

Redes I

Examen 2019

Emiliano Salvatori

Noviembre 2019

1. ¿Cuáles son las diferencias entre una red conmutada de paquetes de una red conmutada de circuitos?

Existen dos métodos fundamentales que permiten transportar los datos a través de una red de enlaces y conmutadores: la **conmutación de circuitos** y la **conmutación de paquetes**. En las redes de conmutación de circuitos, los recursos necesarios a lo largo de una ruta (buffers, velocidad de transmisión del enlace) que permiten establecer la comunicación entre los sistemas terminales están reservados durante el tiempo que dura la sesión entre dichos sistemas terminales. En las redes de conmutación de paquetes, estos recursos no están reservados; los mensajes de una sesión utilizan los recursos bajo petición y, en consecuencia, pueden tener que esperar (es decir, ponerse en cola) para poder acceder a un enlace de comunicaciones.

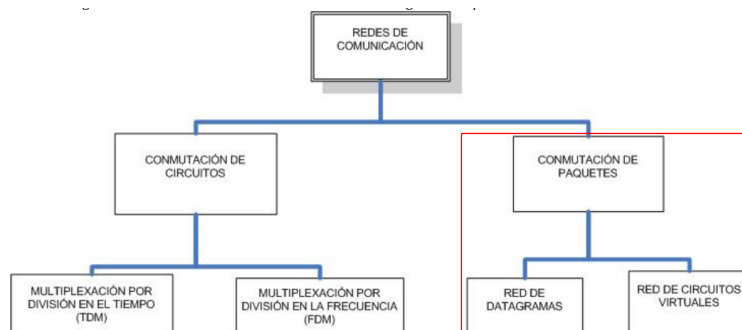
Las omnipresentes redes telefónicas son ejemplos de redes de conmutación de circuitos. Considere lo que ocurre cuando una persona desea enviar información (de voz o faxsímil) a otra a través de una red telefónica. Antes de que el emisor pueda transmitir la información, la red debe establecer una conexión entre el emisor y el receptor.

Cuando la red establece el circuito, también reserva una velocidad de transmisión constante en los enlaces de la red para el tiempo que dure la conexión. Dado que el ancho de banda para esta conexión emisor-receptor ha sido reservado, el emisor puede transferir los datos al receptor a la velocidad constante garantizada.

Veamos qué ocurre cuando un host desea enviar un paquete a otro host a través de Internet: Al igual que con la conmutación de circuitos, el paquete se transmite a través de una serie de enlaces de comunicaciones. Pero con la técnica de conmutación de paquetes, el paquete se envía a la red sin haber reservado ancho de banda. Si uno de los enlaces está congestionado porque otros paquetes tienen que ser transmitidos a través de él al mismo tiempo, entonces nuestro paquete tendrá que esperar en un buffer en el lado del emisor del enlace de transmisión y, por tanto, sufrirá un retardo. Internet realiza el máximo esfuerzo para suministrar los paquetes a tiempo, pero no existe ninguna garantía.

2. Red de Datagramas y Circuitos Virtuales

¿Cuál es la diferencia entre una Red de Datagramas y una de Circuitos Virtuales?. Comparar ambas redes



En las principales arquitecturas de redes de computadoras utilizadas hasta la fecha (Internet, ATM, frame relay, etc.), la capa de red proporciona bien un servicio sin conexión host a host o un servicio orientado a la conexión host a host, pero NO ambos. Las redes de computadoras que sólo proporcionan un servicio de conexión en la capa de red se conocen como redes de **Circuitos virtuales (VC)**; las redes que sólo proporcionan un servicio sin conexión en la capa de red se denominan **Redes de Datagramas**.

Un **Circuito Virtual** (VC por sus siglas en inglés) es un sistema de comunicación por el cual los datos de un usuario origen pueden ser transmitidos a otro usuario destino a través de más de un circuito de comunicaciones real durante un cierto periodo de tiempo, pero en el que la conmutación es transparente para el usuario.

Un ejemplo de protocolo de circuito virtual es el ampliamente utilizado TCP (Protocolo de Control de Transmisión). Es una forma de comunicación mediante conmutación de paquetes en la cual la información o datos son empaquetados en bloques que tienen un tamaño variable a los que se les denomina paquetes. El tamaño de los bloques lo estipula la red. Los paquetes suelen incluir cabeceras con información de control. Estos se transmiten a la red, la cual se encarga de su encaminamiento hasta el destino final. Cuando un paquete se encuentra con un nodo intermedio, el nodo almacena temporalmente la información y encamina los paquetes a otro nodo según las cabeceras de control.

Es importante saber que en este caso los nodos no necesitan tomar decisiones de encaminamiento, ya que la dirección a seguir viene especificada en el propio paquete. En los circuitos virtuales, al comienzo de la sesión se establece una ruta única entre las ETD (entidades terminales de datos) o los host extremos. A partir de aquí, todos los paquetes enviados entre estas entidades seguirán la misma ruta.

Las dos formas de establecer la transmisión mediante circuitos virtuales son los circuitos virtuales conmutados(SVC) y los circuitos virtuales permanentes(PVC). Los circuitos virtuales conmutados (SVC) por lo general se crean ex profeso y de forma dinámica para cada llamada o conexión, y se desconectan cuando la sesión o llamada es terminada.

En una **Red de Datagramas**, cada vez que un sistema terminal desea enviar un paquete marca el paquete con la dirección del sistema terminal de destino y luego introduce el paquete en la red. Esto se hace sin configurar ningún circuito virtual. Los routers de una red de datagramas no mantienen ninguna información de estado acerca de los Circuitos Virtuales (porque no existe ningún circuito virtual).

Cuando un paquete se transmite desde un origen a un destino pasa a través de una serie de routers. Cada uno de estos routers utiliza la dirección de destino del paquete para reenviar dicho paquete. Específicamente, cada router tiene una tabla de reenvío que asigna direcciones de destino a interfaces de enlace; cuando un paquete llega a un router, éste utiliza la dirección de destino del paquete para buscar la interfaz del enlace de salida apropiado en la tabla de reenvío. Después, el router reenvía intencionadamente el paquete a esa interfaz del enlace de salida.

Suponga que los routers de una cierta red están sujetos a condiciones que pueden hacer que fallen con frecuencia. ¿Aconsejaría una red de datagramas o una de circuitos virtuales? ¿Por qué? Justifique

Aconsejaría una **Red de Datagramas** debido a que cada uno de los paquetes enviados no están sujetos a establecer entre los distintos routers una ruta única, la cual en este ejemplo está propensa a fallar. Cuando se utiliza una Red de Datagramas, los paquetes enviados desde un Host A hasta un Host B no tienen por qué seguir una ruta preestablecida, por lo que es más confiable que lleguen los paquetes a pesar de fallos en los enlaces o en los routers, debido a que los paquetes pueden ser enviados por otros enlaces y aún así llegar a destino.

3. Dado un enlace directo dedicado de 2500km de longitud y velocidad de transmisión de 2 GB/s, se envían 50 paquetes de 250 MB cada uno. Si la velocidad de propagación del medio es de $2,5 \times 10^8$ m/segundo:

Explicar qué es el retardo de propagación. Dar la explicación matemática con la que se calcula indicando el significado de cada variable en la misma y calcular cuánto vale para el ejemplo dado.

Una vez que un bit ha entrado en el enlace desde el router A, tiene que propagarse hasta el router B. El tiempo necesario para propagarse desde el principio del enlace hasta el router B es el **Retardo de Propagación**. El bit se propaga a la velocidad de propagación del enlace. Esta velocidad depende del medio físico del enlace (es decir, que el medio sea cable de fibra óptica, cable de cobre de par trenzado, etc) y está comprendido en el rango entre: 2×10^8 m/s y 3×10^8 m/s que es igual o menor que la velocidad de la luz.

El retardo de propagación es igual a la distancia entre dos routers dividida entre la velocidad de propagación. Es decir, el retardo de propagación es igual a $\frac{d}{s}$, donde **d** es la distancia entre el router A y el router B, y **s** es la velocidad de propagación del enlace. Una vez que el último bit del paquete se ha propagado hasta el nodo B, éste y todos los bits anteriores del paquete se almacenan en el router B. A continuación, el router B lleva a cabo el reenvío.

En el ejemplo se tiene se deben convertir las unidades a metros: $2500 * 1000 = 2500000m$

Ahora bien, se tiene que:

$$d = 2500000m$$

$$s = 2 * 10^8$$

Por lo que:

$$\frac{d}{s} = \frac{2500000m}{2 * 10^8 m/s} = 0,01s$$

Explicar qué es retardo de procesamiento en el nodo y cuáles son los factores que lo provocan. ¿En dónde se producen?

El tiempo requerido para examinar la cabecera del paquete y determinar dónde hay que enviarlo es parte del **Retardo de Procesamiento**. Este tipo de retardo se producen en los routers existentes entre una comunicación de un Host A y un Host B.

El retardo de procesamiento también incluye otros factores como el tiempo necesario para comprobar los errores de nivel de bit del paquete que se producen al transmitir los bits del paquete desde el nodo anterior al router A. Los retardos de procesamiento en los routers de alta velocidad suelen ser del orden de los microsegundos o menores. Una vez efectuado el procesamiento nodal, el router dirige el paquete a la cola que precede al enlace que lleva al router B.

4. Dada la red 101.16.52.0/22. Se necesitan definir 5 subredes con las siguientes especificaciones: tres de ellas deben ser aptas para conectar 120 dispositivos cada una, y las dos subredes restantes deben permitir la conexión de 60 dispositivos.

¿Es posible cumplir con las especificaciones solicitadas? Justifique claramente indicando cuántas direcciones disponibles existen para la conexión de dispositivos en total

Para resolver este punto se debe explicitar la disponibilidad que se tiene en la dirección establecida para cada subred y para los distintos host, es por ello que se plantea de la siguiente forma:

$$11111111 \cdot 11111111 \cdot 11111100 \cdot 00000000$$

Los caracteres establecidos en 1 son los que son inamovibles, es decir, los que no pueden ser modificados ya que son los primeros 22 caracteres como bien indica la máscara de red. Lo que nos quedan son 10 dígitos restantes para poder trabajar en las distintas redes que se piden.

Si se toman los últimos dos caracteres del tercer octeto, se tendría $2^2 = 4$ posibles subredes para poder generar; el problema es que en el enunciado se piden **cinco subredes** por lo que habría que sumar un bit más para esto, es decir que se le pide un bit más al cuarto octeto.

Si se toman los primeros tres caracteres en cero, se tendría $2^3 = 8$ posibles subredes, por lo que se estaría pasando de las cinco requeridas, pero es la única posibilidad que hay para esa disposición.

Ahora bien, como se tomaron los primeros 3 dígitos de los 10 que había, entonces quedan otros 7 dígitos para poder obtener la cantidad posible de host que se pueden conectar, por lo que sería $2^7 = 128$ posibles hosts. A estos se le deben restar las dos redes por defecto: **Dirección de Red** y la **Dirección de Broadcast**, por lo que el cálculo final sería: $(2^7) - 2 = 126$ posibles conexiones para host.

Para esclarecer lo anterior, se puede visualizar el siguiente esquema:

	Red			Host
Dirección de red	10	0	0	0
	00001010	00000000	00000000	00000000
Dirección de broadcast	10	0	0	255
	00001010	00000000	00000000	11111111
Dirección host	10	0	0	1
	00001010	00000000	00000000	00000001

Teniendo en cuenta la disposición de bits que se requieren (22) que son los que se sitúan más a la izquierda, se pueden dividir de la siguiente forma la red:

$$nnnnnnnn \cdot nnnnnnnn \cdot nnnnnn00 \cdot 0hhhhhhh$$

Donde:

- Los caracteres de tipo **n** son los establecidos como dirección de red.
- Los caracteres en **0** son los establecidos para identificar las direcciones de subred.
- Los caracteres en **h** son los establecidos para identificar los posibles host a conectar.

Indicar para cada subred: Dirección de Subred, Máscara de Subred, y Dirección de Broadcast

Red 0

Por lo tanto la **Red 0** tendría las siguientes características:

- **Dirección de Red:** 101.16.52.0/22

$$01100101 \cdot 00010000 \cdot 00110100 \cdot 00000000$$

- **Dirección de Broadcast:** 101.16.52.127/22

$$01100101 \cdot 00010000 \cdot 00110100 \cdot 01111111$$

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 1

Por lo tanto la **Red 1** tendría las siguientes características:

- **Dirección de Red:** 101.16.52.128/22

$$01100101 \cdot 00010000 \cdot 00110100 \cdot 10000000$$

- **Dirección de Broadcast:** 101.16.52.255/22

$$01100101 \cdot 00010000 \cdot 00110100 \cdot 11111111$$

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 2

La **Red 2** tendrá las siguientes características:

- **Dirección de Red:** 101.16.53.0/22

$$01100101 \cdot 00010000 \cdot 00110101 \cdot 00000000$$

- **Dirección de Broadcast:** 101.16.53.127/22

$$01100101 \cdot 00010000 \cdot 00110101 \cdot 01111111$$

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 3

La **Red 3** tendrá las siguientes características:

- **Dirección de Red:** 101.16.53.128/22

$$01100101 \cdot 00010000 \cdot 00110101 \cdot 10000000$$

- **Dirección de Broadcast:** 101.16.53.255/22

$$01100101 \cdot 00010000 \cdot 00110101 \cdot 11111111$$

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 4

La **Red 4** tendrá las siguientes características:

- **Dirección de Red:** 101.16.54.0/22

01100101 · 00010000 · 00110110 · 00000000

- **Dirección de Broadcast:** 101.16.54.127/22

01100101 · 00010000 · 00110110 · 01111111

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 5

La **Red 5** tendrá las siguientes características:

- **Dirección de Red:** 101.16.54.128/22

01100101 · 00010000 · 00110110 · 10000000

- **Dirección de Broadcast:** 101.16.54.255/22

01100101 · 00010000 · 00110110 · 11111111

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 6

La **Red 6** tendrá las siguientes características:

- **Dirección de Red:** 101.16.55.0/22

01100101 · 00010000 · 00110111 · 00000000

- **Dirección de Broadcast:** 101.16.55.127/22

01100101 · 00010000 · 00110111 · 01111111

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

Red 7

La **Red 7** tendrá las siguientes características:

- **Dirección de Red:** 101.16.55.128/22

01100101 · 00010000 · 00110111 · 10000000

- **Dirección de Broadcast:** 101.16.55.255/22

01100101 · 00010000 · 00110111 · 11111111

- **Dispositivos disponibles:** $(2^7) - 2 = 126$

5. **Sunponga que los nodos A, B y C están conectados a la misma red LAN de difusión. Si A envía un datagrama IP a B, con cada trama que los encapsula dirigida a la dirección MAC de B:**

¿Procesará el adaptador de C estas tramas? Justificar

¿Pasará el adaptador de C los datagramas IP de dichas tramas a la capa de red C? Justifique

6. ¿Cuáles son los servicios que debe otorgar la capa de enlace? ¿En qué consiste el protocolo CSMA/CD? ¿Qué es una dirección MAC?

Servicios de la Capa de Enlace

Para la Capa de Enlace, nos resultará conveniente referirnos a los hosts y los routers simplemente como **nodos** ya que, no nos va a preocupar especialmente si un determinado nodo es un router o un host. También nos referimos a los canales de comunicación que conectan nodos adyacentes a lo largo de la ruta de comunicaciones con el nombre de enlaces. Para que un datagrama pueda ser transferido desde el host de origen al de destino, debe moverse a través de cada uno de los enlaces individuales que forman la ruta terminal a terminal.

Entre los posibles servicios que un protocolo de la capa de enlace puede ofrecer se incluyen:

- **Entramado:** Casi todos los protocolos de la capa de enlace encapsulan cada datagrama de la capa de red dentro de una trama de la capa de enlace antes de transmitirla a través del enlace. Una trama consta de un campo de datos, en el que se inserta el datagrama de la capa de red, y de una serie de campos de cabecera. (Una trama también puede incluir campos de cola; sin embargo, utilizaremos el término campos de cabecera para referirnos tanto a los de cabecera como a los de cola.) La estructura de la trama está especificada por el protocolo de la capa de enlace.
- **Acceso al enlace:** Un protocolo de control de acceso al medio (MAC, Médium Access Control) especifica las reglas que se utilizan para transmitir una trama a través del enlace. Para los enlaces punto a punto que tengan un único emisor en un extremo del enlace y un único receptor en el otro extremo, el protocolo MAC es muy simple (o no existe): el emisor puede enviar una trama siempre que el enlace esté inactivo. El caso más interesante es cuando hay varios nodos compartiendo un mismo enlace de difusión, en cuyo caso se presenta el denominado problema del acceso múltiple. En ese caso, el protocolo MAC sirve para coordinar la transmisión de las tramas de los múltiples nodos.
- **Entrega fiable:** Cuando un protocolo de la capa de enlace proporciona un servicio de entrega fiable, garantiza que va a transportar cada datagrama de la capa de red a través del enlace sin que se produzcan errores. Recuerde que ciertos protocolos de la capa de transporte (como TCP) también proporcionan un servicio de entrega fiable. De forma similar a los servicios de entrega fiable de la capa de transporte, el servicio de entrega fiable de la capa de enlace suele implementarse mediante reconocimientos y retransmisiones (véase la Sección 3.4). A menudo se utiliza un servicio de entrega fiable de la capa de enlace en aquellos enlaces que suelen presentar altas tasas de error, como por ejemplo en los enlaces inalámbricos, con el objetivo de corregir los errores localmente (en el enlace en el que se producen los errores), en lugar de obligar a que un protocolo de la capa de transporte o de la aplicación realice una retransmisión de datos terminal a terminal. Sin embargo, la entrega fiable en la capa de enlace puede considerarse una sobrecarga innecesaria en aquellos enlaces que tengan una baja tasa de errores de bit, incluyendo los enlaces de fibra, los coaxiales y muchos enlaces de cobre de par trenzado. Por esta razón, muchos protocolos de la capa de enlace para enlaces cableados no proporcionan un servicio de entrega fiable.
- **Control de flujo:** Los nodos situados en cada extremo de un enlace tienen una capacidad limitada de almacenamiento en buffer de las tramas. Esto puede ser un problema cuando el nodo receptor puede recibir las tramas a más velocidad de la que puede procesarlas. Sin un control de flujo, el buffer del receptor puede desbordarse con lo que las tramas se perderían. De forma similar a lo que sucede en la capa de transporte, el protocolo de la capa de enlace puede proporcionar un mecanismo de control de flujo para evitar que el nodo emisor al otro lado del enlace abrume al nodo receptor situado en el otro extremo.
- **Detección de errores:** El hardware de la capa de enlace en un nodo receptor pudiera llegar a decidir, incorrectamente, que un bit contenido en una trama es cero cuando fue transmitido como un uno, y viceversa. Dichos errores de bit se introducen debido a la atenuación de las señales y al ruido electromagnético. Puesto que no existe ninguna necesidad de reenviar un datagrama que contenga un error, muchos protocolos de la capa de enlace proporcionan un mecanismo para detectar dichos errores de bit. Esto se lleva a cabo haciendo que el nodo transmisor incluya bits de detección de errores en la trama y que el nodo receptor realice una comprobación de errores. Recuerde de los Capítulos 3 y 4 que las capas de transporte y de red de Internet también ofrecen una forma limitada de detección de errores: la suma de comprobación

de Internet. La detección de errores en la capa de enlace normalmente es más sofisticada y se implementa en hardware.

- **Corrección de errores:** La corrección de errores es similar a la detección de errores, salvo porque el receptor no sólo detecta si hay bits erróneos en la trama, sino que también determina exactamente en qué puntos de la trama se han producido los errores (y luego corrige esos errores). Algunos protocolos proporcionan corrección de errores en la capa de enlace sólo para la cabecera del-paquete en lugar de para el paquete completo. Habla remos de la detección y corrección de errores en la Sección 5.2.
- **Semiduplex y fullduplex:** Con la transmisión fullduplex, los nodos de ambos extremos de un enlace pueden transmitir paquetes al mismo tiempo. Sin embargo, con la transmisión semiduplex un mismo nodo no puede transmitir y recibir al mismo tiempo.

Protocolo CSMA/CD

El protocolo **CSMA/CD**: protocolo de acceso múltiple de Ethernet. El mismo, realiza lo siguiente:

1. Un adaptador puede comenzar a transmitir en cualquier instante; es decir, no existe el concepto de partición de tiempo.
2. Un adaptador nunca transmite una trama cuando detecta que algún otro adaptador está transmitiendo; es decir, utiliza un mecanismo de sondeo de portadora.
3. Un adaptador que está transmitiendo aborta su transmisión tan pronto como detecta que otro adaptador también está transmitiendo; es decir, utiliza un mecanismo de detección de colisiones.
4. Antes de intentar llevar a cabo una retransmisión, un adaptador espera un intervalo de tiempo aleatorio que normalmente es más pequeño que el tiempo que se tarda en transmitir una trama.

Estos mecanismos proporcionan a CSMA/CD un rendimiento mucho mejor que el del protocolo ALOHA con particiones en un entorno LAN. De hecho, si el retardo máximo de propagación entre estaciones es muy pequeño, la eficiencia de CSMA/CD puede aproximarse al 100 por ciento. Observe también que el segundo y tercer mecanismos de la lista anterior requieren que los adaptadores de Ethernet sean capaces de (1) detectar cuándo algún otro adaptador está transmitiendo y (2) detectar una colisión mientras están transmitiendo. Los adaptadores Ethernet realizan estas dos tareas midiendo los niveles de tensión antes y durante las transmisiones.

Dentro de un adaptador específico, el protocolo CSMA/CD opera de la siguiente forma:

1. El adaptador obtiene un datagrama de la capa de red, prepara una trama Ethernet y la coloca en un buffer del adaptador.
2. Si el adaptador detecta que el canal está inactivo (es decir, durante 96 periodos de bit el adaptador no recibe intensidad de señal procedente del canal), comienza a transmitir la trama. Si el adaptador detecta que el canal está ocupado, espera hasta comprobar que no hay intensidad de señal (más otros 96 periodos de bit) y luego comienza a transmitir la trama.
3. Mientras está transmitiendo, el adaptador monitoriza la presencia de señales procedentes de otros adaptadores. Si el adaptador transmite la trama completa sin detectar ninguna señal procedente de otros adaptadores, concluye que ha terminado su trabajo con esa trama.
4. Si el adaptador detecta intensidad de señal procedente de otros adaptadores mientras está transmitiendo, deja de transmitir su trama y transmite una señal de interferencia (jam) de 48 bits.
5. Después de abortar la transmisión de la trama (es decir, de transmitir la señal de interferencia), el adaptador entra en la fase de espera exponencial (backoff exponencial). Específicamente, a la hora de transmitir una determinada trama, después de experimentar la n -ésima colisión para esa trama, el adaptador selecciona un valor aleatorio para K del conjunto $0, 1, 2, \dots, 2m - 1$, donde $m = \min(n, 10)$. El adaptador espera entonces $K \cdot 512$ periodos de bit y vuelve al Paso nº 2.

Direcciones MAC

En realidad, no son los nodos (es decir, los hosts o routers) los que tienen asignadas direcciones de la capa de enlace, sino que las direcciones de la capa de enlace se asignan a los adaptadores instalados en cada nodo. A las direcciones de la capa de enlace se las denomina de diversas formas como dirección LAN, dirección física o dirección MAC.

En la mayoría de las redes LAN (incluyendo las redes Ethernet y las LAN inalámbricas 802.11), la dirección MAC tiene 6 bytes de longitud, lo que nos da 2^{48} posibles direcciones MAC. Estas direcciones de 6 bytes suelen

expresarse en notación hexadecimal, indicándose cada byte de la dirección mediante una pareja de números hexadecimales.

La dirección MAC de un adaptador es fija. Una propiedad interesante de las direcciones MAC es que nunca puede haber dos adaptadores con la misma dirección. La respuesta es que el IEEE se encarga de gestionar el espacio de direcciones MAC. En particular, cuando una empresa quiere fabricar adaptadores, compra por un precio fijado una parte del espacio de direcciones compuesto por 2^{24} direcciones. IEEE asigna el fragmento de 2^{24} direcciones fijando los primeros 24 bits de una dirección MAC y dejando que la empresa diseñe combinaciones únicas de los últimos 24 bits para cada adaptador.

Como hemos descrito al principio de esta sección, cuando un adaptador de un emisor quiere enviar una trama a otro adaptador de destino, inserta la dirección MAC del de destino en la trama y luego la envía a través de la red LAN. Si la red LAN es una LAN de difusión, la trama será recibida y procesada por todos los demás adaptadores de la LAN. En particular, cada adaptador que reciba la trama comprobará si la dirección MAC de destino contenida en la trama se corresponde con su propia dirección MAC.

Si existe una correspondencia, el adaptador extraerá el datagrama incluido en la trama y lo pasará hacia arriba por la pila de protocolos para entregárselo a su nodo padre. Si no hay una correspondencia entre ambas direcciones, el adaptador descarta la trama, sin pasar el datagrama de la capa de red hacia arriba por la pila de protocolos. De este modo, sólo el nodo de destino será interrumpido cuando se reciba la trama.

7. ¿Cuáles son las dos funciones más importantes de la Capa de Red en una Red de Datagramas? Explicar en qué consiste cada una de ellas, marcando claramente las diferencias entre ambas.

Funciones de la capa de Red

La función de la capa de red es por tanto tremendamente simple: transporta paquetes desde un host emisor a un host receptor. En la realización de esta tarea podemos identificar dos importantes funciones de la capa de red:

- **Reenvío (forwarding):** Cuando un paquete llega al enlace de entrada de un router, éste tiene que pasar el paquete al enlace de salida apropiado. Por ejemplo, un paquete que llega procedente de H1 al router R1 debe ser reenviado al siguiente router de la ruta hacia H2.

Reenvio/Forwarding: Tiene que ver con lo anterior, es mover paquetes desde una entrada del router a la salida del mismo. Cuando un paquete llega al enlace de entrada de un router, éste tiene que pasar el paquete al enlace de salida apropiado.

- **Enrutamiento (routing):** La capa de red tiene que determinar la ruta o camino que deben seguir los paquetes a medida que fluyen de un emisor a un receptor. Los algoritmos que calculan estas rutas se conocen como algoritmos de enrutamiento. Un algoritmo de enrutamiento debe determinar, por ejemplo, la ruta por la que fluirán los paquetes para ir de H1 a H2.

Ruteo/Enrutamiento/Routing: Determina una ruta de una punta a la otra, desde origen a destino. La capa de red tiene que determinar la ruta o camino que deben seguir los paquetes a medida que fluyen de un emisor a un receptor. Los algoritmos que calculan estas rutas se conocen como algoritmos de enrutamiento. Un algoritmo de enrutamiento debe determinar, por ejemplo, la ruta por la que fluirán los paquetes para ir de un Host situado en la ciudad/país A, hasta otro situado en la ciudad/país B.

Ejemplo práctico de Reenvío y Enrutamiento

Un paralelismo con lo anterior se puede poner cuando un usuario quiere realizar un viaje en auto por el país. Decidir la ruta que se tomará para llegar a determinada provincia desde la casa del usuario sería el *enrutamiento*. En cambio, llegado a una ciudad (que sería para el ejemplo como ser un router), decidir por qué calle tomar para dar con la ruta que me llevará al próximo pueblo sería el *reenvío*.

El reenvío hace referencia a la acción local que realiza un router al transferir un paquete desde una interfaz de un enlace de entrada a una interfaz del enlace de salida apropiada. El enrutamiento hace referencia al proceso que realiza la red en conjunto para determinar las rutas terminal a terminal que los paquetes siguen desde el origen al destino.

Para saber la mejor ruta los routers corren algoritmos que van a determinar la mejor ruta para ir de un host a otro, una vez que termina el algoritmo genera una tabla de reenvío, esa tabla está conformada como si fuera un algoritmo de Dijkstra. Evalúa dándole determinado peso entre routers.

Todo router tiene una tabla de reenvío. Un router reenvía un paquete examinando el valor de un campo de la cabecera del paquete entrante y utilizando después ese valor para indexarlo dentro de la tabla de reenvío del

router. El resultado de la tabla de reenvío indica a cuál de las interfaces del enlace de salida del router será reenviado el paquete. Dependiendo del protocolo de la capa de red, este valor de la cabecera del paquete podría ser la dirección de destino del paquete o una indicación de la conexión a la que pertenece el paquete.

Una vez que el router pasa el paquete a otro, se olvida del paquete que envió. Si cada router corre el mismo algoritmo proporcionado por el protocolo que se ejecute, todos llegan a la misma conclusión de que la mejor ruta es una, y en base a ello es que se genera la tabla de reenvío; la problemática es que todos corran el mismo algoritmo y saber cuál usar.

Cada router corre el algoritmo, genera la tabla, determina la mejor ruta, envía los paquetes; los algoritmos se corren cada determinado tiempo, para que se actualice la tabla de ruteo; y todo esto se denomina *Ruteo dinámico*. *Ruteo estático*: es cuando se define vía hardware por dónde debe salir un paquete determinado.

8. Respecto a los algoritmos de enrutamiento de Estado de Enlace y de Vector distancia:

- Explicar en qué consiste cada uno, indicando claramente las diferencias y las coincidencias entre ambos.
- ¿Es necesario que todos los sistemas autónomos utilicen el mismo algoritmo de enrutamiento interno? ¿por qué?.

Algoritmo de enrutamiento de Estado de Enlaces (LS)

Recuerde que en un algoritmo de Estado de Enlaces, la topología de la red y el coste de todos los enlaces **son conocidos**; es decir, están disponibles como entradas para el algoritmo LS.

En la práctica, esto se consigue haciendo que cada nodo difunda paquetes del estado de los enlaces a todos los demás nodos de la red, con cada paquete de estado de enlace conteniendo las identidades y los costes de sus enlaces conectados. En la práctica (por ejemplo, con el protocolo de enrutamiento OSPF de Internet), esto suele conseguirse mediante un algoritmo de difusión de estado de enlaces.

El resultado de difundir la información de los nodos es que todos los nodos tienen una visión completa e idéntica de la red. Cada nodo puede entonces ejecutar el algoritmo LS y calcular el mismo conjunto de rutas de coste mínimo que cualquier otro nodo.

El algoritmo de enrutamiento de Estado de Enlaces que presentamos a continuación se conoce como algoritmo de Dijkstra, en honor a su inventor. El algoritmo de Dijkstra calcula la ruta de coste mínimo desde un nodo (el origen, al que denominaremos *u*) hasta todos los demás nodos de la red. El algoritmo de Dijkstra es iterativo y tiene la propiedad de que después de la *k*-ésima iteración del algoritmo, se conocen las rutas de coste mínimo hacia *k* nodos de destino y entre las rutas de coste mínimo a todos los nodos de destino, estas *k* rutas tendrán los *k* costes más pequeños.

Algoritmo de enrutamiento por Vector Distancia (DV)

Mientras que el algoritmo LS es un algoritmo que emplea información global, el algoritmo por Vector de Distancias (DV) es iterativo, asincrónico y distribuido.

- Es **distribuido** en el sentido de que cada nodo recibe información de uno o más de sus vecinos directamente conectados, realiza un cálculo y luego distribuye los resultados de su cálculo de vuelta a sus vecinos.
- Es **iterativo** porque este proceso continúa hasta que no hay disponible más información para ser intercambiada entre los vecinos. (Además, el algoritmo también finaliza por sí mismo, es decir, no existe ninguna señal que indique que los cálculos deberían detenerse; simplemente se detienen).
- El algoritmo es **asincrónico**, en el sentido de que no requiere que todos los nodos operen sincronizados entre sí.

Como tendremos oportunidad de ver, un algoritmo asincrónico, iterativo, distribuido y que finaliza por sí mismo es mucho más interesante y divertido que un algoritmo centralizado.