

Driver de Red para SODIUM

Federico Zeballos, Diego Pedro, Pablo Sánchez, Walter E. Pirraglia

Universidad Nacional de la Matanza,
Departamento de Ingeniería e Investigaciones Tecnológicas,
Sistemas Operativos Plan 625

Resumen: El documento expuesto a continuación tiene por objetivo ilustrar el desarrollo de un driver de placa de red para el sistema operativo didáctico de la Universidad Nacional de la Matanza, también conocido como SODIUM. Se explicará a continuación los métodos de comunicación y protocolos utilizados por las placas de red en general, la integración de las mismas con el kernel sobre el cual corren, así como también algunas de las distintas versiones de las normas más habituales en los sistemas operativos actuales.

Palabras Clave: IEEE, 802, 802.11, 802.3, network, OSI, Driver, Ethernet

1.- Introducción

La invención de la computadora trajo consigo grandes avances para el ser humano en sus comienzos, haciendo más fácil las tareas repetitivas que se presentasen en la vida cotidiana. Pero a medida que sus usos aumentaban, la necesidad de disponibilidad de los datos que se procesaban en un equipo lo hacía en la misma medida, ya sea porque se dependiera de ellos para la toma de alguna decisión, o que esos mismos datos fueran la entrada de uno o más procesos. Dos computadoras podían estar incluso en la misma sala, pero la disponibilidad de los datos se relegaba a que el operador los llevara de un lado a otro, además de que pronto se noto que los equipos, que en ese entonces eran costosos, pasaban la mayor parte de su tiempo sin realizar tarea alguna, siendo que en otros lugares no se conseguía satisfacer las necesidades de los usuarios relacionadas con los recursos del sistema. Con la aparición de los microcomputadores, la información comenzó a descentralizarse, y el procesamiento de datos era tratado en cada una de las estaciones de trabajo, perdiéndose el control de la información, que ahora era transportada en otros medios hacia los distintos puntos donde era solicitada. La centralización de la información, así como también el almacenamiento de la misma y el aprovechamiento de los recursos que se encontraran en lugares remotos, hizo necesario la disponibilidad de un medio confiable y de alta velocidad que permitiera aprovechar todo el potencial que las nuevas tecnologías presentaban. En este ámbito, surge la idea de de la interconexión de las estaciones de trabajo y las redes locales.

En un principio las redes estaban formadas por conexiones simples que solo intercomunicaban a las estaciones de trabajo, permitiendo a los usuarios el acceso a recursos no locales como archivos, impresoras, etc. Dado que el acceso no era controlado, cualquier usuario podía acceder a todos los recursos, editarlos, sin ninguna clase de seguridad involucrada, causando distinta clase de problemas de integridad de la información. Asimismo, el medio de intercomunicación utilizado se basaba en un solo cable coaxial que interconectaba cada uno de los sistemas, lo cual presentaba un gran problema tanto en la fiabilidad de los datos, los cuales podían sufrir corrupciones en interconexiones de grandes distancias, así como también en la seguridad de la conexión, la cual dependía de un solo cable expuesto a cualquier tipo de accidentes, cortando la conexión entre todos los equipos. Las tecnologías avanzaron, así como también la proposición de medios para la conexión de redes de computadoras, surgiendo varios modelos en los años siguientes.

Pero una nueva necesidad apunto a la especificación de las tecnologías, los medios utilizados para la interconexión de los equipos, y para todo el hardware involucrado en el proceso: la estandarización de las redes de computadoras. Este aspecto permitió no solo que los fabricantes de software y hardware pudieran

desarrollar sus procesos productivos de una manera mas enfocada y eficiente, sino también que esto último permitió la elaboración del hardware en forma que abarcara un publico mas general, el abaratamiento de los costos, y el establecimiento de las redes fuera del ámbito de aquellas compañías lo suficientemente poderosas como para afrontar los costos.

Las redes han evolucionado a través del tiempo, convirtiéndose en un aspecto fundamental de la computación en si, así como también de todas aquellas áreas en las que se usan computadoras como herramientas, ya no solo interconectando equipos en ciudades, sino a través del mundo entero. Obviamente, los adaptadores de redes están fuertemente ligados a este crecimiento, adaptándose y cubriendo las distintas formas en que se pueden establecer las distintas redes, dado que son éstos dispositivos los encargados de llevar a cabo la comunicación física entre las computadoras. Sin embargo es importante destacar que, a pesar de este crecimiento tecnológico y los distintos avances dados en el campo, las normas que regulan las redes si bien han sido adaptadas para cubrir estos aspectos, nunca han sido alteradas de manera importante. Estas normas fueron reguladas en un principio por el Instituto de Ingenieros Eléctricos y Electrónicos, en su norma 802 y diferentes apartados, contemplando las especificaciones de hardware utilizadas para los diferentes tipos de adaptadores que han surgido a través de los años, así como también de los medios utilizados por ellos. Sin embargo, es el apartado creado por la Organización de Estándares Internacionales, conocido como modelo OSI, aquel que todos los fabricantes de hardware y software deberán utilizar como guía para la elaboración de sus productos, y es el que se tendrá en cuenta en el desarrollo del presente documento, dado el marco que representa en la interacción entre los adaptadores, y el software encargado del control de los mismos, así como también del intercambio de información.

1.1- El modelo OSI

El modelo OSI (de sus siglas en ingles, Open System Inteconnection) es un modelo de normativa desarrollada por la Organización de Estándares Internacionales (de sus siglas en ingles ISO) como marco de reglas para la transferencia de datos entre dispositivos. Esta formada por siete capas definiendo las etapas por las cuales deberá pasar la información para poder ir de un dispositivo a otro sobre un medio o red de comunicaciones. Como puede observarse en la figura 1, las distintas capas que componen al modelo OSI son las siguientes



Figura 1 – Capas del Modelo ISO OSI

- ◆ **Capa Física:**
Encargada de las conexiones del dispositivo tanto en lo que se refiere a medio físico como a la transmisión de la información, definiendo el medio físico sobre el cual viajará la comunicación, las características materiales y eléctricas usadas en dicho medio, así como también el establecimiento, mantenimiento y liberación del enlace, garantizando la conexión.
- ◆ **Capa de enlace de datos:**
Encargada de la regulación de la interconexión de los equipos, el acceso al medio de comunicación, la detección de errores sobre la información enviada/recibida y el control del flujo de información.
- ◆ **Capa de Red:**
Encargada de la identificación del enrutamiento que seguirá la información entre una y más redes, haciendo que la información (encapsulada en paquetes) viaje desde el origen al destino.
- ◆ **Capa de Transporte:**
Independientemente del medio físico utilizado, esta capa es la encargada del transporte de la información en si.
- ◆ **Capa de Sesión:**
Encargada del enlace entre los extremos (origen y destino) manteniendo el enlace y restableciéndolo en caso de ser necesario, hasta la finalización de la transmisión.
- ◆ **Capa de presentación:**
Encargada de la representación de la información de manera reconocible, independientemente del conjunto de caracteres que usen los extremos.
- ◆ **Capa de aplicación:**
Encargada del ocultamiento de la complejidad de los servicios de las capas anteriores, permitiendo la definición de protocolos de interacción para el intercambio de datos.

El esquema del estándar OSI ha sido el pilar fundamental sobre el cual se ha basado la construcción de diferentes protocolos de comunicaciones, y es tomado en cuenta al momento de la elaboración tanto de hardware como software que tomará parte en la construcción de redes de computadoras. Sin embargo, la estructuración rígida que presenta este modelo hace que si bien sirva de guía, se deje en segundo plano de manera de implementar mayor flexibilidad en la creación y uso de nuevos protocolos de comunicación.

A los efectos prácticos de este documento, dada la amplia interacción que se tendrá con los niveles más bajos del modelo, se presenta esta información en forma de carácter ilustrativo, teniéndola siempre presente en la elaboración del software necesario para controlar la placa física dentro del sistema operativo indicado. Sin embargo es de remarcar que los adaptadores de red, sus aspectos físicos así como también los datos necesarios para el desarrollo de controladores de los mismos, se encuentran mayormente contemplados en los estándares creados por el Instituto de Ingenieros Eléctricos y Electrónicos, y es en ellos donde los fabricantes y desarrolladores podrán encontrar los detalles específicos correspondientes a cierto adaptador, diferenciados cada uno en los distintos apartados, según la tecnología que utiliza, lo cual se podrá observar a continuación, y comprende el marco específico sobre el cual se basará el desarrollo por este documento contemplado.

1.2 – El estándar IEEE802

Para el año 1980, el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) decidió estandarizar las redes de computadoras bajo la norma 802, abarcando desde el nivel físico, el nivel de enlace y superiores, además de hacer la distinción en el nivel de enlace entre lógico y el de acceso al medio.

La familia de estándares IEEE 802¹ abarcan todo lo concerniente a lo que redes de computadoras se refiere, dividiéndose en diferentes apartados según la tecnología del adaptador y el medio por estos utilizados. Algunas de ellos han visto su uso discontinuado siendo reemplazados por otros cuyas ventajas eran un

¹ Los diferentes apartados de la norma IEEE802 que comprenden las diferentes tecnologías se encuentran explyados en el anexo del documento.

punto sobresaliente, sin embargo, otros de ellos, a pesar de haber sido inventados ya hace tiempo, todavía son parte importante del uso cotidiano en las redes.

El conjunto de reglas establecidos para las redes que se encuentren contempladas bajo las normas mencionadas anteriormente se dedican a regular en particular la funcionalidad de las dos últimas capas del modelo OSI, así como también abarcar algunos aspectos de las capas superiores que entraran en contacto con la administración de la red del apartado en cuestión. Debido a la extensión de cada una de ellas, y las consignas del documento aquí elaborado, se enfocará la atención en los apartados IEEE 802.3 – Ethernet y el IEEE 802.11 – Redes inalámbricas WLAN, haciendo mención a los aspectos más importantes de estas, que serán tenidos en cuenta al momento del desarrollo tanto del documento como el software.

1.3 – Norma IEEE 802.3

Este estándar define el protocolo Ethernet utilizado para redes LAN y MAN. En particular, Ethernet utiliza la especificación de acceso común al medio de comunicación (del inglés MAC), con el protocolo CSMA/CD (Carrier Sense Multiple Access with Collision Detection), en dos velocidades: half y full duplex. De esta manera, la capa física codifica y decodifica los frames o paquetes de información para la transmisión o recepción de los mismos, según sea el caso, y de acuerdo a la velocidad correspondiente.

El estándar IEEE802.3 fue, desde su comienzo, el intento por regular el método de interconexión Ethernet, y ha sufrido varias ampliaciones para reflejar los distintos avances en la velocidad y cambios en la tecnología utilizada, así como también los cambios en el medio de transmisión². En este tipo de protocolo, se utiliza una trama de acuerdo a lo siguiente:

Sincronización	Delimitador Comienzo	Dirección Destino	Dirección Origen	Longitud/Tipo	Datos	FRC
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46-1500 bytes	4 bytes

De donde se desprende que:

Sincronización: intervalo de bits de sincronización de la línea

Delimitador Comienzo: intervalo de bits que indica el comienzo de la trama

Dirección Destino: 3 bytes corresponden al identificador de organización, y los siguientes 3 son utilizados para la dirección MAC hacia donde se dirige el frame

Dirección Origen: ídem anterior

Longitud: Longitud de la trama

Datos: los datos a transmitir

FRC: Secuencia de chequeo de la trama

Luego de la trama, se suele dejar un interface gap, o vacío de 12bytes, correspondiente a la espera que se realiza antes de transmitir nuevamente, aun si el medio estaba libre.

Este protocolo se deberá tener en cuenta al momento de desarrollar software encargado del envío y recepción de paquetes de información en una red de computadoras, dado que refleja en su mayor parte el método actual de transmisión que soportan y sobre el cual se han guiado la mayoría de los fabricantes de adaptadores de red.

1.4 – Norma IEEE 802.11

Esta norma corresponde a la regulación de las transmisiones inalámbricas utilizadas para las redes de computadoras, ya sea en forma de red multidestino, o punto a punto. Esta norma también ha sufrido varias revisiones, correspondiéndose a los adelantos tecnológicos hechos en el campo y reflejando las distintas velocidades que se han alcanzado con ellos².

² Los diferentes subapartados que corresponden a cada uno de los adaptadores desarrollados bajo éste pueden ser observados en el anexo 2 del documento

La información presentada por esta norma permite conocer el método de intercomunicación en comunicaciones sin cableado, tan ampliamente difundidas actualmente, y la cual ha sido tenido en cuenta al momento del desarrollo del software perseguido por el documento, dado el fin buscado.

2.- Desarrollo

Se desarrollará un controlador de adaptador de red tratando de ser lo mas amplio posible, de manera de incluir la mayor cantidad de adaptadores dentro del software a desarrollar. El principal objetivo de esto es no limitar el sistema operativo SODIUM al uso de un adaptador específico, y brindarle la mayor funcionalidad posible.

La intercomunicación de computadoras esta enmarcada en el estándar IEEE 802, el cual si bien no refleja la realidad en forma exacta, los fabricantes de hardware utilizado para la intercomunicación e implementación de redes suelen utilizarlo como método guía y no desviarse mas allá de ella.

Las continuas revisiones del estándar IEEE 802 en lo que se refiere tanto a redes cableadas como redes inalámbricas (se encuentra bajo revisión la sección IEEE 802.11 al momento de realizar este documento) hace que sea necesario la confinación del software a realizar a algunas de las versiones. Esto sumado a las herramientas disponibles y con las que se contaron al momento del desarrollo del controlador, y la amplia difusión de ciertos métodos de interconexión hizo que los desarrolladores de este documento limitaran las opciones a solo dos: IEEE 802.3, conocida también como Ethernet, y la tecnología IEEE 802.11 o conocida también como interconexión inalámbrica. Dada la revisión actualmente de la norma IEEE 802.11, lo que podría suponer un cambio radical o no en la tecnología inalámbrica, incluyendo el método de transmisión y los protocolos utilizados en dicha tecnología, se opto por enmarcar la investigación basado en las herramientas disponibles tanto en forma virtual como física, y desarrollar un controlador de dispositivo orientado a conexiones cableadas, ampliamente difundidas en todo el ámbito, de fácil acceso y menor costo.

Dado que el desarrollo de un controlador genérico implicaría el reconocimiento de dispositivos de diferentes fabricantes, los cuales no siempre comparten el mismo método de funcionamiento, y tratando de evitar errores al momento del reconocimiento de los dispositivos físicos, se decidió confinar el desarrollo a la familia de dispositivos basados en el chipset Intel e1000, dispositivo reconocido por las diferentes herramientas de simulación de software con las cuales se trabajarán. Tanto los emuladores Qemu como Bosch y VMware permiten la especificación de dispositivos para la interconexión de red de esta familia, lo que amplia el rango de dispositivos abarcados por la investigación

El dispositivo de red deberá ser capaz de ser registrado a nivel de Kernel por el sistema operativo SODIUM, así como también poder establecer algún comando básico de intercomunicación con otro dispositivo de la misma familia ubicado en un equipo receptor, haciendo uso de las capacidades del dispositivo en si. A tales efectos se utilizará como guía las especificaciones descriptas por las normas IEEE 802.3 descriptas anteriormente, y desarrolladas en la introducción de este documento.

Actualmente los dispositivos de red, sean integrados o no, se encuentran conectados por medio del puerto PCI, por lo que un entendimiento básico de tal puerto será tenido en cuenta y posteriormente desarrollado para ser añadido en el documento.

2.1- Elaboración

Desde el inicio del sistema, los dispositivos que se encuentren conectados al mismo no están accesibles por el sistema operativo, sino que deberán ser detectados e inicializados previamente para poder ser habilitados. En el caso específico de los dispositivos de red abarcados por este documento, dado que los mismos se conectan al sistema mediante el bus PCI, es el controlador de este bus el que brindará los servicios que serán utilizados para la inicialización de los mismos.

Como se dijo anteriormente, los dispositivos de red se encuentran inicialmente no accesibles, y se podría decir que para el sistema operativo los mismos todavía no existen en el equipo. Así, al inicio del sistema, es el controlador de puertos PCI el encargado de realizar ciertas operaciones con con los registros del bus PCI, de manera de obtener una dirección específica de 32 bits, la cual será utilizada para acceder al bus del dispositivo específico buscado. Esta dirección es denominada CONFIG_ADDRESS, y es la que permite, mediante los valores que tienen ciertas porciones de la misma, la identificación del dispositivo que se

encuentra en el bus conectado, y la especificación del mismo. La estructura de la dirección CONFIG_ADDRESS es la siguiente:

31	30 – 24	23 – 16	15 – 11	10 – 8	7 – 2	1 – 0
Bit Habilitado	Reservado	Numero del Bus	Numero de Dispositivo	Numero de Función	Numero de Registro	00

de la cual se desprende que:

- ◆ BIT 31: bandera de habilitación para determinar cuando acceder al registro CONFIG_DATA.
- ◆ BIT 23-16: permite al software de configuración, elegir a un bus PCI específico en el sistema.
- ◆ BIT 15-11: Selecciona un dispositivo específico en el bus PCI.
- ◆ BIT 10-8: Elige una función específica dentro de un dispositivo (solo si el dispositivo soporta múltiples funciones)
- ◆ BIT 7-2: selecciona el área de 32-bits específica en el espacio de configuración del dispositivo.

Mediante el acceso por esta dirección al bus del dispositivo, el controlador de puertos PCI permitirá leer una estructura de 256 bytes, la cual contiene varios de los datos que el controlador del dispositivo necesita para iniciar una interacción con el mismo, de manera de poder escribir en sus registros los valores necesarios para la inicialización. Cabe destacar que esta estructura no es específica de los dispositivos de red, sino que sea cual fuere la función que cumple el dispositivo, la estructura de registros será la misma, aunque variará el contenido o valores de los registros en sí. La estructura mencionada, esta dividida de acuerdo a lo siguiente:

BYTE OFFSET	BYTE 3	BYTE 2	BYTE 1	BYTE 0
0h	Device ID		Vendor ID	
4h	Status Register		Command Register	
8h	Class Code (020000h)			Revision ID
Ch	BIST (00h)	Header Type (00h)	Latency Timer	Cache Line Size
10h	Base Address 0			
14h	Base Address 1			
18h	Base Address 2			
1Ch	Base Address 3 (unused)			
20h	Base Address 4 (unused)			
24h	Base Address 5 (unused)			
28h	Cardbus CIS Pointer (not used)			
2Ch	Subsystem ID		Subsystem Vendor ID	
30h	Expansion ROM Adress			
34h	Reserved		Cap_Ptr	
38h	Reserved			
3Ch	Max_Latency (00h)	Min_Grant (FFh)	Interrupt Pin (01h)	Interrupt Line

La lectura de los registros presentados anteriormente, permitirá que el dispositivo, sea cual fuere su función, sea identificado mediante dos valores: su deviceid y su vendorid, es decir, mediante el identificador único asignado a cada fabricante de dispositivos, y el identificador de dispositivo, que especifica el modelo del mismo dentro de la familia elaborada por el fabricante detectado. La detección y especificación del dispositivo es esencial para la inicialización del mismo, dado que esta identificación permitirá que el controlador correcto sea invocado al momento de habilitar el dispositivo.

Una vez identificado, la interacción con los registros de dirección base (Base Address Register) permitirá al controlador del dispositivo, setear los valores predeterminados para poder inicializarlo, y de esta manera sea “detectado” por el sistema operativo. Sin embargo, este es solo el primer paso en la habilitación del mismo, dado que la escritura en los registros correspondientes solo la dejará en un estado “DOWN” o abajo, no pudiendo ser configurada mediante el sistema operativo para poder interactuar con las redes o

sistemas externos. Para poder habilitar el dispositivo, y que el mismo sea reconocido por el sistema operativo, es necesario setear el calor correspondiente, especificado según el manual del dispositivo, en el registro de control del mismo. El dispositivo, habiendo detectado la escritura en dicho registro, actuará en consecuencia, habilitando la interacción con el mismo, para luego realizar las tareas correspondientes de configuración de las capas mas elevadas del modelo OSI.

3.- Problemas encontrados

Configuración del Entorno de desarrollo

Si bien la configuración de una maquina virtual que incluya una placa de red de la familia e1000 es relativamente sencilla, el principal inconveniente se presentó al momento de hacer que esa maquina se comunicase con los equipos que están presentes en la red. Para alcanzar tal comportamiento fue necesario proveer a la maquina virtual, de un stack TCP/IP completo, lo cual se logró a través de una interfaz tap, la cual trabaja en la capa 2 del modelo OSI. Esta interfaz es más complicada de configurar, pero nos provee de todas las funcionalidades necesarias para hacer las pruebas que requiera el desarrollo del driver de placa de red. Es necesario aclarar que las interfaces bridge no funcionan con conexiones WI-FI, por lo tanto solo se podrán realizar las pruebas necesarias cuando el equipo se encuentre conectado a redes Ethernet IEEE 802.3 (redes cableadas). Se editaron los archivos /etc/modules y /etc/network/interfaces del SDK provisto para proveer al entorno de las interfaces anteriormente descriptas. La configuración de las interfaces dentro del archivo /etc/network/interfaces es muy importante, puesto que el administrador de redes (NetworkManager) del SDK no administra las interfaces que se encuentran configuradas en dicho archivo. Es necesario evitar que NetworkManager administre estas interfaces de red, porque no administra de forma correcta las interfaces del tipo bridge, y cada vez que realizamos un conexión entre algún equipo de la red y la maquina virtualizada con QEMU se perdía trafico de red. Se deben agregar dos scripts en el directorio herramientas de SODIUM (./herramientas), estos scripts serán invocados por QEMU al momento de iniciar y finalizar la maquina virtual, y su único objetivo es levantar y bajar respectivamente la interfaz “tap” creada a tal efecto. Por último se debe invocar a QEMU, con los parámetros correspondientes para que levante la interfaz tap, en el bridge creado anteriormente.

Configuración del Entorno con Bosch

Después de investigar la configuración del emulador de x86 Bochs, se determina que la configuración de soporte de plugins (Hardware) que el mismo emula se debe realizar antes de la compilación. El comando habilita todo lo necesario para después realizar la compilación. Como se lo realizo en un principio se creyó que el “make install” necesario para la compilación, se encargaba de modificar la copia del binario que poseía el sistema. Como esto no se daba así, siempre se terminaba ejecutando el binario original compilado sin las opciones anteriores. La solución a este problema se dio en principio modificando los binarios viejos del sistema, y colocando los nuevos compilados recientemente.

Al no haber podido configurar la placa NE2000 y antes de encontrar el problema citado en el párrafo anterior, se decidió cambiar por la familia de placas Intel e1000. Esto conlleva también una mejora en material de información e implementaciones. Posterior a esta decisión se intento volver a configurar el bochs para la placa e1000. Es en este momento se incurre en el problema de que la ultima versión disponible de bochs (2.5.1), con la que cuenta el entorno de desarrollo, no cuenta con un plugin para la misma. Tras cierta investigacion se llego a la conclusion de que si bien lo anterior es así, tenemos la posibilidad de descargar del SVN del proyecto Bochs, una versión aun más actual que todavía no ha sido versionada pero que si cuenta con el plugin necesario, por lo que se volvio a reconfigurar el emulador. Para configurar la ejecución de Bochs lo que se realizó es ejecutar el emulador sin parámetros. Dentro del menú se configuró los discos, ya sean duros o cds, floppys y mouse y/o teclados. Se encontró que la manera mas fácil de crear un disco es utilizando la herramienta de Bochs llamada “bximage”. Se puede recurrir también a discos ya instalados, con versiones de S.O. por lo general libres, en la web del proyecto Bochs. Después de haber configurado el entorno de ejecución, fue útil guardar todo en un archivo de configuración, ya que más adelante se nos permitía tocar directamente el mismo en formato texto. Mas allá de lo que hemos hecho en la configuración anterior, si se saltea alguno de los pasos, se debía configurar la placa que nos interesa en el archivo.

Especificación del desarrollo

Al comenzar con la investigación, se trato de enfocar el tema lo mas general posible para así no limitar el desarrollo del software y el sistema operativo SODIUM a un hardware específico. Sin embargo, las herramientas con las que se comenzó el desarrollo y fueron brindadas a tales efectos, sumado al posible cambio de algunas normas, hizo que se especificara mas y mas el tema, hasta limitarlo a una familia de dispositivos, mas que el desarrollo de un driver genérico que abarcara tanto las redes cableadas como inalámbricas.

4.- Impacto del desarrollo

A los efectos del desarrollo de este documento y el software, se han modificado los siguientes archivos:

```
/etc/modules  
/etc/network/interfaces  
/include/kernel/drivers/pci.h  
/kernel/drivers/pci.c  
/include/kernel/drivers/red.h  
/kernel/drivers/red.c
```

5.- Bibliografía

Juan Carlos Bordachar, Lucas Hernan Enriquez, Rodrigo Medina Maciel, “Host USB” 11/2011

<http://standards.ieee.org/about/get/802/802.html>

Andrew S. Tanenbaum, “Redes de Computadoras” 2003, Cuarta edición

Alessandro Rubini, Jonathan Corbet, “Linux Device Drivers”, 2001, Segunda Edición

<http://linuxgazette.net/issue93/bhaskaran.html>

<http://sourceforge.net/projects/e1000/>

<http://www.spinics.net/lists/netdev/msg168340.html>

<http://blog.vodkamelone.de/archives/146-Unbricking-an-Intel-Pro1000-e1000-network-interface.html>

<http://www.kernel.org/doc/Documentation/networking/e1000.txt>

<http://www.redbudcomputer.com/howtos/spam/e1000-nic.txt>

Jordi Iñigo Griera, José María Barceló Orinas, Llorenc Cerdà Alabern, Eric Peig Olivé, Jaume Abella I Fuentes, Guiomar Corral I Torruella, “Estructura de Redes de Computadores”, 2008, Primera Edición

PCI Local Bus Specification - Revision 2.2 - December 18, 1998

PCI/PCI-X Family of Gigabit Ethernet - Controllers Software Developer’s Manual

Tom Shanley and Don Anderson, “PCI System Architecture”, 1999, Cuarta Edición

Intel® 64 and IA-32 Architectures Software Developer’s Manual, Combined Volumes: 1, 2A, 2B, 2C, 3A, 3B and 3C

Anexo

1 – Familia de Estándares IEEE802

La familia de estándares IEEE 02 abarcan todo lo concerniente a lo que redes de computadoras se refiere, dividiéndose en los siguientes apartados:

- IEEE 802.1 – Normalización de interfaz.
- IEEE 802.1D – Spanning Tree Protocol
- IEEE 802.1Q – Virtual Local Area Networks (VLAN)
- IEEE 802.1aq - Shortest Path Bridging (SPB)
- IEEE 802.2 – Control de enlace lógico.
- IEEE 802.3 – CSMA / CD (ETHERNET)
- IEEE 802.4 – Token bus.
- IEEE 802.5 – Token ring.
- IEEE 802.6 – Metropolitan Area Network (ciudad) (fibra óptica)
- IEEE 802.7 – Grupo Asesor en Banda ancha.
- IEEE 802.8 – Grupo Asesor en Fibras Ópticas.
- IEEE 802.9 – Voz y datos en LAN.
- IEEE 802.10 – Seguridad.
- IEEE 802.11 – Redes inalámbricas WLAN.
- IEEE 802.12 – Prioridad por demanda
- IEEE 802.13 – Se ha evitado su uso por superstición
- IEEE 802.14 – Modems de cable.
- IEEE 802.15 – WPAN (Bluetooth)
- IEEE 802.16 - Redes de acceso metropolitanas sin hilos de banda ancha (WIMAX)
- IEEE 802.17 – Anillo de paquete elástico.
- IEEE 802.18 – Grupo de Asesoría Técnica sobre Normativas de Radio.
- IEEE 802.19 – Grupo de Asesoría Técnica sobre Coexistencia.
- IEEE 802.20 – Mobile Broadband Wireless Access.
- IEEE 802.21 – Media Independent Handoff.
- IEEE 802.22 – Wireless Regional Area Network.

2.- Subapartados Concernientes a IEEE802.3

Lo siguiente describe cada uno de los subapartados de la norma correspondiente a las redes de computadoras cableadas o Ethernet:

- Ethernet experimental: 2,85 Mbit/s sobre cable coaxial en topología de bus.
- Ethernet II (DIX v2.0): 10 Mbit/s sobre coaxial fino (thinnet) - La trama tiene un campo de tipo de paquete. El protocolo IP usa este formato de trama sobre cualquier medio.
- IEEE 802.3 10BASE5 10 Mbit/s sobre coaxial grueso (thicknet). Longitud máxima del segmento 500 metros - Igual que DIX salvo que el campo de Tipo se substituye por la longitud.
- IEEE 802.3a: 10BASE2 10 Mbit/s sobre coaxial fino (thinnet o cheapernet). Longitud máxima del segmento 185 metros
- IEEE 802.3b: 10BROAD36
- IEEE 802.3c: Especificación de repetidores de 10 Mbit/s
- IEEE 802.3d: FOIRL (Fiber-Optic Inter-Repeater Link) enlace de fibra óptica entre repetidores.
- IEEE 802.3e: 1BASE5 o StarLAN
- IEEE 802.3i: 10BASE-T 10 Mbit/s sobre par trenzado no blindado (UTP). Longitud máxima del segmento 150 metros.
- IEEE 802.3j: 10BASE-F 10 Mbit/s sobre fibra óptica. Longitud máxima del segmento 1000 metros.

- IEEE 802.3u: 100BASE-TX, 100BASE-T4, 100BASE-FX Fast Ethernet a 100 Mbit/s con auto-negociación de velocidad.
 - IEEE 802.3x: Full Duplex (Transmisión y recepción simultáneos) y control de flujo.
 - IEEE 802.3y: 100BASE-T2 100 Mbit/s sobre par trenzado no blindado (UTP). Longitud máxima del segmento 100 metros
 - IEEE 802.3z: 1000BASE-X Ethernet de 1 Gbit/s sobre fibra óptica.
 - IEEE 802.3ab: 1000BASE-T Ethernet de 1 Gbit/s sobre par trenzado no blindado
 - IEEE 802.3ac: Extensión de la trama máxima a 1522 bytes (para permitir las "Q-tag") Las Q-tag incluyen información para 802.1Q VLAN y manejan prioridades según el estándar 802.1p.
 - IEEE 802.3ad: Agregación de enlaces paralelos.
 - IEEE 802.3ae: Ethernet a 10 Gbit/s ; 10GBASE-SR, 10GBASE-LR
 - IEEE 802.3af: Alimentación sobre Ethernet (PoE).
 - IEEE 802.3ah: Ethernet en la última milla.
 - IEEE 802.3ak: 10GBASE-CX4 Ethernet a 10 Gbit/s sobre cable bi-axial.
- IEEE 802.3an: 10GBASE-T Ethernet a 10 Gbit/s sobre par trenzado no blindado (UTP)

3.- Subapartados Concernientes a IEEE802.11

Lo siguiente describe cada uno de los subapartados de la norma correspondiente a las redes de computadoras no cableadas, o wireless

- IEEE 802.11: versión original del estándar especificando velocidades teóricas de entre 1 y 2 Mbit/s utilizando señales infrarrojas. También define el protocolo CSMA/CA como método de acceso y transmisión de la información
- IEEE 802.11a: utiliza el mismo set de protocolos que la original, operando en la banda de 5 Mhz. Posee 12 canales sin solapamiento, 8 para red inalámbrica y 4 para conexiones punto a punto. La velocidad ronda los 5Mbit/s
- IEEE 802.11b: velocidad máxima de transmisión de 11Mbit/s operando en la banda de 2,4Ghz
- IEEE 802.11c: utilizado para la interconexión de dos redes de diferente tipo
- IEEE 802.11d: Permite que distintos dispositivos intercambien información en rangos de frecuencia según lo que se permite en el país de origen del dispositivo
- IEEE 802.11e: añade características QoS y de soporte multimedia, a la vez que mantiene compatibilidad con los estándares anteriores
- IEEE 802.11f: Utiliza el protocolo IAPP que le permite a un usuario itinerante cambiarse claramente de un punto de acceso a otro mientras está en movimiento sin importar qué marcas de puntos de acceso se usan en la infraestructura de la red. También se conoce a esta propiedad simplemente como itinerancia
- IEEE 802.11g: utiliza la banda de 2,4 Ghz (al igual que el estándar 802.11b) pero opera a una velocidad teórica máxima de 54 Mbit/s, que en promedio es de 22,0 Mbit/s de velocidad real de transferencia, similar a la del estándar 802.11a. Es compatible con el estándar b y utiliza las mismas frecuencias. Buena parte del proceso de diseño del estándar lo tomó el hacer compatibles los dos estándares. Sin embargo, en redes bajo el estándar g la presencia de nodos bajo el estándar b reduce significativamente la velocidad de transmisión
- IEEE 802.11h: proporciona a las redes 802.11a la capacidad de gestionar dinámicamente tanto la frecuencia, como la potencia de transmisión.
- IEEE 802.11i: dirigido a batir la vulnerabilidad actual en la seguridad para protocolos de autenticación y de codificación. El estándar abarca los protocolos 802.1x, TKIP (Protocolo de Claves Integra – Seguras – Temporales), y AES (Estándar de Cifrado Avanzado). Se implementa en WPA2.
- IEEE 802.11j: equivalente al 802.11h, en la regulación Japonesa.
- IEEE 802.11k: permite a los conmutadores y puntos de acceso inalámbricos calcular y valorar los recursos de radiofrecuencia de los clientes de una red WLAN.
- IEEE 802.11n: puede trabajar en dos bandas de frecuencias: 2,4 GHz (la que emplean 802.11b y 802.11g) y 5 GHz (la que usa 802.11a). Gracias a ello es compatible con dispositivos basados en todas

las ediciones anteriores. Además permite alcanzar un mayor rendimiento, con velocidades teóricas de 600Mbit/s.

- IEEE 802.11p: Este estándar opera en el espectro de frecuencias de 5,90 GHz y de 6,20 GHz, especialmente indicado para automóviles. Será la base de las comunicaciones dedicadas de corto alcance (DSRC) en Norteamérica. La tecnología DSRC permitirá el intercambio de datos entre vehículos y entre automóviles e infraestructuras en carretera
- IEEE 802.11r: su principal característica es permitir a la red que establezca los protocolos de seguridad que identifican a un dispositivo en el nuevo punto de acceso antes de que abandone el actual y se pase a él. Permite que la transición entre nodos demore menos de 50 milisegundos. Un lapso de tiempo de esa magnitud es lo suficientemente corto como para mantener una comunicación vía VoIP sin que haya cortes perceptibles.