módulos PCMCIA incluidos en el núcleo

El archivo 'Makefile' incluye comandos para compilar e instalar el controlador Host AP en sistemas que utilizan el PCMCIA incluido en el núcleo (es decir, los módulos no se compilan con el paquete externo pcmcia-cs). Edite la variable KERNEL_PATH en el Makefile para adaptarla a su sistema. Ejecute 'make' (esto compila las fuentes; este paso puede ejecutarse por un usuario que no sea root). Cuando termine la compilación, puede instalar el módulo prism2.o ejecutando 'make install' (como root). Después tendrá que reiniciar cardmgr para que se cargue la nueva configuración de /etc/pcmcia/prism2.conf.

5.2. Sistemas con módulos pcmcia-cs externos

Copie el contenido del subdirectorio 'driver' encima del directorio raíz de pcmcia-cs, de manera que driver/modules/prism2.c termine en el directorio 'modules' del paquete pcmcia-cs:

```
cp -a driver/* /usr/src/pcmcia-cs-3.1.31
```

Después, debería ser capaz de compilar e instalar pcmcia-cs con el controlador prism2.o utilizando los comandos habituales ('make config', 'make all', and 'make install'). pcmcia-cs necesita saber qué controlador usar para cada tarjeta. Esta distribución incluye un fichero propio (/etc/pcmcia/prism2.conf) que enumera algunas de las tarjetas que funcionan con el controlador, aunque la lista no es exhaustiva, y a lo mejor hace falta que añada su modelo en el archivo prism2.conf para poder iniciar el controlador.

El controlador funciona con las Linux Wireless Extensions, y algunos parámetros de configuración pueden verse y modificarse con iwconfig(8) y con iwpriv(8) de las utilidades wireless, por ejemplo, el modo (AP/station; iwconfig 'mode'), el canal (iwconfig 'freq' o 'channel'), WEP (encryption; iwconfig 'key'; sin embargo, el controlador no puede activar WEP en modo Host AP), intervalo de balizas (iwpriv 'beacon_int'), y modo normal IBSS (ad-hoc) vs. modo pseudo IBSS (iwpriv 'pseudo_ibss').

6. Bridging entre redes guiadas (con cables) e inalámbricas.

El controlador Prism2 AP maneja el dispositivo inalámbrico (wlan#), pero no se ocupa de hacer *bridging* entre la red inalámbrica y una red guiada. Si se requiere esta funcionalidad del AP, deberá implementarse con *software* externo.

Los núcleos de Linux ofrecen *Ethernet bridging* (el código se encuentra presente en las versiones 2.4.x, y también en forma de parche aparte para los núcleos 2.2.x). Esta funcionalidad puede utilizarse para construir fácilmente un AP entre la red inalámbrica y la guiada. A continuación hay un ejemplo sencillo de configuración en el que el AP tiene un dispositivo Ethernet y un dispositivo inalámbrico. En caso necesario, consulte <http://bridge.sourceforge.net> para más información.

Ejemplo 1. Configuración del *bridge* entre una red inalámbrica y una red Ethernet

Tras cargar el controlador Prism2 AP:

```
brctl addbr br0
brctl addif br0 eth0
brctl addif br0 wlan0
ifconfig eth0 0.0.0.0
ifconfig wlan0 0.0.0.0
ifconfig br0 192.168.100.200 up
```

el AP pasa paquetes entre la red Ethernet y la red inalámbrica, y puede verse desde cualquiera de estas con la dirección IP 192.168.100.200; las estaciones asociadas a este AP pueden comunicarse entonces con los ordenadores en la red guiada).

7. Monitorización IEEE 802.11

Las tarjetas Prism2 poseen un modo de prueba que puede ser utilizado para monitorizar redes inalámbricas. En este modo, el controlador recibe todas las tramas IEEE 802.11 (incluyendo las tramas de gestión). La monitorización puede iniciarse utilizando iwpriv(8):

```
iwpriv wlan0 monitor 1
```

comienza el modo monitorización y envía las tramas recibidas (incluyendo estructura RX de Prism2 y cabecera 802.11) a espacio de usuario utilizando un *netlink socket*.

```
iwpriv wlan0 monitor 2
```

comienza el modo monitorización y envía las tramas recibidas (incluyendo la cabecera 802.11) a espacio de usuario utilizando un dispositivo de red normal. Esto cambia el tipo de dispositivo a ARPHRD_IEEE80211, de manera que los programas en espacio de usuario sepan como manejar un tipo de cabecera diferente.

```
iwpriv wlan0 monitor 0
```

desconecta el modo monitorización y vuelve a modo Host AP.

Los programas de ejemplo en el directorio 'sniff' pueden utilizarse para recibir tramas monitorizadas enviadas a través de un socket (modo monitor 1 en el ejemplo anterior). `wlansniff.c` es un sencillo parser de tramas 802.11 que muestra datos específicos de las tramas RX de Prism2 y los contenidos analizados de la trama 802.11. `prism2ethereal` es una utilidad que escribe las tramas en un fichero que puede ser leído por Ethereal (por ejemplo) para un análisis más detallado.

Las últimas versiones de [libpcap](#) y [Ethereal](#) entienden ARPHDR_IEEE80211, de manera que el modo monitor 2 del ejemplo anterior también puede ser utilizado para monitorización IEEE 802.11 en tiempo real.

[Jouni Malinen](#)

Traducido al castellano por [Paul Salazar Mora](#).
[MadridWireless](#)