INSTALACIÓN, CONFIGURACIÓN Y ADMINISTRACIÓN

USERS

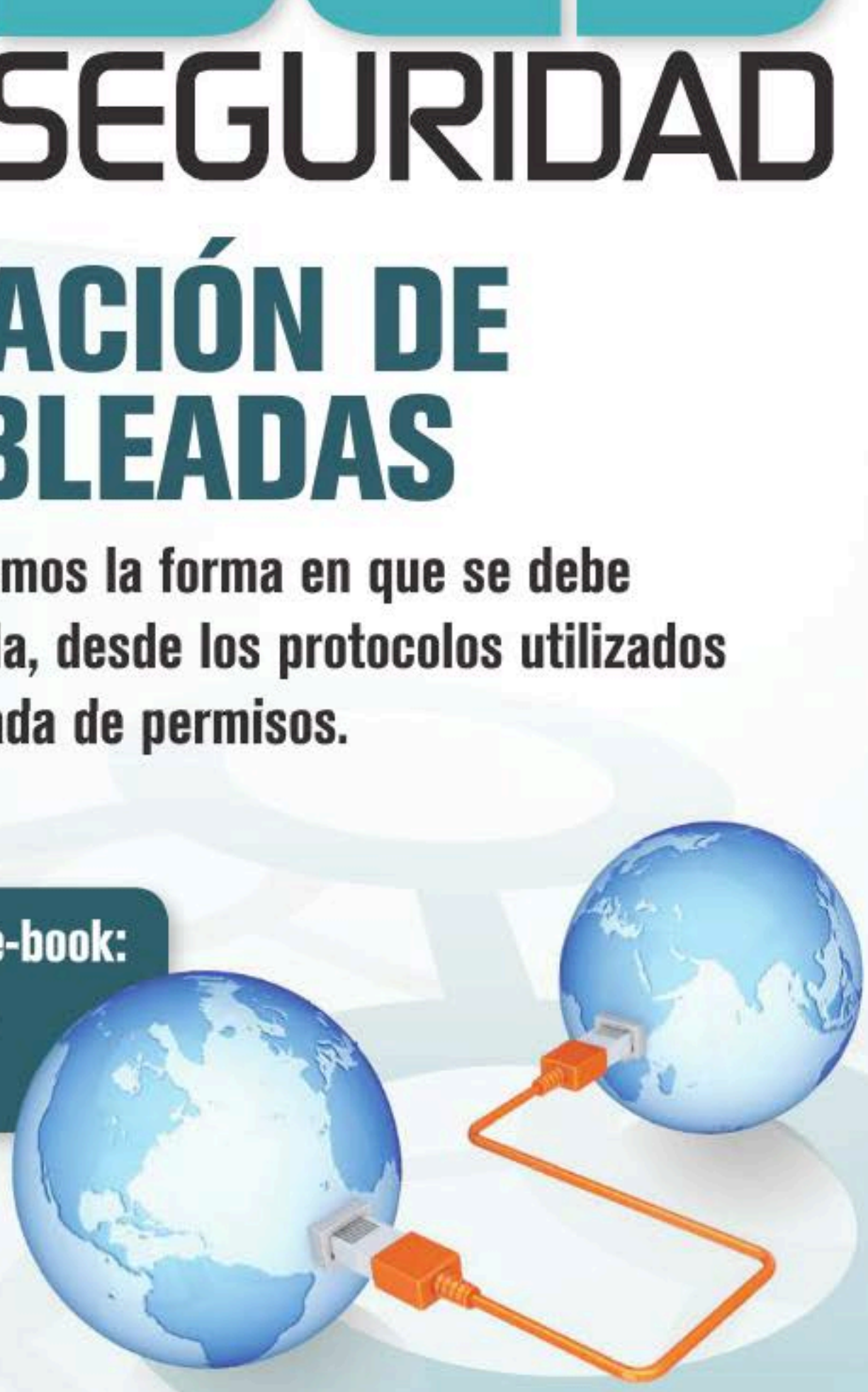
Argentina $ 22.- // México $ 49.-

Técnico en REDES &SEGURIDAD 6

CONFIGURACIÓN DE REDES CABLEADAS

En este fascículo conoceremos la forma en que se debe configurar una red cableada, desde los protocolos utilizados hasta la asignación adecuada de permisos.

Incluye e-book: Asterisk



USERS

Técnico en REDES &SEGURIDAD

Coordinador editorial
Paula Budris

Asesores técnicos
Federico Pacheco
Javier Richarte

Nuestros expertos
Valentín Almirón
José Bustos
Gustavo Cardelle
Rodrigo Chávez
Alejandro Gómez
Javier Medina
Gustavo Martín Moglie
Pablo Pagani
Gerardo Pedraza
Ezequiel Sánchez

Impreso en Sevagraf S.A. Impreso en Argentina.

Técnico en redes y seguridad / coordinado por Paula Budris. - 1a ed. - Buenos Aires: Fox Andina, 2013
576 p. ; 28 x 20 cm. (Users; 22)

ISBN 978-987-1857-78-4

1. Informática. 2. Redes. I. Budris, Paula, coord.
CDD 004.68

En esta clase veremos...

La configuración de redes cableadas, los protocolos y las tecnologías necesarias, así como también algunas consideraciones importantes sobre seguridad.

En la clase anterior, vimos la implementación completa de una red cableada, y analizamos la forma en que podemos solucionar los problemas más comunes. Efectuamos una introducción a la seguridad en los sistemas operativos de red y conocimos los distintos tipos de switch que existen en el mercado. También explicamos qué son los dominios y los puertos lógicos, y, para terminar, aprendimos a administrar las particiones en un disco duro y los alcances de NetBIOS. En esta entrega, nos dedicaremos a profundizar en las tareas de configuración relacionadas con las redes cableadas, revisaremos las tecnologías y los protocolos relacionados, así como también las opciones de seguridad que es necesario tener en cuenta. Aprenderemos a realizar la asignación de permisos y a hacer un booteo remoto. Para concluir, veremos en detalle aspectos de seguridad tan importantes como TCP Handstake.

6

2
El protocolo TCP/IP en profundidad

6
Tecnología Wake On LAN

14
Permisos en grupos de trabajo

20
Conocer los protocolos IP: IGMP e ICMP

El protocolo TCP/IP en profundidad

El conjunto de protocolos TCP/IP hace posible establecer todas las comunicaciones; sin embargo, su protagonismo se encuentra oculto.

Todos los medios de comunicación se encuentran estandarizados con normas y protocolos para optimizar su comprensión y funcionamiento. El conjunto de protocolos **TCP/IP** define la comunicación entre sí de los diferentes tipos de equipos tecnológicos que podemos adquirir. Su nombre está formado por la unión de los protocolos más conocidos: **TCP** (*Transmission Control Protocol*) e **IP** (*Internet Protocol*). El conjunto de protocolos que lo conforman nos permite enviar correos electrónicos, visualizar páginas web, mirar videos online, hacer streaming y acceder al escritorio remoto; en resumen, todas las actividades que podemos efectuar a través de la red, ya sea interna o externa.

Capas TCP/IP

En los protocolos que se dividen en capas, el conjunto de esas capas se denomina pila lógica (**stack**). Entonces, cuando nos referimos a que se agrega cabecera a la pila lógica, estamos diciendo que se agrega información adicional en cada capa del modelo. Al igual que el **modelo OSI**, la **arquitectura TCP/IP** se divide en cuatro capas, que podemos comparar con las de dicho modelo. Las datos que enviamos a través de la red van pasando por las capas del modelo TCP/IP, y a medida que lo hacen, cada una agrega un nuevo encabezado, a la vez que deja los datos preparados para que la capa de su nivel inferior interprete lo que está recibiendo (encabezado más datos originales), y pueda agregar su propio encabezado. Este proceso se conoce como **encapsulación**. Cuando los datos llegan a su destino, sucede el proceso inverso: cada capa lee el encabezado y lo retira, y deja los datos listos para ser enviados a la capa superior. A continuación, conoceremos los detalles de cada una de etas capas.

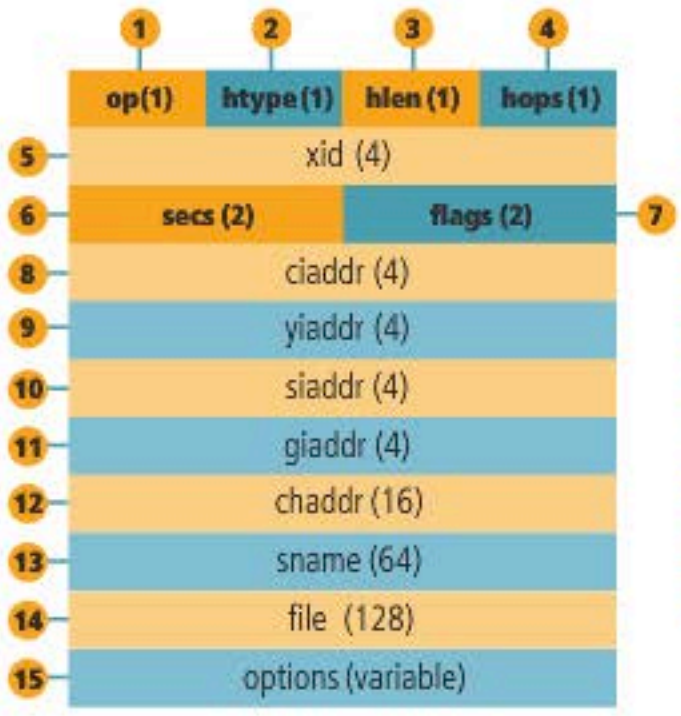
LOS ESTÁNDARES DE LOS PROTOCOLOS SE PUBLICAN EN LAS RFC (REQUEST FOR COMMENTS).

Aplicación

En esta capa se procesan los pedidos de **datos**, o servicios. No se está haciendo referencia a la aplicación de usuario (software) en sí misma, sino a todos aquellos protocolos que proveen de un servicio que interactúa con el software. Por ejemplo, en esta capa se encuentra el protocolo **POP3** (*Post Office Protocol versión 3*), utilizado para el correo entrante independientemente del software cliente de correo que tengamos. Otros protocolos que aparecen en esta capa son: **FTP**, **SNMP**, **IMAP**, etc. En esta capa también se llevan a cabo tareas de control para la comunicación. Se establece una comunicación entre los extremos para efectuar tareas de recuperación de errores en la transferencia e integridad de datos; es decir, se verifica que la

Socket, segmentos, paquetes y tramas

En el campo de las redes, muchas veces nos encontraremos con estos términos: socket **(cuando, además de incluir la dirección IP de destino, se incluye el puerto de destino, el conjunto de datos se denomina socket),** segmento **(cuando se reciben datos para ser enviados, TCP los divide en segmentos, a cada uno de los cuales le agrega la cabecera),** paquetes **(los segmentos anteriores son recibidos por el protocolo IP, que empaqueta los datos agregando las direcciones IP de origen y destino) y** tramas **(en la capa de enlace de datos, los paquetes IP se convierten en tramas Ethernet, para ser enviadas dentro de la LAN).**



1. op: Tipo de mensaje (1:BOOTREQUEST, 2:BOOTREPLY).
2. htype: Tipo de dirección de hardware (1:10 Mbits Ethernet).
3. hlen: Largo de la dirección de hardware (6 para Ethernet).
4. hops: Es utilizado por agentes relay cuando se arranca utilizando un relay.
5. xid: IP de transacción. Número aleatorio para asociar mensajes y respuestas.
6. secs: Utilizado por el cliente. Segundos desde que el cliente comenzó el proceso.
7. flags: Bandera.
8. ciaddr: Client IP Address (dirección IP del cliente). Utilizado si el cliente está en modo RENEW, BOUND o REBINDING.
9. yiaddr: Su (your) dirección IP.
10. siaddr: Es la dirección IP del siguiente servidor que se utilizará en el arranque. Devuelto por el servidor en mensajes DHCPOFFER y DHCPACK.
11. giaddr: Se refiere a la dirección IP del agente de relay. Utilizado cuando se arranca con un servidor relay.
12. chaddr: Dirección de hardware del cliente
13. sname: Nombre de host del servidor opcional. Cadena terminada con un nulo.
14. file: Nombre del archivo de arranque. Cadena terminad con un nulo.
15. options: Se refiere al campo de parámetros opcionales.

En este diagrama podemos observar cada uno de los campos que conforman un paquete DHCP típico.

comunicación pueda establecerse (chequeo de disponibilidad física para entablar la comunicación).

Transporte

Esta capa ofrece los servicios de transporte entre el host emisor y el receptor, al establecer una conexión lógica entre ambos. Los protocolos que intervienen en ella son **TCP** y **UDP** (*User Datagram Protocol*). Los datos que se reciben de la capa superior incluyen un número de puerto que permite identificar a cada protocolo (y, a su vez, al software cliente que tengamos instalado para tal fin). Es importante verificar la configuración de routers y firewalls, ya que podríamos tener bloqueado algún puerto necesario y, entonces, la comunicación no podría establecerse. Los datos recibidos son segmentados en partes iguales, que se envían desde un dispositivo final a otro final (de nuestra PC al access point, del access point al router). Esta capa garantiza el flujo de datos, la fiabilidad del enlace y el control de errores. En caso de que ocurra un error, es posible retransmitir el segmento perdido; esto se realiza gracias al procedimiento de **windowing**, que sirve para asegurar que los segmentos lleguen al receptor. Para lograrlo, el host receptor, al ir recibiendo los segmentos, manda un acuse de recibo al emisor, el cual lleva un control de los segmentos enviados satisfactoriamente; al no recibir el acuse de recibo, se reenvía el segmento perdido.

Internet

En esta capa se define el direccionamiento y la selección de ruta. Los routers usan diferentes protocolos para determinar la mejor ruta que permita establecer la conexión con el destino. Para realizar esta tarea, necesitan la dirección IP como parte del encabezado, para conocer su destino. Debemos tener en cuenta que la dirección IP está formada por dos partes: con la máscara de red se identifica a la propia red de destino; y con la de la dirección IP, al host de destino, utilizando distintos protocolos que le permiten conocer la dirección física o lógica de él.

Acceso a la red

En esta capa se preparan los datos para su transmisión por el medio físico. También se incluyen los de direccionamiento, es decir, la dirección IP asignada a la placa de red, que se identifica por su **dirección MAC**. Por otra parte, en la capa de acceso a la red influyen las topologías de redes y el hardware de conexión, que definen la conexión con el medio y la forma de envío de los datos que corresponden.

Tamaño máximo 65.500 bytes

Aquí vemos la estructura que corresponde a un datagrama o paquete.

TCP vs. UDP

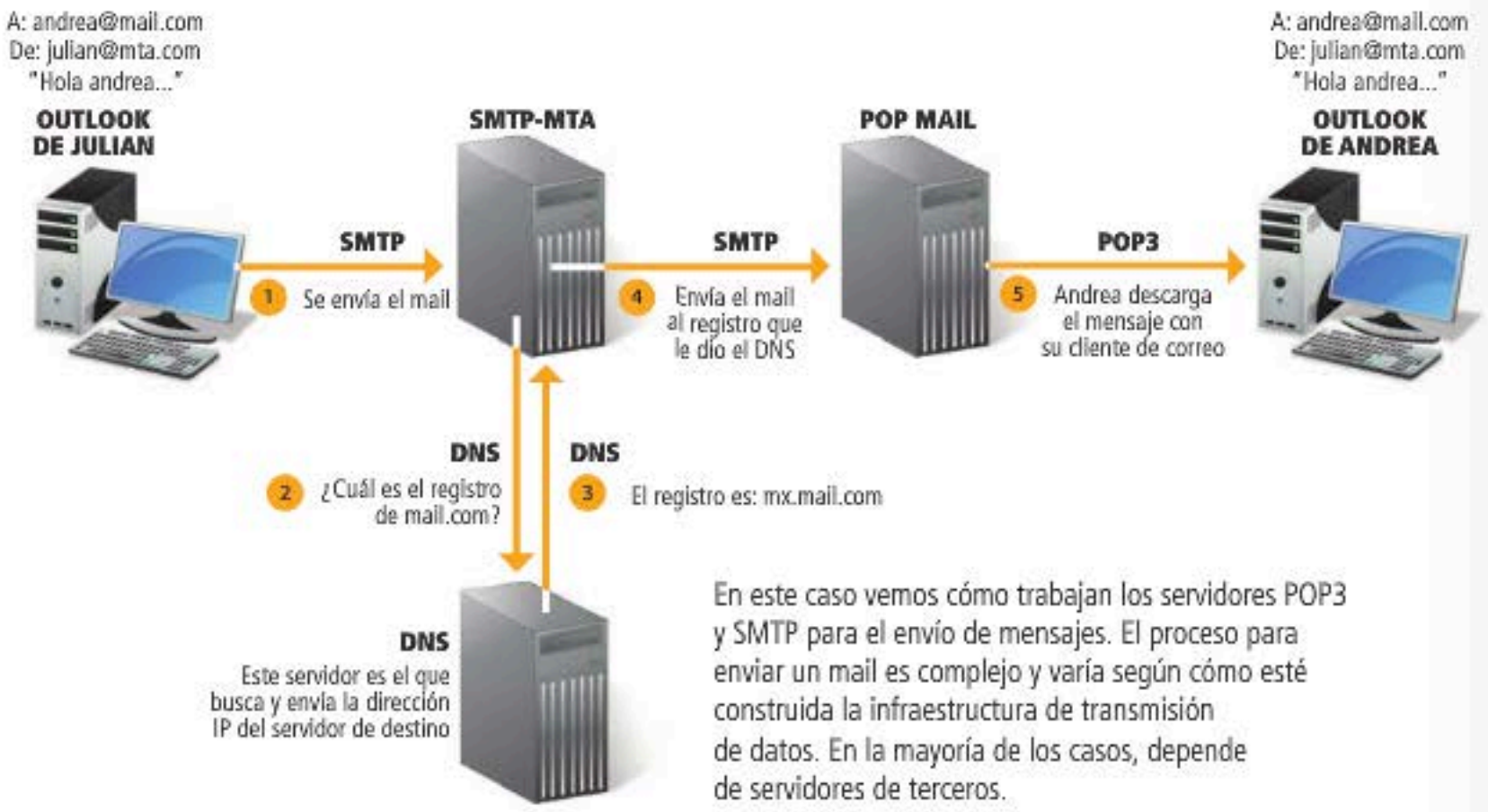
Estos protocolos proveen de servicios a la capa de transporte, pero se diferencian en cuanto a sus funciones específicas. TCP es un protocolo orientado a conexiones, que antes de realizar el envío, comprueba que este sea posible. En la cabecera que agrega a cada segmento de datos, incluye información que sirve para asegurar la entrega en orden, el control de flujo y el de errores. Algunas de las aplicaciones que usan TCP son: navegadores de Internet, clientes de correo y software para transferencia de archivos mediante FTP.

Debemos tener en cuenta que la cabecera de TCP ocupa 20 bytes en total. Sus partes son las siguientes:

- **Puerto de origen (16 bits):** número de puerto utilizado por la aplicación en el host de origen.
- **Puerto de destino (16 bits):** número de puerto que utilizará la aplicación en el host de destino
- **Número de secuencia (32 bits):** número para asegurar la correcta secuencia de los datos que van llegando.
- **Número de acuse de recibo (32 bits):** número de secuencia que incluirá el próximo segmento por recibir. Solo al recibir este número, TCP considera que el segmento anterior fue enviado en forma correcta.
- **Posición de datos (4 bits):** se encarga de indicar dónde comienzan los datos de la capa superior.
- **Reservado (6 bits):** secuencia de bits destinados a un futuro uso; al no ser utilizado, su valor corresponde a 0.
- **Ventana (16 bits):** se trata del número de bytes que el emisor espera aceptar sin acuse de recibo.
- **Suma de control (16 bits):** comprobación interna, calculada por la suma de los campos de cabecera y datos.
- **Puntero urgente:** indica el final de los datos urgentes (si antes estuvo activo el bit de control URG).
- **Opciones:** secuencia que incluye el tamaño máximo de segmento TCP.
- **Datos:** se trata de los datos provenientes de la capa superior.

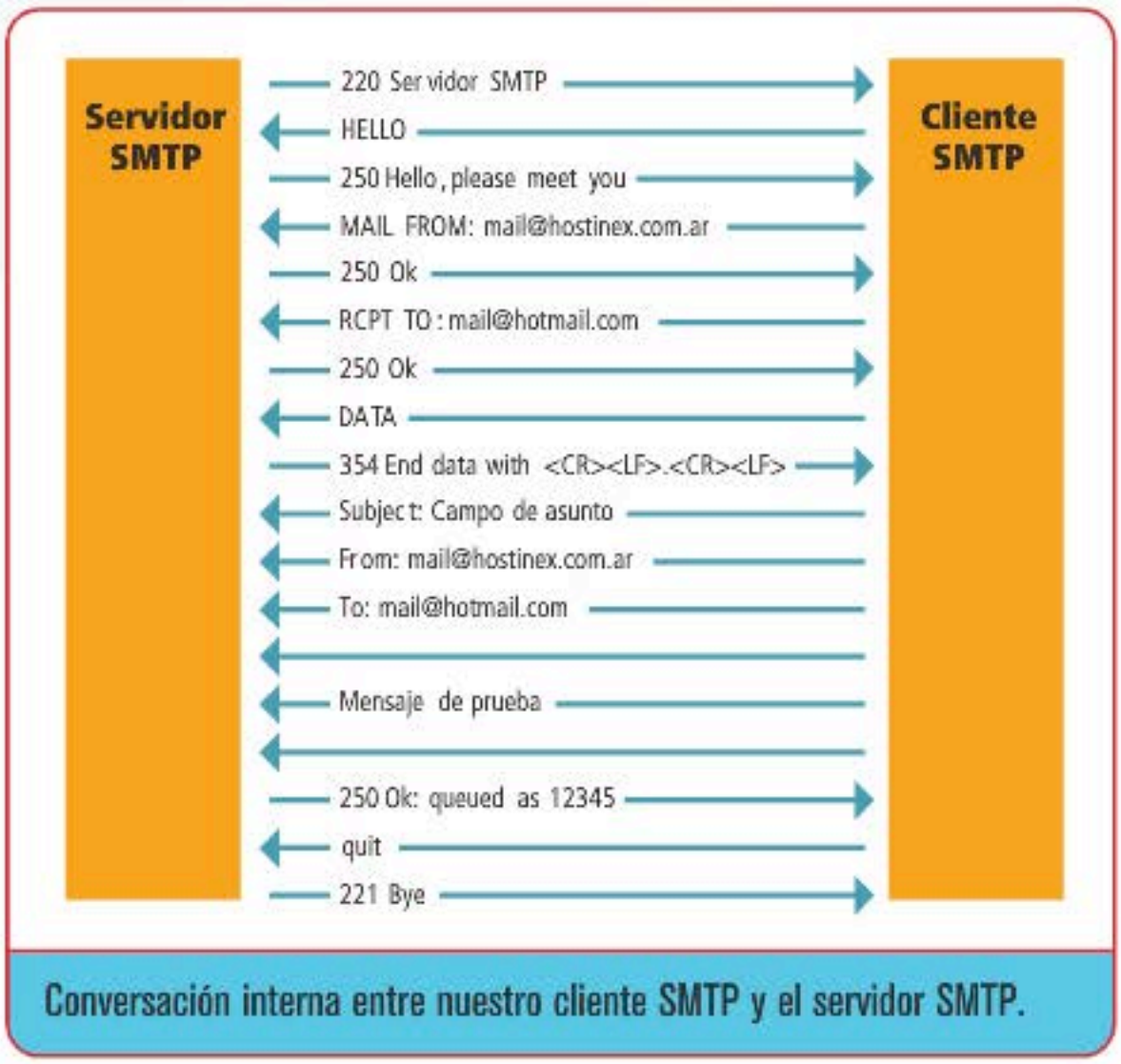
También es necesario considerar la existencia de los bits de control: se trata de bits utilizados para realizar el control de los datos entregados por TCP. Vamos a mencionarlos a continuación:

- **URG:** si su valor es 1, el paquete debe entregarse en forma urgente.
- **ACK:** si el valor es 1, el paquete se presenta como un acuse de recibo.
- **PSH:** si el valor es 1, el paquete puede ser forzado por el emisor para su envío. Utiliza el método denominado PUSH.
- **RST:** si el valor es 1, se realizará el reseteo de la conexión a la red.
- **SYN:** si el valor es 1, indica un pedido para establecer una conexión.



En este caso vemos cómo trabajan los servidores POP3 y SMTP para el envío de mensajes. El proceso para enviar un mail es complejo y varía según cómo esté construida la infraestructura de transmisión de datos. En la mayoría de los casos, depende de servidores de terceros.

En el proceso para el envío de mails intervienen protocolos y pedidos que se realizan en fracción de segundos.



Conversación interna entre nuestro cliente SMTP y el servidor SMTP.

En caso de que esta sea satisfactoria, sirve para sincronizar los números de secuencia.

- **FIN:** si se encuentra un valor 1, indica el fin de la conexión existente.

A diferencia de TCP, UDP no está orientado a la conexión. Ambos utilizan protocolo IP para el envío de datos, pero UDP no usa acuses de recibo ni garantiza su entrega. La cabecera de UDP pesa 8 bytes, y sus partes son las siguientes:

- **Puerto de origen (16 bits):** se trata de un campo opcional, que se utiliza solo si la información que ha sido enviada debe regresar al host emisor.
- **Puerto de destino (16 bits):** especifica el puerto en el host destino y, a su vez, el protocolo al cual UDP debe pasar los datos.
- **Longitud UDP (16 bits):** es el tamaño en bytes que corresponde al datagrama, incluyendo la cabecera de este.
- **Suma de control (16 bits):** también es opcional, pero se utiliza para comprobar que los datos no fueron dañados durante su transmisión.
- **Datos:** se trata de los datos que corresponden a la capa superior.

Protocolos pertenecientes a TCP/IP

El protocolo TCP/IP está formado por varios tipos de protocolos que trabajan en diferentes capas; a continuación, conoceremos algunos de ellos:

- **IP** (protocolo de Internet): recibe los segmentos de TCP, los encapsula en paquetes y agrega las direcciones IP de origen y destino necesarias.
- **ARP** (protocolo de resolución de direcciones): funciona dentro de una LAN, cuando los hosts comienzan a establecer una comunicación. Cuando tenemos varios equipos en una LAN, y cada uno tiene su dirección IP, ARP mapea las direcciones lógicas con las físicas.
- **RARP** (protocolo de resolución de direcciones inverso): cumplía la función inversa de ARP, es decir que, conociendo la dirección física de un host, buscaba su IP.
- **DHCP** (protocolo de configuración dinámica de host): permite a cualquier dispositivo obtener una dirección IP de forma automática asignada por un servidor.
- **HTTP** (protocolo de transporte de hipertexto): es el protocolo más conocido, ya que se utiliza para visualizar las páginas web con nuestro navegador preferido.
- **FTP** (protocolo de transferencia de archivos): se trata de un protocolo orientado a la conexión para la transferencia de archivos en la red.
- **TFTP** (protocolo trivial de transferencia de archivos): es un protocolo sin conexión (utiliza UDP). Se emplea en redes LAN, y es muy usado por los switches y routers empresariales para la reinstalación, actualización o carga de la configuración guardada.
- **SMTP** (protocolo para la transferencia simple de correo electrónico): se encarga de enviar correo electrónico entre servidores.
- **POP3** (protocolo de oficina de correo versión 3): este protocolo se utiliza para recibir los correos electrónicos que fueron enviados por medio del protocolo SMTP.
- **IMAP** (protocolo de acceso a mensajes de Internet): es una variante de POP3, que también podemos utilizar para recibir correos electrónicos o listas de difusión que soporten IMAP. En nuestro cliente de correo, para enviar debemos configurar el servidor SMTP, y para recibir, IMAP o POP3. ■

Aquí podemos observar cómo los protocolos SMTP y POP3 trabajan en conjunto, utilizando un servidor de correo.

Tecnología Wake On LAN

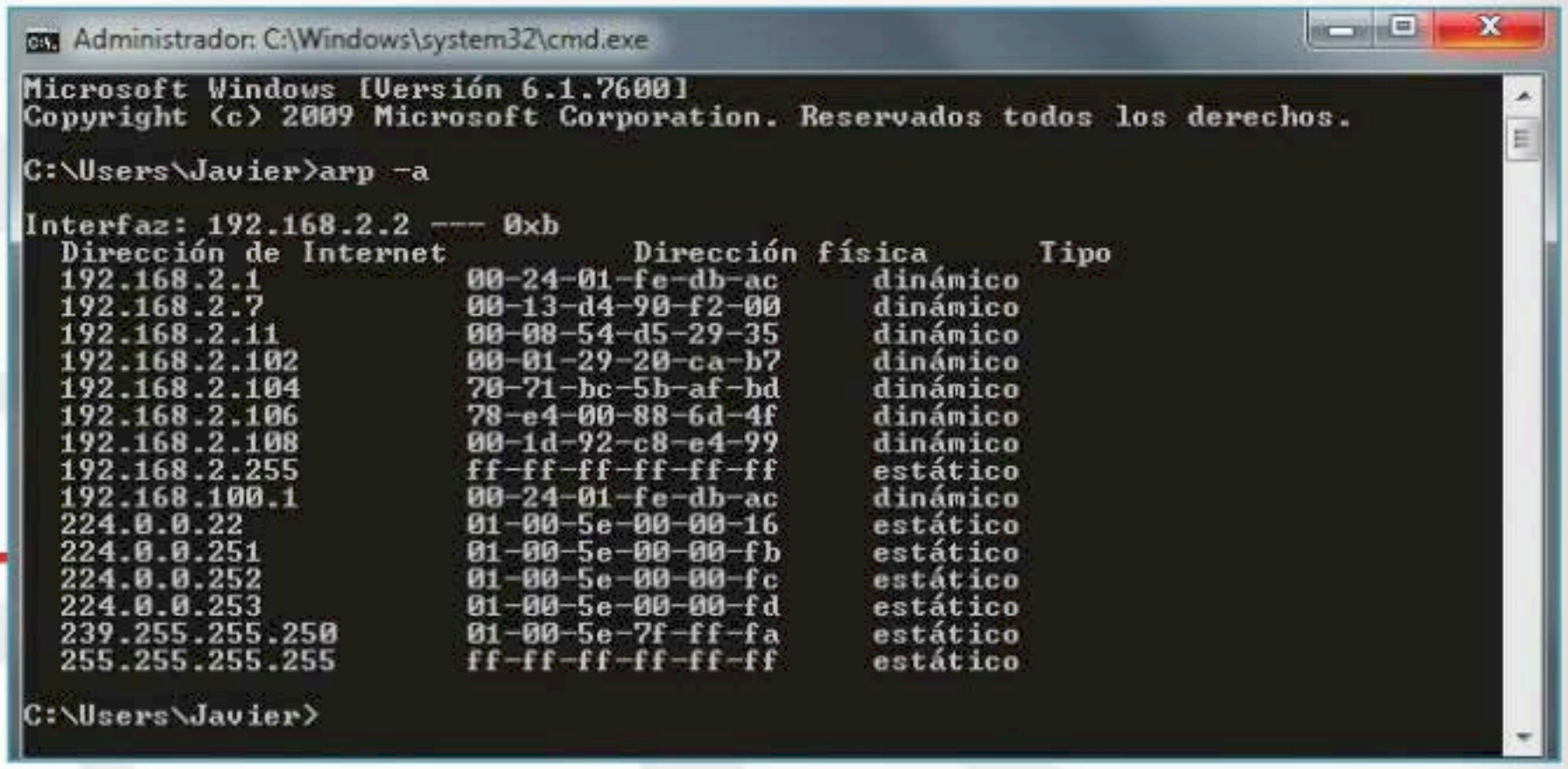
Es una función poco conocida y no muy usada, pero que puede resultar útil tanto en el hogar como en redes empresariales.

Gracias a la tecnología Wake On LAN, es posible arrancar uno o más equipos desde cualquier PC, tablet o smartphone en la red local, muy fácilmente. Hoy en día, los comercios de tipo cibercafé implementan este método para encender desde el servidor central las decenas de equipos que poseen. Esto también puede resultarle útil a un administrador de red, para encender un equipo que esté en una habitación alejada o en otro un piso. Tanto los administradores como los usuarios hogareños que cuenten con una conexión de banda ancha y un router pueden aprovechar este sistema para encender un determinado equipo en la empresa o la propia PC en casa, estando en otro lugar. De este modo, podrán acceder vía VNC, VPN, FTP o HTTP a dicho equipo para transferir archivos, realizar una consulta, disparar un trabajo de backup o, simplemente, para que nuestra aplicación P2P favorita, programada para arrancar automáticamente con el sistema operativo, comience a descargar los elementos en cola. Por ejemplo, el popular software **CiberControl**, para gestión de cibercafés, a partir de su versión 5 posee una función para encender uno o más equipos simultáneamente, desde el servidor principal. Esta función no es otra cosa que la implementación interna de WOL (Wake On LAN).

Qué necesitamos para implementarla

La mayoría de los usuarios cuenta con los elementos necesarios para aprovechar las ventajas de la tecnología Wake On LAN, ya que solo se necesitan dispositivos muy básicos:

- El o los equipos que se encenderán en forma remota deben contar con una interfaz de red incorporada al motherboard. En su defecto, se podrá usar una placa de red PCI, pero será necesario que su firmware soporte este estándar y se precisará un cable



En esta imagen vemos el comando `arp` mostrando la dirección IP de la placa de red, con su respectiva dirección MAC.

que comunique la tarjeta de red con el motherboard, mediante un pequeño conector de tres pines llamado WOL. Si la placa de red es PCI 2.2, no hará falta el cable WOL, porque esta comunicación se realiza internamente vía bus PCI a partir de esa versión.

- La fuente de la PC debe ser ATX. Esto se debe a que las fuentes de este tipo, en una PC apagada, siguen alimentando ciertos dispositivos como el motherboard, la placa de red, el módem, el teclado y el mouse (para lograr el encendido por WOL o WOM —Wake on Modem, también conocida como WOR, Wake on Ring—, hay que hacer clic del mouse o, desde el teclado, pulsar alguna tecla o combinación de ellas, o usar una contraseña).

- La opción **Wake on LAN** o **Wake on PCI Card** debe estar en **Enabled** (habilitada).

Cómo funciona

Tal como mencionamos anteriormente, las fuentes ATX continúan alimentando el motherboard mientras estén conectadas a la tensión de línea y, por lo tanto, también a la placa de red. Podemos notar que, si un equipo con fuente ATX está apagado, el LED de link de la placa de red se encenderá al conectarle un cable UTP proveniente de otra PC, hub, switch o router. De esta manera, la placa de red, al recibir la orden de encendido que coincida con su dirección física, envía una señal a la placa madre, y esta enciende el equipo, tal como si hubiésemos pulsado el botón de **PowerOn**. El encendido se produce cuando la placa de red recibe un pequeño fragmento de datos llamado **Magic Packet** en el puerto TCP o UDP número 7.

LA PLACA DE RED, AL RECIBIR LA ORDEN DE ENCENDIDO, ENVÍA UNA SEÑAL A LA PLACA MADRE.

El Magic Packet tiene un tamaño fijo de 102 bytes, que comienza con 6 bytes, con el valor hexadecimal FF, que sería la dirección de broadcast; en definitiva, significa que se enviará este paquete a todos los equipos de la subred. Luego siguen 16 grupos de 6 valores hexadecimales cada uno, que contienen la dirección física o MAC Address. Por ejemplo, si la placa de red del equipo que queremos posee la dirección física **01:E4:F6:7C:42:9B**, el Magic Packet completo sería de la siguiente forma:

FF FFFFFFFFFF
01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B
01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B
01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B
01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B 01 E4 F6 7C 42 9B

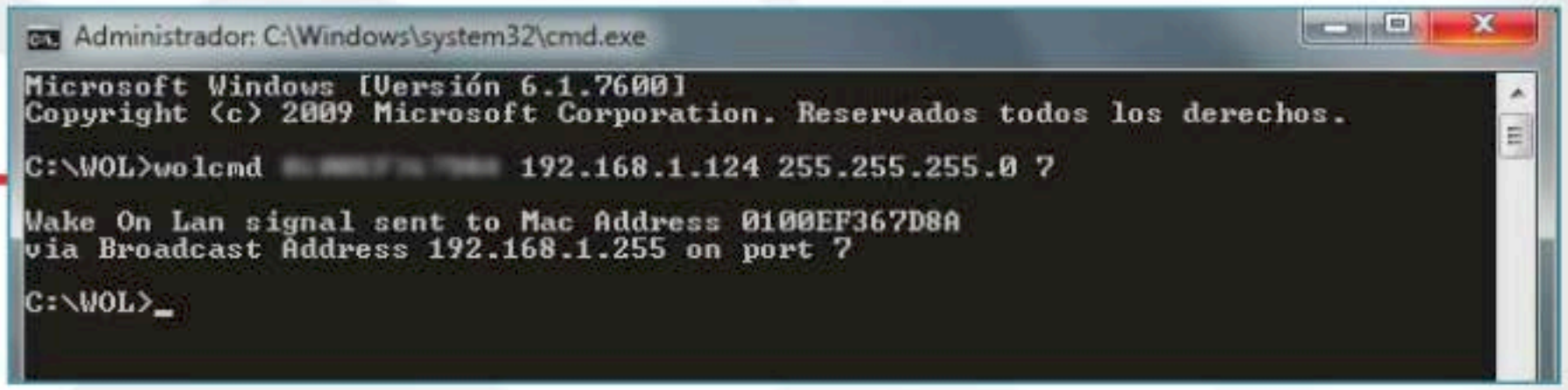
Puesta en marcha

Lo primero que debemos hacer es verificar si contamos con todos los elementos necesarios en nuestra PC. Lo más común es tener una placa de red PCI y no poseer el cable WOL, que se puede adquirir en cualquier comercio de electrónica o relacionado con cables y conectores de PC. Una vez que lo conectamos desde la placa de red hacia el motherboard en el correspondiente conector WOL, activamos la opción **Wake On LAN** o **Wake on PCI Device** del Setup del BIOS. Recordemos que si la placa de red no tiene el conector WOL, es porque cumple con la norma PCI 2.2, y no será necesario usarlo. El paso siguiente es averiguar y tomar nota de la dirección IP, máscara de subred y dirección física o MAC Address de cada uno de los equipos de la red. En Windows, podemos hacerlo ingresando en la consola de comandos y ejecutando el comando **arp –a** o **arp –g**, desde cualquier equipo de la red. Se presentarán en pantalla todas las direcciones IP y sus respectivas MAC de la subred. Si direccionamos los resultados a un archivo de texto, será más práctico y fácil manipular esa información más adelante. Lo hacemos con la siguiente sentencia: **arp –a > macaddress.txt**. Para el caso de GNU/Linux, el comando

Dirección física

La **MAC Address** o **Media Access Control Address** (dirección de control de acceso al medio, o simplemente, dirección física) es una dirección de 48 bits que poseen todos los dispositivos de red, tales como placas de red y otro hardware de conectividad. Esa dirección es única e irrepetible en cada uno de los dispositivos. Al tener 48 bits de longitud, se pueden llegar a obtener 281 billones de direcciones distintas. Originalmente, la MAC Address no podía ser cambiada por el usuario, ya que venía grabada en una memoria ROM. En la actualidad, al venir en memorias EEPROM, la dirección física puede modificarse fácilmente vía software.

Para encender equipos mediante Wi-Fi, debemos asegurarnos de tener estas opciones habilitadas.



En esta ventana de consola se ejecutó el comando wolcmd, podemos damos cuenta que inicia sesión en un equipo remoto.

es el mismo, **arp**, y posee un modificador integrado para almacenar los resultados de la conversión de direcciones en un archivo de texto: se trata de **arp –f nombre_de_archivo** (si no se especifica ningún nombre, la información se guardará en **/etc/ethers** por defecto, en la mayoría de los casos).
Un método alternativo es realizar este mismo procedimiento, pero en cada equipo, mediante el comando **ipconfig /all**. Este detallará los tres números necesarios para cada adaptador de red, en caso de que haya más de uno en el mismo equipo.
Si ya sabemos de memoria la dirección IP y la máscara, podemos ejecutar el comando **getmac**, para visualizar la dirección física.
Con esta tabla de datos que hemos recabado, podemos dar la orden de encendido desde cualquiera de las PCs de la red.
Pero para hacerlo, necesitaremos un pequeño programa argentino llamado **Fusion WOL**, que se descarga en forma gratuita desde **www.fusion-online.com.ar/es/products/wol**.

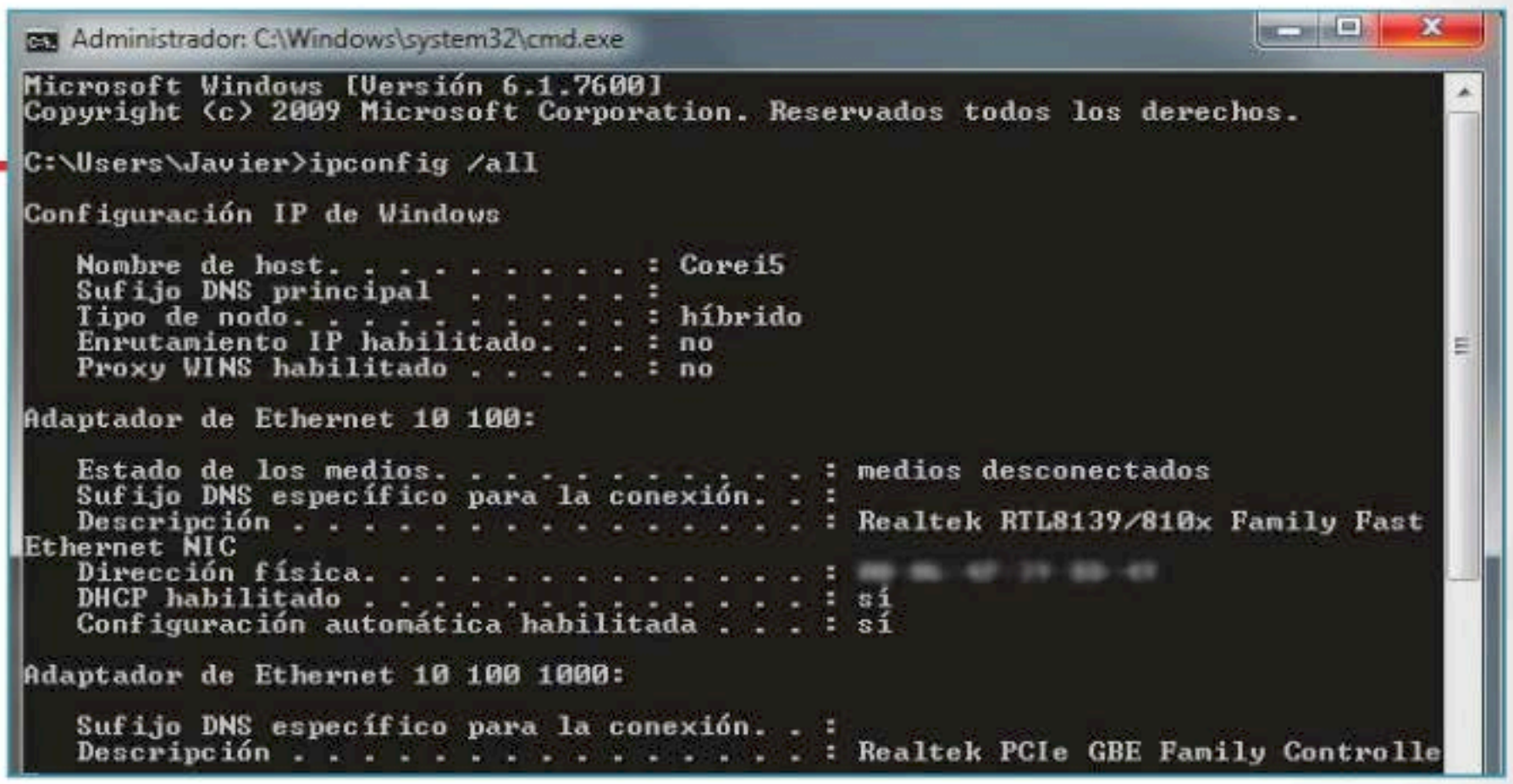
El comando ipconfig muestra más detalles que el comando arp.

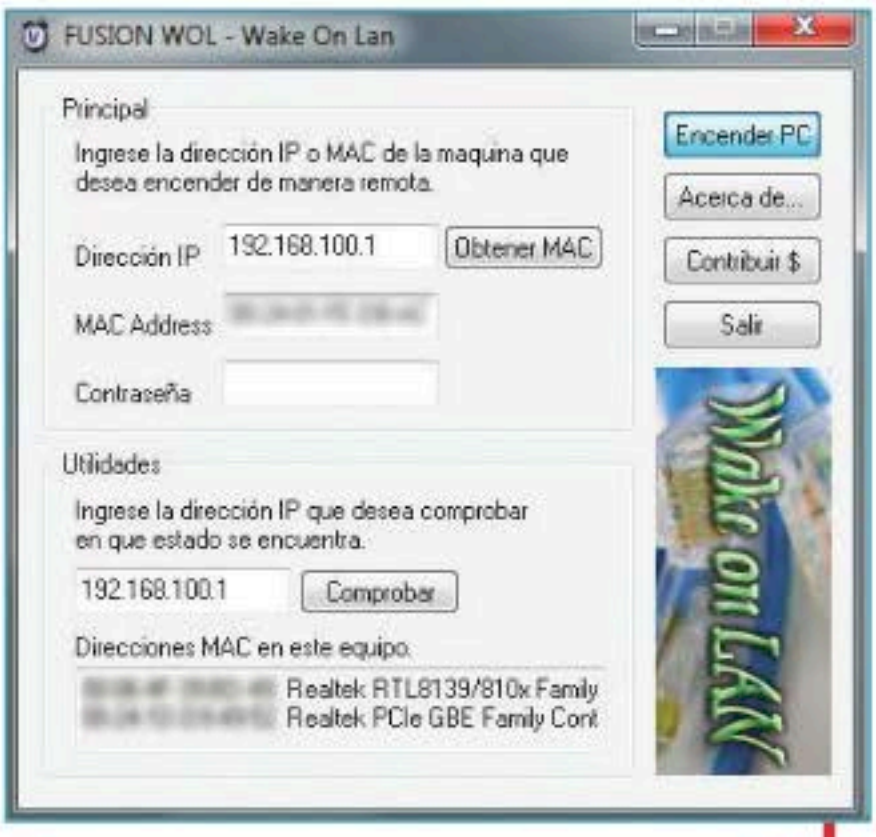
Se trata de una pequeña aplicación que no requiere instalación; al ejecutarla, solo debemos completar el campo MAC Address del equipo que queremos encender (tenemos que averiguarla previamente). Al hacer clic en el botón **Encender PC**, el equipo remoto se encenderá al instante. Existe una versión de consola, llamada **WOLcmd** (**www.depicus.com/wake-on-lan/wake-on-lan-cmd.aspx**). Las sentencias para este comando se ejecutan de la siguiente forma:

wolcmd dirección física dirección IP máscara de subred puerto

Por ejemplo:
wolcmd 0100EF367D8A 192.168.1.124 255.255.255.0 7

La gran ventaja de esta modalidad es la posibilidad de crear scripts y encender decenas o cientos de equipos en pocos segundos y con tan solo un clic o un comando, guardando previamente en un archivo **.CMD** o **.BAT** que contenga las sentencias para encender cada uno de ellos. Esto nos permite ahorrar tiempo y esfuerzo.

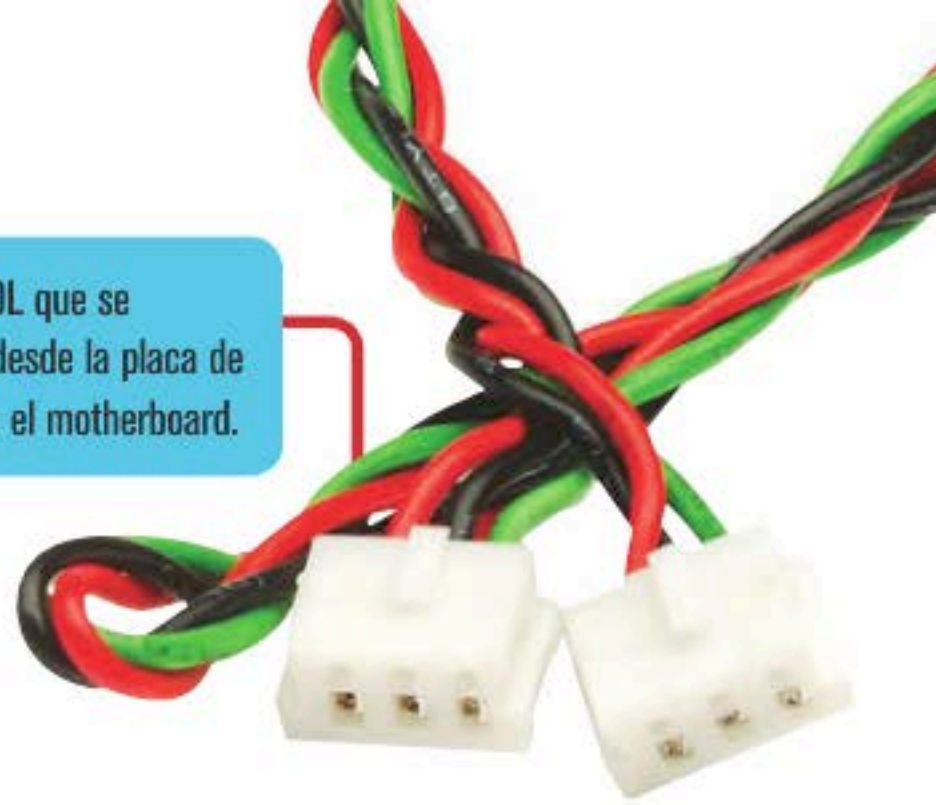




Fusion WOL es un pequeño software diseñado para encender equipos remotos fácilmente.

Firmware DD-WRT

DD-WRT es un firmware o software interno (sistema operativo) como el que llevan todos los routers W-iFi, pero concebido para mejorar las prestaciones de este tipo de dispositivos. Ya lo veremos más adelante y con mayor detalle, pero amerita ofrecer un adelanto en esta oportunidad porque, justamente, DD-WRT brinda un completo apartado dedicado a la función Wake On LAN.
Hace unos cuantos años, Cisco tuvo la idea de usar un firmware basado en Linux en sus routers de la línea Linksys. Al lanzar la primera versión de su legendario modelo WRT54G, tuvo que liberar el código fuente de su sistema operativo al público cuando esto se supo, por estar bajo licencia GPL. A raíz de esto, muchos programadores comenzaron a modificar el firmware original para obtener mayores beneficios, por medio de la habilitación de otras funciones intersantes. Lo mismo sucedió con otros modelos de la misma firma y, más tarde, con equipos de otros fabricantes.
El firmware DD-WRT está basado en GNU/Linux y licenciado bajo GPL2; no es compatible con todos los routers del mercado, pero sí con la mayoría de los modelos inalámbricos de uso hogareño:



Cable WOL que se conecta desde la placa de red hacia el motherboard.

la lista incluye los clásicos DIR-300 y DIR-600 de D-Link, los Linksys WRT54G/GS/GL de Cisco y algunos modelos de equipos TP-Link, entre otros. Es decir, los modelos más comúnmente utilizados por los usuarios son compatibles con este software alternativo.
Para instalarlo, como primera medida, debemos saber si nuestro router forma parte de la lista de dispositivos compatibles con DD-WRT; el listado completo puede consultarse desde el enlace: **www.dd-wrt.com/wiki/index.php/Supported_Devices**.
Una vez que lo verificamos, debemos buscarlo en la base de datos de descargas, ingresando al menos tres letras o números de su marca o modelo, desde el enlace: **www.dd-wrt.com/site/support/router-database**. Luego de ubicar nuestro modelo, hacemos clic sobre el nombre (marca o modelo), lo cual nos enviará a una nueva página con diversas versiones del firmware.

LA MAYORÍA DE LOS USUARIOS CUENTA CON LOS ELEMENTOS NECESARIOS PARA APROVECHAR LAS VENTAJAS DE LA TECNOLOGÍA WAKE ON LAN.

Wake On Internet

Si tenemos una conexión directa (cablemódem) con IP fija, está todo dicho: los datos que debemos ingresar son los mismos que para una red LAN. En caso de contar con más de un equipo o conexión ADSL, debemos tener nuestro router siempre online y con la opción **Subnet Directed Broadcasts** habilitada.
En caso de que el router no disponga de esta opción, habrá que establecer en él las rutas estáticas que cada PC tiene en la red interna (dirección IP <---> dirección física). ■

WoWLAN

El término WoWLAN proviene de **Wireless Wake On LAN**, y es, básicamente, el mismo concepto que el de WOL aplicado a equipos que se conectan a la red mediante interfaces Wi-Fi. Se utiliza igual que en el caso de WOL, pero debemos tener en cuenta que no todas las placas de red inalámbricas soportan el encendido remoto y que debe estar activada la opción **Permitir que este dispositivo reactive el equipo**, dentro de las propiedades de la interfaz Wi-Fi en el **Administrador de dispositivos.**

Tecnología FireWire para equipos en red

Si tenemos puertos IEEE1394 en nuestros equipos, nada mejor que conectarlos vía red por medio de esta interfaz; así tendremos acceso a una alta velocidad en la transferencia de datos.

La tecnología **FireWire** fue desarrollada por la empresa Apple en la década de 1980 con la idea de utilizarla para interconectar discos duros internos en los equipos Mac de aquella época. Luego de unos años, ya en los 90, el IEEE (*Institute of Electrical and Electronics Engineers*) se basó en esta tecnología para crear lo que hoy conocemos como IEEE-1394 o FireWire (Sony la adquirió para sus cámaras DV bajo el nombre de i-Link), utilizada en impresoras, escáneres, discos externos y, sobre todo, en cámaras de video profesional. Otro detalle que hace a FireWire más versátil que USB 2.0 (su principal competidor) es que puede usarse como un dispositivo de red, es decir que, por medio de un cable especial, se pueden interconectar computadoras, y estas pueden compartir sus recursos con las demás (archivos, impresoras y hasta la conexión a Internet).

Estándar

El estándar **FireWire A** posee una tasa de transferencia de 400 Mbps, y **FireWire B**, lanzado más adelante, alcanza los 800 Mbps, valores muy superiores a los de una interfaz Ethernet común y corriente, que es de 100 Mbps. Este sistema permite conectar hasta 63 dispositivos, aunque cabe aclarar que, usando unos aparatos especiales llamados concentradores, esa cifra puede trepar hasta los 1024. Esta tecnología, al igual que USB, también es hot-plug, es decir, los dispositivos se pueden conectar mientras el equipo está encendido y serán detectados automáticamente.
Cada tecnología es buena en su especialidad. USB fue diseñada para interconectar dispositivos como impresoras, escáneres, webcams, pen drives, etc., y logra muy bien su objetivo, sobre todo, el de la universalidad y popularidad. FireWire apunta a la transferencia masiva de datos, por ejemplo, en las cámaras de video DV o en grandes unidades de almacenamiento externo. Quizás esté menos difundido y no sea tan popular como USB, pero en ámbitos profesionales (como la edición de video digital) es muy reconocido. Además de FireWire 400 y FireWire 800, existen dos versiones posteriores menos frecuentes, FireWire 1600 y FireWire 3200, mucho más veloces: de 1,6 y 3,2 Gbps, respectivamente.

TCP/IP over FireWire

TCP/IP over FireWire es un stack que posibilita establecer comunicaciones de red basadas en IP sobre una conexión FireWire. Es decir, se aprovechan las ventajas que tiene FireWire, como la alta velocidad de transmisión de datos y la posibilidad de sostenerla en el tiempo (a diferencia de otras tecnologías, que son más fluctuantes), y se la hace actuar como si fuera una conexión de red Ethernet. Esta conversión es transparente al usuario y se lleva a cabo mediante un driver o controlador que Windows XP y Mac OS X incluyen en forma nativa. Incluso, FireWire A (de 400 Mbps) supera la tasa de transferencia sostenida de una interfaz Ethernet operando a una velocidad máxima teórica de 1 Gbps. Si bien USB 2.0 opera a 480 Mbps, FireWire aprovecha mejor su arquitectura y, aun trabajando a 400 Mbps, obtiene mejores resultados.

IP sobre Bluetooth

De la misma manera en que una capa de red puede implementarse en otras plataformas que no hayan sido concebidas inicialmente para soportarlas, como TCP/IP sobre FireWire, existen otros ejemplos, como IP sobre Bluetooth. Esta implementación permite crear una red para que dos dispositivos que se comunican mediante Bluetooth transfieran datos mediante el protocolo IP, al igual que lo harían si estuviesen enlazados físicamente por un cable Ethernet o inalámbricamente mediante Wi-Fi.

Las placas FireWire B o FireWire 800 duplican la tasa de transferencia de la primera versión de esta tecnología.

Establecer conexión

Para establecer una nueva conexión de red mediante FireWire, solo debemos conectar dos computadoras con un cable FireWire especial (que tenga una ficha macho en cada uno de sus extremos). Los equipos pueden ser Mac o PC, indistintamente; incluso, se puede conectar una PC a una Mac. El sistema operativo puede ser Windows, Mac OS X o GNU/Linux; y también se podrán comunicar entre un sistema y otro si los equipos poseen diferentes plataformas de software.

La configuración del software en Windows es muy simple. Ingresamos en las conexiones de red, donde figura la interfaz FireWire. Hacemos clic derecho sobre su icono, vamos a Propiedades y, allí, establecemos una **dirección IP** y una **máscara de subred**. En Linux, hay que seguir los pasos que indicamos a continuación. Para activar el driver, ingresamos en la consola de comandos y ejecutamos el comando:

```
sudomodprobefirewire-net
```

Para activar automáticamente esta interfaz durante el booteo, hacemos un cambio en </etc/network/interfaces>:

```
auto firewire0
iface firewire0 inetstatic
address192.168.3.x
netmask 255.255.255.0
broadcast192.168.3.255
pre-upmodprobefirewire-net
```

Soporte nativo en Windows

Desde el año 2004, Microsoft decidió no brindar más soporte para redes TCP/IP sobre FireWire para futuras versiones de Windows (desde Vista en adelante), alegando la popularidad y bajo costo de las interfaces de red de 1 Gbps. Por suerte, una compañía llamada **UniBrain** desarrolló sus propios controladores **IP over FireWire** para Windows Vista, Server 2008, 7 y 8 (en sus versiones tanto de 32 como de 64 bits) llamados **drivers ubCore**.

El enlace para descargar los controladores **ubCore** es **www.unibrain.com/download/download.asp**.

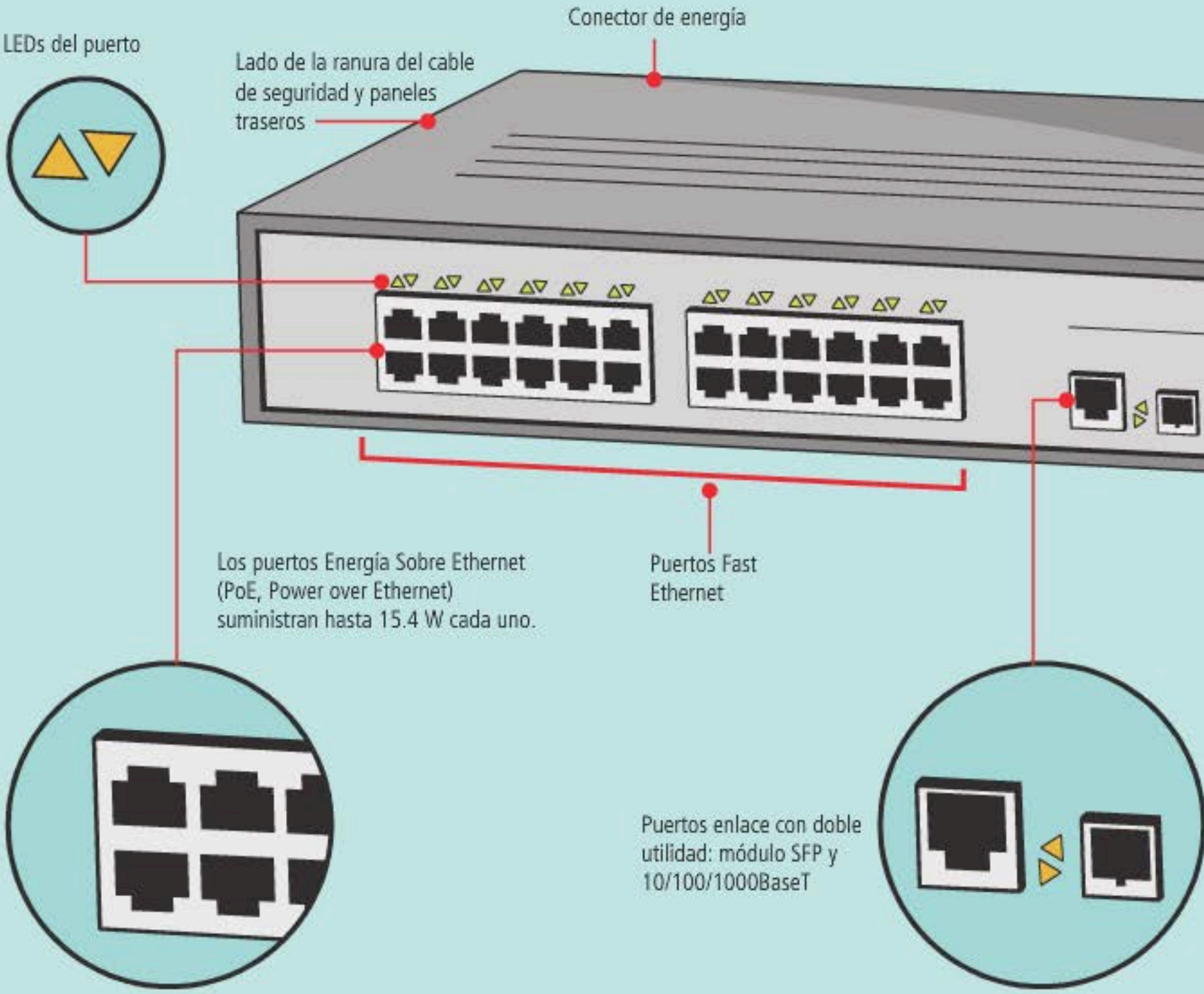
Una vez que los tenemos, los instalamos para lograr la compatibilidad necesaria. ■

Los usuarios de cámaras DV son los principales beneficiados con la tecnología FireWire para transferir archivos a la PC.

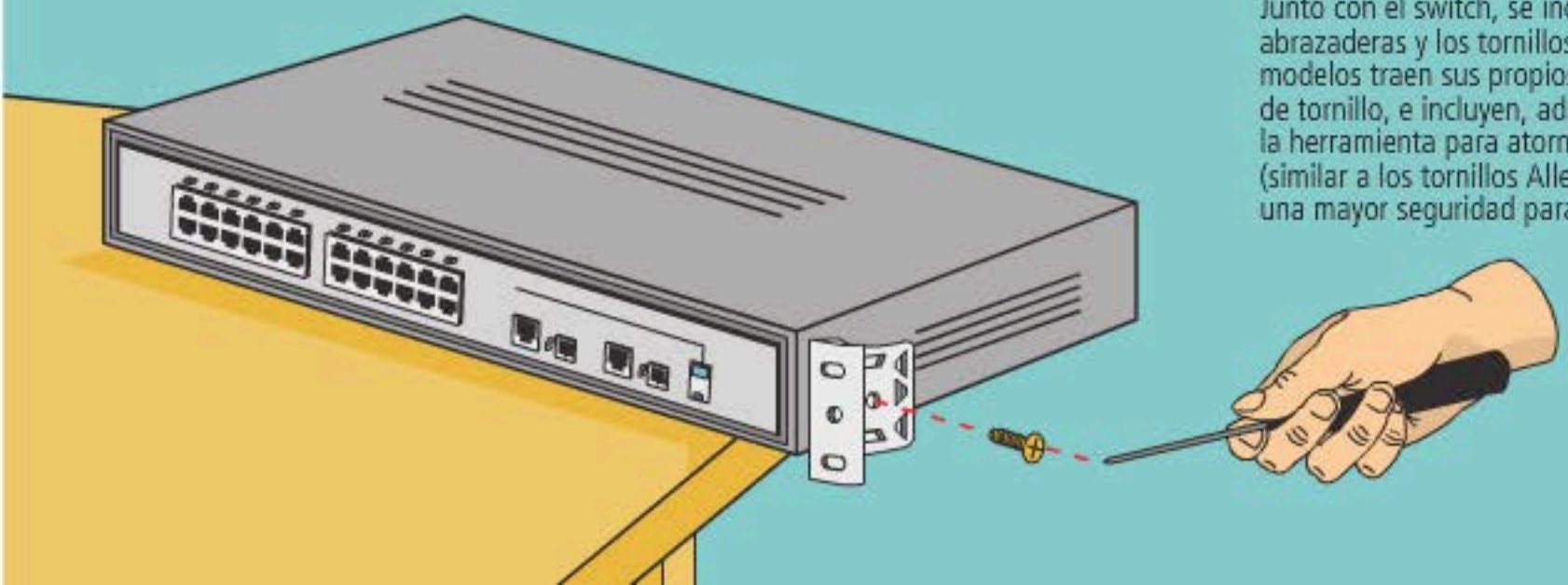
Tecnologías de transmisión de datos

Estándar	Bus	Velocidad máxima	Cantidad de dispositivos	Distancia máxima
Ethernet 10BaseT	Cliente/Servidor	10 Mbps	-	100 m
Ethernet 100BaseTx	Cliente/Servidor	100 Mbps	-	100 m
Ethernet 1000BaseTx	Cliente/Servidor	1 Gbps	-	100 m
USB 2.0	Basado en host	480 Mbps	127	5 m
FireWire A (400)	Punto a punto	400 Mbps	63	4,25 m
FireWire B (800)	Punto a punto	800 Mbps	63	100 m

Montaje del switch



CÓMO ARMARLO | Procedimiento de instalación



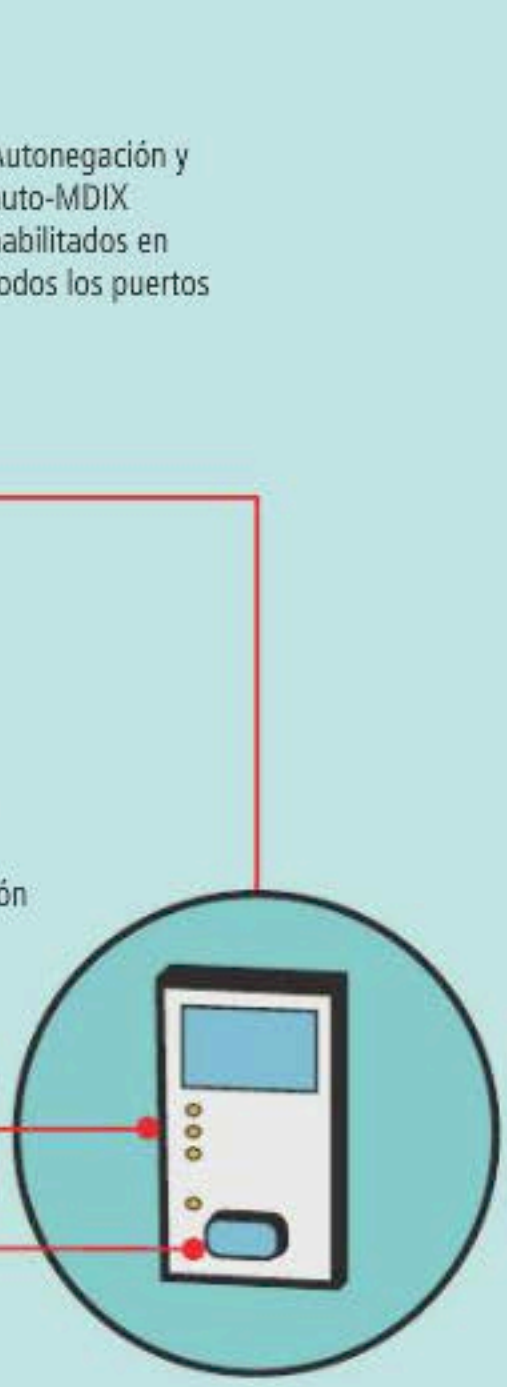
PASO 1
Junto con el switch, se incluyen las abrazaderas y los tornillos. Algunos modelos traen sus propios modelos de tornillo, e incluyen, además, la herramienta para atornillarlos (similar a los tornillos Allen), proveyendo una mayor seguridad para su anclaje.

PARA MONTAR UN SWITCH EN EL RACK EN FORMA CORRECTA DEBEMOS CONSIDERAR ALGUNOS ASPECTOS IMPORTANTES, LOS CUALES REVISAMOS EN ESTAS PÁGINAS.

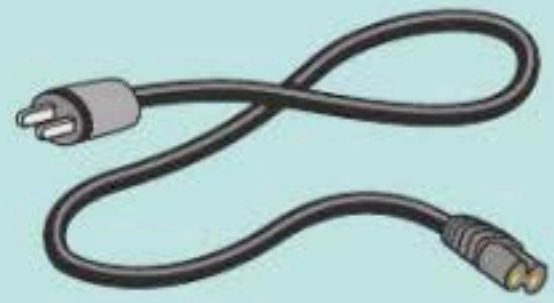
Switch

Autonegación y auto-MDIX habilitados en todos los puertos

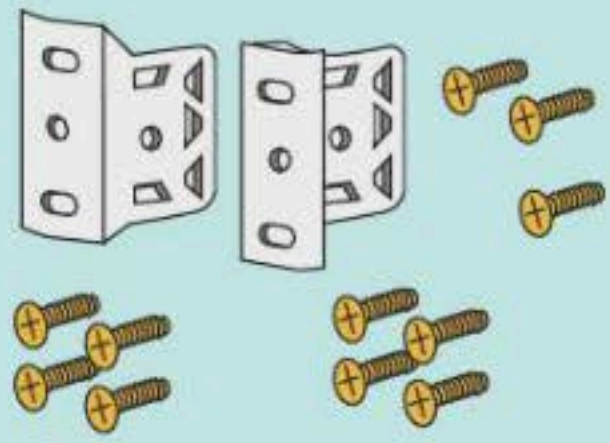
LEDs del switch:
System: Estado de switch
Alert: Eventos detectados
PoE: Estado de PoE
Setup: Modo de configuración



Botón Setup

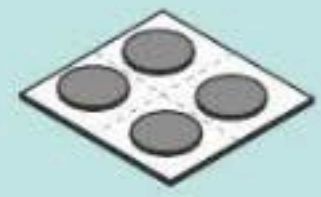


Cable de alimentación

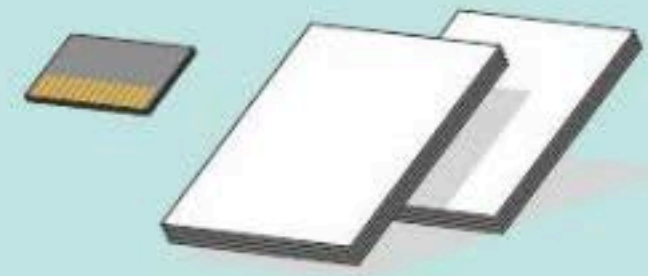


Abrazaderas y tornillos para montar el bastidor

Guía para el cable y tornillo



Patas de montaje



Documentación

PASO 2

Una vez colocado el switch en el rack, debemos poner las trabas de seguridad que, además de ir a presión, también irán atornilladas e impedirán que se pueda retirar el switch del rack. Si al momento de colocar los tornillos dañamos accidentalmente alguno de ellos, lo debemos descartar inmediatamente y colocar uno nuevo. Por más que el switch haya traído tornillos propietarios, siempre podremos usar los convencionales, como los Philips.

Permisos en grupos de trabajo

En estas páginas aprenderemos la forma en que podemos compartir recursos con usuarios o grupos de usuarios y configurar los permisos de acceso adecuados para cada recurso o usuario en particular.

Cuando tenemos varios usuarios de una red, tal vez no queramos compartir toda la información con ellos, sino solo algunas carpetas específicas. En sistemas Windows es posible hacer frente a esta tarea realizando la creación de usuarios locales, a los cuales se les otorgan distintos **privilegios**, tal como explicaremos en las siguientes secciones.

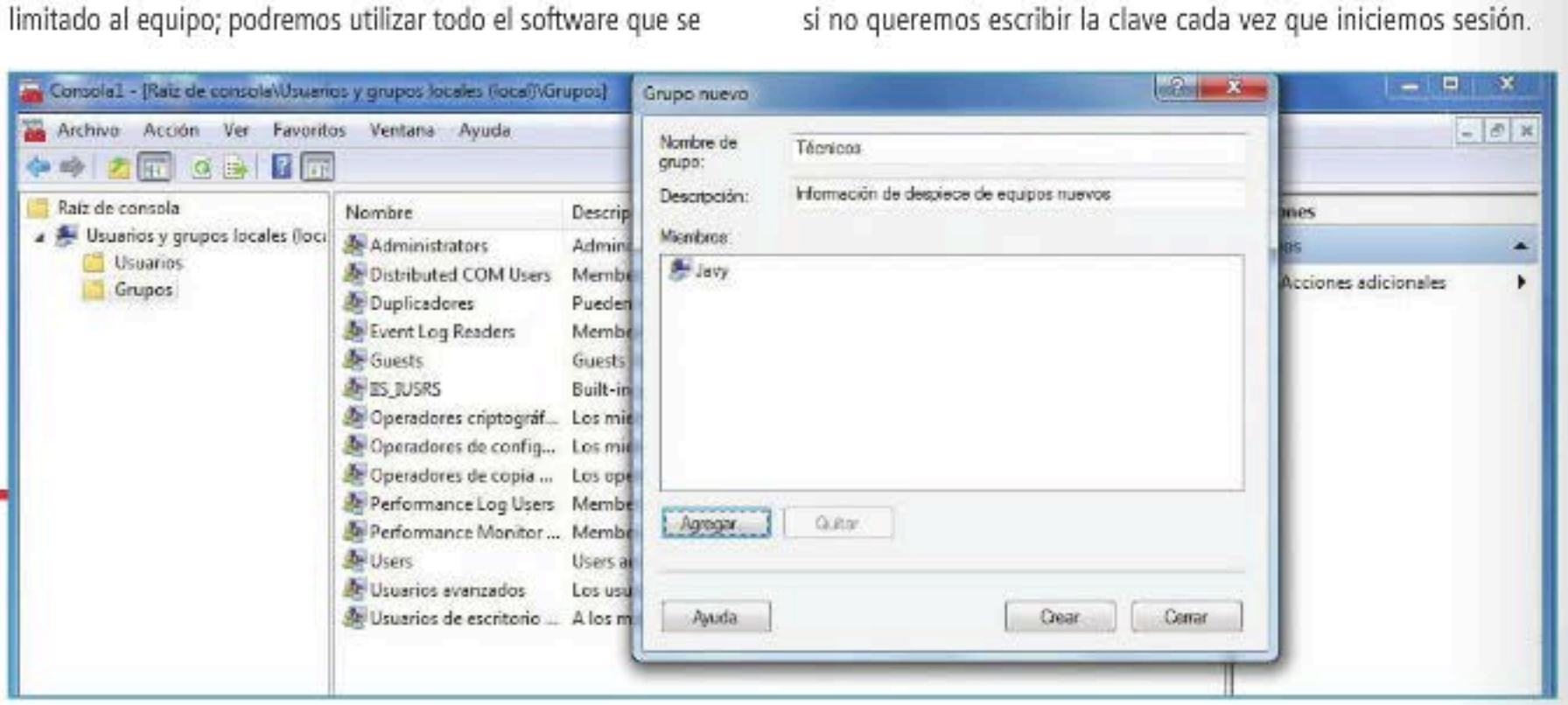
Privilegios

En Windows encontraremos diferentes privilegios para usuarios locales, los cuales responden a necesidades específicas:

- **Administrador**: con esta cuenta tendremos acceso a todo el sistema, de modo que podremos instalar y desinstalar programas, o realizar otras modificaciones.
- **Usuario estándar**: con este tipo de usuario, tenemos acceso limitado al equipo; podremos utilizar todo el software que se encuentra instalado, pero no, desinstalar o instalar nuevas aplicaciones, y las modificaciones que hagamos se limitarán solo a la configuración de nuestro usuario personal.
- **Invitado**: este tipo de cuenta puede activarse para aquellas personas que no usan diariamente el equipo, pero lo necesitan durante un tiempo. No posee carpeta personal en la cual se almacenen documentos, y no admite efectuar modificaciones; solo puede utilizar el software instalado.

Compartir con usuarios específicos

Para compartir carpetas o archivos de manera personalizada en nuestra red, debemos establecer privilegios. Por ejemplo, supongamos que la red está compuesta por tres equipos (de escritorio o notebooks). En todos ellos debemos crear el usuario de los otros equipos con su respectiva contraseña. Podemos configurar el inicio automático para nuestro usuario si no queremos escribir la clave cada vez que iniciemos sesión.



Desde Management Console, podemos crear diferentes grupos y agregar usuarios según qué información queramos compartir.

Es conveniente crear una carpeta especial en la que guardaremos los documentos que queramos compartir con los otros usuarios. Por ejemplo, creamos la carpeta **Compartido** dentro de **Mis Documentos**. Luego, hacemos clic derecho sobre ella, vamos a la pestaña **Seguridad** y a **Editar**. Pulsamos en **Agregar**, y en la siguiente ventana, vamos escribiendo el nombre de los usuarios con los cuales deseamos compartirla. En la parte inferior de donde aparecen los usuarios, podemos configurar de forma personalizada los permisos que queremos otorgarles sobre lectura y escritura para la carpeta compartida. Para terminar, aceptamos todos los cambios.

EN WINDOWS ENCONTRAMOS DIFERENTES PRIVILEGIOS PARA APLICAR A USUARIOS LOCALES.

Windows Server

Si estamos utilizando un equipo dentro de una empresa, seguramente necesitaremos un nombre de usuario y contraseña para iniciar sesión. Este nombre se da de alta desde el servidor interno de la organización, y según nuestras actividades, tendremos diferentes permisos para realizar nuestra actividad. En este caso, podemos unirnos a un Grupo en el Hogar, pero no podremos compartir nuestros archivos ni crear un grupo nuevo para que los demás equipos se unan a él.

Grupos

Otra opción es asociar todos los usuarios que hemos creado en un **grupo común**, por ejemplo, si queremos compartir una carpeta con algunas personas del trabajo, y otra con nuestra familia.
Una vez que hemos creado todos los usuarios que necesitamos, los reunimos en grupos. Para crear un grupo de usuario, hacemos clic en **Inicio**, escribimos MMC y apretamos **ENTER**. Se abrirá la ventana de **Management Console**. Si en el menú de la derecha no vemos el ítem llamado **Usuarios y Grupos Locales**, lo añadimos desde **Archivo/Agregar o quitar complementos**, seleccionándolo de la lista. Aceptamos los cambios, y el ítem mencionado ya aparecerá en la derecha. Al desplegar **Usuario y grupos locales**, elegimos **Grupos**. Desde el menú **Acción** creamos los grupos necesarios; para nuestro ejemplo, uno **Trabajo** y otro **Familia**. Luego, en ellos seleccionamos los usuarios que correspondan. Una vez hecho esto, hacemos clic derecho sobre la carpeta que deseamos compartir y vamos a **Propiedades**, como vimos anteriormente, solo que en vez de ir seleccionando los usuarios de a uno y escribir sus nombres, indicamos el del grupo. Los permisos que concedamos se aplicarán a todos los usuarios que forman parte del grupo.

Con un clic derecho sobre la carpeta que queremos compartir, agregamos usuarios específicos.

Grupo en el Hogar

A partir de Windows 7, contamos con la función de **Grupo en el Hogar**, que permite configurar de forma casi automática los recursos y archivos que deseamos compartir en nuestra red local. Para crear un grupo, hacemos clic en **Inicio**, escribimos **Grupo Hogar** y presionamos **ENTER**. Se abre una ventana que nos indica si ya formamos parte de uno, o nos da la opción de crear uno nuevo. Al crear un nuevo **Grupo**, nos pregunta qué deseamos compartir (música, imágenes, videos o impresoras). El asistente configura los recursos seleccionados, y nos muestra en pantalla una contraseña para el grupo en cuestión.
Los demás equipos se deben unir a ese grupo, para lo cual, una vez más, desde **Grupo en el Hogar**, hacemos clic en **Unirse**. Se nos preguntará qué recursos queremos compartir y el ingreso de la contraseña que se nos otorgó anteriormente.
A partir de ese momento, si abrimos el explorador de Windows, en el menú de la derecha aparecerá el ítem **Grupo en el Hogar**, y veremos, por nombre de usuario, los equipos que lo componen. Si somos usuarios de dos equipos del grupo —por ejemplo, disponemos de uno portátil y uno de escritorio—, debemos tener un nombre de usuario que nos ayude a identificar a cada uno, como **UsuarioPC** para el desktop y **UsuarioNet** para la netbook.

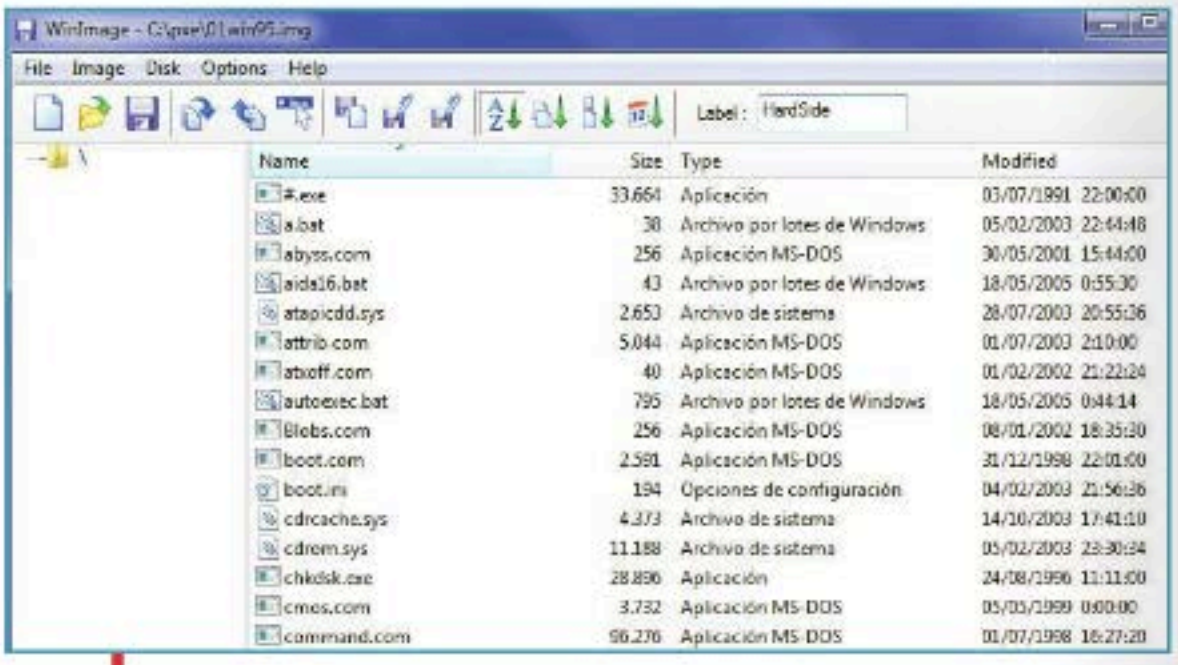
Quitar o agregar carpetas al Grupo en el Hogar

Para quitar una carpeta o archivo que ya no deseemos compartir en el grupo, simplemente hacemos clic derecho en el archivo, y en el menú contextual vamos a **Compartir con**, donde elegimos **Nadie**. De esta forma, podemos excluir carpetas o archivos específicos del grupo. De un modo similar, si tenemos una carpeta en otra ubicación, como en otro disco rígido, y queremos compartirla con el grupo, del menú contextual **Compartir con**, pulsamos en **Grupo en el Hogar**, y veremos dos opciones: compartir solo lectura, o lectura y escritura. En el último caso, los usuarios que se conecten a esta carpeta compartida tendrán la libertad completa para agregar o borrar los archivos que consideren necesarios. ■

Booteo remoto: entorno PXE y protocolo TFTP

¿Es posible bootear una PC sin disco duro, ni unidad óptica, ni llave USB? Sí, mediante la placa de red y un servidor de booteo.

El significado de la sigla **PXE** es **Preboot eXecution Environment**, o entorno de ejecución previa al arranque. Es un protocolo cliente/servidor que combina otros dos: DHCP, para asignar direcciones IP automáticamente; y TFTP, para realizar la transferencia de archivos de inicio (bootstrap) y otros adicionales, empleando los puertos UDP 67 al 69.



Aquí vemos WinImage, el software que nos ayudará a preparar los archivos IMG de booteo.

Aplicaciones

Los usos que se le pueden dar a este sistema alternativo de arranque son variados. A continuación mencionamos algunos ejemplos de ellos:

- **Booteo en equipos sin CD-ROM ni floppy** (caso muy común en las estaciones de trabajo de la mayoría de las empresas y cibercafés). A la hora de hacer o recuperar una imagen de disco vía red en este escenario, PXE se torna imprescindible.

- **Recuperación de desastres**: en casos de emergencia, se lo utiliza para bootear vía red utilidades como antivirus, software de diagnóstico o recuperación de datos, herramientas de partición, clonación e imagen de discos duros.

- **Booteo práctico y cómodo**: se trata de una opción muy útil para los administradores de red en una empresa, que pueden alojar todas las herramientas booteables en un servidor y acceder a ellas desde cualquier equipo que las requiera, sin necesidad de transportar discos de arranque, ya sea CD, DVD o unidades USB; así se reduce el tiempo y el esfuerzo del administrador de la red.

WinImage

La aplicación WinImage **permite crear, editar y guardar imágenes booteables de discos de inicio de plataformas DOS/Win9x. Se encarga de utilizar los formatos .IMA, .IMG e .IMZ, y permite inyectar archivos dentro de esas imágenes de disco. Además de que podemos editar discos de inicio de Win9x, nos será de utilidad para generar discos de arranque destinados a otros fines, como software de diagnóstico de hardware y algunos discos de emergencia, por ejemplo.**

Cómo funciona

Como primera medida, debemos tener habilitada la opción de booteo por red en el **Setup** del BIOS del equipo cliente; es decir, el primer **Boot Device** debe estar establecido en la opción **LAN**. De esta forma, el BIOS rastreará el bootstrap a través de la placa de red, ya sea que esté incorporada al motherboard o en formato de tarjeta discreta. En caso de no tener placa de red onboard, podremos usar una convencional, pero esta deberá tener el chip de **BOOT ROM** colocado en el zócalo. Este firmware se encarga de disparar y hacer funcionar el sistema **PXE**. Por su parte, las placas incorporadas traen este firmware integrado desde fábrica.

Una vez que el BIOS inició el booteo vía PXE, el firmware de la placa de red dispara la búsqueda del servidor DHCP presente en la red. Cuando lo encuentra, consulta si cuenta con funcionalidades PXE y, en caso afirmativo, procede a la petición de la ubicación del NBP (*Network Bootstrap Program*). Este se transfiere vía TFTP y se aloja en la memoria RAM del equipo cliente, donde se ejecuta exactamente igual que un disquete o CD-ROM de inicio.

EL SIGNIFICADO DE LA SIGLA PXE ES "ENTORNO DE EJECUCIÓN PREVIA AL ARRANQUE".

Manos a la obra

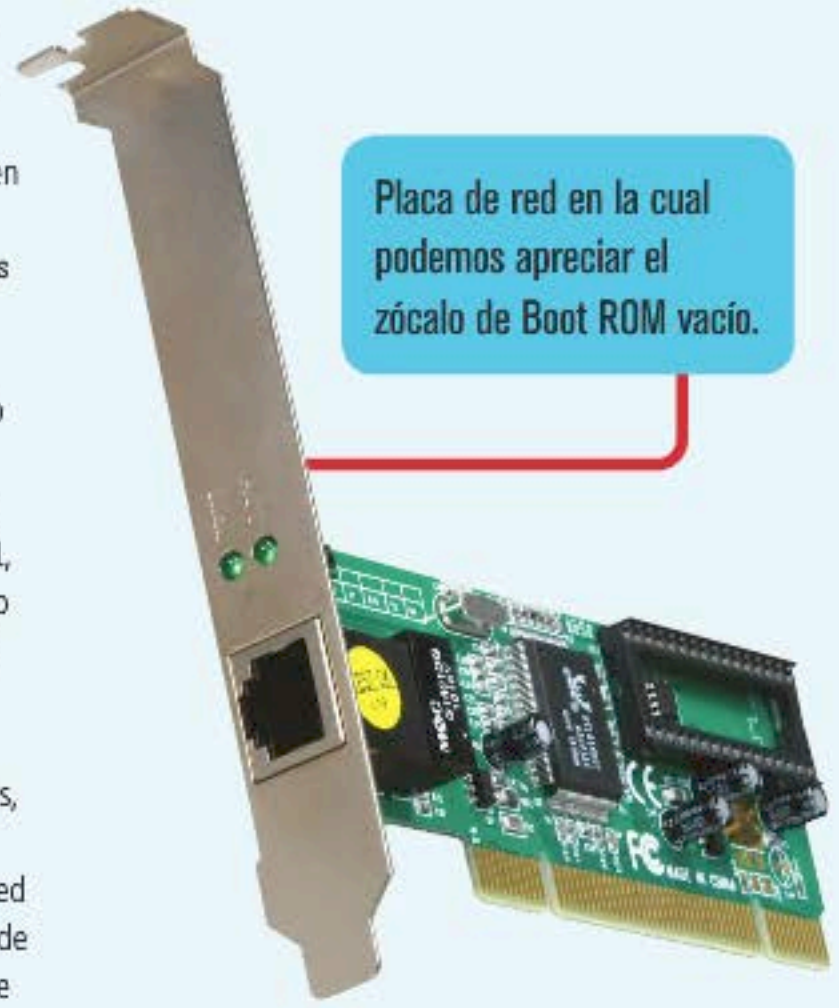
Necesitaremos el software **TFTPD32** (o **TFPTD64**) para cubrir las necesidades de servidor DHCP y TFPT. Esta aplicación es gratuita y puede descargarse desde este vínculo: **http://tftpd32.jounin.net/tftpd32_download.html**. Además, precisamos otros archivos, como **pxelinux.0**; **memdisk** (archivo sin extensión), que se obtiene de paquetes como **SYSLINUX** (**www.kernel.org/pub/linux/utils/boot/syslinux/syslinux-4.06.zip**); y dos archivos de configuración, cuyos contenidos detallaremos más adelante.

El programa TFTP32 debe ejecutarse en el equipo que hará las veces de servidor.

En esta imagen podemos ver una tarjeta Ethernet con el zócalo de Boot ROM ocupado por un chip EEPROM.

En realidad, este software no requiere instalación, por lo que, simplemente, descomprimimos el archivo que descargamos en la ubicación **C:\pxe**. También extraemos los archivos necesarios mencionados anteriormente (**pxelinux.0** y **memdisk**) en el mismo directorio. Ingresamos en esa carpeta y ejecutamos el archivo **tftpd32.exe**. En caso de poseer más de una interfaz de red, en el campo **Server interface** seleccionamos la que nos une con el o los equipos que bootearán vía red. Para este ejemplo, utilizamos una placa de red con una dirección Clase A (10.0.0.10), pero es posible usar una de Clase B o C, como la clásica 192.168.0.1. En ese caso, debemos realizar el reemplazo de los dos primeros octetos en las direcciones: **192.168.** por **10.0**. Ingresamos en el botón inferior **Settings** y nos dirigimos a la solapa **DHCP**. En el campo **IP pool starting address** colocamos el número de IP que da inicio al rango de asignación de direcciones por DHCP (debe ser distinto de la IP de nuestra interfaz de red). En **Size of pool**, ponemos un valor de **10**, que es el rango de direcciones IP que asignará el DHCP. En **Boot File**, escribimos **pxelinux.0** (es el bootloader incluido en el zip de SYSLINUX). En los dos campos siguientes, **DNS/WINS Server** y **Default router**, ingresamos la dirección de la placa de red local (en caso de compartir la conexión de Internet) o bien la de la puerta de enlace predeterminada o router de la red local, si es que hay uno. En el cuadro **Mask**, escribimos **255.0.0.0** para una dirección IP de Clase A o **255.255.255.0** para una de Clase C. Ahora debemos dirigirnos a la solapa **TFTP**. Activamos las casillas **PXE Compatibility** y **Use tftpd32 only on this interface**, y también seleccionamos la dirección IP de misma placa que antes (en este caso, 10.0.0.10).

Al aceptar los cambios, volvemos a la solapa **DHCP Server** del menú principal, donde pulsamos en el botón **OK**.



Placa de red en la cual podemos apreciar el zócalo de Boot ROM vacío.

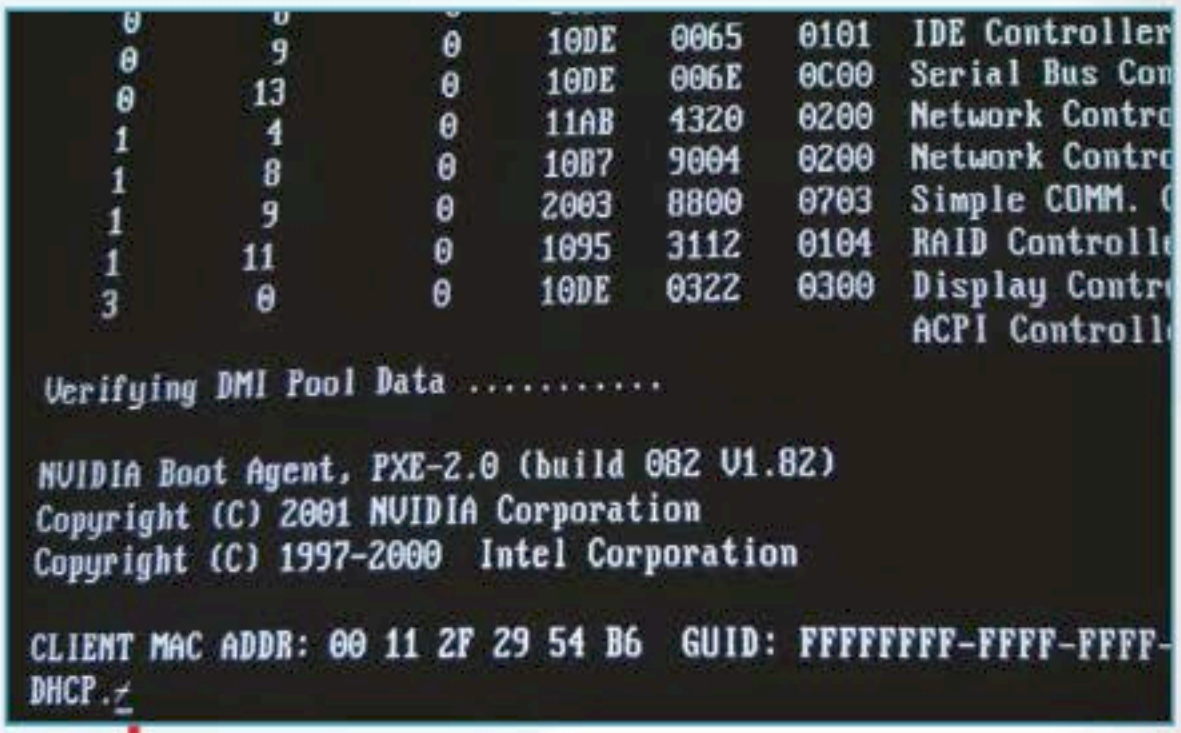
Archivos de configuración

Dentro de la carpeta C:\pxe, creamos un archivo de texto plano llamado menu (sin extensión) con el siguiente contenido (a modo de ejemplo, según las necesidades de cada administrador y de cada caso):

```
Menú de PXE / Booteo Remoto:
Seleccione una opción de arranque:
  1: Disco de inicio Windows 95
  2: Disco de inicio Windows 98
  3: Ghost
  4: PartitionMagic
  5: RIP (Linux)
  6: F-Prot Antivirus (DOS)
  7: Data LifeGuard Tools
```

Ese será el menú que aparecerá ante nuestros ojos al bootear vía red.
Los números asociados a cada opción de arranque servirán como referencia para bootear una u otra opción, pulsando en el teclado el que corresponda.
Además, dentro de la misma carpeta C:\pxe, creamos un directorio con el nombre pxelinux.cfg, en el cual guardamos un archivo de texto plano bajo el nombre default, con el siguiente texto:

```
PROMPT 1
DEFAULT 4
TIMEOUT 200
DISPLAY menu
label 1
KERNEL memdisk
APPEND initrd=01win95.img
label 2
KERNEL memdisk
APPEND initrd=02win98.img
label 3
KERNEL memdisk
APPEND initrd=03ghost.img
label 4
KERNEL memdisk
APPEND initrd=04pqmagic.img
label 5
KERNEL memdisk
APPEND initrd=05riplinux.img
label 6
KERNEL memdisk
APPEND initrd=06fprot.img
label 7
KERNEL memdisk
APPEND initrd=07dlg.img
```



Equipo iniciando a través de la placa de red, a la espera de la asignación de dirección IP.

Luego de crear y guardar estos archivos en sendas carpetas, guardamos las imágenes de disquetes en la ubicación C:\pxe, en formato IMG o IMA. Para generar las imágenes de disco booteables existe un software llamado **WinImage**, con el cual se podrán leer los disquetes de arranque que se necesiten y se guardarán en la carpeta mencionada como archivos IMA. La versión 8 de **WinImage** puede descargarse desde **www.winimage.com**, y probarse por un período de 30 días (trialware).

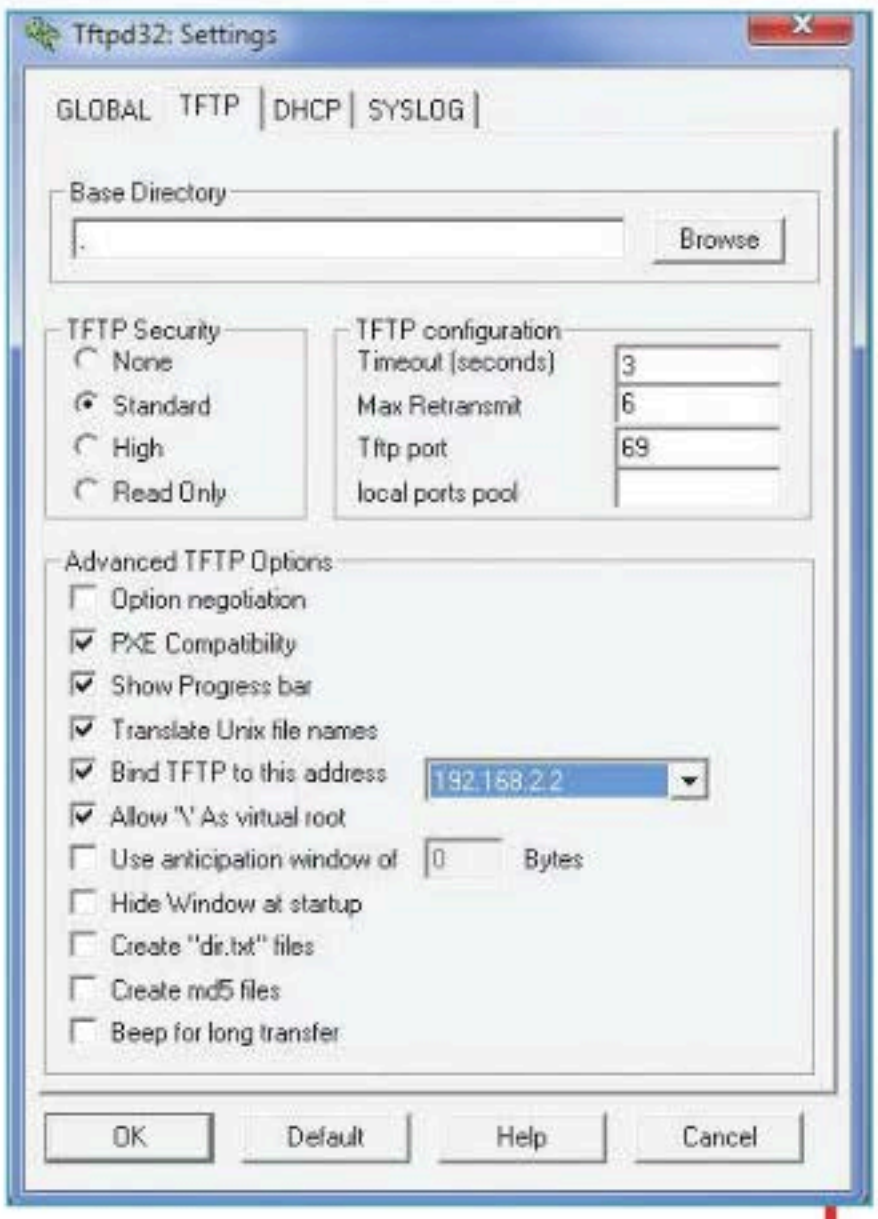
Imágenes booteables

La forma más sencilla de crear archivos booteables es con la aplicación **WinImage**, porque es capaz de leer disquetes booteables y guardar un archivo de imagen en formato IMA. También admite cambiar su contenido agregando, quitando o modificando archivos, etc.
Para este ejemplo, utilizamos las imágenes de disquetes booteables de Windows 95, 98 y Me. Si ya no utilizamos estos sistemas para generar esos discos de inicio, podemos descargar las imágenes booteables de Internet.
Este menú y estas aplicaciones se presentan a modo de ejemplo, pero podemos buscar nuestros programas preferidos de diagnóstico y rescate, generar el correspondiente disquete de arranque y capturar la imagen con **WinImage** para guardarla como archivo. Esta herramienta es muy fácil de utilizar: solo debemos insertar el disquete en la unidad, ir al menú Disk y elegir la opción Read disk. Una vez copiado a memoria, guardamos el archivo resultante con la función Save del menú File. Todos los archivos imagen deberán guardarse dentro de la carpeta C:\pxe.

SYSLINUX

SYSLINUX es un **bootloader** para iniciar todo tipo de sistemas. Su nombre es un tanto engañoso, ya que esta solución no se emplea habitualmente para bootear distribuciones GNU/Linux.
En estas páginas, por razones de espacio, se desarrolla un menú basado en modo texto, pero este poderoso y versátil entorno permite mostrar las diferentes opciones de booteo en red mediante un vistoso y llamativo menú en modo gráfico, con interesantes funciones para aplicar en nuestro trabajo. Este entorno de booteo ofrece una amplia posibilidad de scripting para ejecutar varios módulos.
SYSLINUX está formado por varios módulos, cada uno de ellos encargado de resolver distintas cuestiones, que vemos a continuación:

- **MEMDISK**: dispara la carga de antiguas imágenes de disco, como discos de arranque de DOS, Win9x y similares.



Captura de pantalla correspondiente a la configuración DHCP del menú de TFTPD32.

▶ **ISOLINUX**: es el módulo encargado de bootear imágenes ISO, muy útil para iniciar aplicaciones de rescate de sistemas operativos, discos de emergencia, etc.

▶ **EXTLINUX**: se ocupa de iniciar entornos Linux basados en sistemas de archivo ext2, ext3, ext4, etc.

Scripting con SYSLINUX

Opcionalmente, podemos mejorar un poco la apariencia de nuestro menú de booteo PXE. Luego del encabezado del script, una serie de comandos definen el aspecto del menú. Por ejemplo, veamos el siguiente fragmento de script:

```
MENU TITLE Menu de booteo por red
MENU WIDTH 63
MENU MARGIN 1
MENU ROWS 35
```

La primera línea muestra en pantalla un nombre para el menú. La segunda define el ancho, en caracteres, que tendrá la pantalla (el máximo es 80). La tercera especifica el margen, en caracteres, que tendrá el texto, contando a partir del borde izquierdo. La cuarta línea indica el número de filas que podrá aparecer en pantalla, con un máximo de 35.

Asignar colores en el menú de SYSLINUX puede ser un tanto complejo al principio, pero es muy versátil: contamos con una paleta de 16 millones de colores con un canal alfa para controlar las transparencias a gusto. Veamos ahora un ejemplo del comando `MENU COLOR`:

```
menu color title      1;37;44  #00000000 #00000000 none
```

Los códigos numéricos separados por ";" corresponden al formato y el estilo. Los códigos numéricos ubicados a continuación corresponden a dígitos hexadecimales de: canal alpha (opacidad), rojo, verde y azul, respectivamente (`#AARRBBGG`). Por último, hay un modificador que define la sombra del texto y tiene cuatro posibles opciones: `none` (sin sombra), `std` (sombra estándar), `all` (sombra frontal y de fondo) y `rev` (sombra invertida).

Tarea cumplida

Cada usuario escogerá las herramientas booteables esenciales según su necesidad, siguiendo los ejemplos aquí provistos. Los nombres de estas deben escribirse en el archivo `menú`, mientras que los archivos de imagen y sus nombres deben declararse en el archivo default, ubicado en el directorio `pxelinux.cfg`.

Ejecutamos **TFTP32**, que estará listo para servir a otro u otros equipos de la red. En ellos, tendremos que ingresar en el Setup del BIOS y, en la secuencia de arranque, establecer la opción LAN como primer método de booteo. Al encender el equipo, la placa de red buscará un servidor DHCP durante unos segundos; al encontrarlo, TFTP32 le asignará su dirección y le enviará la información de inicio, que presenta en pantalla el menú de opciones de arranque, de donde podremos elegir la necesaria en cada caso.

Placas de red sin Boot ROM

Ciertas placas de red no poseen la función PXE integrada y/o no tienen su chip de Boot ROM en el zócalo para tal fin. En ese caso, es posible emular ese firmware mediante un disquete.

Desde **http://rom-o-matic.net/gpxe/gpxe-1.0.1/contrib/rom-o-matic**, indicamos nuestro modelo de placa de red y seleccionamos el formato de archivo de disquete booteable, el cual, una vez descargado, deberá escribirse en el disquete mediante **rawwrite** (**http://nosetup.org/programa/113**). ■

Algunas placas de red no poseen zócalo Boot ROM, pero traen un diminuto chip que cumple esa función.

Conocer los protocolos IP: IGMP e ICMP

IGMP e ICMP son los protocolos que hacen posible la comunicación, y en ocasiones, nos ayudan a resolver problemas.

Los protocolos denominados **IGMP** e **ICMP** pertenecen a la capa de red y utilizan el protocolo IP para enviar los datos requeridos. A lo largo de estas páginas, vamos a realizar la revisión de cada una de sus características y alcances, así como también la forma correcta de configurar su funcionamiento para lograr los mejores resultados y aprovechar su potencial en la resolución de problemas.

Protocolo IGMP

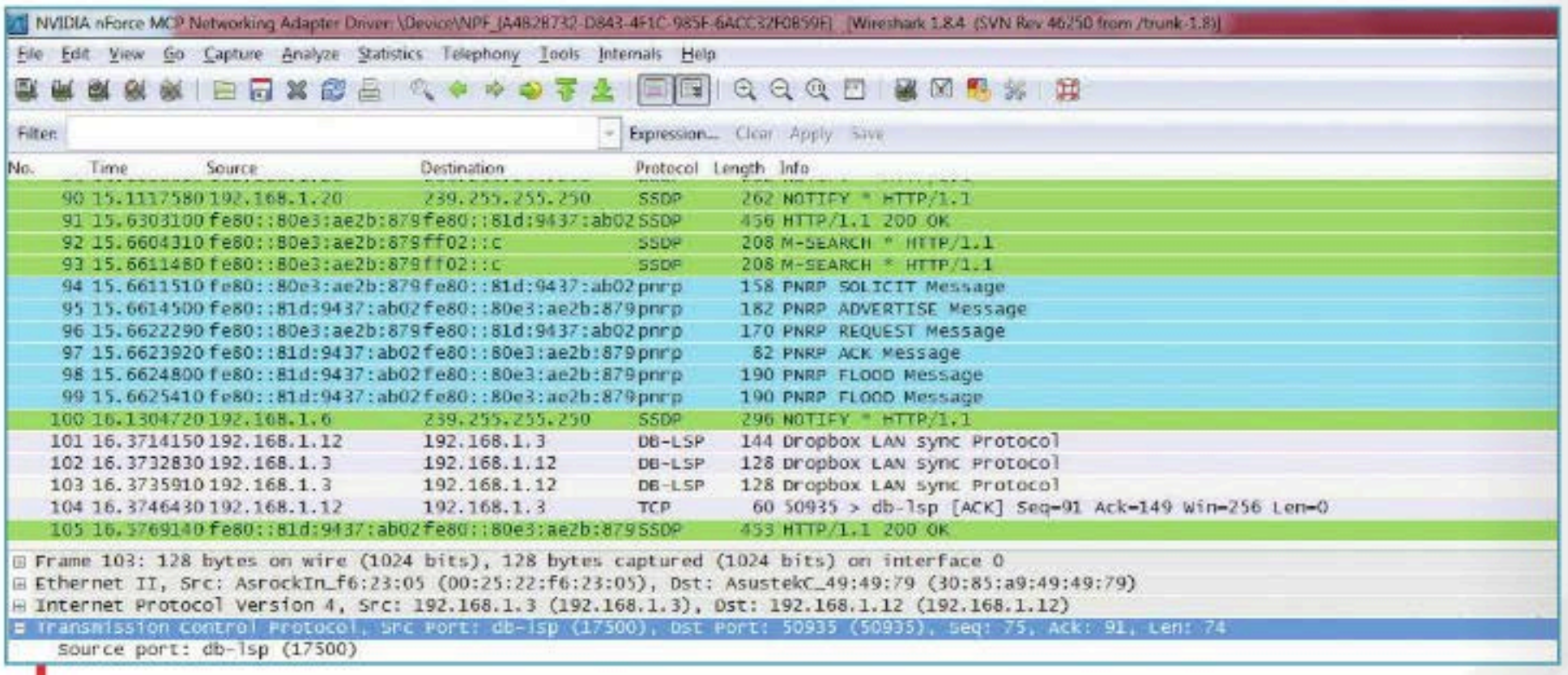
El protocolo de **administración de grupo de Internet** define cómo se procesarán los paquetes IP de multidifusión, aquellos datos que serán recibidos por un grupo de hosts.

El proceso de **multidifusión** es la transmisión de un datagrama IP a un grupo de hosts identificados por una sola dirección IP de destino. Se trata de una dirección IP especial, utilizada para tal fin. A su vez, tampoco se garantiza que los datagramas transmitidos sean recibidos por todos los miembros del grupo de multidifusión. Lo permanencia de un host en el grupo es dinámica: pueden unirse o dejar el grupo en cualquier momento; incluso, para enviar datos por multidifusión, no es necesario que pertenezcan al grupo de destino. Un host que tiene más de una placa de red puede pertenecer a varios grupos de multidifusión según la cantidad de placas de red que tenga instalada.

Para pertenecer a un grupo, el host envía su petición como mensajes IGMP. El protocolo IGMP utiliza un método de pregunta y respuesta para definir la pertenencia. Los **routers multidifusión** envían consultas determinando el tiempo por medio de datagramas multidifusión, recibidos por los hosts miembros.

Protocolo ICMP

El protocolo de **control de mensajes de Internet** también funciona en conjunto con el protocolo IP. A veces suele confundírselos, ya que cualquier sistema que ofrezca soporte para IP en general también lo tiene para ICMP. El protocolo



WireShark nos ofrece datos precisos sobre un paquete TCP, como sus puertos de origen y destino, y tamaño del paquete.

ICMP utiliza mensajes de control y error durante el proceso de transmisión de paquetes IP, para ayudarnos a resolver algunos problemas de red. El formato de mensaje de ICMP está compuesto de la siguiente manera:

- **Tipo**: indica el tipo de mensaje o error; por ejemplo, destino inalcanzable, respuesta de eco, tiempo excedido, etc.
- **Código**: información adicional específica sobre el tipo de mensaje.
- **Suma de control**: similar a la comprobación que realiza IP.
- **Misceláneos**: se trata de información utilizada por diferentes mensajes; incluso, puede no ser usado, en cuyo caso su valor queda en 0.
- **Cabecera y datos**: contiene la cabecera del datagrama IP, y los primeros 64 bits de datos del paquete enviado.

Podemos considerar a ICMP como un protocolo de preguntas y respuestas, que antes de realizar el envío, comprueba si se puede enviar datos y espera la respuesta para decidir qué acción tomar. Algunos tipos de mensaje que utiliza ICMP son los siguientes:

0: respuesta de eco
1 – 2: sin asignar
3: destino inalcanzable
4: disminución del tráfico desde el origen
5: redireccionar (cambio de ruta)
8: eco
11: tiempo excedido
12: problema de parámetros
30: traza de ruta

Al ingresar una IP no válida, vemos un error de servidor no encontrado. No es el error 404, de página no encontrada.

Una vez conocido el tipo de mensaje, el código de mensaje nos provee de información específica con la cual podremos determinar cuál es el problema y buscar su solución. Algunos de los códigos utilizados por ICMP son los siguientes:

0: red inalcanzable
1: host inalcanzable
2: destino no dispone el protocolo solicitado
3: no se puede llegar al puerto de destino
4: se aplica defragmentación, pero está indicado no fragmentar
5: ruta de origen incorrecta
6: red de destino desconocida
7: host de destino desconocido
8: host de origen aislado
9: comunicación con la red de destino está prohibida por reglas administrativas
10: comunicación con el host de destino está prohibida por reglas administrativas
11: red de destino inalcanzable por el tipo de servicio
12: host de destino inalcanzable por el tipo de servicio
13: comunicación prohibida por reglas administrativas

Cómo funciona ICMP

ICMP envía mensajes cuando detecta un error, el más común de los cuales es el destino inalcanzable. Cuando ejecutamos **ping x.x.x.x**, estamos probando la conexión. Si el host remoto acepta, recibiremos la respuesta de que los paquetes han sido enviados. Si la IP a la que pedimos la consulta tiene bloqueadas las peticiones de eco, o no existe, el destino será inalcanzable.

Un concepto importante es el de **MTU** (*Maximum Transmission Unit*, o unidad máxima de transmisión). El MTU está relacionado con la capa física de datos. Por ejemplo, Ethernet acepta MTU de 1500 bytes; si de otro dispositivo recibe una MTU mayor, se lo fragmentará en partes pequeñas que sí puede soportar su MTU. Para evitar la fragmentación se usa *Path MTU Discovery*. Su función es enviar un paquete con la opción de no ser fragmentado; cuando un dispositivo en la red detecte que su MTU es mayor, como no podrá fragmentarlo, responderá con un mensaje de ICMP. Este proceso se repite con menores tamaños de MTU, hasta que todos los dispositivos lo acepten. ■

Comando ping

Recordemos que el comando **ping** es una petición de eco, que envía una consulta y espera una respuesta. A su vez, este comando es de capa 3. Supongamos que tenemos instalado un firewall en nuestro equipo y hemos bloqueado varias páginas de Internet. Si realizamos **ping** a una de ellas, responderá que el destino es alcanzable; y si deseamos ver la página web desde nuestro navegador, no podremos hacerlo, porque este utiliza protocolos de capa 7.

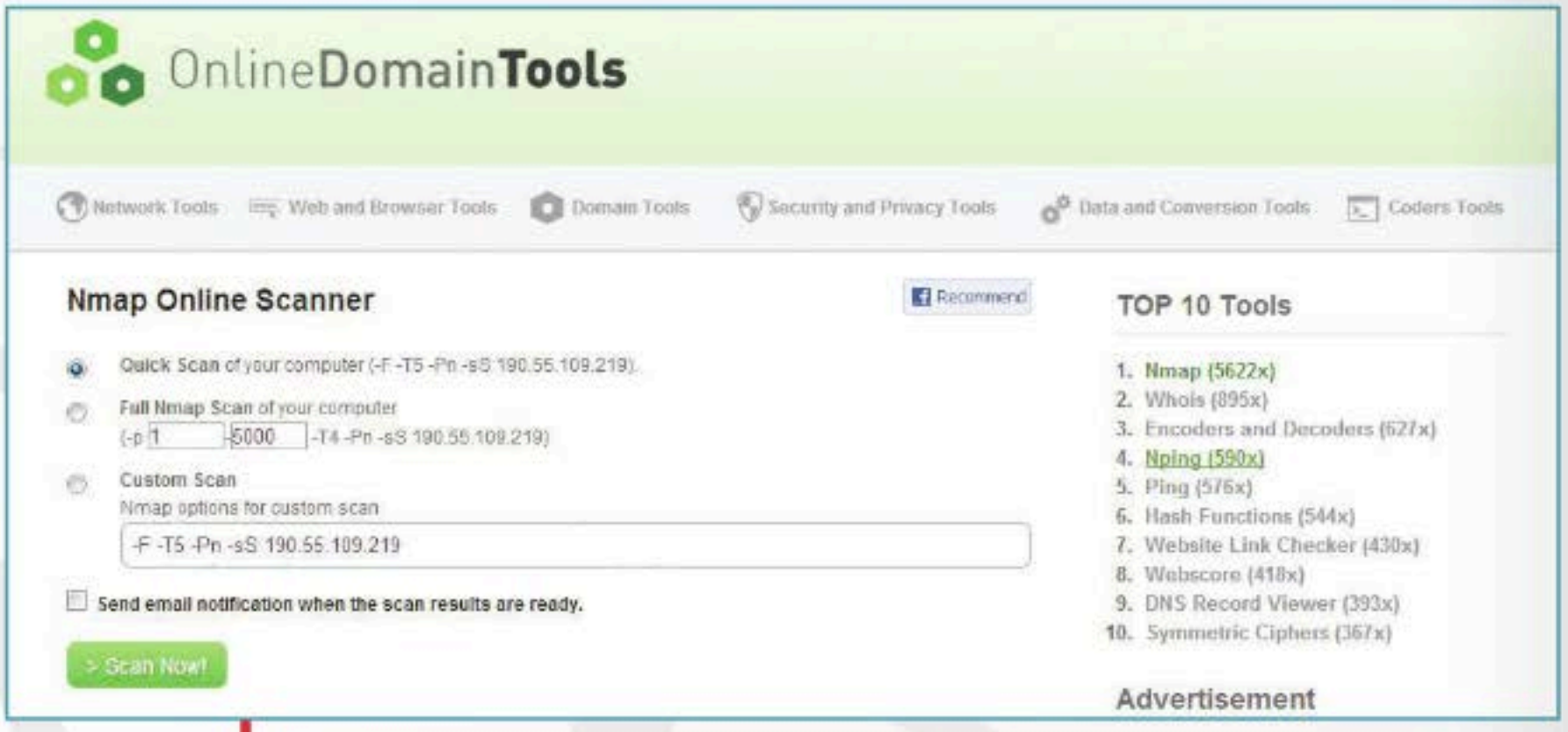
Seguridad: TCP handshake y headers

El diseño de los protocolos TCP/IP y sus métodos de funcionamiento son muy importantes en seguridad de redes, aquí los conoceremos.

El funcionamiento de los protocolos TCP/IP es de suma importancia en la seguridad informática, pues muchos problemas se han heredado de su diseño original. Además, las herramientas realizan tareas específicas que se basan en la manipulación de paquetes y encabezados (**headers**), de manera tal que puedan obtenerse las respuestas esperadas en caso de utilizar técnicas activas. Dado que muchos fabricantes utilizan los protocolos TCP/IP sin ajustarse estrictamente a sus especificaciones, en ciertas oportunidades surgen anomalías, que pueden ser reconocidas por los sistemas de detección de intrusiones (IDS) o detectadas por los analizadores de protocolos, como **Wireshark**.

Saludo

Recordando algunos conceptos básicos sobre TCP, para iniciar una comunicación tenemos el conocido **saludo de tres vías de TCP** o **TCP 3-way handshake**. Se trata de un método a partir del cual dos dispositivos se ponen de acuerdo para establecer una conexión entre ellos. Para hacerlo, utilizan una serie de campos de la cabecera TCP: por un lado, se usan los flags **TCP SYN** y **TCP ACK**; y por otro, los campos correspondientes a los número de secuencia y de acuse de recibo. Como primer paso, el cliente envía una petición de conexión al servidor activando el flag **SYN** y transfiere un número de secuencia generado en forma pseudoaleatoria (**seq=x**). En caso de que el puerto no esté abierto en el servidor, este le envía un paquete al cliente con el flag **RST** activado, para indicar el rechazo del intento de conexión. Como segundo paso, si del lado del servidor el puerto está abierto, este responderá a la petición **SYN** válida con un paquete que tenga el flag **SYN** y el **ACK** activado, además de incluir en el campo de acuse de recibo el número de secuencia del cliente incrementado en 1 (**ack=x+1**), y en el campo de número de secuencia, uno nuevo, esta vez, generado de modo pseudoaleatorio por el servidor (**syn=y**).



Interfaz online para NMAP, para ejecutar escaneos sin tenerlo instalado localmente.

Para finalizar, como tercer y último paso, el cliente responderá con un paquete que posea el flag ACK activado, y en el campo de acuse de recibo, el número de secuencia del servidor incrementado en 1 (seq=y+1). También es necesario finalizar la conexión TCP de manera correcta, siguiendo los pasos preestablecidos por el estándar, en cuatro etapas. Primero, el host A indica que quiere finalizar la conexión enviando un paquete con el flag FIN activo. El host B responde con un paquete con el flag ACK para confirmar la recepción, y al terminar de enviar los paquetes pendientes, da por finalizada la conexión. Luego, este manda un paquete con el flag FIN levantado al host A, que responderá con un paquete ACK para confirmar la recepción. Así, ambos dan por concluida la conexión.

Flags

En pocas palabras, podemos resumir la función de cada uno de los flags de la siguiente manera:

- **SYN** (*Synchronization*): utilizado para indicar el intento de una nueva conexión.
- **ACK** (*Acknowledgement*): además de los datos que pueda contener el paquete, sirve como confirmación de un paquete anterior.
- **FIN** (*Finalization*): indica que se desea cerrar la conexión y se queda a la espera de que el otro host esté listo para hacerlo.
- **URG** (*Urgent*): indica que el paquete contiene datos urgentes (se trata de una función que no es muy usada).
- **PSH** (*Push*): indica que se debe vaciar el buffer de transmisión o recepción y pasar los datos a la pila.
- **RST** (*Reset*): le dice al otro extremo de la conexión que ha habido algún tipo de problema con la sincronización de la conexión, y que se cerrará.

El ejemplo más concreto de la manipulación deliberada de los encabezados se encuentra en los escáneres de puertos, que utilizan las más diversas técnicas para determinar si un sistema está disponible (**pingsweep** o barrido de ping) o, fundamentalmente, para saber si un puerto está abierto. Además, realizan otras funciones, como la detección del sistema operativo, la recopilación de leyendas de aplicaciones (**banner grabbing**), y demás.

Estado de los puertos

El hecho de que un puerto esté abierto implica que el equipo objetivo acepta peticiones de él. Está filtrado cuando un firewall u otro dispositivo de red lo enmascara, y previene que se determine si está abierto o no. Finalmente, se encuentra cerrado cuando el puerto no admite conexiones, es decir, responde con un paquete TCP que tiene habilitado el flag RST. Las técnicas de escaneo surgen a partir de la activación de uno o varios flags de la cabecera TCP.

La manera más sencilla de identificar el estado de un puerto —es decir, de saber si está abierto, cerrado o filtrado— es tratando de conectarse a él. Si está abierto, según lo visto a partir del 3-way handshake, responderá un ACK; si está cerrado, responderá un RST; en tanto que si está filtrado, no se recibirá ningún tipo de respuesta. En caso de que el puerto esté abierto, continuamos con los otros dos pasos y establecemos la conexión.

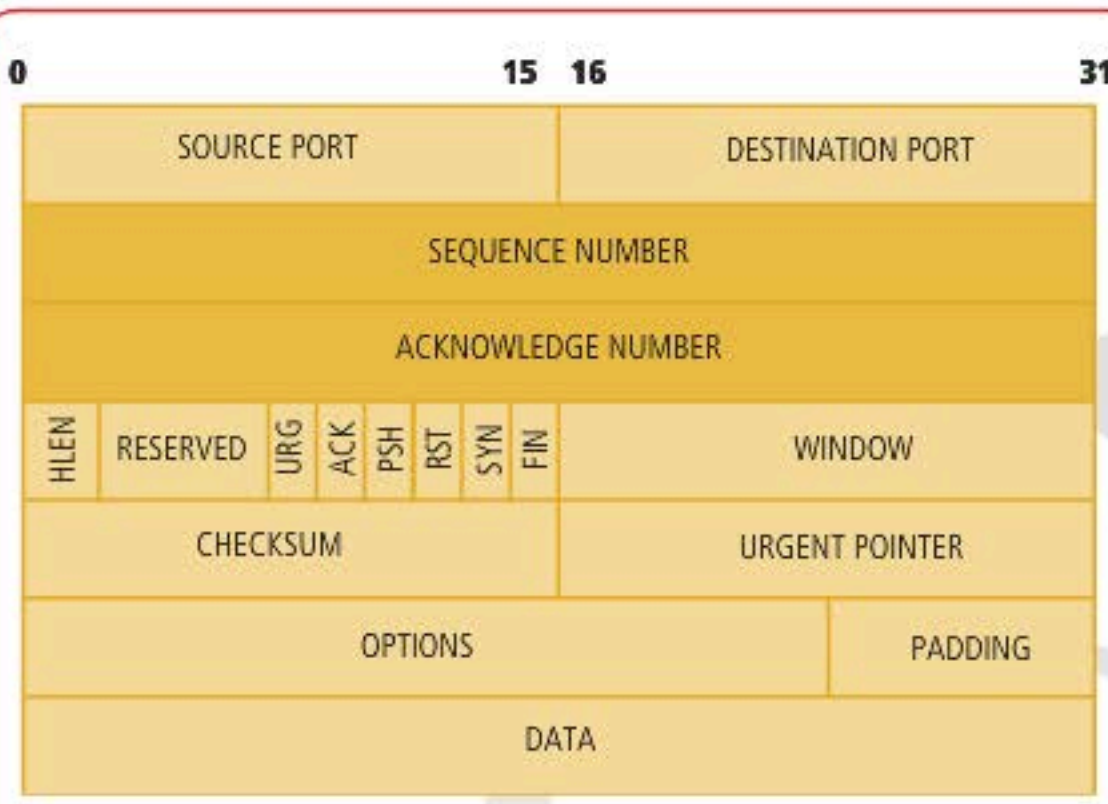
Este escaneo es conocido como **TCP Connect**. Si bien esta forma es válida y efectiva, desde el punto de vista del atacante, es muy ruidosa, porque deja muchos registros en el objetivo y es fácilmente detectable.

Por todo lo mencionado, existen variantes a dicho escaneo, y la más conocida es **SYNScan**. La diferencia con la anterior es que, en vez de responder el último paso con un paquete que tenga el flag ACK activado y finalmente establecer la conexión, envía un RST de modo tal de cortar la conexión.

Este nuevo escaneo deja menor cantidad de registros en el objetivo.

Otras técnicas

Otro tipo de escaneo es el llamado **FINscan**, también conocido como



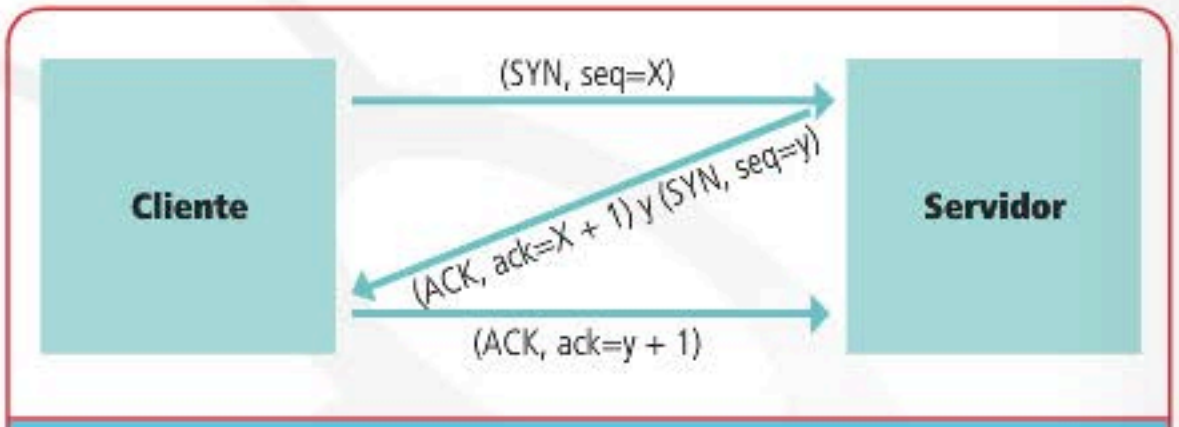
En este diagrama vemos la cabecera del protocolo TCP. Los campos implicados en el **TCP handshake** son el número de secuencia, el número de acuse de recibo, y los flags SYN y ACK.

Stealthscan, por ser muy sigiloso y discreto entre las técnicas más comunes. Se basa en el envío de un paquete FIN al puerto del sistema remoto (de ahí su nombre). El estándar TCP/IP indica que, al recibir un paquete FIN en un puerto cerrado, se debería responder con un paquete RST. Entonces, si recibimos RST por respuesta, implica que el puerto está cerrado, y en caso de no recibir respuesta (o sea, que se ignore el paquete FIN), el puerto podría encontrarse abierto. En los sistemas Windows, un puerto cerrado ignora los paquetes FIN, por lo que este escaneo dará resultados erróneos.

Otra técnica es **UDPscan**, que no se basa en TCP sino en UDP, y consiste en mandar un paquete UDP vacío (0 bytes de datos) al puerto en cuestión. Si este se encuentra cerrado, el sistema responderá con un paquete ICMP de tipo 3 (destino inalcanzable). En caso de no responder, el puerto podría estar abierto.

LA TÉCNICA DE ESCANEO DE PUERTOS SIRVE COMO BASE PARA LA POSTERIOR FASE DE ESCANEO DE VULNERABILIDADES EN UN SISTEMA REMOTO.

También existe **ACKscan**, orientado a identificar puertos filtrados, en estado "silencioso", o equipos detrás de un firewall que bloquee los intentos normales de conexión (SYN). Se basa en el envío de paquetes ACK con números de secuencia y confirmación aleatorios, de manera que, al recibir el paquete, si el puerto se encuentra abierto, responderá con un RST; y si está cerrado, con un RST. Si no se obtiene respuesta estando el equipo en línea, significa que el puerto está filtrado. Por su parte, el escaneo llamado **Nullscan** se basa en enviar un paquete con todos los flags desactivados. Si el puerto está cerrado, responderá un RST; si no recibe nada como respuesta, se trata de un puerto abierto o filtrado.



Esquema de los tres pasos del 3-way handshake, o saludo de tres vías.

Otros protocolos

Otro de los protocolos de la capa de transporte es UDP (*User Datagram Protocol*), mucho más simple que TCP y no orientado a conexión. No tiene flags como TCP. El último protocolo interesante para mencionar en este apartado es ICMP (*Internet Control Message Protocol*), que trabaja también en capa 4 y está orientado a control de errores. Normalmente, un paquete ICMP se utiliza para avisar de eventos. Por ser TCP un protocolo orientado a conexión, tiene estados definidos según el momento de conexión en que se encuentre el socket. Estos estados son:

- **Listen:** espera de conexiones y escucha de un puerto.
- **Syn-Sent:** se produce al enviar un paquete SYN y comenzar el handshake.
- **Syn-Received:** se produce en el segundo paso del handshake.
- **Established:** se produce al completarse el handshake y permanece en él durante el tiempo que dura la conexión.
- **Fin-Wait-1:** se produce cuando se envía FIN pero no se ha confirmado.
- **Fin-Wait-2:** se produce cuando se recibe la confirmación del FIN.
- **Close-Wait:** cuando se recibe el paquete FIN pero todavía hay datos para enviar.
- **Closing:** se produce cuando dos hosts quieren finalizar la conexión a la vez.
- **Last-Ack:** ocurre cuando el último en haber enviado el paquete FIN pendiente de confirmación entra en estado Last-Ack.
- **Time-Wait:** ocurre cuando, una vez enviados los paquetes FIN, el primero manda la confirmación y entra en estado Time-Wait, para esperar la recepción. ■

PRÓXIMA ENTREGA

7

INSTALACIÓN DE REDES INALÁMBRICAS

En el próximo número En el próximo número aprenderemos a realizar la instalación de redes inalámbricas y a configurar interfaces Wi-Fi. También revisaremos las opciones de seguridad.

Técnico en
REDES &SEGURIDAD

PROFESORES EN LÍNEA
profesor@redusers.com

SERVICIOS PARA LECTORES
usershop@redusers.com

SOBRE LA COLECCIÓN

CURSO VISUAL Y PRÁCTICO QUE APORTA LOS SABERES NECESARIOS PARA FORMAR TÉCNICOS EXPERTOS EN REDES Y SEGURIDAD. INCLUYE UNA GRAN CANTIDAD DE RECURSOS DIDÁCTICOS COMO INFOGRAFÍAS, GUÍAS VISUALES Y PROCEDIMIENTOS REALIZADOS PASO A PASO.

Con la mejor metodología para llevar adelante el montaje y mantenimiento de las redes informáticas y con los aspectos clave para brindarles la protección necesaria, esta obra es ideal para aquellos aficionados que deseen profundizar sus conocimientos y para quienes quieran profesionalizar su actividad.

REDES &SEGURIDAD 00006
9 789871 857784

CONTENIDO DE LA OBRA

1 Introducción a las redes informáticas
2 Tipos de redes y topologías
3 Dispositivos de red
4 Instalación de redes cableadas
5 Puesta en marcha de una red cableada
6 CONFIGURACIÓN DE REDES CABLEADAS
7 Instalación de redes inalámbricas
8 Configuración de redes inalámbricas
9 Seguridad en redes cableadas e inalámbricas
10 Configuración avanzada de routers
11 Recursos compartidos y dispositivos multimedia
12 Seguridad física de la red
13 Impresoras de red
14 Hardware de servidores
15 Administración de Windows Server
16 Administración de sistemas Linux
17 Administración y asistencia remota
18 Servidores web y FTP
19 Servidores de mail
20 Servidores de archivos e impresión
21 Servidores adicionales
22 VLAN, VPN y trabajo remoto
23 Telefonía IP
24 Cámaras IP