

Apéndice - Anotaciones
Redes de Comunicaciones
TCP/IP

Laura M. Castro Souto

Primer Cuatrimestre
Curso 2000/2001

Capítulo 1

Tecnología Ethernet

Las redes Ethernet definen un esquema de direccionamiento de 48 bits. Cada computadora conectada a una red Ethernet es asignada a un número único de 48 bits conocido como *dirección Ethernet*. Para asignar una dirección, los fabricantes de hardware de Ethernet adquieren bloques de direcciones Ethernet (el IEEE maneja dicho espacio de direcciones y las asigna conforme se necesitan) y las asignan en secuencia conforme fabrican el hardware de interfaz Ethernet. De esta manera no existen dos unidades de hardware de interfaz que tengan la misma dirección Ethernet.

1.1. Paquetes Ethernet

Los paquetes Ethernet son de longitud variable, pero no menor de 64 bytes ni mayor de 1518:

CRC *Cyclic Redundancy Check (Verificación por Redundancia Cíclica)* Sirve para detectar errores de transmisión, se computa en emisor y receptor como una función de la trama de datos.

Preámbulo 64 bits que alternan ceros y unos para ayudar a la sincronización de los nodos de recepción.

Tipo 16 bits que identifican el tipo de datos, permitiendo así múltiples protocolos; los datos se autoidentifican y el sistema operativo decide qué módulo de software de protocolo se usa para procesarlos.

Los protocolos TCP/IP son muy flexibles porque casi cualquier tecnología subyacente puede usarse para transferir tráfico de información TCP/IP.

Los routers no utilizan el host destino sino la red destino cuando enrutan un paquete.

Capítulo 2

Direcciones Internet

Para las direcciones, los diseñadores del TCP/IP eligen un esquema análogo al direccionamiento en las redes físicas, en el que cada host en la red tiene asignada una dirección de número entero de 32 bits, llamada su *dirección de Internet ó dirección IP*. La parte inteligente del direccionamiento en una red es que los números enteros son seleccionados con cuidado para hacer eficiente el enrutamiento. De manera específica, una dirección IP codifica la identificación de la red a la que se conecta el host, así como la identificación de un host único en esa red.

Cada host en una red de redes TCP/IP tiene asignada una dirección de número entero de 32 bits que se utiliza en todas las comunicaciones con dicho host.

Los bits de dirección IP de todos los hosts de una red comparten un prefijo común.

Conceptualmente, cada dirección es un par (*netid*, *hostid*) donde *netid* identifica una red y *hostid* un dentro de la red.

Definida una dirección IP, se puede determinar su tipo según los 3 bits de orden, de los que son necesarios sólo 2 para distinguir entre los 3 tipos primarios.

Debido a que las direcciones IP codifican tanto una red como un host en dicha red, no especifican una computadora individual, sino una conexión a la red.

Por lo tanto, un router que conecta cierto número de redes tiene cierto número de direcciones IP distintas, una para cada conexión de red.

Por regla, nunca se asigna un campo *hostid* igual a 0 a un host individual; en vez de eso, una dirección IP con campo *hostid* a 0 se utiliza para referirse a la red en sí misma.

Las direcciones IP se pueden usar para especificar la difusión (*broadcast*); estas direcciones se transforman en difusión por hardware, si ésta se encuentra disponible. Por norma, una dirección de difusión tiene todos los bits del campo *hostid* a 1.

La dirección 127.0.0.0, valor del rango tipo A, se reserva para loopback. Cuando algún programa utiliza la dirección loopback como destino, el software de protocolo en una computadora regresa los datos sin generar tráfico a través de alguna red. Así pues, un paquete enviado a la dirección 127 nunca debe aparecer en ninguna red; un host o un router nunca deben difundir información de enrutamiento ni de

TIPO	DIRECCIÓN MÁS BAJA	DIRECCIÓN MÁS ALTA
A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

accesibilidad para el número de red 127, pues no es una dirección de red.

2.1. Resumen de reglas especiales de direccionamiento

Capítulo 3

Transformación de direcciones Internet en direcciones Físicas (ARP)

El objetivo es diseñar un software de bajo nivel que oculte las direcciones físicas y permita que programas de un nivel más alto trabajen sólo con direcciones de la red de redes. Sin embargo, la comunicación debe llevarse a cabo por medio de redes físicas, utilizando cualquier esquema de direcciones físicas proporcionado por el hardware, ¿cómo transformar la dirección Internet (alto nivel) de un host en su dirección física? Este problema se conoce como *problema de asociación de direcciones*.

Cada interfaz Ethernet tiene asignada una dirección física de 48 bits desde la fabricación del producto. En consecuencia, cuando el hardware falla y se necesita reemplazar una interfaz Ethernet, la dirección física de la máquina cambia; ¡además no hay posibilidad de codificarla en una dirección IP de 32 bits!

Los diseñadores de los protocolos TCP/IP utilizan un protocolo de bajo nivel para asignar direcciones en forma dinámica. Conocido como *Protocolo de Asociación de direcciones (ARP)*, proporciona un mecanismo razonablemente eficaz y fácil de mantener.

El protocolo de Asociación de Direcciones (ARP) permite que un host encuentre la dirección física de otro host dentro de la misma red física con sólo proporcionar la IP del objetivo: transmite por broadcast un paquete especial que pide al hosta que posee dicha IP que responda con su dirección física. Todos los host lo reciben, pero sólo el host adecuado reconoce su propia IP y envía una respuesta que contiene su dirección Ethernet.

No se hace todo por multicast (todas las transmisiones) porque es muy caro en costes de comunicación, además las máquinas mantienen una caché.

3.1. Refinamientos ARP

En cada difusión ARP el transmisor incluye su asignación de dirección IP como dirección física (es probable que si A solicita la dirección Ethernet de B para enviarle algo, B necesite luego la de A para responderle); los receptores actualizan su caché (memoria intermedia) antes de procesar un paquete ARP.

Cuando a una máquina se le reemplaza la interfaz de Ethernet y cambia su dirección física, notifica a las otras su nueva dirección al reiniciarse.

Pensamos ARP como parte del sistema físico de red, no como parte de los protocolos de red de redes, es un protocolo de bajo nivel que oculta el direccionamiento físico subyacente de red, al permitir que se asigne una dirección IP arbitraria a cada máquina.

Capítulo 4

Determinación en el arranque de una dirección Internet (RARP)

Por lo general, la dirección IP de una máquina se mantiene en el área secundaria de almacenamiento, donde el sistema operativo la encuentra en el momento del arranque. ¿Cómo puede una máquina que no cuenta con disco permanente determinar su IP? El problema es crítico para las estaciones de trabajo que almacenan sus archivos en un servidor remoto, ya que dichas máquinas necesitan una dirección IP antes de poder utilizar protocolos TCP/IP estándar para transferencia de archivos a fin de obtener su imagen inicial de arranque.

La máquina que necesita conocer su dirección *sabe* comunicarse, puede usar su dirección física, recurre de manera temporal al direccionamiento físico de red, envía una solicitud a un servidor identificándose como destino (en realidad hace un broadcast al que responderán uno o más servidores), éste busca en su base de datos y contesta.

Una máquina sin disco utiliza un protocolo TCP/IP para red de redes llamado RARP (*Protocolo Inverso de Asociación de Direcciones*) a fin de obtener su IP desde un servidor. Es el caso opuesto a la asociación de direcciones: dada una dirección física (única y uniforme), se necesita un esquema que permita que un servidor la transforme en una dirección de Internet.

Capítulo 5

Protocolo Internet: entrega de datagramas sin conexión

El servicio más importante en la red de redes consiste en un sistema de entrega de paquetes, que se define técnicamente como un sistema de entrega de paquetes sin conexión y con el mejor esfuerzo. Es un servicio *no confiable* porque la entrega no está garantizada, los paquetes se pueden perder, duplicar, retrasar o entregar sin orden (pueden viajar por diferentes rutas), pero el servicio no detectará estas condiciones ni informará al emisor o al receptor. El servicio es llamado *sin conexión* porque cada paquete es tratado de forma independiente de todos los demás. Se dice que el servicio trabaja con base en una entrega con el mejor esfuerzo posible porque el software de red de redes hace un serio intento por entregar los paquetes, no se descartan caprichosamente, la no confiabilidad aparece sólo cuando los recursos están agotados o la red subyacente falla.

El protocolo que define el mecanismo de entrega sin conexión y no confiable es conocido como *Protocolo Internet (IP)*. Define la unidad básica para la transferencia de datos utilizada a través de una red TCP/IP, es decir, especifica el formato exacto de todos los datos que pasarán por ella. El software IP realiza la función de enrutamiento. Por último, IP incluye un conjunto de reglas sobre la forma en que hosts y routers deben procesar los paquetes, cómo y cuándo generar mensajes de error, condiciones de descarte de los paquetes...

5.1. Encapsulación de Datagramas

Capítulo 6

Enrutamiento IP

El *enrutamiento* es el proceso de selección de un camino sobre el que se mandarían paquetes y el *router* es la computadora que hace la selección. El enrutamiento ocurre a muchos niveles.

De forma ideal, el software de enrutamiento examinaría aspectos como la carga de la red, la longitud del datagrama o el tipo de servicio que se especifica en su cabecera para seleccionar el mejor camino. Sin embargo, la mayor parte del software de enrutamiento es mucho menos sofisticado y selecciona rutas basándose en suposiciones sobre los caminos más cortos.

6.1. Entrega Directa e Indirecta

Debido a que las direcciones IP de todas las máquinas dentro de una sola red incluyen un prefijo común y como la extracción de dicho prefijo se puede realizar mediante unas cuantas instrucciones máquina, la comprobación de que una máquina se puede alcanzar directamente es muy eficiente.

Los routers en una red TCP/IP forman una estructura cooperativa e interconectada. Los datagramas pasan de un router a otro hasta que llegan a uno que los puede entregar de forma directa.

El algoritmo usual de enrutamiento IP emplea una *tabla de enrutamiento* en cada máquina para almacenar información sobre posibles destinos y cómo alcanzarlos. Por lo común, esta tabla contiene pares (N, R) donde N es la dirección IP de una *red* de destino y R es la dirección IP del *siguiente salto* (router) en el camino hacia N .

Para ocultar información, mantener reducidas las tablas de enrutamiento y tomar las decisiones de enrutamiento de forma eficiente, el software de enrutamiento IP sólo puede guardar información sobre las direcciones de las redes de destino, no sobre las direcciones de hosts individuales.

Otra técnica utilizada es asociar muchos registros a un *router por defecto* (u *omisión*). La idea es hacer que el software de enrutamiento IP busque primero en la tabla de enrutamiento para encontrar la red de destino. Si no aparece una ruta en la tabla, las rutinas de enrutamiento envían el datagrama a un *router asignado por defecto*. Este enrutamiento por omisión es de gran ayuda cuando un sitio tiene pocas direcciones locales y sólo una conexión con el resto de la red de redes.

Es importante entender que, a excepción de la disminución del tiempo de vida y de volver a computar la suma de verificación (*checksum*), el enrutamiento IP no altera el datagrama original. En particular, las direcciones origen y destino permanecen sin alteración. ¿Dónde almacena IP la dirección de salto siguiente, entonces? En el datagrama no existe un lugar reservado para ella. De hecho, IP no almacena esa dirección. Después de ejecutar el algoritmo de enrutamiento, IP pasa el datagrama y la dirección de salto siguiente al software de interfaz de red, responsable de la red física sobre la que el datagrama se debe enviar. El software de interfaz de red transforma la dirección de salto siguiente en una dirección física, crea una trama usando esa dirección física, pone el datagrama en la porción de datos de la trama y envía el resultado.

6.2. Manejo de los datagramas entrantes

Cuando un datagrama IP llega a un host, el software de interfaz de red lo entrega al software IP para su procesamiento. Si la dirección de destino del datagrama corresponde a la dirección IP del host, el software IP del host acepta el datagrama y lo pasa al software de protocolo de alto nivel apropiado para su procesamiento posterior. Si la dirección IP de destino no corresponde, el host descarta el datagrama.

A diferencia de los hosts, en los routers, si la dirección de destino IP no coincide, IP lo enruta usando el algoritmo estándar y la información en la tabla local de enrutamiento.

La determinación sobre si un datagrama IP alcanzó su destino final no es tan trivial. Recuérdese que hasta un host puede tener muchas conexiones físicas, cada una con su propia dirección IP. Cuando llega un datagrama IP, la máquina debe comprobar la dirección de destino con la IP de cada una de sus conexiones de red. Una máquina también debe aceptar datagramas que se transmitieron por broadcast o multicast si la dirección IP de destino es la dirección IP de broadcast de su red o la dirección IP de multicast de su grupo multicast. En cualquier caso, si la dirección no corresponde a ninguna de las direcciones de la máquina local, IP disminuye el campo de tiempo de vida en la cabecera del datagrama, descartándolo si el contador llega a cero, o computando un nuevo checksum y enrutando el datagrama si la cuenta es positiva.

Capítulo 7

IP: mensajes de error y de control (ICMP)

El protocolo IP por sí mismo no tiene nada para ayudar al transmisor a comprobar la conectividad ni para ayudarle a aprender sobre dichas fallas.

Para permitir que los routers en una red reporten los errores o proporcionen información sobre circunstancias inesperadas, los diseñadores agregaron a los protocolos TCP/IP un mecanismo de mensajes de propósito especial, conocido como *Protocolo de Mensajes de Control Internet (ICMP)*, que se considera parte obligatoria del IP y se debe incluir en todas sus implementaciones.

Al igual que el resto del tráfico, los mensajes ICMP viajan a través de la red en la porción de los datos de los datagramas IP, pero su destino final no es un programa de aplicación ni un usuario en la máquina destino, sino el software IP de la misma. Esto es, cuando llega un mensaje de error ICMP, el módulo de software ICMP lo maneja. Por supuesto, si el ICMP determina que un protocolo de un nivel más alto o un programa de aplicación causaron un problema, notificará al módulo apropiado.

Aunque en principio fue diseñado para permitir que los routers reportasen a los hosts las causas de los errores en la entrega, el ICMP no se restringe sólo a ellos, cualquier máquina puede enviar un mensaje ICMP a cualquier otra.

Cuando un datagrama causa un error, el ICMP sólo puede reportar la condición del error a la fuente original del datagrama; la fuente debe relacionar el error con un programa de aplicación individual o tomar alguna otra acción para corregir el problema.

Los mensajes ICMP requieren 2 niveles de encapsulación. Cada mensaje ICMP viaja a través de la red en la porción de datos de un datagrama IP, que a su vez viaja a través de la red física en la porción de datos de una trama.

Hay una excepción en los procedimientos de manejo de errores si un datagrama IP que lleva un mensaje ICMP causa un error. Esta excepción, diseñada para evitar el problema de tener mensajes de error sobre mensajes de error, especifica que los mensajes ICMP no se generan por errores resultantes de datagramas que llevan mensajes de error ICMP.

Como se enrutan exactamente igual que los demás, también pueden perderse, descartarse,...

A pesar de que los mensajes ICMP se encapsulan y envían mediante IP, ICMP no se considera un protocolo de nivel más alto, sino una parte obligatoria de IP. La razón de usar IP para entregar ICMP es que quizá se necesite viajar a través de muchas redes para llegar al destino final, de modo que no se pueden entregar sólo por medio de transporte físico.

Capítulo 8

Subnetting y Supernetting

8.1. Proxy ARP

Es una de las técnicas que se usan para transformar un sólo prefijo IP de red en dos direcciones físicas, que sólo se aplica en redes que utilizan ARP para convertir direcciones de red en direcciones físicas.

La ventaja que aporta es que oculta completamente los detalles de las conexiones físicas. La desventaja, que sólo es utilizable en redes que usen ARP.

8.2. Subnetting

El direccionamiento de subred es una parte obligatoria del direccionamiento IP. Conceptualmente, agregar subredes sólo cambia ligeramente la interpretación de direcciones IP. En vez de dividir la dirección IP de 32 bits en un prefijo de red y un sufijo de host, el direccionamiento de subred (*subnetting*) divide la dirección en una porción de red y una porción local. La interpretación de la porción de red permanece igual que en las redes que no utilizan subnetting. La interpretación de la porción local de una dirección se somete al criterio de la localidad (dentro de las limitaciones del estándar formal para el direccionamiento de subred).

El resultado es una forma de *direccionamiento jerárquico* que lleva al correspondiente *enrutamiento jerárquico*.

¿Cómo se tiene que dividir la parte local para que el enrutamiento sea eficiente? Para permitir la máxima autonomía, el estándar TCP/IP de subred permite que la partición se seleccione basándose en cada red particular.

El estándar especifica que una localidad que utiliza subnetting debe escoger una *máscara de subred* de 32 bits para cada red. Los bits en la máscara de subred se indican como 1 si la red trata el bit correspondiente de la dirección IP como parte de la dirección de red, y como 0 si se trata como parte del identificador de host.

Todas las subredes en una dirección IP de red deben ser contiguas, las máscaras de subred deben ser uniformes a través de todas las redes y todas las máquinas deben participar en el subnetting.

8.3. El algoritmo de Enrutamiento con Subnetting

Al igual que el algoritmo estándar de enrutamiento IP, el algoritmo de enrutamiento en subredes basa sus decisiones en una tabla de enrutamiento. Una tabla convencional contiene registros de la siguiente forma:

$$\left(\underbrace{\text{dirección de red}}_{\text{IP de la red destino}}, \quad \underbrace{\text{dirección de salto siguiente}}_{\text{IP del siguiente router por el que se ha de pasar}} \right)$$

El algoritmo estándar sabe que una dirección está dividida en una porción de red y una porción local, ya que los primeros 3 bits codifican el tipo y formato de la dirección (A, B, C ó D). Con subnetting no es posible decidir qué bits corresponden a la red ni cuáles al host sólo con la dirección IP.

El algoritmo modificado que se utiliza con las subredes guarda información adicional en la tabla de enrutamiento. Cada registro de la tabla contiene un campo adicional que especifica la máscara de subred utilizada con la red:

$$(\text{máscara de subred}, \text{dirección de red}, \text{dirección del salto siguiente})$$

Cuando el algoritmo modificado elige rutas, utiliza la máscara de subred para extraer los bits de la dirección de destino y compararlos con el registro en la tabla; realiza un AND con los 32 bits de la dirección IP de destino y el campo de la máscara de subred de cada registro, verifica si es igual al valor del campo de dirección de red y si es así enruta el datagrama a la dirección especificada en el campo de dirección de salto siguiente del registro (que será, igual que en el algoritmo estándar, accesible directamente).

Un host puede obtener la máscara de subred para una red al enviar una solicitud de máscara de subred ICMP al router en dicha red, solicitud que puede hacerse por broadcast si no se conoce la dirección del router.

8.4. Supernetting

El direccionamiento de superred trata las direcciones IP como números enteros arbitrarios y permite que un administrador de red asigne un grupo de números contiguos a una localidad o a una red dentro de una localidad. Los hosts y routers que utilizan el direccionamiento de superred necesitan software de enrutamiento no convencional que entienda los rangos de direcciones.

Capítulo 9

Estratificación de Protocolos por Capas

9.1. Modelo ISO de 7 capas

9.2. Modelo TCP/IP

Los protocolos de comunicación utilizan técnicas de *multiplexado* y *demultiplexado* a través de la jerarquía de capas. Cuando envía un mensaje, la computadora fuente incluye bits extras que codifican el tipo de mensaje, el programa de origen y los protocolos utilizados. En el extremo de recepción, la máquina destino se vale de la información extra para guiar el proceso.

Demultiplexado de tramas entrantes basado en el campo de tipo que se encuentra en la cabecera de la trama.

Demultiplexado en la capa Internet. El software IP selecciona un procedimiento apropiado para manejar un datagrama, basándose en el campo de tipo de protocolo, localizado en la cabecera del datagrama.

Capítulo 10

Protocolo de Datagrama de Usuario (UDP)

En vez de pensar en un proceso como destino final, imaginaremos que la máquina contiene un grupo de puntos abstractos de destino, llamados *puertos de protocolo*. Cada uno de ellos se identifica por medio de un número entero positivo. El sistema operativo proporciona un mecanismo de interfaz que los procesos utilizan para especificar o acceder a un puerto.

La mayor parte de los sistemas operativos proporciona un acceso síncrono a los puertos, lo que significa que los cómputos se detienen durante una operación de acceso a puerto. Por ejemplo, si un proceso intenta extraer datos de un puerto antes de que llegue cualquier valor, el sistema operativo detiene (bloquea) temporalmente el proceso hasta que lleguen datos. Una vez que esto sucede, el sistema operativo pasa los datos al proceso y lo vuelve a iniciar. En general, los puertos tienen *memoria intermedia*, para que los datos que llegan antes de que un proceso esté listo para aceptarlos no se pierdan. Para lograr la colocación en memoria intermedia, el software de protocolo, localizado dentro del sistema operativo, coloca los paquetes que llegan de un puerto de protocolo en particular en una cola de espera (finita) hasta que un proceso los extraiga.

Para comunicarse con un puerto externo, un transmisor necesita saber tanto la dirección IP de la máquina de destino como el número de puerto de protocolo del destino dentro de la máquina. Cada mensaje debe llevar el número del *puerto de destino* de la máquina a la que se envía, así como el número de *puerto origen* de la máquina fuente a la que se deben direccionar las respuestas. Por lo tanto, es posible que cualquier proceso que recibe un mensaje conteste al transmisor.

El protocolo de datagrama de usuario (UDP) proporciona un servicio de entrega sin conexión y no confiable, utilizando IP para transportar mensajes entre máquinas (proporciona, pues, la misma semántica de entrega de datagramas, no emplea acuses de recibo para asegurarse de que llegan los mensajes, no ordena los mensajes entrantes ni proporciona retroalimentación para controlar la velocidad a la que fluye la información... los mensajes UDP pueden, por tanto, perderse, duplicarse o llegar sin orden, pueden también llegar más rápido de lo que el receptor los puede procesar). Emplea IP para llevar mensajes, pero agrega la capacidad para distinguir entre varios destinos dentro de una computadora host, ya que además de los datos cada mensaje UDP contiene tanto el número de puerto de destino como el número de puerto de origen.

10.1. Encapsulación de UDP y estratificación por capas de protocolos

Estratificación conceptual por capas de UDP entre programas de aplicación e IP.

Datagrama UDP encapsulado en un datagrama IP para su transmisión a través de una red. El datagrama se encapsula en una trama cada vez que viaja a través de una red.

La capa IP sólo es responsable de transferir datos entre un par de hosts dentro de una red de redes, mientras que la capa UDP solamente es responsable de diferenciar entre varias fuentes o destinos dentro de un host.

Capítulo 11

Servicio de Transporte de Flujo Confiable (TCP)

Si UDP es el primero, el *Protocolo de Control de Transmisión* (TCP) es el segundo servicio más importante y mejor conocido de nivel de red.

11.1. Características

La interfaz entre los programas de aplicación y el servicio TCP/IP de entrega confiable se puede caracterizar por 5 funciones:

- *Orientación de flujo de datos*: pensamos en los datos como un flujo de bits; el servicio de entrega de flujo en la máquina destino para al receptor exactamente la misma secuencia de bytes que le pasa el transmisor en la máquina de origen.
- *Conexión de circuito virtual*: Antes de poder empezar la transferencia, los programas de aplicación, transmisor y receptor, interactúan con sus respectivos sistemas operativos, informándose de la necesidad de realizar una transferencia de flujo. Una vez que se establecen todos los detalles, los módulos de software de protocolo (que se habrán comunicado mediante mensajes por la red, verificando existencia, autorización y disponibilidad) informan a los programas de aplicación que se estableció una *conexión* y que la transferencia puede empezar.
- *Transferencia con memoria intermedia*: cuando se transfieren datos, cada aplicación utiliza paquetes del tamaño que encuentre adecuado. En el extremo receptor, el software de protocolo entrega el flujo de datos en el mismo orden en que se enviaron. El software de protocolo puede, no obstante, dividir a su vez el flujo en paquetes, para hacer más eficiente el tráfico por la red. Para aplicaciones en las que los datos se deben entregar aunque no se llene una memoria intermedia, el servicio de flujo proporciona un mecanismo de *empuje* (*push*) que las aplicaciones usan para forzar una transferencia. En un receptor, hace que TCP ponga los datos a disposición de la aplicación sin demora.
- *Flujo no estructurado*: El servicio de flujo TCP/IP no está obligado a formar flujos estructurados de datos. Los programas de aplicación que utilizan el servicio de flujo deben entender el contenido del flujo y ponerse de acuerdo sobre su formato antes de iniciar una conexión.
- *Conexión Full Duplex*: Las conexiones proporcionadas por el servicio de flujo TCP/IP permiten la transferencia concurrente en ambas direcciones (*full duplex*).

11.2. Proporcionando Confiabilidad

¿Cómo lo hace el software de protocolo si el sistema subyacente de comunicación sólo ofrece entrega no confiable? Usando una técnica fundamental: el *acuse de recibo positivo con retransmisión*, que requiere que el receptor se comunique con el origen y le envíe un mensaje de *acuse de recibo* (ACK) conforme recibe los datos. El transmisor guarda un registro de cada paquete que envía y espera un *acuse de recibo* antes

de enviar el siguiente paquete, a la vez que arranca un temporizador cuando lo envía y lo *retransmite* si dicho temporizador expira antes de que llegue el acuse de recibo.

Un protocolo simple de acuses de recibo positivos ocupa una gran cantidad sustancial de ancho de banda de red debido a que debe retrasar el envío de un nuevo paquete hasta que reciba un acuse de recibo del paquete anterior.

Capítulo 12

Otros protocolos de Enrutamiento

12.1. Protocolo Pasarela-a-Pasarela (GGP)

Los routers núcleo iniciales utilizan un protocolo de vector-distancia conocido como *Gateway-to-Gateway Protocol* para intercambiar información de enrutamiento. La información de intercambio de rutas en el GGP consiste en un conjunto de pares (N, V) , donde N es una dirección de red IP y V una distancia medida en saltos. Puede decirse que un router que usa GGP *anuncia* las redes que puede alcanzar y el costo para alcanzarlas. El GGP mide las distancias en *saltos de router*, donde un router se define en 0 saltos si está conectado directamente, un salto para redes que están conectadas a través de otro router,...

12.2. Enrutamiento Enlace-Estado (SPF)

La principal alternativa a los algoritmos de vector-distancia es una clase de algoritmos conocidos como *enlace-estado (link-state)*, *Shortest Path First (Primero la Ruta más Corta)* ó *SPF*. Estos algoritmos requieren que cada router participante tenga información sobre la topología completa. Un router que participa en un algoritmo SPF realiza 2 tareas: prueba activamente el estado de todos los routers vecinos y difunde periódicamente la información del estado del enlace hacia otros routers.

12.3. Protocolo de Pasarela Exterior (EGP)

A dos routers que intercambian información de enrutamiento se les llama *vecinos exteriores* si pertenecen a dos sistemas autónomos diferentes y *vecinos interiores* si pertenecen al mismo sistema autónomo. El protocolo que emplea vecinos exteriores para difundir la información de accesibilidad a otros sistemas autónomos se le conoce como *Protocolo de Pasarela Exterior (Exterior Gateway Protocol)* ó *EGP* y los routers que se utilizan se conocen como *routers exteriores*.

12.4. Enrutamiento Interior

Un solo router puede utilizar dos diferentes protocolos de enrutamiento simultáneamente, uno para la comunicación al exterior del sistema autónomo y otro para la comunicación al interior del sistema autónomo.

12.4.1. Protocolo de Información de Enrutamiento (RIP)

El protocolo subyacente RIP es consecuencia directa de la implantación del enrutamiento de vector-distancia para redes locales. En principio, divide las máquinas participantes en *activas* y *pasivas (silenciosas)*. Los routers activos anuncian sus rutas a los otros; las máquinas pasivas listan y actualizan sus rutas con base a estos anuncios, pero no anuncian. Sólo un router puede correr RIP en modo activo, un host debe utilizar el modo pasivo.

Un router que corre RIP de modo activo difunde un mensaje cada 30 segundos. El mensaje contiene información tomada de la base de datos de enrutamiento actualizada. Cada mensaje consiste de pares,

donde cada par contiene una dirección de red IP y un entero que representa la distancia hacia esta red. RIP utiliza una *métrica de conteo de saltos* (*hop count metric*) para medir la distancia hacia un destino.

Debe ser obvio que utilizar el conteo de saltos para calcular la trayectoria más corta no siempre produce resultados óptimos.

Para prevenir que los routers oscilen entre dos o más trayectorias de costos iguales, RIP especifica que se deben conservar las rutas existentes hasta que aparezca una ruta nueva con un costo estrictamente menor.

12.4.2. Protocolo de SPF Abierto (OSPF)

Este nuevo protocolo, *Open SPF*, propone varios objetivos ambiciosos:

- Incluye un *enrutamiento de servicio de tipo*, los administradores pueden instalar múltiples rutas hacia un destino dado, uno por cada tipo de servicio.
- Proporciona *balance de carga*, si un administrador especifica múltiples rutas hacia un destino con el mismo costo, OSPF distribuye el tráfico entre ellas de la misma manera (protocolos como RIP calculan una sola ruta por destino).
- Para permitir el crecimiento y hacer las redes de una localidad fáciles de manejar, OSPF permite que una localidad divida sus redes y routers en subconjuntos llamados *áreas*.
- El protocolo OSPF especifica que todos los intercambios entre routers deben ser *autenticados*, para garantizar que sólo routers confiables difundan información de enrutamiento.
- El OSPF permite a los routers intercambiar información de enrutamiento aprendido desde otras localidades (externas).

Capítulo 13

Multidifusión (Multicast) Internet (IGMP)

13.1. Difusión por hardware (Broadcast)

Muchas tecnologías de hardware tienen mecanismos para enviar paquetes hacia destinos múltiples simultáneamente. La entrega por difusión (*multicast*) significa que la red entrega una copia de un paquete para cada destino. En tecnologías como Ethernet, la difusión puede completarse con la transmisión de un solo paquete. En las redes compuestas por conmutadores con conexiones punto a punto, el software tiene que implantar la difusión enviando copias de los paquetes a través de conexiones individuales hasta que todas las computadoras han recibido una copia.

En el caso de la mayor parte del hardware, el usuario especifica la entrega de difusión enviando el paquete hacia una dirección de destino especial y reservada, llamada *dirección de difusión*.

La mayor desventaja de la difusión es que toda difusión consume recursos en todas las máquinas.

13.2. Multidifusión por hardware (Multicast)

Algunas tecnologías de hardware soportan una segunda forma de entrega de multipunto, menos común, llamada *multidifusión*, que permite que cada máquina decida si quiere participar en ella. Por lo general, una tecnología hardware reserva un conjunto extenso de direcciones para usarse con la multidifusión y cuando un grupo de máquinas quiere comunicarse selecciona una para utilizarla durante la comunicación. Luego de configurar el hardware de interfaz de red para reconocer la dirección de multidifusión seleccionada, todas las máquinas en el grupo recibirán una copia de cada paquete enviado hacia tal dirección.

El direccionamiento de multidifusión puede considerarse una generalización de todas las otras formas de difusión: podemos pensar en el direccionamiento de difusión como en una forma de multidifusión en la que cada máquina es un miembro de un grupo de multidifusión, y el intercambio entre dos hosts como un grupo formado sólo por las dos máquinas.

13.3. Multidifusión IP

La *multidifusión IP* es la abstracción de red del hardware de multidifusión, permite la transmisión de un datagrama IP a un conjunto de hosts que forma un sólo grupo de multidifusión, que pueden incluso estar en redes físicas separadas¹ (éstos también pueden perderse, borrarse, duplicarse o entregarse sin orden).

La pertenencia a un grupo de multidifusión IP es un proceso dinámico. Un host puede unirse o abandonar un grupo en cualquier momento. Además, un host puede ser miembro de un número indeterminado de grupos de multidifusión. Un host puede enviar datagramas a un grupo sin ser un miembro.

Cada grupo de multidifusión tiene una dirección de multidifusión única (de clase D) y se corresponden siempre aun cuando el grupo no tenga actualmente miembros (por eso se dice que son direcciones *bien conocidas*). Algunas están disponibles para usos temporales (*grupos transitorios de multidifusión*, que se crean cuando son necesarios y se desechan cuando el número de miembros llega a cero).

¹Routers especiales, *de multidifusión*, envían estos datagramas.

Para transformar una dirección de multidifusión IP en su correspondiente dirección de multidifusión Ethernet, se colocan los 23 bits de orden menor de la dirección de multidifusión IP en los 23 bits de orden inferior de la dirección de multidifusión Ethernet especial $01,00,5E,00,00,00_{16}$.

Antes de que un router de multidifusión pueda difundir información a los miembros de multidifusión, debe determinar si uno o más hosts en la red local se han unido a un grupo de multidifusión. Para hacerlo, los routers de multidifusión y los hosts utilizan el IGMP (*Protocolo de Administración de Grupos de Internet, Internet Group Management Protocol*) para comunicar información a los miembros del grupo.

IGMP es análogo a ICMP, utiliza también datagramas IP para transportar mensajes y proporciona un servicio utilizado por IP. Sin embargo, aun cuando el IGMP se vale de datagramas IP para transportar mensajes, pensamos a éste como una parte integral del IP, no como un protocolo separado. Además, IGMP es un estándar para el TCP/IP.

Conceptualmente, el IGMP tiene dos fases:

1. Cuando un host se une a un nuevo grupo de multidifusión envía un mensaje IGMP para la dirección de multidifusión “todos los hosts”, declarando su ingreso. Los routers de multidifusión local reciben el mensaje y establecen el enrutamiento necesario para difundir la información hacia otros routers de multidifusión.
2. Debido a que la pertenencia al grupo es dinámica, los routers de multidifusión local muestrean de manera periódica a los hosts en la red local para determinar qué hosts se mantienen como miembros de qué grupos. Si en un grupo no se reportan miembros después de varios muestreos, el router de multidifusión asume que no hay hosts en la red que se mantengan en el grupo y deja de anunciar miembros del grupo a otros routers de multidifusión y de transmitir mensajes para el grupo.

Capítulo 14

BOOTP y DHCP

El protocolo RARP que ya vimos tiene 3 inconvenientes:

1. Como opera a bajo nivel, su uso requiere de un acceso directo al hardware de red, de modo que puede resultar difícil o imposible para un programador de aplicaciones construir un servidor.
2. Aun cuando RARP requiere un intercambio de paquetes entre una máquina cliente y una computadora que responda a las solicitudes, la réplica contiene sólo una pequeña parte de información: la dirección IP del cliente de 32 bits. Esto puede ser molesto en redes como Ethernet que imponen un tamaño de paquete mínimo.
3. Como RARP emplea una dirección hardware de computadora para identificar una máquina, no puede usarse en redes con asignación dinámica de direcciones de hardware.

Para sortear algunas de estas dificultades de RARP, se desarrolló BOOTP (*Bootstrap Protocol*) y más recientemente DHCP (*Dynamic Host Configuration Protocol*) como su sucesor.

Dado que utiliza IP y UDP, BOOTP ha de implantarse con un programa de aplicación. Como RARP, BOOTP opera dentro de un paradigma cliente-servidor y requiere sólo de un intercambio de paquetes. No obstante, BOOTP es más eficiente que RARP porque un solo mensaje BOOTP especifica muchos más aspectos necesarios para arranque.

¿Cómo se puede usar IP antes de conocer la propia dirección IP? Un programa de aplicación puede utilizar la dirección IP de difusión límite (*broadcast*: 255,255,255,255) para obligar a IP a difundir un datagrama en la red local antes de que IP haya descubierto la dirección IP de la red local o la dirección IP de la máquina.

Aun cuando pueda no ser obvio, el servidor puede tener que utilizar también la dirección de difusión límite para su réplica aunque conozca la IP del cliente, ya que es el cliente mismo quien la desconoce.

14.1. Política de retransmisión BOOTP

BOOTP confiere toda la responsabilidad de la confiabilidad de la comunicación al cliente. Sabemos que como UDP usa IP para la entrega, los mensajes pueden retrasarse, perderse, entregarse fuera de orden o duplicarse. Además, dado que IP no proporciona una suma de verificación para los datos, el datagrama UDP puede llegar con algunos bits alterados. Para protegerse contra esto, BOOTP requiere que UDP utilice checksum. También especifica que solicitudes y réplicas han de enviarse con el bit de *no fragmentar* activado a fin de adaptarse a los clientes que tengan una memoria pequeña para reensamblar datagramas. BOOTP también permite réplicas múltiples: las acepta y procesa primero.

Para manejar datagramas perdidos, BOOTP utiliza la técnica convencional de *tiempo límite* (*time out*) y *retransmisión*. Cuando el cliente transmite una solicitud, inicia un temporizador. Si no llega ninguna réplica antes de que el tiempo expire, el cliente retransmite la solicitud. Las especificaciones aconsejan comenzar con un time out aleatorio de entre 0 y 4 segundos y duplicarlo tras cada intento. Cuando alcanza los 60 segundos no se incrementa más, pero se sigue estableciendo el valor aleatoriamente. Esto pretende evitar que BOOTP añada tráfico excesivo y congestione la red.

14.2. Procedimiento de arranque de dos pasos

BOOTP utiliza un procedimiento de arranque de dos pasos. No proporciona una imagen de memoria a los clientes, sólo la información necesaria para obtenerla (además de su dirección IP, la dirección de un router y la de un servidor). El cliente entonces usa un segundo protocolo (por ejemplo TFTP) para obtener la imagen de memoria. Este procedimiento, aunque pueda parecer innecesario, permite una clara separación entre configuración y almacenamiento. Un servidor BOOTP no necesita correr en la misma máquina que almacena las imágenes.

14.3. Configuración dinámica de host

Para manejar la asignación de direcciones de manera automática se diseñó DHCP (*Dynamic Host Configuration Protocol, protocolo de configuración dinámica de host*), que extiende BOOTP de dos formas: permite que una computadora adquiera toda la información que necesita en un solo mensaje (por ejemplo, que obtenga además de su IP, la máscara de subred. . .) y que una computadora posea una dirección IP de forma rápida y dinámica. Cada vez que una nueva computadora se conecta a la red, contacta al servidor y solicita una dirección. El servidor selecciona una de las direcciones especificadas por el administrador como disponibles para tal fin y la asigna a la computadora.

DHCP permite tres tipos de asignación de direcciones:

- *Configuración manual* (el administrador puede configurar una dirección específica para una máquina específica).
- *Configuración automática* (puede asignar una dirección permanente cuando una computadora es conectada por primera vez a la red).
- *Configuración dinámica completa* (el servidor “presta” una dirección para una computadora por tiempo limitado).

Como el BOOTP, DHCP utiliza la identidad del cliente para decidir cómo proceder.

Capítulo 15

Sistema de Nombre de Dominio (DNS)

Los primeros sistemas de computadoras forzaban a los usuarios a entender direcciones numéricas; el enlace de redes introduce el direccionamiento universal así como el software de protocolo para transformar direcciones universales en direcciones de hardware de bajo nivel. Los usuarios necesitan nombres simbólicos y significativos para nombrar las máquinas.

En una red de redes TCP/IP, la jerarquía de nombres de máquinas se asigna de acuerdo con la estructura de la organización que obtiene la autoridad para dividir el espacio de nombres y no necesariamente de acuerdo con la estructura de las interconexiones de red física.

El mecanismo que implanta una jerarquía de nombres de máquina para las redes TCP/IP se conoce como *Domain Name System* (*Sistema de Nombres* o *Nomenclatura de Dominio* o DNS).

El DNS tiene dos aspectos conceptualmente diferentes. El primero es abstracto, especifica la sintaxis del nombre y las reglas para delegar la autoridad respecto a los nombres. El segundo es concreto, especifica la implantación de un sistema de computación distribuido que transforma eficientemente los nombres en direcciones.

15.1. Asociación de nombres de dominio en direcciones

El esquema de nombres de dominio incluye un sistema distribuido, confiable y de propósito general para asociar nombres en direcciones. El sistema está distribuido en sentido técnico (un conjunto de servidores, que opera en varias localidades de manera conjunta, resuelve el problema de la asociación de nombres en direcciones), es eficiente en el sentido de que la mayor parte de los nombres se puede asociar localmente (sólo unos pocos requieren tráfico de redes), es de propósito general porque no se encuentra restringido a nombres de máquina y es confiable porque si una máquina falla, prevendrá al sistema para que opere correctamente.

El mecanismo de dominio para la asociación de nombres en direcciones consiste en sistemas independientes y cooperativos llamados *servidores de nombres*. Un servidor de nombres es un programa servidor que ofrece la asociación nombre-a-dirección, asociando nombres de dominio en direcciones IP. A menudo el software servidor se ejecuta en un procesador dedicado y la máquina se conoce como servidor de nombre. Un software cliente, llamado *solucionador de nombres* (*name resolver*) utiliza uno o más servidores de nombre cuando traduce un nombre.

¿Cómo encuentra un cliente un servidor de nombres? ¿Cómo encuentra un servidor de nombres a otros que puedan responder solicitudes que él no puede? Un cliente debe saber cómo contactar al último servidor de nombre. Para asegurarse de que el servidor de nombres puede alcanzar a otros, también el sistema de dominio requiere que cada servidor conozca la dirección del último servidor en la raíz.

Los servidores de nombres de Internet utilizan una memoria inmediata (*name caching*) para optimizar los costos de búsqueda para nombres no locales, donde se mantienen los nombres más utilizados recientemente así como un registro de dónde fue obtenida la información para la asociación de nombres.

Índice general

1. Tecnología Ethernet	2
1.1. Paquetes Ethernet	2
2. Direcciones Internet	3
2.1. Resumen de reglas especiales de direccionamiento	4
3. Transformación de direcciones Internet en direcciones Físicas (ARP)	5
3.1. Refinamientos ARP	5
4. Determinación en el arranque de una dirección Internet (RARP)	6
5. Protocolo Internet: entrega de datagramas sin conexión	7
5.1. Encapsulación de Datagramas	7
6. Enrutamiento IP	8
6.1. Entrega Directa e Indirecta	8
6.2. Manejo de los datagramas entrantes	9
7. IP: mensajes de error y de control (ICMP)	10
8. Subnetting y Supernetting	11
8.1. Proxy ARP	11
8.2. Subnetting	11
8.3. El algoritmo de Enrutamiento con Subnetting	12
8.4. Supernetting	12
9. Estratificación de Protocolos por Capas	13
9.1. Modelo ISO de 7 capas	14
9.2. Modelo TCP/IP	14
10. Protocolo de Datagrama de Usuario (UDP)	16
10.1. Encapsulación de UDP y estratificación por capas de protocolos	17
11. Servicio de Transporte de Flujo Confiable (TCP)	18
11.1. Características	18
11.2. Proporcionando Confiabilidad	18
12. Otros protocolos de Enrutamiento	20
12.1. Protocolo Pasarela-a-Pasarela (GGP)	20
12.2. Enrutamiento Enlace-Estado (SPF)	20
12.3. Protocolo de Pasarela Exterior (EGP)	20
12.4. Enrutamiento Interior	20
12.4.1. Protocolo de Información de Enrutamiento (RIP)	20
12.4.2. Protocolo de SPF Abierto (OSPF)	21
13. Multidifusión (Multicast) Internet (IGMP)	22
13.1. Difusión por hardware (Broadcast)	22
13.2. Multidifusión por hardware (Multicast)	22
13.3. Multidifusión IP	22

14. BOOTP y DHCP	24
14.1. Política de retransmisión BOOTP	24
14.2. Procedimiento de arranque de dos pasos	25
14.3. Configuración dinámica de host	25
15. Sistema de Nombre de Dominio (DNS)	26
15.1. Asociación de nombres de dominio en direcciones	26