La arquitectura en capas de TCP/IP

Asignatura de Redes

Resumen de Nico Matovelle

Índice

Introducción

Conceptos

IPv4 – representaciones:

- Formato textual: 193.110.128.200
- Formato binario (uint32_t): $193 + 110 \times 2^8 + 128 \times 2^{16} + 200 \times 2^{24}$
- Puede sustituirse por un nombre de host.

Puertos – entregan los datos a la aplicación (o servicio) correcta. Cada aplicación se identifica con un número de puerto. Es un entero de 16 bits (uint16_t).

Sockets – es la interfaz entre la aplicación y la capa de transporte (como el buzón). Se construye a partir de las direcciones IP que conecta y los puertos. Se identifica con un número. Pueden estar orientados a conexión o no.

Servidor – conjunto de programa y computador que proporciona un servicio.

Cliente – conjunto de programa y computador que solicita el servicio del servidor.

Hosts – sistemas terminales, origen y destino de las transmisiones.

Enlaces – medios físicos por los que se realizan las transmisiones.

Routers – dispositivos que interconectan cables.

Protocolos – reglas y formatos para las comunicaciones (estándares de internet). Están descritos en los RFC. Hay dos tipos

- Básicos – para que funcione internet: TCP/IP, UDP…
- De aplicación – para otras aplicaciones: HTTP, SMTP…

Proveedores de servicios de internet (ISP)

- De baja escala (residenciales) – proporcionan acceso a internet a los usuarios (como el teléfono)
- De alta escala (nacionales o internacionales) – interconectan proveedores de baja escala y líneas de larga distancia.

Servicios orientados a conexión o sin conexión

Orientados a conexión

Fases

- Establecimiento de conexión
  - El cliente solicita una conexión
  - Se fijan los parámetros

    - Ambos extremos se preparan para la transmisión
- Transmisión de datos
- Desconexión

Características

- Segmentación (TCP recoge los datos que la aplicación escribe en el socket y los transforma en paquetes MSS)
- Transferencia fiable (si el receptor no envía ACK, confirmaciones, el emisor manda un paquete de nuevo)
- Control de flujo (permite que el receptor controle la tasa de envío del emisor)
- Control de gestión (permite que la tasa de envío del emisor se ajuste a las capacidades de la red)

Sin conexión

No hay fase de establecimiento de conexión, ni confirmaciones (no se sabe si el paquete llega) ni controles de flujo o de gestión. Es más rápido pero menos fiable.

El correspondiente en internet sería UDP.

Tipos de redes

Redes de Conmutación

De circuitos

Son redes en las que para su conexión se reservan unos recursos de hardware que no pueden ser usados por otras transmisiones. Al desconectarse, se liberan todos los recursos.

Pueden ser:

- Sin multiplexación (solo pueden hacer una transmisión por cada enlace de cada vez)
- Con multiplexación (reparten la capacidad del enlace entre varias transmisiones, ya sea con división de frecuencia (FDM) o con división del tiempo (TDM), creando ranuras temporales)

Son derrochadoras de recursos ya que los ocupan aunque la transmisión no los use.

De paquetes

Características

- No reservan recursos, se asignan y comparten bajo demanda.
- Trabajan con paquetes, no mandan todo de una vez (segmentación)
- Los paquetes contienen una cabecera con información de control (para llegar a su destino, ACK...)

Los routers funcionan como conmutadores de paquetes (store-and-forward: reciben el paquete completo, lo procesan y lo almacenan en la cola de envío)

Retardos en estas redes

- De procesamiento (examinar la cabecera y establecer la dirección del paquete)
- De espera en la cola
- De transmisión (tamaño de paquete / tasa de transmisión)
- De propagación (longitud del enlace / velocidad de propagación)

Segmentación – reducen el tiempo de transmisión, permiten que se intercalen unas transmisiones con otras y en caso de error es menos costoso solucionarlo.

Tipos

- Datagramas – sin conexión y encaminamiento en función del destino.
  - Cada paquete incluye en la cabecera la IP de destino
  - Reenvío: el router examina la cabecera y lo coloca en la salida más apropiada
  - No mantienen información de estado (aunque sean de la misma transmisión, los paquetes se transmiten de forma independiente)
  - Internet es una red de datagramas
- Circuitos virtuales – orientadas a conexión y encaminamiento en función del número de circuito virtual.
  - Se establece la conexión planificando una ruta al destino, su número de circuito virtual
  - Reenvío: los routers examinan el número de CV y envían a todos por el mismo sitio
  - Los routers mantienen información de estado (en la tabla de circuitos virtuales)

Redes de difusión

Son las redes Ethernet, las inalámbricas... Todos los hosts reciben las transmisiones y sólo el destinatario procesa la transmisión.

Acceso a internet

Acceso residencial

Módem

Usa la línea telefónica como si fuese una llamada de voz

Fases

- Establece la conexión (llama al número del ISP)
- Modulación (convierte la señal digital en modulada)
- Demodulación (el receptor realiza la operación inversa)

Problema: ancho de banda de frecuencia máximo = 4KHz → 56 kbps

DSL

Aprovecha todo el ancho de banda de frecuencias del cable telefónico (1MHz). Divide la frecuencia (FDM) en tres canales independientes: voz, subida y bajada (el ancho de banda de bajada es mucho mayor que los otros). Esto permite velocidades de transmisión de hasta 30 Mbps (VDSL2).

Cable HFC (híbrido, fibra y coaxial)

Consta de:

- Cabecera final (centraliza todas las conexiones)
- Líneas troncales de fibra óptica (conectan la cabecera con los nodos de fibra)
- Nodos de fibra (cabeceras intermedias)
- Ramales de cable coaxial (dan servicio de TV, teléfono e internet)

Acceso empresarial

Suele ser mediante una LAN (de Ethernet) conectada a un router que conecta a un ISP con enlace dedicado (con conexión telefónica a parte).

Medios de transmisión

Medios guiados

Cable de cobre de par trenzado

Para el teléfono suelen ser dos hilos de cobre trenzados. En cambio, para LAN se usa un cable de 4 pares trenzados (cable de climpar). Tienen problemas de pérdida de energía debido a la radiación (que se reduce al trenzarlos) y a la resistencia (la señal se convierte en calor).

Pueden ser STP (cable apantallado, shielded con una capa de metal) o UTP (sin apantallar, que puede ser de mejor o peor calidad, siendo por ejemplo la 5 la más alta, para conexiones de 100 Mbps).

Cable coaxial

Constituido por dos conductores concéntricos, uno central sólido y otro formado por una malla de hilos que recubre al primero, y estando los dos separados con plástico. La ventaja de ser concéntricos es que no emiten radiación.

Existen versiones de 50 ohmios para transmitir señales digitales en redes locales y versiones de 75 ohmios, mejores, que se utilizan en banda ancha en las redes de cable HFC.

Fibra óptica

Transmiten señales luminosas en vez de señales eléctricas, por lo que evitan las pérdidas por radiación y permiten usarse en distancias de 100 km sin repetidores. Están hechas de vidrio o

plástico transparente muy puro y son muy finas, por lo que pueden ir miles en un mismo conducto.

El problema es que son más difíciles de instalar y el precio de los dispositivos que operan con ellas es mucho más elevado.

Medios no guiados

La atmósfera y el espacio. Para el wifi y esas cosas :P

Arquitectura en capas

- Facilita el diseño de protocolos
- Divide la comunicación en tareas independientes
- Las capas superiores se usan para los servicios inferiores
- Tiene modularidad: hay que respetar las especificaciones de cada capacidad
- La principal que vamos a estudiar es la TCP/IP. Consta de cada de Aplicación, de Transporte, de Red, de Enlace y Física.
    - Tipo de cabecera de un paquete:
        - Cabecera de enlace
        - Cabecera IP (con IP de origen e IP de destino)
        - Cabecera TCP o UDP (con puerto de origen y puerto de destino)

Explicamos las capas y sus componentes a continuación.

Capa de aplicación

Es la capa en la que se localizan los procesos que se comunican entre sí mediante mensajes. Existen protocolos de aplicación como HTTP, SMTP...

Se podría representar con un esquema en el que dos aplicaciones, cliente y servidor, se comunican mediante una relación de petición y respuesta.

Los protocolos de comunicación de esta capa son necesarios ya que es imprescindible la comprensión de los mensajes que se envían. Estos protocolos facilitan la programación de las funciones de envío y recepción. En estos protocolos se debe especificar:

- El tipo de mensajes que se intercambian (petición, respuesta...)
- Las reglas de cómo y cuándo se envían los mensajes
- La sintaxis y la semántica del mensaje (especificar los campos de los que consta el mensaje y cómo rellenamos cada uno)

Los protocolos que se definen en la capa de aplicación son los que más tarde usan los protocolos de la capa de transporte. Estos pueden ser de transferencia de datos fiable (TCP, orientado a conexión) o no fiable (UDP, más rápido pero sin conexión). Además, el funcionamiento mejor o peor de los servicios de la capa de aplicación depende también del ancho de banda y de la temporalización.

A continuación trataremos protocolos de los dos tipos, de TCP (HTTP, SMTP, POP3, FTP) y de UDP (DNS).

HTTP (HyperText Transfer Protocol)

Es un protocolo que define la comunicación entre los servidores web y los clientes web. Usa TCP y por defecto, el puerto 80.

Tipos de conexiones

- Conexiones no persistentes – usan una conexión TCP distinta para transferir cada objeto, como imágenes, documentos html, scripts... (HTTP/1.0)
    - En serie – esperan a que acabe la anterior conexión para iniciar una nueva
    - En paralelo – inician varias conexiones TCP a la vez
- Conexiones persistentes – pueden transferir varios objetos (e incluso procedentes de varias páginas) con una misma conexión TCP
    - Sin entubamiento – el cliente debe esperar a recibir un objeto para pedir el siguiente
    - Con entubamiento – el cliente puede hacer peticiones de varios objetos antes de recibir los anteriores (HTTP/1.1)

Tiempo de transferencia

El primer objeto de una transferencia siempre tarda 2*RTT + $t_{transmisión}$. Los siguientes dependen del tipo de conexión.

Un RTT (Round-Trip Time) es el tiempo necesario para que un paquete pequeño (en este caso la solicitud de conexión TCP y la petición del primer paquete) vaya desde el cliente al servidor y vuelva. El $t_{transmisión}$ depende del tamaño del archivo que se esté enviando.

Tipos de mensajes – tanto los mensajes de petición como los de respuesta tienen una sección de cabecera con información en ascii del tipo de mensaje y opciones, y un cuerpo, con los datos en binario del objeto, del contenido del formulario...

Mensajes de petición

cabecera	línea de petición	método	sp	URL	sp	version	cr	lf
	líneas de cabecera	nombreCampo			sp	valor	cr	lf
	línea en blanco	cr	lf					
cuerpo		cuerpo						

El significado de estos campos es el siguiente:

- método: GET (página normal) o POST (formulario)
- sp: espacio, en formato ASCII7
- URL: la dirección del objeto que se requiere
- version: versión de HTTP en uso
- cr: retorno de carro, \r
- lf: salto de línea \n
- nombreCampo: puede haber varias líneas de este tipo. Esta variable puede ser Host (para el nombre del servidor), Connection (que puede valer "close", para conexiones no persistentes), User-agent (nombre del navegador en uso), Accept-languaje (el idioma preferido por el solicitante)
- valor: para cada posible nombreCampo también tiene que especificarse el valor que toma
- cuerpo: caso de que método=POST contiene los datos del formulario

Mensajes de respuesta

cabecera	línea de petición	versión	sp	cod	sp	frase	cr	lf
	líneas de cabecera	nombreCampo			sp	valor	cr	lf
	línea en blanco	cr	lf					
cuerpo		objeto						

El significado de estos campos es similar al de los mensajes de petición, pero:

- cod: es la codificación de la frase (200, OK; 404, Not found; 400, Bad Request)
- nombreCampo: ahora puede ser Connection (igual que en los de petición), Date (fecha de envío), Server (servidor que envía el objeto), Last-Modified (del objeto), Content-Length (en bytes), Content-Type (en formato tipo de objeto / extensión)

- cuerpo: los datos del objeto que se envía

FTP (File Transfer Protocol)

Define la comunicación con un servidor de archivos. Es una comunicación con estado (mantiene información durante toda la sesión) que usa dos conexiones TCP paralelas:

- **Conexión de control** – usa el puerto 21 y sirve para enviar los comandos (nombre y clave, dir, put, get...) y recibir las respuestas. Es una conexión persistente y usa ASCII7. Los comandos constan de 4 caracteres en mayúsculas y son los siguientes:
  - USER – nombre de usuario
  - PASS – clave
  - LIST – lista de archivos
  - RETR – nombre de fichero (traer fichero)
  - STOR – nombre de fichero (almacenar fichero)

  Las respuestas son códigos de 3 dígitos con frase explicativa.
- **Conexión de datos** – usa el puerto 20 para transmitir los archivos en respuesta a los comandos. Es no persistente y usa también ASCII7

Protocolos de correo electrónico

Para enviar correo al servidor de correo o entre servidores se usa SMTP (o HTTP si usamos un navegador). Para acceder a este correo, usamos POP3, IMAP o HTTP.

SMTP (Simple Mail Transfer Protocol)

Funcionamiento

- El agente de usuario (AU) se comunica con el servidor, en el que hay un buzón de correo
- Entre los servidores se conectan usando el puerto 25 y conexiones TCP persistentes con ASCII7
- Si el servidor destino está fuera de servicio, se reintenta (30 minutos)

Tipos de mensajes

- Comandos – HELO (nombre servidor), MAIL FROM (dirección remitente), RCPT TO (dirección destino), DATA (contenido), QUIT
- Respuestas – con códigos numéricos con frases aclaratorias
- Datos – son el contenido de los correos, con todos los objetos encapsulados

SMTP es un protocolo inseguro, ya que no pide nombre ni clave de usuario. En comparación con HTTP los dos transmiten archivos con conexiones TCP, pero HTTP es un protocolo de demanda (en SMTP ofertas el archivo), además de que en HTTP puede haber archivos binarios, en archivos diferentes y no tiene estado.

POP3 (Post Office Protocol)

No se usa SMTP porque es un protocolo de oferta, y nosotros solo queremos solicitar los correos cuando nos conectemos en vez de tener el ordenador siempre encendido recibiendo.

También tiene tres **tipos de mensajes**

- Comandos (de 4 caracteres) – user (nombre de usuario), pass (palabra clave), list, retr (número de correo, para traer correo), dele (número de correo), quit
- Respuestas – OK o ERR, con frases explicativas
- Datos – la lista de mensajes, los contenidos de los correos... o el correo completo en un único mensaje

Otras características

POP3 usa el puerto 110 con conexiones TCP persistentes. Necesita cuenta con contraseña y se puede usar para descargar o borrar los correos o simplemente descargarlos y mantenerlos en el servidor.

DNS (Domain Name System)

Sirve para traducir nombres de hosts a direcciones IP y viceversa. Consta de numerosos servidores de nombres distribuidos por internet formando parte de una gran base de datos distribuida de forma jerárquica. Es un protocolo sin conexión que usa UDP por el puerto 53 y no tiene estado. La razón de que se use UDP es que en caso de que una respuesta no sirva o no llegue bien, se prefiere enviar otra distinta en vez de seguir probando con la misma.

Servicios proporcionados

- Traducir nombres de hosts (de IP a alias y viceversa)
- Informa de cuáles son los servidores autorizados para un dominio
- Alias de servidores de correo (simplificar direcciones: @usc.es en vez de @smtp.usc.es)
- Distribución de carga: un host puede tener asignadas varias IPs o varios servidores (que hacen de espejo, teniendo el mismo contenido) y pueden devolver las IPs de forma rotatoria para no sobrecargar una sola.

Tipos de servidores

- Locales – atienden las consultas de los hosts (las nuestras. En otras redes pueden establecer nombres de host, pero no están registrados en el DNS mundial)
- Autorizados – para que un host sea accesible en internet debe estar registrado en uno de estos. Normalmente pertenecen a un ISP y cada host debe estar en dos servidores autorizados por fiabilidad (estos también se comportan como locales)
- Intermedios o LTD – con información de niveles intermedios (nicomt.com, usc.es...)
- Raíz – debe de haber como 12 por todo internet. Tienen la información de los dominios de primer nivel (es, com, org...)

Tipos de consultas

- Consultas recursivas – cada servidor DNS interroga al siguiente
- Consultas iterativas – nuestro servidor DNS local contacta con cada uno de los servidores

Caché – los servidores almacenan copias locales de las correspondencias que obtienen, que solo se borran después de cierto tiempo sin usar. Pueden haber copias en cualquier nivel de la jerarquía, incluso en los hosts locales.

Mensajes: consultas y respuestas

- Cabecera con información de control
  - Identificación: 16 bits que identifican la consulta y su correspondiente respuesta
  - Señales: 4 bits que indican si es consulta o respuesta (y tipo de ésta)
  - Tamaño de los campos del cuerpo
- Cuerpo de 4 campos
  - Cuestiones: una o varias preguntas
  - Respuestas: para cada cuestión puede haber varias respuestas
  - Servidores autorizados a los que se puede consultar para un determinado dominio
  - Información adicional

cabecera	identificación	señales
	n. cuestiones	n. resupestas
	n. s. autoriz.	n. i. adicional
cuerpo	cuestiones	
	respuestas	
	servidores autorizados	
	información adicional	

Hay formatos ya están definidos llamados **registros de recurso**. Son los siguientes:

- A – nombre de host (devuelve dirección IP)
- NS – dominio (devuelve servidor autorizado para ese dominio)
- CNAME – alias (devuelve nombre del host)
- MX – alias de correo (devuelve el servidor de correo)

Distribución de contenidos

Los accesos a servidores centralizados pueden ser lentos debido a que el camino de los mensajes sea lento o a que el servidor esté sobrecargado. Como solución, se distribuyen (y duplican) los contenidos en distintas zonas y se dirigen las peticiones al servidor de menor tiempo de respuesta. Los métodos son los siguientes

Caché web o servidor proxy

Consiste en que hay servidor intermedio (proxy) por el que pasan todas las peticiones de los hosts de una red. En él se mantienen copias de los contenidos durante cierto tiempo. Para su uso, el

usuario debe configurar su navegador y suelen estar proporcionados por el ISP. Este sistema permite un esquema jerárquico.

CDNs (Content Distribution Networks)

Es decir, redes de distribución de contenidos. Hay empresas que poseen centros de hosts de internet (empresas CDN) y alquilan su infraestructura. La empresa CDN replica los contenidos de sus clientes y los mantiene actualizados, proporcionando un mecanismo para que el contenido pueda ser entregado por el servidor CDN más rápido. El acceso a los contenidos puede ser mediante la redirección de objetos o mediante el balanceo de las peticiones usando el DNS (seleccionar el servidor CDN más rápido para hacer la traducción del nombre de host a IP).

P2P (Peer 2 Peer)

Redes en las que todos los usuarios son servidores y clientes mediante una aplicación. Se necesita un nodo de arranque para comenzar el reparto, y un directorio que puede estar centralizado o no (el centralizado tiene más problemas) o directamente no haberlo (inundación de consultas en la red, muy ineficiente). BitTorrent es el sistema más famoso, que tiene una red propia para cada archivo, y para cada archivo, un tracker. No necesita construir un directorio (la información están en los .torrent). Recordar el trabajo (escalabilidad, tablas hash distribuidas dht, skype...)

Capa de transporte

Es en la que se preparan los mensajes para que se puedan transmitir fuera del ordenador. En esta capa se recogen los datos de la aplicación origen y forma paquetes. En el destino se comprueba que todo llega correctamente y reconstruye el mensaje. Esta capa proporciona una comunicación lógica entre los procesos de capas (la capa de transporte de origen tiene una comunicación lógica con la capa de transporte de destino). Hay dos protocolos de transporte: TCP y UDP.

En el origen la diferencia es que TCP fragmenta los mensajes en segmentos y después, tanto UDP como TCP les añaden la cabecera de la capa de transporte.

Multiplexación y demultiplexación

Los mensajes pasan de la capa de aplicación a la capa de transporte a través de un socket en el que los procesos escriben y leen. La capa de transporte recoge los mensajes el socket y los traslada al socket destino.

Multiplexación – consiste en recorrer todos los sockets abiertos, procesar los mensajes y enviarlos a la capa de red

Demultiplexación – es recoger los segmentos que llegan de la capa de red, reconstruir los mensajes y colocarlos en los sockets destino.

La identificación del socket destino se hace a través de los números de puerto de la cabecera del segmento. Suelen ser enteros de 16 bits. Los que están de 0 a 1023 los usa el administrador para los servicios bien conocidos.

Multiplexación con sockets sin conexión (UCP)

En este caso la identificación del socket se realiza mediante la pareja de dirección IP y puerto destino. Los segmentos de distintos hosts que se entregan en el mismo puerto son recogidos por el mismo proceso. El puerto origen se usa para que el proceso que recoge los datos sepa después a quién responder.

Multiplexación con sockets con conexión (TCP)

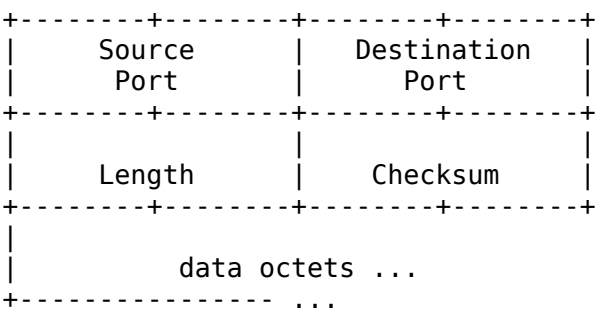
La identificación del socket se hace con la tupla de parejas de dirección IP y puerto origen y destino. Esto permite que pueda haber varias conexiones de distintos hosts a un mismo puerto y los atiendan procesos distintos. Un ejemplo de esto es telnet, que permite varias conexiones por el mismo puerto entre dos hosts.

Para este sistema se necesitan dos tipos de sockets: el socket servidor, que es el que espera las conexiones, y los sockets de conexión, que se encargan de la transmisión de datos (práctica de redes del convertidor a mayúsculas).

UDP (User Datagram Protocol)

UDP hace casi lo mínimo que debería hacer un protocolo de transporte: multiplexación / demultiplexación y comprobar errores. Está descrito en el RFC 768 de la siguiente manera:

```
                0      7 8     15 16    23 24    31
```



En el origen, UDP le añade una cabecera al mensaje para formar un segmento. En el destino, comprueba si el paquete llegó sin errores. En caso de que contenga errores, se suele descartar.

Los campos de la cabecera son: los puertos de origen y destino, la longitud total de todo el segmento en bytes y una suma de comprobación (incluyendo la cabecera).

Características

- Es un protocolo sin conexión: no hay acuerdo previo entre emisor y receptor, por lo que no hay retardo en el establecimiento de la conexión. En caso de errores, no se retransmiten los mensajes.
- Tampoco tiene estado: cada segmento se envía independientemente de los demás (no hay segmentación, así que no es UDP quien reconstruye los mensajes). Esto hace que el procesamiento sea más rápido y requiera de menos recursos (permite más clientes).
- Tiene las cabeceras más pequeñas (para mayor velocidad y menor espacio de disco)
- Exige más control de la aplicación en caso de que se quiera añadir una cabecera propia o se quiera numerar los datos. Tampoco incluye control de congestión por ejemplo.

Es un sistema propio de las aplicaciones que prefieren velocidad antes que fiabilidad: transmisión de audio, vídeo, DNS, protocolos de encaminamiento (RIP), administración de red (SNMP), servidores de archivos remotos (NFS)... También lo usan algunas aplicaciones (no todas) de flujos multimedia y telefonía por internet. La causa de las aplicaciones multimedia usando TCP es que hay organizaciones que bloquean el tráfico UDP por no tener mecanismos de control de congestión.

Suma de comprobación (checksum)

- Se toman todas las palabras de 16 bits del segmento (incluida la cabecera y alguna parte de la cabecera IP. Lógicamente, la parte de checksum no la podemos coger) y se suman.
- Se cogen los 16 bits más significativos y se les suma la parte que quedó de acarreo (siempre que la longitud del número sea mayor de 16 bits, es decir, puede que sea necesario hacer este paso varias veces).
- Se hace el C1 del resultado y se guarda en el campo de checksum.

Fundamentos de una transmisión fiable

Se basa en los protocolos de retransmisión de los paquetes con errores.

Se trata de los protocolos **ARQ (Automatic Repeat reQuest**, solicitud automática de repetición). Existe el que tras mandar un paquete para y espera a que el receptor confirme la recepción. Se puede dara también el enviar varios paquetes antes de recibir las confirmaciones (ARQ de ventana

deslizante). Cuando pasan N paquetes con una confirmación todavía pendiente, se envía también ese paquete sin confirmar.

Las confirmaciones de recepción son los **ACKs (ACKnowledgment number)**.

Estos protocolos se realizan con enlaces bidireccionales (full duplex) y necesitan numerar los paquetes para funcionar.

Casos del ARQ parar y esperar

- Sin errores: el receptor devuelve un ACK con el número del siguiente paquete
- Pérdida de paquetes: el receptor no devuelve el ACK y el emisor retransmite el paquete
- Paquete con errores: similar al anterior
- Pérdida de un ACK: el emisor retransmite el paquete. El receptor lo recibe por duplicado, lo descarta y vuelve a mandar el ACK
- Timeout: paquetes y ACKs que no llegan a tiempo: acaban llegando duplicados y se ignoran.

NAK

Un **NAK (Negative–Acknowledge Character)**, representado con dos ACKs iguales. Indica la recepción de un paquete con errores (ya no espera al timeout para reenviar). Cuando un paquete con errores llega, se devuelve el ACK del último correcto. Si vence el temporizador, se envían continuamente duplicados de paquetes. Para evitar este problema, son tres ACKs los que equivalen a un NAK. TCP usa este sistema con alguna variación.

Ventana deslizante

Si después de cada transferencia esperamos al ACK reduciremos bastante la velocidad de transmisión. La utilización del enlace con este sistema (ARQ parar y esperar) será mucho menor que 1: $U = t_{trans} / (RTT + t_{trans})$

Esto se resuelve con entubamiento: el emisor envía N paquetes antes de recibir confirmaciones. Ahora el tiempo útil en el emisor será $N*t_{trans}$ en vez de solo t_{trans} , por lo que U podrá llegar a 1 cuando $N*t_{trans} >= t_{trans} + RTT$, es decir, cuando $N >= 1 + RTT/t_{trans}$.

El rango de números de secuencia que se use con este sistema debe abarcar al menos el doble del tamaño de la ventana emisora. Además, emisor y receptor deben almacenar más de un paquete.

Una ventana emisora es el conjunto de N paquetes que el emisor puede enviar o están pendientes de confirmación. La ventana receptora es el conjunto que el receptor puede aceptar o está procesando. Estas ventanas se van desplazando a medida que el emisor reciba las confirmaciones.

Tipos

- Retroceder N (Go Back N): el receptor solo acepta paquetes en orden, por lo que si un paquete llega con errores, se descartan los siguientes y se vuelven a enviar. Pasa lo mismo si expira el temporizador. La ventaja es que los ACKs pueden ser acumulativos: un ACK implica la recepción de un paquete y todos sus previos.
- Repetición selectiva: el receptor acepta paquetes fuera de orden, por lo que solo se

retransmiten los paquetes erróneos, pero hay que confirmarlos uno a uno.

TCP (Transmission Control Protocol)

Es un protocolo que aplica los principios de transmisión fiable. Emplea números de 32 bits que identifican bytes (no segmentos) empezando por un número x aleatorio e incrementando secuencialmente. Los ACKs usan estos números para indicar el siguiente byte que se desea recibir.

En este protocolo se usa un fenómeno de **superposición** (piggybacking), con el que en un mismo segmento ACK se pueden transferir también datos. Para evitar que se agote el tiempo de espera máximo, si no se tienen datos que enviar, se envía solo el ACK.

Para asegurar una **transferencia fiable de datos** el emisor usa temporizadores para la retransmisión, ventana deslizante y ACKs acumulativos. Se recomienda usar un temporizador único, el cual se reinicia cuando llega un ACK. Si un ACK no llega a tiempo, se retransmite solo el segmento no confirmado, se reinicia el temporizador y se espera al ACK.

Cuando se produce un **vencimiento del temporizador** el emisor además de transmitir el segmento duplica el tiempo disponible. Si vuelve a vencer el tiempo, vuelve a duplicar el tiempo. En otros casos se actualiza la duración del temporizador con unas fórmulas más ajustadas que emplean el RTT calculado en anteriores transferencias.

Para **retransmitir los paquetes de forma rápida**, TCP no tiene NAKs, pero sí que interpreta 3 ACKs repetidos como un NAK. Cuando el receptor recibe el paquete siguiente al que esperaba, manda un ACK. El emisor no hace caso y sigue mandando el siguiente. Cuando llega éste al receptor, éste vuelve a mandar otro ACK más igual, por lo que el emisor ya emplea una retransmisión rápida para mandar el paquete perdido antes de que venza el temporizador. Ante esto, TCP distingue dos eventos de pérdidas: vencimiento del temporizador (lo cual se considera como un problema grave pues se han perdido los paquetes y los ACKs) y los ACKs triplicados (lo cual es un problema leve, ya que por lo menos los ACKs han llegado).

TCP podría considerarse un **ARQ intermedio** entre retroceder N y repetición selectiva, ya que los ACKs son acumulativos (el ACK de un segmento implica los ACKs de los anteriores) pero puede aceptar también segmentos en desorden y esperar a que lleguen después los intermedios.

El **control de flujo** es el mecanismo que permite al receptor indicar al emisor el ritmo al que puede recibir datos. En el momento de la conexión, el receptor indica el tamaño de su ventana de recepción y el emisor fija a este valor su ventana de envío. Esto se hace mediante un campo en la cabecera TCP, lo que permite modificar también este valor en cada transmisión.

Estructura de la cabecera TCP

	16 bits								16 bits
	Puerto origen								Puerto destino
	Número de secuencia (SN)								
20 bytes	Agradecimiento superpuesto (AN)								
	Long. Cabecera	Reservado	URG	ACK	PSH	RST	SYN	FIN	Ventana otorgada (AW)
	Suma de comprobación								Puntero urgente
	Opciones + relleno (tamaño libre)								

- Puertos de origen y destino – el destino es estándar y lo elige la aplicación, mientras que el

origen se decide aleatoriamente

- Número de secuencia – identificación del número de byte por el que empieza el segmento. No se empieza numerando desde 0 si no desde un número aleatorio
- Agradecimiento superpuesto – reconoce el número de bytes recibidos correctamente siempre que el bit ACK está activado. En caso de no estarlo, este campo no se usa
- Longitud de la cabecera – hay que especificarla dado que existe el campo de opciones
- Indicadores
  - URG – el segmento tiene datos urgentes (urgent)
  - ACK – el segmento tiene un reconocimiento (acknowledgment)
  - PSH – el segmento fue empujado (push)
  - RST – se solicita reinicio de la conexión (reset)
  - SYN – se solicita inicio de conexión (synchronize)
  - FIN – se solicita fin de conexión (finished)
- Ventana otorgada – indica cuántos bytes de datos puede recibir (usada para control de flujo)
- Suma de comprobación – suma de 16 bits en C1 de la cabecera TCP y cuerpo (y algunos campos más de la cabecera IP). Se calcula en los dos lados de la transferencia para comprobar que se haya transmitido correctamente.
- Puntero urgente – indica la posición de los datos urgentes dentro del segmento. En caso de que los haya, URG está a 1.
- Opciones – usos varios, como negociar el tamaño máximo de los segmentos.

Transmisión de datos

- Conexión
  - Solicitud de conexión (SYN=1, num. secuencial = x)
  - Contestación de aceptación (SYN=1, ACK=1, num. secuencial = y, ack = x+1)
  - Confirmación de aceptación (ACK=1, num. secuencial = x+1, ack = y+1) e inicio del envío de datos (ACK=1, num. secuencial = x+1, ack = y+1, datos).
- Se realiza la transmisión
  - La aplicación escribe los datos en el socket, que los va acumulando hasta enviar un segmento (por superar el MSS, maximun segment size; cuando la aplicación lo fuerza, push; cuando el temporizador llega a 0)
  - TCP genera el segmento y se lo pasa a la siguiente capa
- Desconexión
  - Solicitud de desconexión (FIN=1)
  - Aceptación de desconexión (ACK=1) y solicitud de desconexión (FIN=1)

- Aceptación de desconexión (ACK=1)

Control de congestión

Los recursos de la red siempre deben repartirse entre las diferentes peticiones. Si hay demasiados paquetes en la red se producen retardos en las transmisiones, lo cual provoca muchas pérdidas de paquetes. Normalmente esta congestión se produce por desbordamiento de memoria en los routers. Para controlarla se suelen usar dos técnicas: reservar recursos antes de que ocurra o dejar que ocurra y después resolverla.

El origen de una congestión puede estar en la capacidad de velocidad finita C del router con dos emisores. Mientras la tasa de transmisión está entre 0 y C/2 todo se recibe correctamente. Cuando se supera, el enlace no puede proporcionar paquetes a esa velocidad, por lo que la cola del router no deja de aumentar.

Este origen también puede estar en la carga ofrecida por el router con dos emisores y misma velocidad que el anterior. Cuando la tasa de transmisión supera el C/2, la tasa entregada disminuye porque algunos paquetes son duplicados.

Como última posibilidad estudiada, se puede dar que, con varios emisores, routers y enlaces, con tasa de transmisión elevada, los buffers de los routers se llenan y disminuye la tasa entregada. Esto hace que la tasa, en el límite, tienda a 0.

En TCP/IP el mecanismo de control de congestión recae en TCP, que considera que hay congestión cuando expira un temporizador o cuando se reciben ACKs triplicados.

Para actualizar la ventana de congestión y adecuarla a la transferencia en curso, se estima periódicamente el RTT y se actualiza según la fórmula:

tasa de envío (en B/s) = ventana de congestión / RTT

Mecanismos para actualizar la ventana de congestión

- Inicio lento – cuando se inicia una conexión TCP, el emisor empieza fijando su variable de tamaño de ventana de congestión a MSS/RTT (bits/seg). Esto es una velocidad muy baja, por lo que a cada intervalo de tamaño RTT, el tamaño de ventana se duplica, aumentando exponencialmente hasta que se pierde un segmento.
- Incremento aditivo/decremento multiplicativo (AIMD) – cuando la transferencia ha sido iniciada, pueden perderse segmentos. Cuando la pérdida se anuncia con 3 ACKs, se considera un problema leve, y se reduce la ventana de congestión a la mitad (decremento multiplicativo) pero sin hacerla descender nunca de 1 MSS. Si se ha resuelto el problema, la ventana de congestión aumentará linealmente (incremento aditivo) hasta que se produzca otra pérdida.
- Vencimiento del temporizador – este evento indica que la congestión es grave, dado que se han perdido incluso los 3 ACKs, por lo que el tamaño de la ventana se reduce a 1 MSS y aumenta como en inicio lento hasta la mitad del punto en el que se produjo el vencimiento y a partir de ese punto empieza a aumentar linealmente.

Imparcialidad

Entre conexiones TCP, siempre se dará la misma capacidad de enlace a cada una. Si una aplicación usa una conexión y otra usa cinco, la capacidad de la segunda será cinco veces mayor. Entre una conexión TCP y una UDP, la UDP siempre acapara la mayor parte de la velocidad.

Capa de red

Es la capa encargada de buscar las rutas y hacer llegar los paquetes de un host a otro. Los elementos principales de esta capa son los routers. A partir de esta capa (junto con las inferiores) se implementa también en sistemas no finales (routers).

Conceptos

Reenvío (forwarding) – mecanismo por el cual, cuando un paquete llega a un router, éste lo hace pasar a la interfaz de salida apropiada.

Encaminamiento o enrutado (routing) – mecanismo que determina la ruta que debe seguir un paquete que se envía de un emisor a un receptor. Esto se realiza mediante algoritmos de encaminamiento.

Tablas de reenvío – tablas que almacenan la información para el envío de paquetes: están determinadas por los algoritmos de encaminamiento y permiten que se asigne el valor del campo de la cabecera a la interfaz de salida apropiada.

Redes de conmutación de paquetes

Redes de datagramas

Redes en las que cada paquete incluye en su cabecera la IP destino, lo que provoca que para hacer el reenvío el router tenga que mirar la cabecera y decidir la salida más apropiada. No mantienen información de estado, es decir, en una secuencia de paquetes cada uno puede ser encaminado de forma independiente.

La capa de red en **internet** se llama IP. Es de tipo datagrama, por lo tanto sin estado, y no es del todo fiable: no garantiza la entrega de paquetes (ni orden) ni el tiempo de la entrega, pero siempre intenta entregar el mayor número de paquetes aunque algunos se pierdan. La ventaja es que permite la interconexión de redes de diferentes tecnologías (de ahí su nombre, ínter-red).

Redes de circuitos virtuales

Establecen la conexión planificando una ruta al destino (VC, Virtual Circuit, circuito virtual). A cada paquete se le escribe un identificador de VC que luego los routers emplean para el envío, por lo que pueden mantener información de estado (en una tabla de circuitos virtuales).

Tablas de circuitos virtuales – cada segmento de la transmisión mantiene una tabla de encaminamiento formada con nodos que tienen la siguiente información:

- Interfaz de entrada del circuito virtual (sitio por el que debe entrar el paquete en ese nodo)
- VCI (Virtual Circuit Identifier)
- Interfaz de salida por la que los paquetes de ese VC dejan el nodo
- Identificador de salida del circuito virtual (siguiente VCI)

Un paquete que llega por una interfaz con un VCI se coloca en la interfaz indicada en la tabla con el siguiente VCI.

Para la **construcción de una tabla** de VC se ejecutan los siguientes pasos:

- A envía una Petición de llamada a B, que se va mandando por los distintos routers con VCIs aleatorios y las interfaces que éstos deciden más convenientes para llegar a B.
- B devuelve una Llamada aceptada con el mismo VCI de entrada que él recibió, y así los routers por los que ésta pasa van rellenando la tabla de VC con los VCI y las interfaces que usaron anteriormente.
- El nodo con el que se había comunicado A le manda un ACK a A con el VCI que A debe emplear.
- A partir de entonces, A manda a B el resto de paquetes con el VCI que le indicó el nodo.

Algoritmos de encaminamiento

También algoritmos de rutado, son los encargados de encontrar el camino mínimo entre el origen y el destino. Cada host está conectado a un router (router por defecto) y el problema se limita a encontrar el camino mínimo entre los distintos routers. Este problema es equivalente a encontrar el camino mínimo en un grafo, en el cual los routers son los nodos y los enlaces son las aristas (con un peso asignado, el coste, determinado por la distancia, velocidad, carga de enlace...).

Clasificación de los algoritmos

- Globales (de estado de los enlaces, EE) – en estos, cada nodo dispone de toda la información sobre la red (nodos y coste de todos los enlaces), por lo que a partir de esa información, cada nodo puede calcular su tabla de encaminamiento.
- Descentralizados (de vector de distancias, VD) – cada nodo colabora con los otros para hacer el cálculo del camino mínimo. Un nodo intercambia información con sus vecinos, por lo que solo conocen la distancia a los demás y por dónde empezar.
- Estáticos o dinámicos – los estáticos solo cambian cuando cambia el estado de la red o se modifican manualmente. Los dinámicos, en cambio, se ejecutan periódicamente de forma automática (son los usados en internet).
- Sensibles o insensibles a la carga – en los sensibles, los costes de los enlaces varían en dinámicamente en función de la carga, por lo que puede resultar que haya paquetes atrapados en un ciclo. En internet los algoritmos son insensibles a la carga.

Algoritmo de encaminamiento EE

Variante del algoritmo de Dijkstra, **Forward Search**. Se aplica a cada nodo de la red. El router tiene el grafo entero gracias a los LSPs (link state packet, tabla de routers vecinos y coste de cada uno) que recibe de los otros routers. Anexo de AED:

El algoritmo de Dijkstra busca el camino más corto entre dos vértices. Se aplica a un grafo G dirigido y valorado, tal que cada arco tiene un valor A(i,j) >= 0. Este algoritmo tiene complejidad de orden $O(n^2)$.
Se trata de un algoritmo voraz, que usa los siguientes conjuntos:

- C: conjunto de vértices Candidatos
- S: conjunto de vértices Seleccionados
- D: vector de Distancias que almacena la longitud del camino especial (parte del vértice origen y tiene todos los vértices de S excepto el último) más corto desde el origen hasta cualquier vértice de G

- A: matriz de Aristas con sus pesos
- P: vector que para cada elemento siempre contiene el Predecesor inmediato en el camino especial del vértice 1 al j

Algoritmo – C empieza teniendo todos los vértices excepto el origen y S solo tiene el vértice origen (el 1). D tiene para cada j el valor A(1,j)

- Seleccionar el vértice v con D(v) mínimo. Lo sacamos de C y lo ponemos en S.
- Para cada vértice de C actualizamos D tal que D(j) = min(D(j), D(v)+A(v,j))
- Introducimos en P el predecesor inmediato de v en su camino especial.
- Continuamos hasta terminar todos los vértices.
- Para recuperar un camino, solo hay que ir mirando los predecesores en el vector P.

Con el uso de este algoritmo se asegura una rápida estabilidad y no se genera mucho tráfico, además de que se puede responder rápidamente a cambios en el estado de la red. El problema es que la cantidad de LSPs almacenada en cada nodo puede ser bastante grande, lo cual genera problemas de escalabilidad.

Algoritmo de encaminamiento VD

Se trata de un algoritmo en el que todos los nodos deben participar (se trata de una especie de algoritmo de Floyd). Pasos:

- Inicialmente, los nodos solo conocen el coste a sus vecinos
- Iterativamente, los nodos comunican a sus vecinos sus distancias a otros nodos (distancias de z: $d_{z,i}$), computando distancias a sus nuevos nodos o actualizando las que tienen a menores valores (almacenando el vecino que envió dicha información (siguiente salto).
- Ese paso iterativo se repite hasta que el algoritmo converge

Este algoritmo se actualiza periódicamente con los vecinos, reaccionando frente a una disminución del coste de un enlace con una actualización rápida de las tablas y frente a un aumento, con problemas que se resuelven con diversas técnicas (horizonte dividido e inverso envenenado)

Es un algoritmo iterativo que puede necesitar demasiadas iteraciones, lo que lo hace peor que EE. Además es menos robusto, ya que si un nodo calcula mal sus distancias, todos usarán esos valores incorrectos.

Encaminamiento jerárquico

Los sistemas autónomos (SA) son las regiones en las que se dividen las grandes redes, como internet y suelen estar operados por empresas u organismos.

En estos SA, los routers solo conocen el encaminamiento en su región, y existen routers pasarela frontera que centralizan todo el tráfico del SA.

Por esto, existen dos niveles de encaminamiento: los intradomino, con los que cada SA eligen su propio algoritmo, y los interdominio, que usan un algoritmo común para todos los SA.

Encaminamiento en internet

A parte de los protocolos de aplicación, se usan los siguientes:

RIP (Routing Information Protocol)

Protocolo intra-autónomo propio de intradominios o interno al SA. Es de tipo VD y como protocolo de red usa UDP/IP por el puerto 520. Considera el coste de los enlaces como 1 y que la distancia máxima es 15. Permite el envío de mensajes RIP solo a nodos vecinos. Estos mensajes pueden ser:

- Mensajes de petición RIP: solicitan información
- Mensajes de respuesta RIP: envían una lista de hasta 25 redes internas al SA (estos también se envían de forma automática cada 30. En caso de que no se reciba respuesta en 180 segundos, se considera caído).

OSPF (Open Shortest Path First)

Protocolo intra-autónomo propio de intradominios de tipo EE, que como protocolo de red usa uno propio con el puerto 89. Usa un algoritmo libre basado en el estado de los enlaces. Se trata de un protocolo más avanzado que RIP y que pretende reemplazarlo. Los mensajes OSPF se difunden a todos los nodos cuando se producen cambios y cada 30. Gracias a este sistema, cada router obtiene la información completa del SA.

Además, se pueden enviar mensajes HELLO a cada vecino para comprobar si siguen en línea, y existe la posibilidad de interrogar a un vecino para obtener toda la información. Los costes de cada mensaje los establece el administrador.

Es un protocolo más seguro, ya que implementa un protocolo de transporte propio con todos los mensajes autentificados y que solo considera los routers autenticados.

También permite que, como siempre se elige el camino más corto primero, se pueda repartir el tráfico entre caminos del mismo coste.

A mayores da soporte de jerarquía, permitiendo subdividir los SA en áreas: cada área ejecuta OSPF, y entre esas áreas la comunicación se hace mediante routers de frontera de área, al cual se encaminan todos los paquetes que van hacia fuera del área. Estos routers están interconectados entre sí en un área troncal. Para salir fuera del SA también existe un router frontera.

BGP (Border Gateway Protocol)

Es el protocolo inter-autónomo propio de interdominios o entre SAs. En internet es el protocolo estándar en su versión BGP4. Usa como protocolo de red el TCP/IP con el puerto 179. Solo comunica routers pasarela de frontera: entre routers BGP vecinos emplea E-BGP y para routers BGP del mismo SA se usa I-BGP.

Usa un algoritmo de encaminamiento similar al vector de distancias, solo que intercambia rutas completas (vector de rutas). Cada SA se identifica por un número de sistema autónomo único. Los administradores de cada SA pueden elegir sus políticas de encaminamiento (no redirigir paquetes de otro SA...)

IP (Internet Protocol)

Es un protocolo basado en datagramas que ofrece servicio sin conexión. La fiabilidad de la conexión recae en capas superiores (TCP) y está diseñado para interconectar redes diferentes.

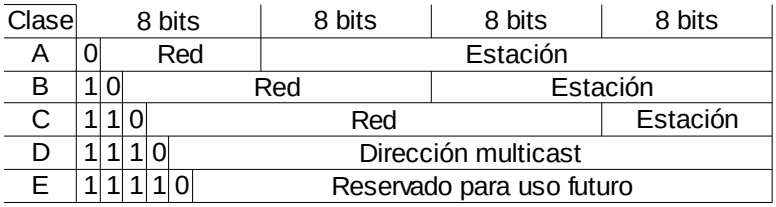
Este protocolo de internet define el formato de las direcciones, de los datagramas y las acciones de

los routers en base a los campos de los datagramas. Además, internet necesita de un protocolo de encaminamiento y de un protocolo de mensajes de control de Internet (ICMP).

Direccionamiento IPv4

Cada nodo de una red tiene una dirección IP por interfaz. La interfaz es la unión de un host o router con un enlace (físicamente, un puerto para conectarle el cable). Estas direcciones IP se codifican mediante 4 bytes, normalmente escritos en forma decimal.

Nunca ninguna dirección IP puede estar completamente a 0 o a 1. Estas se clasifican de la siguiente forma:



- Clase A: 126 ($=2^7 - 2$) redes con más de 16 millones ($=2^{24} - 2$) de estaciones cada una. (Pensadas para organizaciones grandes como países).
- Clase B: 2^{14} redes con $2^{16} - 2$ estaciones cada una. (Pensadas para organizaciones medianas, como empresas grandes).
- Clase C: 2^{21} (más de dos millones) de redes con 254 ($=2^8 - 2$) estaciones cada una. (Pensadas para el resto de organizaciones pequeñas).
- Direcciones multicast: usadas para transmitir un mensaje a un grupo de hosts de la red.

Direcciones especiales reservadas

- La dirección de base propia de cada red es con los bits de estación a 0
- La dirección de broadcast (también de cada red) es con los bits de estación a 1
- Reservadas por el IANA
  - 0.0.0.0 → "Esta red" - sirve para arrancar sistemas sin disco (en el protocolo DHCP, es también la dirección de encaminamiento por defecto)
  - 127.0.0.0 – 127.255.255.255 → la propia estación, dirección de loopback (suele usarse la 127.0.0.1).
  - 240.0.0.0 – 255.255.255.254 → reservadas para uso futuro
  - 255.255.255.255 → difusión a toda la red en el protocolo DHCP (IP de broadcast)

Subredes

Puede existir el problema de que el número de estaciones de una red sea demasiado grande, por lo que puede ser difícil de administrarla. Esto se soluciona dividiendo la red en subredes que se gestionen de forma independiente pero que actúen como una sola de cara al exterior.

Por tanto, una parte del campo estación la usaremos para indicar la subred. Para delimitarla, emplearemos máscaras, que antes eran un número de 32 bits expresado de la misma forma que las

direcciones IP y con los N números más significativos a 1. Así, la máscara de 27, que se denota como /27, se escribía 255.255.255.224.

Ejemplo: a qué subred pertenece 193.168.17.133 /27? → Hacemos un AND del número de estación y cogemos los n.estación – (32 – n.máscara) primeros bits: **100**00000 → subred 4

Los 5 últimos bits se refieren a la posición de la estación en la subred (identifican un host). Así tendríamos 2^3 subredes con 2^5 -2 estaciones por subred, es decir, podríamos direccionar 240 estaciones.

Cada una de estas subredes tendrá una dirección base y una dirección de broadcast.

Redes sin clase, direcciones CIDR (Classless Inter–Domain Routing)

En 1993 se suprimen las clases y el número de red pasa a especificarse con un sufijo: /r significa que esa dirección tiene r bits de red y 32-r para indicar la estación. El número de red se sigue indicando de la misma manera: los bits de red con el número de la red y el resto a 0. Para broadcast se hace igual.

Ahora las redes sin clase serían A/8, B/16 y C/24. Además, se pueden establecer subredes empleando los primeros bits de identificación de estación en números de subred. Así, si se quieren obtener 8 subredes, debemos sacrificar 3 bits de número de identificación de estaciones para crearlas.

Agregación de rutas

Es un proceso que realizan los routers por el cual toman un grupo de direcciones de redes contiguas (bloque CIDR) y las resumen en una sola dirección de red común a todas esas redes.

La ventaja principal es la optimización del enrutamiento en grandes redes corporativas, ya que los routers tienen que mantener menos entradas en sus tablas de enrutamiento y en consecuencia ganan estabilidad. Si un router tiene conectadas 10 redes contiguas, solo publicará el resumen de la ruta CIDR a sus vecinos.

Para obtener la dirección IP base de agregación de rutas de debemos comprobar que sean redes contiguas y, a continuación, hacer la operación AND sobre las direcciones base de esas redes. Para obtener la dirección de broadcast debemos poner todos los bits que tenían comunes al hacer la operación AND a 1.

Formato del datagrama IP

<table>
<tr><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>3 bits</td><td>5 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td></tr>
<tr><td>Versión</td><td>IHL</td><td colspan="2">Tipo de servicio</td><td colspan="5">Longitud total</td></tr>
<tr><td colspan="4">Identificación</td><td>Indic.</td><td colspan="4">Desplazamiento del fragmento</td></tr>
<tr><td colspan="2">Tiempo de vida</td><td colspan="2">Protocolo</td><td colspan="5">Código de comprobación de cabecera</td></tr>
<tr><td colspan="9">Dirección origen</td></tr>
<tr><td colspan="9">Dirección destino</td></tr>
<tr><td colspan="9">Opciones y relleno</td></tr>
<tr><td colspan="9">Datos</td></tr>
</table>

Campos:

- Versión: tipo de IP (v4 ó v6)
- IHL: longitud de la cabecera en palabras de 32 bits (no es constante, el mínimo es 5)
- Tipo de servicio: indicar si se prefiere fiabilidad, rutas de gran capacidad, rapidez...
- Longitud total: longitud del datagrama (todos los fragmentos incluidos), incluyendo cabecera y datos.
- Identificación: identificación del datagrama.
- Indic.: indicadores
    - El primero no se utiliza
    - NF – activado indica que no se quiere que el paquete se fragmente
    - MF – activado indica que hay más fragmentos del mismo datagrama
- Desplazamiento del fragmento: identificación de cada fragmento.
- Tiempo de vida (TTL, Time To Live) – el emisor lo inicializa en un valor (máximo 255) y cada vez que un router lo procesa, reduce en uno el valor. Cuando el TTL llega a 0, el datagrama se destruye.
- Protocolo – número que identifica qué protocolo de capa de transporte se usa para que al llegar al destino, la capa de red sepa a dónde mandar el paquete.
- Código de comprobación de cabecera (checksum): es una suma considerando palabras de 16 bits y en C1. Solo incluye la cabecera IP y puede cambiar si se fragmenta, puesto que los datos ya se comprueban en la cabecera TCP.

Fragmentación

En cada red hay un MTU (Maximun Transmission Unit), que se aplica a los datagramas que se envían por esa red. Por eso mismo puede ser necesario fragmentar los datagramas en unidades más pequeñas (contando con sumarles después 20 bits de cabecera TCP y 20 de cabecera IP) ya en la capa de transporte.

En caso de que al final se fragmente un datagrama, se emplea el identificador MF en cada fragmento para indicar que quedan más partes de un datagrama hasta que llega el último, que no tiene MF a 1.

En caso de que el datagrama tenga su bit NF activo, el datagrama no se fragmenta y se descarta.

Direccionamiento IPv6

El nacimiento de IPv6 viene dado porque la capacidad de IPv4 llegará pronto al límite, necesita simplificarse el protocolo para que los encaminadores sean más eficientes y se necesita proporcionar mayor seguridad. Consta de direcciones de 128 bits ($3.4 * 10^{38}$ direcciones), no tiene clases, permite envíos multicast, servicios en tiempo real y servicios de autenticación y seguridad.

Estas nuevas direcciones de 128 bits se representan con 8 campos en hexadecimal, habiendo una forma compacta si hay cadenas de ceros:

47CD:0000:0000:0000:0000:0000:A456:0124 → 47CD::A456:0124

Además, se permiten las direcciones IPv4 en formato de IPv6: ::FFFF:192.168.0.1

También permiten el uso de máscaras para la parte de red y de host, indicándose éstas de la misma forma que en IPv4. Y se han establecido direcciones unicast, multicast (comienzan por ff) y anycast (a cualquier host de una red).

Cabecera

Consta de 40 bytes siempre (desaparece la longitud de cabecera), en 8 campos. Se han eliminado:

- Opciones: se indican en cabeceras adicionales
- Fragmentación: no se fragmenta nada. Si un mensaje es muy grande, se informa con un mensaje ICMP
- Numeración de los paquetes: ya no necesitan numerarse
- Suma de comprobación: no nos interesa, queda en la capa de transporte y en la de enlace y nos ahorramos procesamiento en cada router

Se añade una clase de tráfico y una etiqueta de flujo (de 8 y 20 bits respectivamente), para gestión de la QoS (Quality of Service, para controlar velocidades dentro de un router)

<table>
<tr><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td><td>8 bits</td></tr>
<tr><td>Versión</td><td colspan="2">Clase de tráfico</td><td colspan="5">Etiqueta de flujo</td></tr>
<tr><td colspan="4">Longitud de carga útil</td><td colspan="2">Cabecera siguiente</td><td colspan="2">Límite de saltos</td></tr>
<tr><td colspan="8">Dirección origen</td></tr>
<tr><td colspan="8">Dirección destino</td></tr>
<tr><td colspan="8">Cabecera siguiente / datos</td></tr>
</table>

La transición de IPv4 a IPv6 debe ser gradual y coexistirán las dos. Para ello, se hace uso de los túneles (tunneling) para encapsular paquetes IPv6 cuando deban pasar por routers no soportados. Para ello se añaden cabeceras de IPv4 a estos paquetes, y se eliminan al llegar a zonas de IPv6.

ICMP (Internet Control Messages Protocol)

Es un protocolo usado para que hosts y routers puedan informarse sobre errores o estado de la red y que funciona sobre IP, pero no se garantiza su entrega. Para enviarlo, se encapsula en un datagrama IP. Estos mensajes tiene un tipo, un código y contienen los primeros 8 bytes del datagrama que causó el mensaje.

Tipos

- Destino inalcanzable: lo envía un nodo a la estación origen cuando no puede alcanzarse el destino o cuando un datagrama no puede fragmentarse (tipo 3)
- Tiempo excedido: lo envía el nodo a la estación origen cuando el contador de un datagrama

llega a 0

- Ralentizar fuente: para limitar el número de datagramas introducidos en la red y evitar congestión
- Solicitud de eco y Respuesta de eco: se utilizan para ver si un destino es alcanzable y se encuentra activo (uso en ping)
- Problema de parámetro: indica que se ha detectado un valor ilegal en un campo de la cabecera
- Redirigir: se utiliza cuando un nodo se da cuenta de hay un mejor camino para enviar el datagrama.
- Marca de tiempo y Respuesta a marca de tiempo: para medir el retardo de la red
- Petición de máscara de dirección y Respuesta a la máscara de dirección: empleadas para conocer la máscara de red.

DHCP (Dynamic Host Configuration Protocol)

Es un protocolo de asignación de direcciones IP a los hosts. Al administrador de un equipo se le asigna una dirección de forma estática, pero puede haber asignaciones dinámicas mediante este protocolo. Cuando se inicia, solicita al servidor una IP temporal. Lo usan en los ISPs cuando no se tienen direcciones para todos los abonados y en las redes inalámbricas. Este protocolo devuelve a cada host su dirección IP, el gateway por defecto y los servidores DNS disponibles.

Pasos

- Descubrir el servidor DHCP (mediante un mensaje DHCPDISCOVER, que consta de IP origen a todo 0 y IP destino a todo 1)
- En la respuesta del servidor se devuelve una IP, una máscara de red... y un tiempo de concesión
- A continuación, el cliente realiza la petición de una de las ofertas DHCP (en caso de que haya varias)
- Y el servidor contesta con un ACK DHCP para confirmar la solicitud.

NAT (Network Addres Translation)

Es un sistema que permite usar la misma IP válida en varios ordenadores. Tiene direcciones especiales para uso en redes privadas que los routers de internet ignoran. Son las siguientes:

- Clase A: 10.0.0.0/8
- Clase B: 172.16.0.0/12
- Clase C: 192.168.0.0/16

Un servidor NAT necesita dos interfaces con dos IPs distintas: una válida para conectar al exterior y otra con una IP privada para conectar a la red interna. Los ordenadores de la red privada tendrán como gateway la IP privada del NAT, que encaminará de forma transparente los paquetes que le lleguen de una red a otra cambiando la IP origen y el puerto origen de los paquetes por su propia IP

y un puerto libre. Tras cada redirección, se almacena una entrada en una tabla para poder después redirigir la entrada al host correcto.

Capa de enlace

Es la capa encargada de los detalles de bajo nivel de la transmisión de cada paquete entre los extremos del enlace: se encarga de transmitir los bloques de bits de un lado a otro de un enlace. A esta pertenecen los protocolos dependientes de la tecnología de red (difusión, circuitos virtuales, conmutación de circuitos, conmutación de paquetes...). Como unidad de medida (PDU, Protocol Data Unit) de la capa de enlace se usan las tramas (también frames, marcos).

Tipos de enlace

- Punto a punto – debe existir un emisor y un receptor a ambos extremos del enlace
- Difusión – medio de transmisión compartido por varios emisores

En un host, la capa de enlace está implementada en la tarjeta de red. Éste se conecta a ese adaptador y le transfiere los datos a ser transmitidos por el enlace. Luego la capa de enlace añade su propia cabecera y envía los datos en forma de tramas.

Los protocolos de esta capa definen el formato de las tramas que se envían y las acciones de los nodos cuando envían o reciben tramas. Servicios posibles de un protocolo de capa de enlace:

- Entramado (delimitado de tramas) – es la forma de encapsular los datagramas
- Acceso al enlace: mediante el protocolo MAC se controla el acceso al medio
- Entrega fiable: gestiona las confirmaciones y retransmisiones
- Control de flujo: permite limitar el envío de tramas para que otros nodos no se desborden
- Detección de errores (más sofisticada que en capas superiores): necesita atenuar las señales, el ruido electromagnético, introduce bits de detección de errores (bits de paridad)... estas detecciones están implementadas en hardware.
- Corrección de errores: muchos de los errores detectados puede corregirlos mediante bits de paridad, checksums y CRC (Comprobación de Redundancia Cíclica, otra forma de autocorregirse)
- Half-duplex y full-duplex: permitir la transmisión en uno o dos sentidos respectivamente.

Modelo IEEE 802

Los principales tipos de LANs están definidas por el IEEE 802, que establece un modelo para la capa de enlace, la cual queda dividida en dos subcapas:

- LLC (Logical Link Control, Control de Enlace Lógico) – necesaria para la gestión de acceso a un medio compartido
- MAC (Media Access Control, Control de Acceso al Medio) – se pueden ofrecer varias opciones MAC para el mismo LLC

Capa LLC

Es la interfaz con las capas superiores, que añade una cabecera adicional y controla los errores y el flujo de datos por los enlaces. Está definida con el modelo IEEE 802.2 y puede ofrecer tres formas de funcionamiento:

- Sin conexión ni confirmaciones (no incluye control de flujo ni de errores, tampoco garantiza la recepción de datos y deja el control de recepción para capas superiores)
- Sin conexión con confirmaciones (confirman las tramas que llegan, pero no establece una conexión previamente)
- Con conexión y confirmaciones (establece una conexión lógica y hay control de flujo y de errores)

Capa MAC

Es la capa que ensambla los datos en tramas con capos de dirección y detección de errores. Cuando las tramas llegan a su destino, esta capa las desensambla, detecta los errores y reconoce la dirección. Para un mismo modelo de LLC puede haber distintos modelos MAC. Los más usados son:

- Ethernet (IEEE 802.3)
- Token bus (IEEE 802.4)
- Token ring (IEEE 802.5)
- MAN (IEEE 802.6)
- Wireless (IEEE 802.11)

Al final, las tramas tienen el siguiente formato:

Control MAC	Destino MAC	Origen MAC	DSAP	SSAP	Control LLC	Datagrama IP	FCS

- DSAP – Destination Service Access Point
- SSAP – Source Service Access Point
- FCS – Frame Check Secuence

Direcciones MAC Ethernet

La arquitectura TCP/IP considera dos direcciones para cada host: una dirección MAC que tiene sentido en el enlace o LAN (propia de cada dispositivo) y una dirección IP, que tiene sentido en internet.

En la LAN, los adaptadores usan las direcciones MAC. Fuera de ésta, se eliminan las cabeceras MAC y el paquete viaja usando las conexiones IP.

Direcciones Ethernet

Todos los nodos Ethernet (IEEE 802.3) tienen una dirección propia que los identifica. Suele estar grabada en una memoria ROM dentro del adaptador Ethernet. Estas direcciones constan de 6 bytes expresados en hexadecimal. Para asegurar que nunca se repiten, cada fabricante tiene un código único para sus direcciones.

Entre estas direcciones, también hay dos direcciones Ethernet especiales

- Broadcast: con todos los bits a 1.
- Multicast: con el bit menos significativo del primer byte a 1.

Un nodo aceptará por tanto las tramas en las que la dirección destino sea su propia dirección

Ethernet (unicast), la dirección de broadcast, la dirección de multicast y, en caso de que se configure en modo promiscuo (con ifconfig eth0 promisc), puede aceptarlas todas.

ARP (Address Resolution Protocol)

Es el protocolo que permite traducir las direcciones IP en MAC, el cual mantiene una tabla (caché ARP) con correspondencias IP/MAC. Cuando ARP recibe una IP:

- Si está en la tabla, devuelve la MAC correspondiente.
- Si no está, emite una señal de broadcast con esa IP
  - El adaptador al que corresponda esa IP responde con su MAC
  - Esa MAC se almacena en la caché y ya se puede usar

Cada entrada de la caché se suele eliminar a los 15 minutos.

Ethernet

Este tipo de red LAN es el más sencillo y más común. Ofrece un servicio no fiable y es una re de difusión. Puede tener topología de bus (un cable coaxial que puentea cada adaptador) o de estrella (un par trenzado que conecta cada adaptador con el centro, lo usado en la actualidad).

Este protocolo funciona sobre cable coaxial, par trenzado y fibra óptica, pudiendo ofrecer muchas velocidades (10Mbps, 100Mbps, 10Gbps...)

Formato de cabecera

La trama tiene el formato de MAC, pero Ethernet tiene un formato específico para el "Control MAC"

- Una cabecera con 7 bytes rellenos cada uno con la secuencia 10101010
- Un SFD (Start of Frame Delimiter), un byte con la secuencia 10101011

Por tanto, una trama con tamaño mínimo, sin cabecera ni SFD, sería de 64 bytes. Una con tamaño máximo llegaría a los 1528 bytes.

Difusión

Debe haber un protocolo para decidir quién transmite: Ethernet usa CSMA/CD (Carrier Sense Mulpile Access with Colission Detection, acceso múltiple por detección de portadora con detección de colisión). Funciona así:

- Para recibir todos los adaptadores escuchan continuamente por el cable
- Para transmitir
  - El adaptador escucha el medio
    - Si está libre, transmite
    - Si está ocupado, espera a que quede libre, dejando un pequeño intervalo de seguridad.

Aún así, en la transmisión se pueden producir colisiones (coinciden dos señales de datos a la vez).

Detección de colisiones

El nodo emisor escucha el cable mientras transmite. De todas formas, hay un tiempo de vulnerabilidad, ya que desde que el nodo ocupa el medio hasta que su transmisión llega a los otros nodos puede pasar un tiempo durante el que los otros nodos ven el medio libre y creen que pueden transmitir. En este caso, los datos quedan alterados. Para que esto se dé, el $tam_{trama} > 2tam_{prop}$, lo cual implica un tamaño de trama mínimo o una longitud de enlace máxima.

Respuesta a colisiones

Cuando un nodo detecta una colisión, termina de transmitir la trama y emite una secuencia de 32 bits (jamming sequence). Tras esto, detiene la transmisión y usa el algoritmo exponential backoff.

Exponential backoff (espera exponencial binaria)

- Divide el tiempo en ranuras discretas de longitud de tamaño $T = 2tam_{prop\_max}$.
- Las estaciones esperan un tiempo, 0 o T antes de reintentar la transmisión.
- Si se detecta otra colisión, se selecciona aleatoriamente entre 0, T, 2T o 3T
- A partir de 10 colisiones, el tiempo se escoge entre 0 y 1023T
- Después de 16 colisiones seguidas, el controlador desiste e informa del fallo

Tecnologías Ethernet

Repetidores

Son dispositivos con una sola capa (física) que trabaja sobre bits individuales. Siempre tienen 2 o más interfaces, y lo que hacen es copiar bits que llegan por una interfaz al resto de interfaces (excepto por donde llegó). Sirve para transmisiones a largas distancias.

Topología bus (obsoleta)

Consiste en un bus de cable coaxial con conectores T (una entrada y salidas a los dos lados) con terminadores en los extremos. No permite más de 4 repetidores y limita el número de adaptadores de cada segmento. Su nomenclatura usa el formato [Mbps][tipo de transmisión][centenas de m.]

Ejemplo: 10base2 – 10 Mbps, banda de base y segmento de 200 m.

Topología estrella

Existe un centro (hub o conmutador) con par trenzado o fibra óptica al que está conectado cada nodo. Cada nodo tiene par trenzado o fibra óptica de entrada y salida. Esta topología tiene distancia limitada, a 100m en caso de par trenzado, y a partir de 1000 Mbps obliga al uso de 4 pares trenzados. Su funcionamiento del protocolo es similar al del bus. Para nombrarlo tiene asignada la letra T al par trenzado y las letras F,S,L y E a la fibra óptica.

Ejemplo: 1000base-T (1 Gbps Ethernet de par trenzado)

Hubs (concentradores)

Son dispositivos de una capa (física) que trabajan a nivel de bits individuales. Hoy en día están obsoletos, y lo único que hacen es regenerar el bit y enviarlo por todas las interfaces por las que llegó. En caso de que lleguen a la vez por distintos interfaces, el hub informa de colisión.

Bridges (puentes) y Switches (conmutadores)

Son dispositivos de 2 capas (enlace y física) que trabajan a nivel de tramas Ethernet. Tienen la capacidad de procesar los distintos capos de las tramas Ethernet (extraer la dirección destino, determinar si tienen errores...). También disponen de colas en las interfaces de salida.

La diferencia reside en que los bridges tienen pocas interfaces, mientras que los switches tienen decenas, por lo que los bridges han quedado obsoletos.

Estos dispositivos tienen capacidad de autoaprendizaje también, es decir, aprenden la localización de los adaptadores y crean una tabla de reenvío con entradas para algunos adaptadores. En cada una de estas entradas el dispositivo almacena información de cada trama: la dirección Ethernet, la interfaz y el instante de creación.

- Al principio, la tabla está vacía y utiliza difusión
- Después va examinando las tramas que llegan
  - La interfaz de llegada indica la localización del adaptador
  - La dirección de origen indica la identidad del adaptador
- Tras unos minutos sin usarse, las entradas se borran

Funcionamiento

- Cuando el adaptador destino está en la tabla, reenvía las tramas solo por la interfaz indicada
  - El resto de adaptadores no verán esta transmisión, por lo que produce **aislamiento de tráfico** y evita colisiones
  - Si la interfaz origen coincide con el destino el conmutador descarta la trama (**filtrado**)
- Por lo tanto, estos dispositivos permiten mantener una **tasa de transmisión agregada** (permiten más de una transmisión a la vez)
  - Permiten múltiples transmisiones simultáneas (solo en el caso de que las interfaces origen y destino sean distintas)
  - En el caso de muchas interfaces, la tasa de transmisión agregada debe ser elevada

Switches frente a routers

Ambos son dispositivos de almacenamiento y reenvío. La diferencia está en que:

- Los switches trabajan a nivel de capa de enlace, manteniendo tablas de conmutación, filtrando y con algoritmos de aprendizaje.
- Los routers trabajan con cabeceras IP (a nivel de capa de red), manteniendo tablas de rutas y con algoritmos de encaminamiento.

Las ISPs al vender dispositivos, suelen juntar en uno solo las funciones de router, switch y NAT.

VLANs (Virtual Local Area Network)

O Virtual LAN, son redes de área local virtuales. Están pensados para solucionar los problemas de las redes institucionales de antes, en las que cada switch formaba una red LAN propia. Tenían los siguientes inconvenientes:

- Si un usuario se cambiaba físicamente de departamento, no podía seguir conectado al anterior
- Solo hay un dominio de broadcast único (para tramas de mensajes ARP o DHCP)
- Hacían un uso ineficiente de los switches (cada uno solo tenía pocos puertos)

Estos problemas se abordan con switches compatibles con VLANs, que soporten el estándar IEEE 802.3Q (añade unos campos a la cabecera). Estos switches permiten definir múltiples LANs virtuales sobre una única red fija.

VLANs basadas en puertos

Dividen los puertos del conmutador en grupos, asignando a cada grupo una VLAN (y manteniendo una tabla de puertos/VLAN para solo entregar tramas entre puertos de la misma VLAN). En caso de cualquier cambio, permiten reconfiguración por software. Tienen las siguientes características:

- Permiten aislamiento de tráfico – sólo se entregan tramas entre puertos de la misma VLAN (que se pueden definir por direcciones MAC)
- Pertenencia dinámica – permiten la asignación dinámica de los puertos a las VLANs
- Reenvío entre VLANs mediante encaminamiento (en la práctica, se combian routers y switches)

WLAN (Wireless LAN)

Especificación IEEE 802.11

Estas son las especificaciones usadas para cada adaptador de red wifi. Cada dispositivo suele tener varias. Por orden de peor a mejor:

- 11b (con 2.4 GHz, DSSS y hasta 11 Mbps)
  - Muy poco alcance y con interferencias
- 11a (con 5GHz OFDM y hasta 54 Mpbs)
  - Todavía menos alcance pero no da tantas interferencias, solo para uso militar
- 11g (con 2.4GHz, OFDM/DSSS y hasta 54 Mbps)
  - Con más alcance que 11a
- 11n (nuevo estándar, hasta 600 Mbps)
  - En realidad no da tanta velocidad. Tiene más alcance que las otras y menos interferencias

La nexus 7, por ejemplo, tiene Wi-Fi 802.11 b/g/n.

Redes simples (ad–hoc)

Son las que usaban las PSP para jugar uno contra otro. Conexiones de igual a igual, permiten comunicar estaciones mientras están en su radio de alcance.

Redes distribuidas (managed)

Se componen de una LAN troncal cableada (distribution system) que conecta los servidores y los puntos de acceso (AP, Access Point). Cada AP da servicio a un número de estaciones móviles.

MACA (Multiple Access witch Colission Avoidance)

También denominado CSMA/CA, es un protocolo de acceso al medio (MAC). Funciona así:

- Un host que quiera transmitir sondea el medio
  - Si está libre, espera un gran intervalo de tiempo (DIFS, Distributed Inter Frame Space)
    - Si sigue libre, transmite
    - Si está ocupado, continúa esperando
  - Si sigue ocupado, usa el algoritmo de espera exponencial binaria (Exponential backoff)
- No tiene detección de colisiones, confía en los ACKs
  - Entre la recepción de la trama y el envío del ACK se espera un intervalo corto (SIFS)
  - Todos los hosts esperan la transmisión del ACK
- Usan las tramas de control para asegurar la transmisión
  - Envían primero un RTS (Request To Send, trama de petición de envío)
  - El destino responde con una CTS (Clear To Send, trama de reserva de canal)
  - Esto hace que los demás hosts tengan que esperar a que la transmisión se complete

Redes ATM (Asynchronous Transfer Mode)

Es el tipo de red con la que trabajan los ISPs (Internet Service Provider, compañías telefónicas) y las redes troncales de internet. Establecen un modo de transferencia asíncrono y están diseñadas para operar a alta velocidad. Pueden transmitir datos y llamadas telefónicas, y sus switches son capaces de operar a velocidades de Tbps. Está integrada en la arquitectura TCP/IP.

Tipos de servicio

- CBR (Constant Bit Rate) – reservan y garantizan una tasa de transmisión, con retardos y pérdidas bajo ciertos límites. Es adecuado para transmitir audio y vídeo.
- ABR (Available Bit Rate) – se suele garantizar un mínimo, pero el máximo no está siempre disponible. Es lo que hacen las ISPs cuando ofrecen hasta x velocidad.
- UBR (Unespecified Bit Rate) – solo se transmiten los paquetes cuando el resto de servicios de la red dejan recursos

- VBR (Variable Bit Rate) – pueden ser para aplicaciones en tiempo real (VBR-rt) o no.

Características

- Usan paquetes muy pequeños y sencillos para garantizar su conmutación a altas velocidades (53 bytes, 5 de cabecera y 48 de datos)
- Son redes de circuitos virtuales (VC) orientadas a conexión
  - Primero hacen solicitud de conexión
  - Planifican la ruta
  - Las celdas llevan el número de canal virtual
  - Al desconectar se eliminan los canales virtuales
- No hay ACKs ni retransmisiones, pero las celdas tienen control de errores de la cabecera

Capas

La capa ATM puede funcionar sobre cualquier capa física. Posee una capa de adaptación (AAL, ATM Adaption Layer) para permitir que otros protocolos usen la red ATM. Para TCP/IP, a la entrada fragmenta los datagramas para que quepan en celdas y los reensambla a la salida. Para el audio y vídeo, agrupa datos hasta rellenar una celda.

Identificación de circuito virtual

Define dos niveles de conexión:

- VCC (Virtual Channel Connection) – canal virtual (para los circuitos virtuales, VC)
- VPC (Virtual Path Connection) – camino virtual. Son conjuntos de VCCs con los mismos extremos, para facilitar su gestión.

Estructura de celdas

4	8	16	3	1	8	Resto
GFC	VPI	VCI	Tipo	CLP	HEC (CRC-8)	Datos

- GFC (Generic Flow Control) – control de flujo para la QoS (solo en la interfaz usuario-red)
- VPI – Virtual Path Identifier
- VCI – Virtual Connection Identifier
- Tipo – Indica el tipo de paquete (operaciones, datos...)
- CLP (Cell Loss Priority) – se deja como ejercicio propuesto al lector un aparato que convierta todos tus paquetes de red ATM para ponerle este valor a 1 (te compensa, que así siempre envían tus paquetes :P)
- CRC – byte para Comprobación de Redundancia Cíclica
- Datos – nadie sabe lo que lleva ahí, es secreto :|

Capa física

Es la capa que trabaja a nivel de bits, convirtiéndolos en señales eléctricas. Es la que define las características físicas del medio de transmisión.

Y sí, terminamos :D