

Redes I

Examen Integrador 06/2018

Emiliano Salvatori

Noviembre 2019

1. ¿Cuáles son las características de una Red de Circuitos Conmutados y una Red de Paquetes Conmutados?

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.

Conmutación de circuitos

La **conmutación de circuitos** es un tipo de conexión que realizan los diferentes nodos de una red para lograr un camino apropiado para conectar dos usuarios de una red de telecomunicaciones. A diferencia de lo que ocurre en la conmutación de paquetes, en este tipo de conmutación se establece un canal de comunicaciones dedicado entre dos estaciones. Se reservan recursos de transmisión y de conmutación de la red para su uso exclusivo en el circuito durante la conexión. Ésta es transparente: una vez establecida parece como si los dispositivos estuvieran realmente conectados.

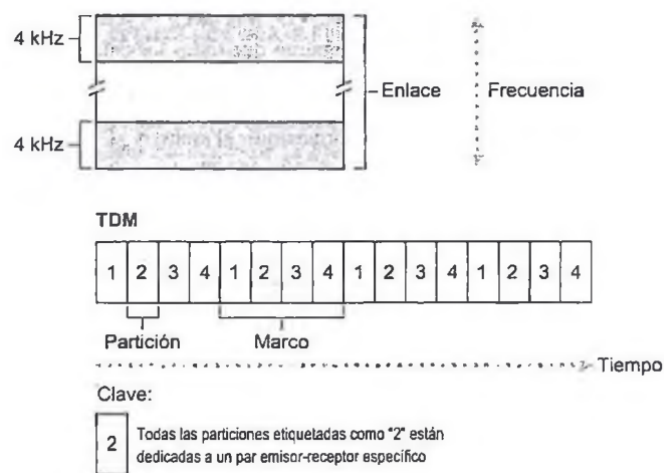
La comunicación por conmutación de circuitos implica tres fases: el establecimiento del circuito, la transferencia de datos y la desconexión del circuito. Una vez que el camino entre el origen y el destino queda fijado, queda reservado un ancho de banda fijo hasta que la comunicación se termine. Para comunicarse con otro destino, el origen debe primero finalizar la conexión establecida. Los nodos deben tener capacidad de conmutación y de canal suficiente como para gestionar las conexiones solicitadas; los conmutadores deben contar con la inteligencia necesaria para realizar estas reservas y establecer una ruta a través de la red.

En esta red, los cuatro conmutadores de circuitos están interconectados mediante cuatro enlaces. Cada uno de los enlaces tiene n circuitos, por lo que cada enlace puede dar soporte a n conexiones simultáneas. Cada uno de los hosts (por ejemplo, los PC y estaciones de trabajo) está conectado directamente a uno de los conmutadores. Cuando dos hosts desean comunicarse, la red establece una conexión terminal a terminal dedicada entre ellos. Por tanto, para que el host A envíe mensajes al host B, la red tiene que reservar en primer lugar un circuito en cada uno de los dos enlaces. Dado que cada enlace tiene n circuitos, para cada enlace utilizado por la conexión terminal a terminal, la conexión obtiene una fracción $\frac{1}{n}$ del ancho de banda del enlace para el tiempo de duración de la conexión.

1.0.1. Multiplexación en Redes de Conmutación de Circuitos

Con **FDM**, el espectro de frecuencia de un enlace se reparte entre las conexiones establecidas a lo largo del enlace. Específicamente, el enlace dedica una banda de frecuencias a cada conexión durante el tiempo que ésta dure. En las redes telefónicas, esta banda de frecuencias normalmente tiene un ancho de 4 kHz (es decir, 4.000 hercios o 4.000 ciclos por segundo). El ancho de esta banda se denomina lógicamente ancho de banda. Las estaciones de radio FM también emplean la multiplexación FDM para compartir el espectro de frecuencias entre 88 MHz y 108 MHz, teniendo cada estación asignada una banda de frecuencias específica.

En un enlace **TDM**, el tiempo se divide en marcos de duración fija y cada marco se divide en un número fijo de particiones. Cuando la red establece una conexión a través de un enlace, la red dedica una partición de cada marco a dicha conexión. Estas particiones están dedicadas para uso exclusivo de dicha conexión con una partición disponible para utilizar (en cada marco) para transmitir los datos de la conexión.



1.1. Conmutación de paquetes

Un paquete es un grupo de información que consta de dos partes: los datos propiamente dichos y la información de control, que indica la ruta a seguir a lo largo de la red hasta el destino del paquete. La **conmutación de paquetes** es un método de agrupar los datos transmitidos a través de una red digital en paquetes que se componen de un encabezado y una carga útil. Los datos en el encabezado son utilizados por el hardware de red para dirigir el paquete a su destino donde la carga útil es extraída y utilizada por el software de la aplicación. La conmutación de paquetes es la base principal de las comunicaciones de datos en redes informáticas de todo el mundo.

Las aplicaciones distribuidas intercambian mensajes para llevar a cabo sus tareas. Los mensajes pueden contener cualquier cosa que el diseñador del protocolo desee. Los mensajes pueden realizar una función de control (por ejemplo, los mensajes de saludo "Hola" del ejemplo anterior sobre establecimiento de la comunicación) o pueden contener datos, como por ejemplo un mensaje de correo electrónico, una imagen JPEG o un archivo de audio MP3. En las redes de computadoras modernas, el origen divide los mensajes largos en fragmentos de datos más pequeños que se conocen como paquetes. Entre el origen y el destino, cada uno de estos paquetes viaja a través de los enlaces de comunicaciones y de los conmutadores de paquetes (de los que existen dos tipos predominantes: los routers y los switches de la capa de enlace). Los paquetes se transmiten a través de cada enlace de comunicaciones a una velocidad igual a la velocidad de transmisión máxima del enlace.

La mayoría de los conmutadores de paquetes emplean el método de transmisión de almacenamiento y reenvío en las entradas de los enlaces. Transmisión de almacenamiento y reenvío significa que el conmutador tiene que recibir el paquete completo antes de poder comenzar a transmitir el primer bit del paquete al enlace de salida. Por tanto, los conmutadores de paquetes de almacenamiento y reenvío añaden un retardo de almacenamiento y reenvío en la entrada de cada enlace existente a lo largo de la ruta que debe seguir el paquete. Veamos el tiempo que se tarda en enviar un paquete de L bits desde un host a otro host en una red de conmutación de paquetes. Supongamos que existen Q enlaces entre los dos hosts, y que la velocidad en cada uno de ellos es igual a R bps. Suponemos que éste es el único paquete presente en la red. En primer lugar, el paquete tiene que enviarse a través del primer enlace que sale del host A, lo que consume un tiempo de $\frac{L}{R}$ segundos. A continuación, tiene que ser transmitido por cada uno de los $Q - 1$ enlaces restantes; es decir, se tiene que almacenar y reenviar $Q - 1$ veces, añadiéndose cada vez un retardo de almacenamiento y reenvío de $\frac{L}{R}$. Por tanto, el retardo total es igual a $\frac{Q \cdot R}{L}$.

Cada conmutador de paquetes tiene varios enlaces conectados a él y para cada enlace conectado, el conmutador de paquetes dispone de un buffer de salida (también denominado cola de salida), que almacena los paquetes que el router enviará a través de dicho enlace. El buffer de salida desempeña un papel clave en la conmutación de paquetes. Si un paquete entrante tiene que ser transmitido a través de un enlace, pero se encuentra con que el enlace está ocupado transmitiendo otro paquete, el paquete entrante tendrá que esperar en el buffer de salida. Por tanto, además de los retardos de almacenamiento y reenvío, los paquetes se ven afectados por los retardos de cola del buffer de salida. Estos retardos son variables y dependen del nivel de congestión de la red. Puesto que la cantidad de espacio en el buffer es finita, un paquete entrante puede encontrarse con que el buffer está completamente lleno con otros paquetes que esperan a ser transmitidos. En este caso, se producirá una pérdida de paquetes, bien el paquete que acaba de llegar o uno que ya se encuentra en la cola será descartado.

2. Dado un paquete de longitud igual a 2000bytes, que se transmite a través de un enlace a una distancia de 2500km, siendo la velocidad de propagación de $2,5 \cdot 10^8$ m/s y la velocidad de transmisión de 2Mb/s

Suponiendo que los paquetes se transmiten de manera que el primero que llega es el primero que sale, lo que es una práctica común en las redes de conmutación de paquetes, nuestro paquete sólo puede ser transmitido después de que todos los paquetes que hayan llegado antes que él hayan sido transmitidos. Sea la longitud del paquete igual a L bits y la velocidad de transmisión del enlace del router A hasta el router B igual a R bits/segundo. Entonces, por ejemplo, para un enlace Ethernet a 10 Mbps, la velocidad es $R = 10$ Mbps; para un enlace Ethernet a 100 Mbps, la velocidad será $R = 100$ Mbps. Por lo que:

$$R_T = \frac{L}{R}$$

2.1. Calcule el retardo de transmisión. ¿Depende este retardo de la longitud del paquete?

Siendo R_T el **Retardo de Transmisión**, L la **Longitud del paquete**, y R la **Velocidad de Transmisión**, entonces se tiene:

$$R_T = \frac{2000b}{2048b/s} \approx 0,98s$$

En el **Retardo de Transmisión** depende directamente de la longitud del paquete que se esté tratando de transmitir, ya que el paquete será transmitido sólo después de que todos los paquetes que hayan llegado antes que él hayan sido transmitidos.

2.2. Calcule el retardo de propagación. ¿Depende este retardo de la longitud del paquete? ¿y de la velocidad de transmisión?

Una vez que un bit ha entrado en el enlace, 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 es igual o menor a la velocidad de la luz.

El **Retardo de propagación** es igual a la distancia entre dos routers dividida entre la velocidad de propagación.

Siendo R_P el **Retardo de propagación**, D la **Distancia** entre dos routers y V_p la **Velocidad de Propagación** entonces se tiene:

$$R_P = \frac{D}{V_p} = \frac{2500000m}{2,5 \cdot 10^8 m/s} \approx 9259,25s$$

En el **Retardo de Propagación** NO importa la longitud del paquete, sino que lo que importa es el medio físico del enlace y la distancia entre los dos dispositivos (routers en este caso) que tienen que intercambiar los paquetes.

3. ¿Cuáles son las características principales 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.

Reenvío/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.

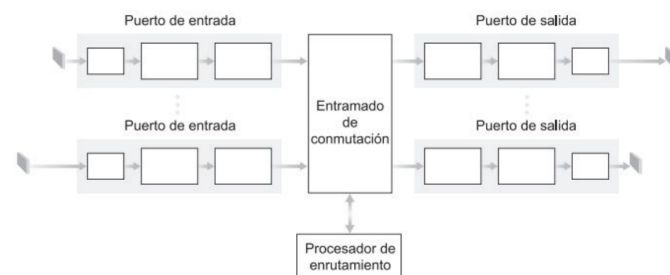
Todo router tiene una tabla de reenvío. Un router reenvía un paquete examinando el valor de un campo de la cabecera del paquete entrante y utilizando después ese valor para indexarlo dentro de la tabla de reenvío del router. El resultado de la tabla de reenvío indica a cuál de las interfaces del enlace 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.

4. ¿Cuáles son las funciones de un router? Realice un diagrama de bloques de un Router y describa sus módulos.

La función principal de un router es la *transferencia real de paquetes desde los enlaces de entrada de un router a los apropiados enlaces de salida*.



En un router se pueden identificar cuatro componentes:

1. **Puertos de entrada:** El puerto de entrada realiza varias funciones. Lleva a cabo las funciones de la capa física (representadas por el recuadro situado más a la izquierda del puerto de entrada y el recuadro más a la derecha del puerto de salida en la Figura) consistentes en la terminación de un enlace físico de entrada a un router. Realiza las funciones de la capa de enlace de datos (representadas por los recuadros centrales de los puertos de entrada y de salida) necesarias para interoperar con las funciones de la capa de enlace de datos en el lado remoto del enlace de entrada. También realiza una función de búsqueda y reenvío (el recuadro más a la derecha del puerto de entrada y el recuadro más a la izquierda del puerto de salida)

de modo que un paquete reenviado dentro del entramado de conmutación del router emerge en el puerto de salida apropiado. Los paquetes de control (por ejemplo, paquetes que transportan la información del protocolo de enrutamiento) son reenviados desde un puerto de entrada al procesador de enrutamiento. En la práctica, suelen agruparse varios puertos en una única tarjeta de línea (line card) dentro del router.

2. **Entramado de conmutación:** El entramado de conmutación conecta los puertos de entrada del router a sus puertos de salida. Este entramado de conmutación está completamente contenido dentro del router.
3. **Puertos de salida:** Un puerto de salida almacena los paquetes que le han sido reenviados a través del entramado de conmutación y los transmite al enlace de salida. Así, el puerto de salida lleva a cabo la función inversa de la capa física y de la capa de enlace de datos que el puerto de entrada. Cuando un enlace es bidireccional (es decir, transporta tráfico en ambas direcciones), un puerto de salida del enlace normalmente estará emparejado con otro puerto de entrada de dicho enlace en la misma tarjeta de línea.
4. **Procesador de enrutamiento:** El procesador de enrutamiento ejecuta los protocolos de enrutamiento mantiene la información de enrutamiento y las tablas de reenvío y realiza funciones de gestión de red dentro del router.

5. ¿Para qué se utiliza el comando Tracert? ¿Cómo se utiliza?

El comando **tracert** es un comando del símbolo del sistema que se utiliza para mostrar varios detalles sobre la ruta que toma un paquete desde el equipo o dispositivo en el que se encuentra hasta el destino que se especifique.

Tracert determinará la ruta llevada a un destino. Para ello, envía mensajes de solicitud de eco del Protocolo de mensajes de control de Internet (ICMP) al destino. Al enviar el tráfico al destino, aumentará de manera creciente los valores del campo Tiempo de vida (TTL) para ayudar a encontrar la ruta a esa dirección de destino.

El comando **tracert**, o **tracert** es similar al comando **ping**, pero este proporciona información sobre la ruta que toma un paquete. Tracert envía paquetes a un destino, pidiendo a cada enrutador de Internet que responda cuando pase el paquete. Esto le mostrará los paquetes de ruta que toma cuando los envía entre su ubicación y un destino.

5.1. Ejemplo

```
tracert 192.168.1.1
```

En el ejemplo anterior, el comando **tracert** se usaba para mostrar la ruta desde la computadora en red en la cual el comando **tracert** está siendo ejecutado por un dispositivo de red, en este caso, un router en una red local, que tiene asignada la dirección IP 192.168.1.1.

6. Detalle las características del protocolo IP. ¿A qué capa pertenece? ¿qué tipos de campos y/o servicios hay en el formato de un datagrama?

El protocolo de la **Capa de Red** de Internet es el **Protocolo IP (Internet Protocol)**. IP proporciona una comunicación lógica entre hosts. El modelo de servicio de IP es un servicio de entrega de mejor esfuerzo (best effort). Esto quiere decir que IP hace todo lo que puede por entregar los segmentos entre los hosts que se están comunicando, pero no garantiza la entrega. En particular, no garantiza la entrega de los segmentos, no garantiza que los segmentos se entreguen en orden y no garantiza la integridad de los datos contenidos en los segmentos. Por estas razones, se dice que IP es un servicio no fiable. Además, sabemos que todos los hosts tienen al menos una dirección de capa de red, que se conoce como dirección IP. Lo único que necesitamos saber es que todo host tiene una dirección IP asociada.

Los campos clave de los datagramas IPv4 son los siguientes:

- **Número de versión:** Estos 4 bits especifican la versión del protocolo IP del datagrama. A partir del número de versión el router puede determinar cómo interpretar el resto del datagrama IP. Las distintas versiones de IP utilizan distintos formatos de datagrama.
- **Longitud de cabecera:** Puesto que un datagrama IPv4 puede contener un número variable de opciones (las que se incluyen en la cabecera del datagrama IPv4), estos 4 bits son necesarios para determinar dónde comienzan realmente los datos del datagrama IP. La mayoría de los datagramas IP no contienen opciones, por lo que el datagrama IP típico tiene una cabecera de 20 bytes.

- **Tipo de servicio:** Los bits del tipo de servicio (TOS, Type of service) se incluyeron en la cabecera de IPv4 con el fin de poder diferenciar entre distintos tipos de datagramas IP (por ejemplo, datagramas que requieran en particular un bajo retardo, una alta tasa de transferencia o una entrega fiable). Por ejemplo, puede resultar útil diferenciar datagramas en tiempo real (como los utilizados en aplicaciones de telefonía IP) del tráfico que no es en tiempo real (como por ejemplo el tráfico FTP). El nivel específico de servicio que se proporcione es una política que determinará el administrador del router.
- **Longitud del datagrama:** Es la longitud total del datagrama IP (la cabecera más los datos) en bytes. Puesto que este campo tiene una longitud de 16 bits, el tamaño máximo teórico del datagrama IP es de 65.535 bytes. Sin embargo, los datagramas rara vez tienen una longitud mayor de 1.500 bytes.
- **Identificador, indicadores, desplazamiento de fragmentación:** Estos tres campos tienen que ver con lo que se denomina fragmentación IP. Es interesante comentar que la nueva versión de IP, IPv6, NO permite la fragmentación en los routers.
- **Tiempo de vida:** El campo Tiempo de vida (TTL, Time-To-Live) se incluye con el fin de garantizar que los datagramas no estarán eternamente en circulación a través de la red (debido, por ejemplo, a un bucle de enrutamiento de larga duración). Este campo se decrementa en una unidad cada vez que un router procesa un datagrama. Si el campo TTL alcanza el valor 0, el datagrama tiene que ser descartado.
- **Protocolo:** Este campo sólo se emplea cuando un datagrama IP alcanza su destino final. El valor de este campo indica el protocolo específico de la capa de transporte al que se pasarán los datos contenidos en ese datagrama IP. Por ejemplo, un valor de 6 indica que los datos se pasan a TCP, mientras que un valor igual a 17 indica que los datos se pasan a UDP. Puede obtener una lista de todos los valores posibles en [IANA Protocol Numbers 2009]. Observe que el número de protocolo especificado en el datagrama IP desempeña un papel análogo al del campo que almacena el número de puerto de un segmento de la capa de transporte. El número de protocolo es el elemento que enlaza las capas de red y de transporte, mientras que el número de puerto es el componente que enlaza las capas de transporte y de aplicación. En el Capítulo 5 veremos que la trama de la capa de enlace también contiene un campo especial que enlaza la capa de enlace con la capa de red.
- **Suma de comprobación de cabecera:** La suma de comprobación de cabecera ayuda a los routers a detectar errores de bit en un datagrama IP recibido. Esta suma de comprobación se calcula tratando cada pareja de 2 bytes de la cabecera como un número y sumando dichos números utilizando aritmética de complemento a 1. Como se ha visto en la Sección 3.3, el complemento a 1 de esta suma, conocida como suma de comprobación Internet, se almacena en el campo Suma de comprobación. Un router calcula la suma de comprobación de cabecera para cada datagrama IP recibido y detecta una condición de error si la suma de comprobación incluida en la cabecera del datagrama no coincide con la suma de comprobación calculada. Normalmente, los routers descartan los datagramas en los que se ha detectado que existe un error. Observe que la suma de comprobación tiene que volver a calcularse y almacenarse en cada router, ya que el campo TTL y, posiblemente, también el campo de opciones pueden cambiar. Una cuestión que suele plantearse en este punto es ¿por qué TCP/IP lleva a cabo una comprobación de errores tanto en la capa de transporte como en la capa de red? Existen varias razones para esta redundancia. En primer lugar, fíjese en que en la capa IP sólo se calcula la suma de comprobación para la cabecera IP, mientras que la suma de comprobación TCP/UDP se calcula sobre el segmento TCP/UDP completo. En segundo lugar, TCP/UDP e IP no necesariamente tienen que pertenecer a la misma pila de protocolos. En principio, TCP puede ejecutarse sobre un protocolo diferente (por ejemplo, ATM) e IP puede transportar datos que no se pasarán a TCP/UDP.
- **Direcciones IP de origen y de destino:** Cuando un origen crea un datagrama, inserta su dirección IP en el campo de dirección IP de origen e inserta la dirección del destino final en el campo de dirección IP de destino. A menudo el host de origen determina la dirección de destino mediante una búsqueda DNS.
- **Opciones:** El campo de opciones permite ampliar una cabecera IP. La idea original era que las opciones de cabecera rara vez se emplearan: de ahí la decisión de ahorrar recursos no incluyendo la información de los campos opcionales en la cabecera de todos los datagramas. Sin embargo, la mera existencia de opciones complica las cosas, ya que las cabeceras de datagrama pueden tener una longitud variable, por lo que no puede determinarse a priori dónde comenzará el campo de datos. Además, dado que algunos datagramas pueden requerir el procesamiento de opciones y otros no, la cantidad de tiempo necesario para procesar un datagrama IP en un router puede variar enormemente. Estas consideraciones cobran una particular importancia en el procesamiento IP realizado en los hosts y routers de altas prestaciones. Por estas razones y otras, las opciones IP fueron eliminadas en la cabecera de IPv6.

- **Datos (carga útil):** Finalmente, llegamos al último campo y más importante. En la mayoría de las circunstancias, el campo de datos del datagrama IP contiene el segmento de la capa de transporte (TCP o UDP) que va a entregarse al destino. Sin embargo, el campo de datos puede transportar otros tipos de datos, como por ejemplo mensajes ICMP.

7. Suponga un MTU de un cierto enlace está limitado a 1500bytes. Suponiendo una cabecera IP de 20 bytes ¿Cuántos datagramas se necesitarían para enviar un archivo MP3 de 5 millones de bytes?

Siendo L_t la Longitud total datagrama, E_e el Límite del enlace, y C_{IP} la Cabecera IP, se tiene:

$$\frac{L_t}{E_e} * C_{IP}$$

8. ¿Qué algoritmos de enrutamiento conoce? Detalle sus principios de funcionamiento
9. Suponiendo que cada nodo conoce inicialmente los costos a cada uno de sus vecinos. Utilizando el algoritmo de Vector Distancias, especifique las entradas de la tabla de distancias para el nodo 2
10. ¿Cuáles son los servicios que debe otorgar la Capa de Enlace?
11. ¿En qué consiste el protocolo CSMA/CD?
12. ¿Qué es un Switch? ¿Cómo funciona?