Seguridad

IP es un protocolo sin conexión, por lo tanto, carece de seguridad en la entrega de paquetes. Cuando una comunicación que utiliza el protocolo IP necesita seguridad en la transferencia de paquetes de datos, esta debe ser proporcionada por otro protocolo de capa superior.

Ampliación

Cada una de estas funciones da origen a una subcapa, la primera función es propia de la subcapa de control de acceso al medio o **MAC** *(Media Access Control)*, la segunda lo es de la subcapa de control de enlace lógico **LLC** *(Logical Link Control)*, aunque normalmente esta subcapa toma el nombre de la capa OSI que la incluye: enlace de datos o **DLL** *(Data Link Layer)*.

3. La familia de protocolos TCP/IP

Por su frecuencia de uso, debemos detenernos especialmente en los protocolos que constituyen esta familia, especialmente en el protocolo IP, en el nivel de red, y el protocolo TCP, en la capa de transporte. Hay muchos más protocolos, pero la importancia de estos dos ha hecho que a toda la arquitectura de protocolos utilizados tanto en sistemas UNIX, como actualmente en muchos otros sistemas, se le llame familia de protocolos TCP/IP.

3.1. Los protocolos básicos en TCP/IP

La arquitectura TCP/IP no se fija en el nivel 2 de OSI, lo asume en lo que llama nivel de red, pero las instalaciones habituales de redes TCP/IP utilizan redes Ethernet en el nivel 2 de OSI.

El nivel de enlace asegura una conexión libre de errores entre dos ordenadores de la misma red. Fundamentalmente organiza los bits en forma de tramas y los pasa a la capa física para que sean transmitidos al receptor a través del medio de transmisión.

Cabe distinguir dos funciones en esta capa:

- Como en muchas redes de área local los canales están compartidos por muchos nodos, ¿cómo saber que el canal está libre? Y si lo está, ¿cómo sabe un nodo si puede o no apropiarse de los recursos de la red?
- Puesto que los bits deben ser agrupados en tramas, ¿cómo confeccionar esas tramas? Además, ¿cómo saber si las tramas recibidas son correctas?

Aunque TCP/IP no sigue la arquitectura OSI, se pueden establecer paralelismos como los que aparecen en la Fig. 3.10.

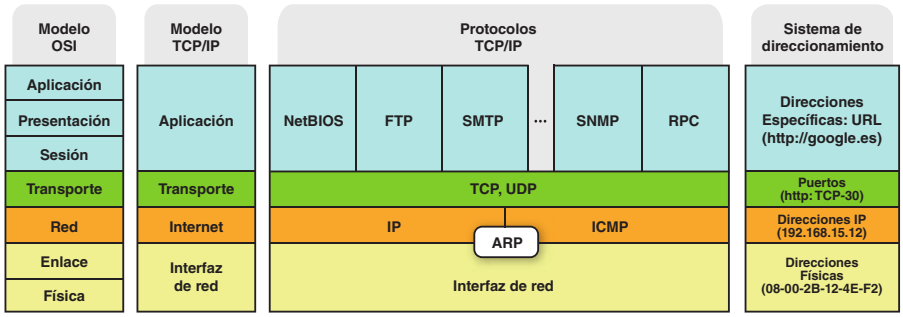

Fig. 3.10. *Estructura de capas de la arquitectura TCP/IP y su relación con OSI. Se especifican algunos ejemplos de protocolos en cada capa y un ejemplo del sistema de direccionamiento utilizado en cada nivel.*

CEO

SMR_RL_AAbad_03_NivelAccesoMedio.docx
Documento que contiene información detallada sobre funciones y protocolos utilizados en el nivel 2 de OSI.

Vocabulario

Bloque de datos: conjunto de datos que posee una estructura interior perfectamente definida.
Segmento: es el bloque de datos definido en el nivel de transporte (nivel 4 de OSI).
Paquete: es el bloque de datos propio del nivel de red (nivel 3 de OSI).
Datagrama: es un tipo de paquete (nivel 3) utilizado en servicios de comunicaciones sin conexión.

A. Protocolo IP

IP (*Internet Protocol*) es el protocolo de nivel de red en ARPANET, el sistema de comunicaciones que tradicionalmente han utilizado los sistemas UNIX y que nació a principios de los años 80. Lo más relevante de IP para el administrador de red es que proporciona un sistema de direcciones para que cada nodo de la red quede identificado por una dirección de cuatro números enteros separados por puntos (o 32 bits) denominada dirección IP o de nivel 3, para distinguirla de la dirección MAC (física) o de nivel 2 que se compone de 12 dígitos hexadecimales.

El protocolo IP acepta **bloques de datos** procedentes de la capa de transporte (por ejemplo, desde el protocolo TCP que opera en el nivel de transporte) de hasta 64 Kbytes. Cada bloque de datos, que en este nivel se denominan **segmentos,** debe ser transferido a través de la red **(Internet)** en forma de **datagramas.** Para llevar a cabo este transporte, normalmente la capa de red debe fraccionar los datagramas en un conjunto de **paquetes** IP, que deben ser ensamblados en el destino para que el mensaje sea al final reconstruido con fidelidad. Al ser IP un protocolo sin conexión, cada paquete puede seguir una ruta distinta a través de la internet. El protocolo de capa superior (TCP) será el encargado de la gestión de errores.

B. Protocolo ICMP

ICMP (*Internet Control Message Protocol,* Protocolo de mensajes de control entre redes) es un protocolo que expresa en un único paquete IP algún evento que se produce en la red. Por tanto, se trata de un protocolo de supervisión. Cualquier red TCP/IP debe utilizar el protocolo ICMP.

En la dirección **http://es.wikipedia.org/wiki/Internet_Control_Message_Protocol** puede encontrarse el formato de los paquetes ICMP así como detalles del funcionamiento orgánico del protocolo.

Ampliación

En ICMP son posibles, entre otros, mensajes como los siguientes:

- Destino inalcanzable. Se utiliza cuando una subred se da cuenta de que no puede alcanzar otra red solicitada por un datagrama IP, o bien, es alcanzable, pero no en las condiciones especificadas en el paquete IP.
- Tiempo excedido. El campo contador del tiempo de vida de un paquete IP ha descendido hasta 0 y ha sido drenado (retirado) de la red.
- Problemas en parámetros. El valor asignado a un parámetro de una cabecera IP es imposible. Esto suele determinar un error en la transmisión o en las pasarelas de la red.
- Enfriar fuente. Este mensaje se envía a un transmisor para que modere la velocidad de transmisión de paquetes.

C. Protocolo TCP

TCP (*Transmission Control Protocol* o protocolo de control de transmisión) fue especialmente diseñado para realizar conexiones en redes inseguras. TCP es un protocolo de capa de transporte adecuado para proporcionar seguridad a IP.

La seguridad del protocolo TCP le hace idóneo para la transmisión de datos por sesiones, para aplicaciones cliente-servidor y para servicios críticos como el correo electrónico.

La seguridad en TCP tiene un precio que se manifiesta en forma de grandes cabeceras de mensajes, y de la necesidad de confirmaciones de mensajes para asegurar las comunicaciones. Estas confirmaciones generan un tráfico sobreañadido en la red que ralentiza las transmisiones en beneficio de la seguridad.

Los puntos de acceso al servicio (SAP de OSI) en la capa de transporte en TCP/IP se llaman **sockets** o conectores TCP/IP y son extraordinariamente útiles en la programación de aplicaciones de red.

Detrás de cada socket activo se implanta un servicio de red. Cuando alguien en la red requiere de ese servicio, manda mensajes al socket o puerto que identifica a ese servicio. Algunos servicios tienen necesidad de más de un socket para su funcionamiento. Por ejemplo, 80 es el puerto que identifica las peticiones de red hacia un servidor web.

D. Protocolo UDP

UDP (*User Datagram Protocol* o protocolo de datagrama de usuario) es un protocolo de transporte sin conexión, es decir, permite la transmisión de mensajes sin necesidad de establecer ninguna conexión y, por tanto, sin garantías de entrega. Actúa simplemente como una interfaz entre los procesos de los usuarios de la red y el protocolo IP. Se utiliza en transmisiones rápidas que no necesitan seguridad en la transmisión.

UDP no impone el uso de confirmaciones puesto que su objetivo no es la seguridad y esto hace de él un protocolo de transporte de mucho mayor rendimiento que TCP, y también más inseguro.

En la Tabla 3.1 se pueden observar algunas de las características que diferencian a TCP de UDP, a pesar de que ambos operan en el nivel 4 equivalente del modelo OSI o capa de transporte en el modelo TCP/IP.

TCP	UDP
Es un protocolo confiable	No es confiable
Orientado a la conexión	No establece una conexión inicial
Lleva gestión de las retransmisiones y control de flujo	No gestiona restransmisiones
Secuencia numéricamente los segmentos (paquetes de datos enviados o recibidos)	No gestiona un secuenciamiento de segmentos
Admite segmentos de acuse de recibo	No incorpora acuse de recibo

Tabla 3.1. *Diferencias sustanciales entre TCP y UDP.*

Ampliación

TCP acepta bloques de datos (TPDU, *Transport Protocol Data Unit*) de cualquier longitud, procedentes de las capas superiores o de los procesos de los usuarios, y los convierte en fragmentos de 64 Kbytes como máximo que pasa a la capa de red, quien a su vez puede volver a fraccionarlos para su transmisión efectiva. Cada uno de los bloques de datos –frecuentemente se les denomina **segmentos**– se transmite como si fuera un datagrama separado con entidad propia. TCP es el responsable de ensamblar los datagramas recibidos por el receptor, ya que la red IP puede desordenarlos al utilizar caminos diversos para alcanzar su destino. IP no garantiza que los datagramas lleguen a su destino, por lo que es necesaria una entidad superior (TCP) que se encargue de ello a través de un sistema de temporizadores y retransmisiones en caso de problemas.

Ampliación

También existe el protocolo RARP *(Reverse ARP)*, que es el protocolo inverso del ARP, es decir, localiza la dirección lógica de un nodo a partir de la dirección física del mismo. Fundamentalmente es utilizado en estaciones de trabajo sin disco, que han conseguido su sistema operativo a través de la red.

Vocabulario

Dirección MAC o **dirección física:** es la dirección lógica de una interfaz de red en el nivel 2. Se compone de 12 cifras hexadecimales.

Investigación

En **http://personales.upv.es/rmartin/TcpIp/cap02s01.html** tienes unas descripción de la familia de protocolos TCP/IP y de cómo se relacionan entre sí algunos de ellos. Interesa que leas este documento o alguno similar para que te habitúes a asociar correctamente los niveles de la familia de protocolos TCP/IP con los protocolos concretos que se utilizan en cada nivel. También puedes ayudarte de la página de Wikipedia localizada por la voz «familia de protocolos de Internet».

E. Protocolo ARP

ARP (*Address Resolution Protocol* o protocolo de resolución de direcciones) no es un protocolo relacionado directamente con el transporte de datos sino que complementa la acción del TCP/IP pasando desapercibido a los ojos de los usuarios y de las aplicaciones de la red.

Como el protocolo IP (equivalente al nivel 3 del modelo OSI) utiliza un sistema de direccionamiento que utiliza el sistema operativo que no tiene nada que ver con las **direcciones MAC** (nivel 2 OSI) que utilizan las tarjetas de las redes de área local, hay que arbitrar un mecanismo de asignación de direcciones IP (cuatro números separados por puntos) a direcciones MAC propias del nivel de enlace. De esto se encarga el protocolo ARP, que funciona del siguiente modo:

Cuando un host quiere transmitir un paquete IP necesita averiguar la dirección MAC del host destinatario cuya dirección es la dirección de destino del campo «dirección de destino» del paquete IP. Para ello genera un paquete de petición ARP que difunde por toda la red. Todos los nodos de la red detectan este paquete y solo aquel host que tiene la dirección IP encapsulada en el paquete ARP contesta con otro paquete ARP de respuesta con su dirección MAC. De este modo el host emisor relaciona dirección IP y dirección MAC, guardando estos datos en una tabla residente en memoria para su uso en transmisiones posteriores.

Puede encontrarse más información detallada sobre este protocolo en la dirección **http://es.wikipedia.org/wiki/Address_Resolution_Protocol.**

3.2. El direccionamiento de red en TCP/IP

El sistema de direccionamiento IP es muy peculiar y ampliamente aceptado por la comunidad mundial. Cada dirección IP consta de 32 bits agrupados en grupos de 8 bits. Una dirección IP se expresa con cuatro números decimales separados por puntos. Cada uno de estos números varía entre 0 y 255, aunque hay algunas restricciones. Un ejemplo de dirección IP sería 128.100.3.67.

A. Clases de subredes

Como IP es un protocolo pensado para la interconexión de subredes, cada dirección IP codifica una red y un host dentro de esa red. Atendiendo a los primeros bits de cada dirección se averigua el tipo de subred de que se trata (en cuanto a su volumen) y de su dirección concreta. Los bits restantes codifican el host de que se trata dentro de esa subred. De las cinco clases de subredes, solo tres sirven para el direccionamiento particular de los nodos de la red (Fig. 3.11):

Fig. 3.11. *Estructura de los bits para las direcciones IP de las redes de clase A, B, C y D. Las direcciones de clase E están reservadas para aplicaciones futuras o para uso experimental.*

- **Redes de clase A.** Se codifican la subred y los 24 restantes la identificación del host dentro de esa subred. Los valores posibles para la subred varían entre 1 y 126, que coincide con el valor del primer byte de la dirección, es decir, hay 126 subredes posibles de tipo A. Cada una de ellas puede contener 16.777.214 hosts distintos. Este sistema de direccionamiento se utiliza, por tanto, para subredes muy grandes.
- **Redes de clase B.** Se caracterizan porque los dos primeros bits de la dirección son 10. Los 14 bits siguientes codifican la subred, desde 128 a 191 para el primer byte de la dirección, por tanto, son posibles 16.384 subredes de tipo B. Cada una de estas subredes puede contener 65.534 hosts distintos, los codificados por los 16 bits restantes del campo de dirección.

- **Redes de clase C.** Se caracterizan por tener sus tres primeros bits con el valor 110. Los 21 bits siguientes codifican la subred y los 8 restantes el host dentro de la subred. El primer byte de la dirección de una subred de clase C tiene un valor comprendido entre 192 y 223. Es posible codificar 2.097.151 subredes distintas de 254 hosts distintos cada una.

Cuando el campo de dirección comienza por la secuencia 1110, se entiende que los 28 bits restantes codifican una dirección de multidifusión, es decir, una dirección especial en donde el destinatario no es único (direcciones de clase D). Las direcciones que comienzan por 1111 se reservan para protocolos especiales como los de administración de grupos de Internet, multitransmisión y otras futuras implementaciones o uso experimental (direcciones de clase E). El valor 127 para el primer byte de una dirección IP está reservado para pruebas de bucle cerrado, es decir, para las comunicaciones entre procesos dentro de la misma máquina.

Al actual protocolo IP se le suele llamar IPv4 para distinguirlo de otra especificación que se empieza ahora a implantar: se trata del protocolo IPv6. Con IPv4 se utilizan direcciones de red de 32 bits, lo que es claramente insuficiente cuando todas las redes se integran entre sí como en el caso de Internet. Aunque tiene muchas más ventajas añadidas en las que aquí no entraremos, IPv6 viene a resolver este asunto, pues su sistema de direccionamiento es de 128 bits. Gran parte de los sistemas operativos modernos así como los dispositivos de red más aventajados ya vienen preparados para la migración de IPv4 a IPv6.

B. Máscaras de subred

Una **máscara** de subred es una secuencia de 32 bits que sirve para distinguir con facilidad qué parte de una dirección codifica la subred (una subdivisión o grupo de la red total) y qué parte el host. Una máscara se construye poniendo a 1 los bits que pertenecen a la subred y a 0 los bits que pertenecen a la identificación del host. Este modo de asignación permite multiplicar extraordinariamente los distintos tipos de subredes. Así una subred de clase A vendría determinada por la máscara 11111111 00000000 00000000 00000000, es decir, 255.0.0.0. Una subred de clase B tendría la máscara 255.255.0.0 (11111111 11111111 00000000 00000000). La subred de clase C tendría la máscara 255.255.255.0. Son posibles combinaciones cualesquiera de los bits para generar subredes y hosts dentro de las subredes siempre que tanto los «1» como los «0» aparezcan consecutivos.

En la Tabla 3.2 se pueden observar los significados de los diferentes códigos **CIDR** y cuántos hosts se pueden identificar en cada subred. La última columna (máscara equivalente) se refiere a la máscara equivalente al CIDR.

Frecuentemente, para facilitar la notación, suele expresarse la dirección IP en formato **CIDR** (*Classless Inter-Domain Routing,* Encaminamiento Inter-Dominios sin Clases), que consiste en escribir la dirección IP en su forma habitual (cuatro números enteros separados por 1) seguida de otro entero cuyo valor es el número de 1 seguidos de la máscara. Estos dos elementos deben ir separados por el símbolo «/». Un ejemplo de notación CIDR sería 128.100.3.67/24, que significaría que el interfaz de red que posee la dirección IP 128.100.3.67 tiene una máscara 255.255.255.0 (24 unos seguidos de otros 8 ceros) y, que por tanto, pertenece a la red 128.100.3.0 o simplemente 128.100.3.

Laboratorio

Identificación de las subredes de la instalación de red

Proseguimos en la investigación de la red de área local que es objeto de nuestro estudio particular para determinar cómo es su sistema de direccionamiento TCP/IP.

Observando las propiedades del protocolo TCP/IP en cada uno de los ordenadores de la red nos daremos cuenta de que las estaciones y servidores que se comunican entre sí comparten el mismo sistema de direccionamiento, permaneciendo ligados a la misma máscara o al menos a máscaras compatibles entre sí.

Identifica todas las subredes de la instalación de red así como los dispositivos que se encargan de comunicar las distintas subredes que hayas localizado.

CEO

SMR_RL_AAbad_03_RedesIPSocket.docx

Documento que contiene:

1. Ejemplo sobre cómo dos nodos saben que están o no en la misma red IP.
2. Ampliación del concepto de socket.

Vocabulario

Dirección IP: conjunto de cuatro números de ocho bits que identifican unívocamente la dirección de nivel 3 de un ordenador en una red TCP/IP.

Máscara IP: es una secuencia de unos y ceros, ambos contiguos, que sirve para denotar en las redes TCP/IP qué identifica la red (secuencia inicial de «1») y qué la subred o conjunto de nodos (secuencia final de «0»).

CIDR: es una mejora del sistema de direccionamiento IP que permite una mayor flexibilidad a la hora de asignar rangos de direcciones por el método de extender las clases de red.

CIDR	Clases C	Clases B	Clases A	Hosts*	Máscara
/32	1/256			1	255.255.255.255
/31	1/128			2	255.255.255.254
/30	1/64			4	255.255.255.252
/29	1/32			8	255.255.255.248
/28	1/16			16	255.255.255.240
/27	1/8			32	255.255.255.224
/26	1/4			64	255.255.255.192
/25	1/2			128	255.255.255.128
/24	1			256	255.255.255.000
/23	2			512	255.255.254.000
/22	4			1024	255.255.252.000
/21	8			2048	255.255.248.000
/20	16			4096	255.255.240.000
/19	32			8192	255.255.224.000
/18	64			16384	255.255.192.000
/17	128			32768	255.255.128.000
/16	256	1		65536	255.255.000.000
/15	512	2		131072	255.254.000.000
/14	1024	4		262144	255.252.000.000
/13	2048	8		524288	255.248.000.000
/12	4096	16		1048576	255.240.000.000
/11	8192	32		2097152	255.224.000.000
/10	16384	64		4194304	255.192.000.000
/9	32768	128		8388608	255.128.000.000
/8	65536	256	1	16777216	255.000.000.000
/7	131072	512	2	33554432	254.000.000.000
/6	262144	1024	4	67108864	252.000.000.000
/5	524288	2048	8	134217728	248.000.000.000
/4	1048576	4096	16	268435456	240.000.000.000
/3	2097152	8192	32	536870912	224.000.000.000
/2	4194304	16384	64	1073741824	192.000.000.000
/1	8388608	32768	128	2147483648	128.000.000.000

Tabla 3.2. *Descripción de los códigos CIDR. Fuente: http://www.vitessenetworks.com.mx*

3.3. Protocolos TCP/IP de nivel superior

En el nivel superior de la arquitectura TCP/IP hay una infinidad de protocolos. Aquí nos vamos a referir a los más comunes, pero existen casi tantos protocolos distintos como tipos de aplicaciones o servicios de nivel de aplicación:

- **FTP.** Es utilizado para la descarga o carga de ficheros en Internet. Define dos canales de comunicación, uno para el gobierno de esta y otro para la transferencia de datos. Pone en marcha el diálogo entre un cliente FTP y un servidor FTP.
- **HTTP.** Es el protocolo utilizado por los navegadores para el acceso a las páginas web.
- **SNMP.** Es uno de los protocolos de la familia TCP/IP utilizados para la gestión de la red. En cada entidad de la red, se habilitan unos agentes que recogen información y que envían a un gestor central desde donde se puede visualizar.
- **RPC.** Es el protocolo de la capa de aplicación en la arquitectura TCP/IP que se encarga de establecer diálogos entre las aplicaciones clientes y sus equivalentes servicios. Se trata de un protocolo básico para la arquitectura de las aplicaciones cliente-servidor.
- **SMTP.** Es el protocolo básico para el intercambio de mensajes de correo electrónico entre servidores de correo o el que usa la aplicación cliente de correo para enviar mensajes al servidor al que se conecta.
- **POP.** Es el protocolo de comunicaciones de alto nivel que se encarga de descargar mensajes de correo electrónico desde el servidor de correo en donde se encuentra el buzón a la bandeja de entrada del cliente de correo. La versión actual del protocolo POP es 3, por ello se denota como POP3.
- **IMAP.** Es un protocolo semejante a POP, pero con algunas funcionalidades añadidas que lo hacen recomendable en situaciones de congestión. Por ejemplo, permite descargar el correo electrónico solo a petición del usuario una vez leída la cabecera del mensaje.

La mayor parte de los protocolos de nivel superior tienen asociado uno o más números de puerto en sus sockets de comunicación, por ejemplo, FTP-21, HTTP-80, SMTP-25, POP-110, etc., aunque esta asociación puede ser alterada por las aplicaciones o por el administrador de la red.

Claves y consejos

Las aplicaciones de SNMP son muy útiles a los administradores de la red porque permiten la configuración de los parámetros de la red desde una consola central, además de recoger estadísticas de utilización de los recursos.

Ejemplos

Acceso desde el explorador a un servidor web

Con este ejemplo vamos a tratar de comprender cómo un explorador de Internet utiliza el sistema de direccionamiento y la tecnología de sockets asociados a puertos de comunicaciones para resolver la exploración de una página web.

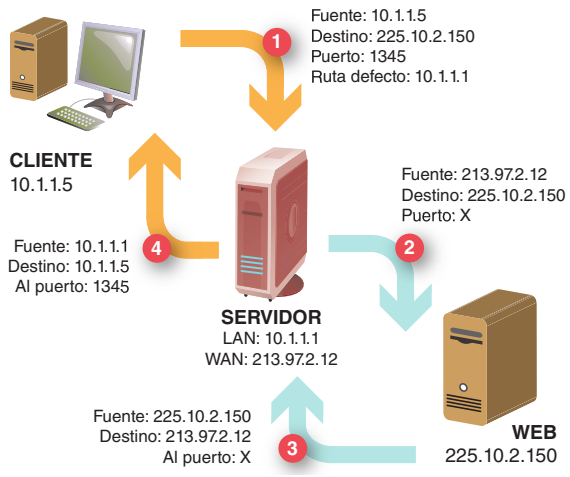

Fig. 3.12. Cliente web que accede a un servidor web a través de un encaminador.

En la Fig. 3.12 está representado el acceso de un cliente con dirección IP 10.1.1.5 con un explorador a un servidor web que reside en Internet con dirección 225.10.2.150, utilizando como intermediario un servidor que hace la función de encaminador de paquetes entre la red local en la que se encuentra el cliente e Internet en donde se encuentra el servidor. Describamos su funcionamiento.

En el paso 1, el cliente hace una petición con destino 225.10.2.150, dejando abierto el puerto 1345. Este paquete es capturado por el servidor-encaminador y envía en su nombre (dirección 213.97.2.12) el paquete al servidor web dejando abierto otro puerto «x» de su interfaz de red externo (paso 2).

El servidor web procesa la petición y devuelve (paso 3) la página a quien se la pidió que fue 213.97.2.12 por el puerto que le dejó abierto que denominamos «x».

En el cuarto paso, el encaminador pone el paquete en la red interna, enviándolo a quien le solicitó su servicio de encaminamiento por el puerto que le dejó abierto que era el 1345.

Aquí tenemos un ejemplo de un cliente que utiliza un servicio (de encaminamiento) para acceder a otro servicio informativo de páginas web.

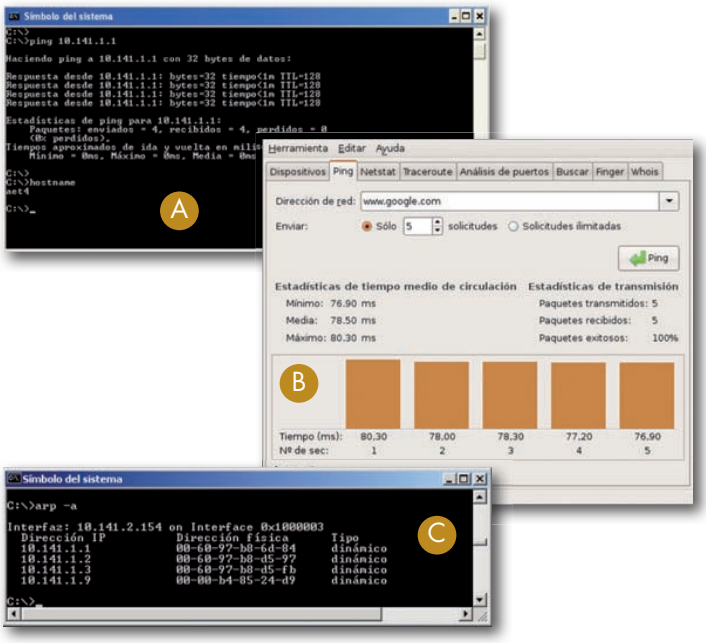

Fig. 3.13. *A) Ejecución del comando ping sobre el nodo 10.141.1.1, y verificación del nodo local con hostname en una estación Windows. B) Utilidad gráfica de ping en un sistema Linux sobre www.google.com. C) Ejecución del comando ARP.*

3.4. Utilidades propias de redes TCP/IP

Las siguientes utilidades son comunes en los sistemas UNIX. Otros sistemas operativos las incorporan en alguna medida si llevan instalado TCP/IP. El nombre exacto y los calificadores de las órdenes varían según los sistemas y las versiones. La ayuda del sistema operativo será de gran utilidad para concretar exactamente el formato de cada orden.

A. Utilidad ping

Ping (*Packet Internet Groper,* Tanteo de paquetes Internet) es una utilidad que sirve para enviar mensajes a una dirección de red concreta que se especifica como argumento con el fin de realizar un test a la red utilizando el protocolo ICMP. El nodo destinatario nos reenviará el paquete recibido para confirmarnos que se realiza el transporte entre los dos nodos correctamente. Además, proporciona información añadida sobre la red, como se puede ver en la Fig. 3.13, A y B.

Ping puede configurar varios parámetros cuando se ejecuta desde la línea de comandos, por ejemplo, es posible indicarle cuántos paquetes queremos enviar, qué información vamos a enviar con cada paquete, el tamaño de cada paquete enviado, etc. Tendremos que recurrir a la ayuda del comando ping en cada sistema para asegurarnos de la sintaxis exacta de la orden.

Hay que tener en cuenta que la utilidad ping varía dependiendo de la versión IP que ejecuta la red. Si no se especifica lo contrario, siempre se supone que se trata de la versión 4 (IPv4).

En la ejecución sobre Windows se pueden distinguir dos secciones. La primera proporciona información sobre la configuración IP del nodo: nombre (aet1), dominio al que pertenece, etc. En la segunda sección se especifican los parámetros de configuración del adaptador de red: tipo de tarjeta (3Com EtherLink XL), dirección física o MAC, dirección IP, etc.

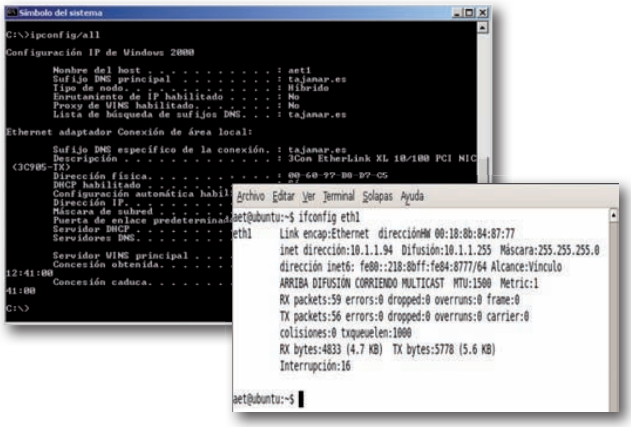

Fig. 3.14. *Respuesta del sistema operativo de red al comando ipconfig/all en una estación de trabajo Windows (arriba). Visualización de la configuración de red para la interfaz eth1 en una estación Linux mediante ifconfig (abajo).*

B. Utilidad arp

ARP (*Address Resolution Protocol,* Protocolo de resolución de direcciones), es una utilidad sirve para asignar automáticamente direcciones IP a direcciones físicas, es decir, para gestionar el protocolo ARP. En la parte superior de la Fig. 3.13-C se interroga al sistema mediante ARP cuáles son las direcciones IP que tiene resueltas, es decir, de las que conoce su dirección física y cuál fue el tipo de asignación.

C. Utilidad ipconfig de Windows e ifconfig/iwconfig de Linux

Configura la dirección del host o bien proporciona información sobre la configuración actual. Por ejemplo, la ejecución del comando siguiente proporciona información sobre la tarjeta Ethernet 3Com EtherLink XL 10/100 PCI (Fig. 3.14, arriba).

La utilidad equivalente en Linux es ifconfig para las redes cableadas e iwconfig para las redes inalámbricas, aunque la mayor parte de las distribuciones ya permiten configurar muchos de los parámetros que admiten a través de la interfaz gráfica. La ejecución del comando «ipconfig help» en Windows e «ifconfig –h» y «iwconfig –h» en Linux nos proporcionarán la ayuda necesaria para la utilización de la orden, ejemplos incluidos. En general, cualquiera de estas órdenes tiene su propia ayuda con el calificador HELP para Windows y –h o --help para Linux.

D. Utilidad netstat

Netstat *(Network status)*, proporciona información sobre el estado de la red. El comando ejecutado en la Fig. 3.15 sobre Windows obtiene información estadística sobre los paquetes de red enviados y recibidos. Como se ve, sobre Linux, la orden puede proporcionar muchas otras informaciones como el estado de las conexiones, lo que hay al otro lado de cada conexión, etcétera.

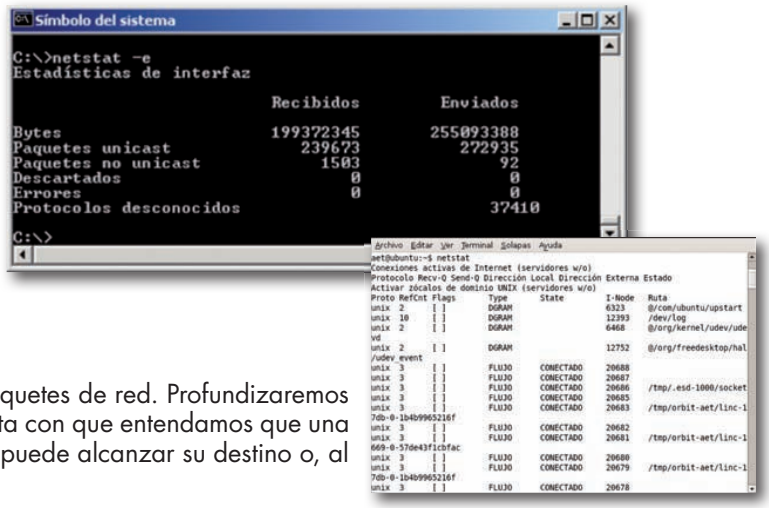

Fig. 3.15. *Respuesta del sistema al comando netstat en Windows y en Linux.*

E. Utilidad route

Sirve para determinar las rutas que deben seguir los paquetes de red. Profundizaremos en el concepto de rutas más adelante. De momento, basta con que entendamos que una ruta indica el camino apropiado por el que un paquete puede alcanzar su destino o, al menos, aproximarse a él.

Para manejar las tablas de rutas, en sistemas Windows suele utilizarse la orden ROUTE, mientras que en sistemas Linux hay una gran diversidad de órdenes y utilidades, aunque la más usual es «ip route», que admite una multitud de parámetros que deberemos consultar en cada versión para utilizarlo con propiedad.

Por ejemplo, si imprimimos las rutas disponibles para un nodo tendremos la siguiente salida (Fig. 3.16-A):

F. Utilidad tracert

Se utiliza para controlar los saltos de red que deben seguir los paquetes hasta alcanzar su destino (Fig. 3.16 B y C). Además proporciona información sobre otros parámetros de la internet. Cuando el número de saltos es 1, esto quiere decir que la red es plana, es decir, se trata de una red de área local.

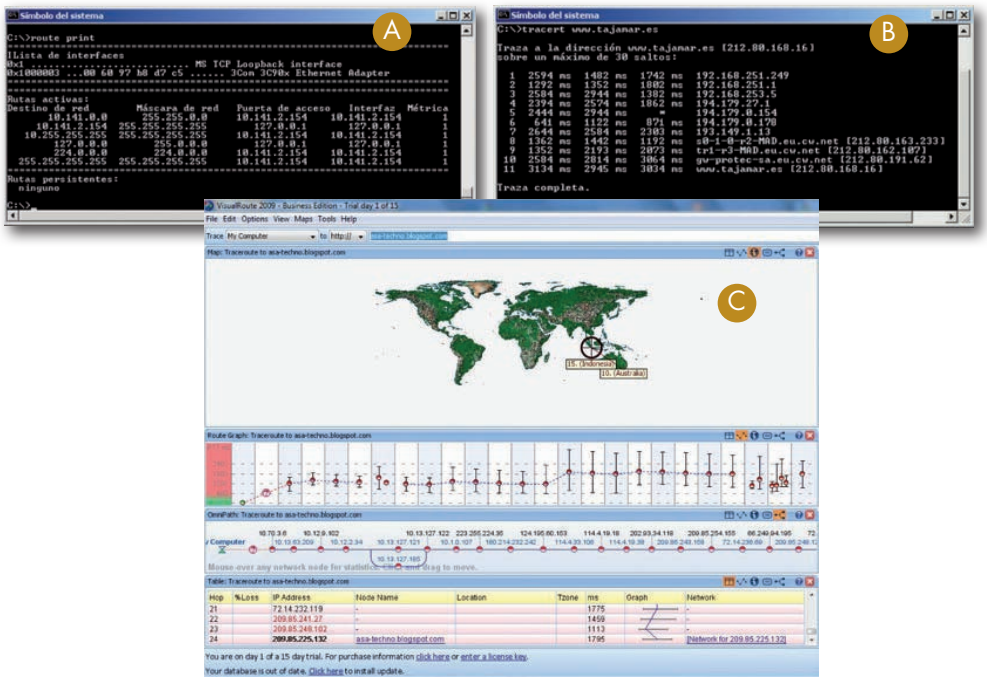

Fig. 3.16. *A) Respuesta del sistema al comando route. B) Comando tracert sobre Windows en el que se pueden observar 11 saltos. C) Utilidad gráfica VisualRoute.*

En la Fig. 3.16 A podemos apreciar tres secciones. En la primera, se especifican las interfaces de red que posee el nodo en el que se ejecuta route. En la segunda sección se describen las rutas activas en ese momento, núcleo de la tabla de enrutamiento. En la tercera sección se describen, si existen, las rutas persistentes. En Linux es habitual interrogar al sistema sobre las rutas con la orden «ip route».

Truco

Para realizar una de estas conexiones anónimas se suele utilizar como nombre de usuario la palabra «anonymous» y como contraseña, suele ser una buena costumbre teclear la dirección de correo electrónico del usuario que pretende beneficiarse del servicio ftp.

G. Utilidades ftp y tftp

La utilidad ftp sirve para intercambiar ficheros entre dos nodos de la red utilizando el protocolo FTP. FTP también tiene su parte de cliente y su parte de servidor. Cuando se ejecuta el cliente ftp, aparece el identificador de utilidad «FTP>» sobre la que se ejecutan los comandos ftp: listar, traer (bajar) o dejar (subir) ficheros, etc. Previamente a la utilización del FTP para realizar transferencias, es necesario preparar una conexión segura a través del protocolo TCP. Esto se realiza con el comando open seguido de la dirección IP o el nombre DNS del host remoto. El comando tftp es similar al ftp, más fácil de configurar, pero con menos prestaciones.

La utilización de un servidor ftp exige tener acceso al servidor a través de un nombre de usuario y una contraseña que nos asignará el administrador del sistema remoto. Muchos servidores en Internet tienen información pública a la que se accede sin necesidad de tener cuenta en el equipo, permitiendo conexiones de usuarios anónimos.

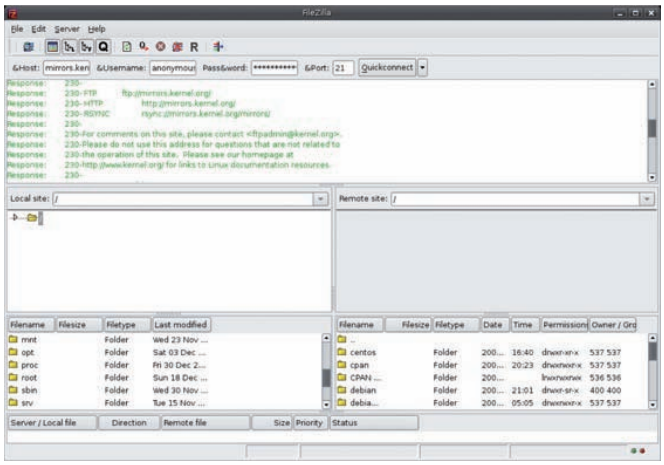

Fig. 3.17. *Ejemplo de cliente FTP gráfico.*

En la Fig. 3.17 se puede ver un ejemplo de Filezilla, un cliente gráfico típico de ftp, que tiene versiones tanto para Windows como para Linux. En la ventana izquierda aparece el sistema de ficheros local. A la derecha, una vez realizada la conexión aparecerá el sistema de ficheros remoto. Las operaciones de copiado se realizan arrastrando los ficheros o directorios desde un lado hacia el otro.

Filezilla dispone tanto de la versión cliente (la representada en la figura) como versión servidor. Es gratuita y se puede descargar desde **http://filezilla-project.org/.**

H. Utilidades telnet y ssh

Sirve para realizar conexiones remotas interactivas en forma de terminal virtual a través del protocolo de alto nivel TELNET. El comando va acompañado de la dirección IP del nodo remoto o de su dirección DNS.

Los servidores Windows implementan un servidor TELNET, que sirve sesiones en forma de ventanas emuladoras DOS a los clientes TELNET que se conectan a ellos desde su red, lo que es muy interesante para ejecutar scripts en máquinas remotas.

En el mundo Linux, la utilidad equivalente más moderna es ssh. Esta es una de las utilidades más versátiles que tiene su parte de cliente y de servidor. Puede ejecutar aplicaciones remotas, copiar ficheros, crear sesiones remotas gráficas, crear túneles de comunicación, etc. Funcionalmente, ssh puede sustituir las conexiones remotas de TELNET, pero la gran ventaja de ssh es que cifra las conexiones, por lo que es mucho más seguro que TELNET.

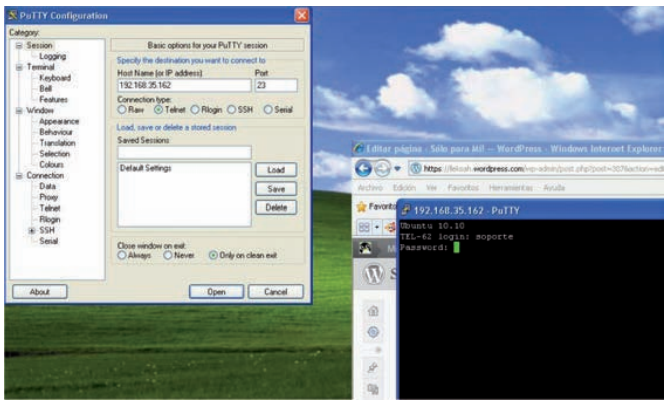

Fig. 3.18. *Ejemplo de ejecución de TELNET desde un sistema operativo de Microsoft mediante PuTTY.*

Investigación

En la página **http://www.guia-ubuntu.org/index.php?title=Servidor_ssh** tienes información sobre ssh. Es interesante, aunque sea más propio de sistemas que de redes, que te familiarices con esta tecnología leyendo algunos documentos técnicos. Otra página para comenzar el estudio es **http://es.wikipedia.org/wiki/Ssh**

Truco

Existen aplicaciones gratuitas que se pueden instalar en sistemas operativos de Microsoft que incorporan clientes de conexión remota como telnet, ssh y otros. Por ejemplo, PuTTY (Fig. 3.18).