Universidad de Concepción
Facultad de Ingeniería
Depto. de Ingeniería Eléctrica

Apuntes de Redes de Datos

543483

Autor : Jorge E. Pezoa Núñez
Semestre : 2001-I
Fecha : 18 de junio de 2001

Índice

1. Introducción y Marco de Referencia **5**
1.1. ¿Qué Son las Redes de Datos? ¿Para Qué Sirven? 5
1.2. Clasificación de las Redes 5
1.2.1. Clasificación por Tecnología de Transmisión 5
1.2.2. Clasificación por Escala 6
1.3. Modelos de Referencia 6
1.3.1. El Modelo OSI 6
1.3.2. El Modelo TCP/IP 8
1.4. Protocolos, Interfaces, Servicios y Tipos de Servicios 9
1.5. Ejemplos de una Red de Datos 11
1.5.1. Ejemplo de LAN: Red del DIE 11
1.5.2. Ejemplo de WAN: Red Reuna 2 12

2. Protocolos LAN **14**
2.1. Introducción 14
2.2. Protocolos LAN y Modelo OSI 14
2.3. Topologías 14
2.4. El Medio Físico 15
2.4.1. Cable Coaxial 15
2.4.2. Par Trenzado 16
2.4.3. Fibra Óptica 17
2.4.4. Enlaces de Radio 18
2.4.5. Enlaces de Microondas 18
2.4.6. Infrarrojos y Ondas Milimétricas 19
2.4.7. Enlaces Satelitales 19
2.5. Capa de Enlace de Datos 19
2.5.1. Protocolos de Transmisión Confiable 21
2.5.2. Protocolo Punto a Punto 24
2.5.3. El Problema de la Asignación del Canal 26
2.5.4. Protocolos de Acceso Múltiple Sin Detección de Portadora 27
2.5.5. Protocolos de Acceso Múltiple Con Detección de Portadora 28
2.5.6. Protocolos de Contención Limitada 30
2.5.7. Protocolos de Redes Inalámbricas 31
2.5.8. Protocolos Token Passing 32
2.6. Estandarización de Redes LAN 34
2.7. Tecnologías Ethernet 34
2.7.1. Especificación IEEE 802.3 y Ethernet 35
2.7.2. Especificación IEEE 802.3u Fast Ethernet 39
2.7.3. Gigabit Ethernet 41
2.8. Token Bus/IEEE 802.4 42
2.9. Token Ring/IEEE 802.5 45

2.10. 100VGAnyLAN/IEEE 802.12 . . . 47
2.11. Interfaces FDDI . . . 50
2.12. LAN Emulation: ATM y su Interconexión LAN . . . 51
2.13. Subcapa de Control de Enlace Lógico . . . 55
2.14. Dispositivos LAN . . . 57
2.14.1. Repetidores . . . 57
2.14.2. Hubs . . . 58
2.14.3. Bridges . . . 58
2.15. LAN Conmutadas . . . 62
2.15.1. Store and Forward vs Cut-Through . . . 63
2.15.2. LANs virtuales o VLANs . . . 63

3. Nivel de Red **67**
3.1. Introducción . . . 67
3.2. Algoritmos de Ruteo . . . 68
3.2.1. El Principio de Optimalidad . . . 68
3.2.2. Ruteo por el Camino Más Corto y Métricas . . . 69
3.2.3. Ruteo Basado en el Flujo . . . 69
3.2.4. Flooding . . . 70
3.2.5. Ruteo por Vector de Distancia . . . 70
3.2.6. Ruteo por Estado del Enlace . . . 70
3.2.7. Ruteo Jerárquico . . . 71
3.2.8. Ruteo Broadcast . . . 71
3.2.9. Ruteo Multicast . . . 72
3.3. Algoritmos de Control de Congestión . . . 72
3.3.1. Principios Generales del Control de Congestión . . . 73
3.3.2. Factores que Influyen en la Congestión . . . 74
3.3.3. Traffic Shaping y Traffic Policing . . . 74
3.3.4. Control de Admisión . . . 75
3.3.5. Choke Packets . . . 75
3.3.6. Descarte de Paquetes . . . 75
3.4. El Protocolo IP . . . 76
3.4.1. Fragmentación . . . 79
3.4.2. Direcciones IP . . . 81
3.4.3. División en Subredes . . . 84
3.4.4. Protocolos de Control IP . . . 85
3.5. Conceptos de Ruteo . . . 90
3.5.1. Sistema Autónomo . . . 91
3.5.2. Protocolos de Ruteo Interior . . . 91
3.5.3. Protocolos de Ruteo Exterior . . . 97
3.6. Classless Inter-Domain Routing: CIDR . . . 99
3.7. IPv6 . . . 101
3.7.1. Direcciones en IPv6 . . . 102

3.7.2. Encabezado IPv6 . . . 103
3.8. IP Clásico sobre ATM . . . 106

4. Tecnologías WAN **110**
4.1. Introducción . . . 110
4.2. Enlaces Punto-a-Punto . . . 110
4.3. Conmutación de Circuitos y de Paquetes . . . 111
4.4. Circuitos Virtuales WAN . . . 112
4.5. Servicios de Marcado . . . 113
4.6. Dispositivos WAN . . . 113
4.7. Encapsulado y Tunneling . . . 115
4.7.1. Virtual Private Network (VPN) . . . 115

5. Nivel de Transporte **117**
5.1. Introducción . . . 117
5.1.1. Direccionamiento . . . 118
5.1.2. Primitivas de Servicio de Transporte . . . 119
5.2. Elementos de Protocolos de Transporte . . . 119
5.3. Protocolos TCP y UDP . . . 123
5.3.1. Protocolo TCP . . . 123
5.3.2. Protocolo UDP . . . 127

1. Introducción y Marco de Referencia

1.1. ¿Qué Son las Redes de Datos? ¿Para Qué Sirven?

Red de Datos: conjunto de computadores, equipos de comunicaciones y otros dispositivos que se pueden comunicar entre sí, a través de un medio en particular. Objetivos principales:

1. La información debe ser entregada de manera confiable y sin daños en los datos.
2. La información debe entregarse de manera consistente.
3. Los equipos que forman la red deben ser capaces de identificarse entre sí.
4. Debe existir una manera estandarizada de nombrar e identificar las partes de la red.

Las redes, entre otras cosas, sirven para:

- Compartir recursos y ahorrar dinero.
- Aumentar la disponibilidad de la información.
- Permitir el acceso a información a una gran cantidad de usuarios (Internet).

1.2. Clasificación de las Redes

1.2.1. Clasificación por Tecnología de Transmisión

Redes de Difusión (Broadcasting): existe un sólo canal o medio de comunicación, que es compartido por todos los dispositivos de la red.

Redes de Punto-a-Punto: consisten en múltiples conexiones entre pares individuales de máquinas.

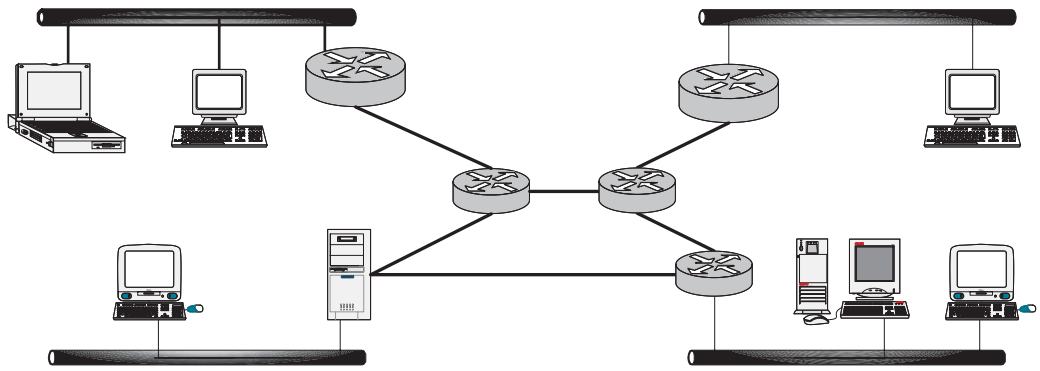

Figura 1: Ejemplo de Clasificación de Redes

1.2.2. Clasificación por Escala

LAN: son el punto de contacto de los usuarios finales. Su finalidad principal es la de intercambiar información entre grupos de trabajo y compartir recursos tales como impresoras y discos duros. Se caracterizan por tres factores: extensión (de unos cuantos metros hasta algunos kilómetros), su tecnología de transmisión (cable de par trenzado UTP o coaxial, fibra óptica, portadoras con infrarojo o láser, radio y microondas en frecuencias no comerciales) y su topología (anillo, bus único o doble, estrella, árbol y completas). Las velocidades en las LAN van desde los 10 Mbps hasta 622 Mbps.

Los estándares más comunes son el IEEE 802.3 llamado **Ethernet** y el IEEE 802.5 llamado **Token Ring**. *Ethernet* opera entre 10 y 1000 Mbps. En este estándar, todos los nodos escuchan todos los paquetes que circulan por la red, sacan una copia y examinan el destinatario. Si el destinatario es el nodo mismo, lo procesa y si no lo descarta para escuchar el siguiente. Para enviar un paquete sensa el medio para saber si está libre; de ser así procede a enviar el dato. Si ocurre que dos nodos enviaron un paquete al mismo tiempo, se provoca una colisión y cada nodo vuelve a retransmitir su paquete después de esperar un tiempo aleatorio. *Token Ring* opera entre 4 y 16 Mbps y utiliza un token o testigo, que permite , al nodo que lo posee, enviar paquetes a la red mientras los otros escuchan. Una vez que un nodo termina de enviar paquetes, pasa el token a otro nodo para que éste transmita.

MAN: corresponde es una versión más grande de una LAN en cuanto a topología, protocolos y medios de transmisión, que por ejemplo puede cubrir un conjunto de oficinas corporativas o empresas en una ciudad. En general, cualquier red de datos, voz o video con una extensión de una a varias decenas de kilómetros puede ser considerada una MAN. El estándar IEEE 802.6 define un tipo de MAN llamado DQDB que usa dos cables half-duplex por los cuales se recibe y transmiten voz y datos entre un conjunto de nodos. Un aspecto típico de las MAN es que el medio físico es de difusión, lo que simplifica el diseño de la red.

WAN: son redes que se expanden en una gran zona geográfica, por ejemplo, un país o continente. Los beneficiarios de estas redes son los que se ubican en nodos finales que son quienes corren aplicaciones de usuario. A la infraestructura que une los nodos de usuarios se le llama *subred* y abarca diversos aparatos de red (llamados *routers o ruteadores*) y líneas de comunicación que unen las diversas redes.

En la mayoría de las WAN se utilizan una gran variedad de medios de transmisión para cubrir grandes distancias. La transmisión puede efectuarse por microondas, por cable de cobre, fibra óptica o alguna combinación de los anteriores. Sin importar el medio, los datos en algún punto se convierten e interpretan como una secuencia de unos y ceros para formar frames de información, luego estos frames son ensamblados para formar paquetes y los paquetes a su vez construyen archivos o registros específicos de alguna aplicación.

1.3. Modelos de Referencia

1.3.1. El Modelo OSI

La ISO ha definido un modelo de 7 capas que describe cómo se transfiere la información desde una aplicación de software a través del medio de transmisión hasta una aplicación en

otro elemento de la red.

Capa Física. La capa física tiene que ver con el envío de bits en un medio físico de transmisión y se asegura de éstos se transmitan y reciban libres de errores. También describe los eléctricos y mecánicos asociados con el medio y los conectores así como los tiempos aprobados para enviar o recibir una señal. También especifica si el medio permite la comunicación simplex, half duplex o full duplex.

Capa de Enlace. En esta capa se toman los bits que entrega la capa física y los agrupa en algunos cientos o miles de bits para formar los frames. En este nivel se realiza un chequeo de errores y si devuelven acknowledges al emisor. La Capa de Enlace es la encargada de detectar si un frame se pierde o daña en el medio físico. De ser éste el caso, debe de retransmitirlo, aunque en ocasiones dicha operación provoca que un mismo frame se duplique en el destino, loa que obliga a esta capa a detectar tal anomalía y corregirla. En este nivel se decide cómo accesar el medio físico.

Capa de Red. Se encarga de controlar la operación de la subred. Su tarea principal es decidir cómo hacer que los paquetes lleguen a su destino dados un origen y un destino en un formato predefinido por un protocolo. Otra función importante en este nivel es la resolución de cuellos de botella. En estos casos se pueden tener varias rutas para dar salida a los paquetes y en base a algunos parámetros de eficiencia o disponibilidad se eligen rutas dinámicas de salida.

Capa de Transporte. La obligación de la capa de transporte es tomar datos de la capa de sesión y asegurarse que dichos datos llegan a su destino. En ocasiones los datos que vienen de la capa de sesión exceden el tamaño máximo de transmisión *(*Maximum Transmission Unit o MTU) de la interfaz de red, por lo cual es necesario partirlos y enviarlos en unidades más pequeñas, lo que origina la fragmentación y ensamblado de paquetes cuyo control se realiza en esta capa. Otra función en esta capa es la de multiplexar varias conexiones que tienen diferentes capacidades de transmisión para ofrecer una velocidad de transmisión adecuada a la capa de sesión.

La última labor importante de la capa de transporte es ofrecer un mecanismo que sirva para identificar y diferenciar las múltiples conexiones existentes, así como determinar en qué momento se inician y se terminan las conversaciones (esto es llamado *control de flujo*).

Capa de Sesión. Esta capa establece, administra y finaliza las sesiones de comunicación entre las entidades de la capa de presentación. Las sesiones de comunicación constan de solicitudes y respuestas de servicio que se presentan entre aplicaciones ubicadas en diferentes dispositivos de red. Estas solicitudes y respuestas están coordinadas por protocolos implementados en esta capa. Otro servicio de este nivel es la sincronización y el establecimiento de puntos de chequeo. Por ejemplo, si se hace necesario transferir un archivo muy grande entre dos nodos que tienen una alta probabilidad de sufrir una caída, es lógico pensar que

una transmisión ordinaria nunca terminaría porque algún interlocutor se caerá y se perderá la conexión. La solución es que se establezcan cada pocos minutos un punto de chequeo de manera que si la conexión se rompe más tarde se pueda reiniciar a partir del punto de chequeo, lo cual ahorrará tiempo y permitirá tarde o temprano la terminación de la transferencia.

Capa de Presentación. La capa de presentación provee servicios que permiten transmitir datos con alguna sintaxis propia para las aplicaciones o para el nodo en que se está trabajando. Como existen computadores que interpretan sus bytes de una manera diferente que otras (Big Endian versus Little Endian), es en esta capa donde es posible convertir los datos a un formato independiente de los nodos que intervienen en la transmisión.

Capa de Aplicación. En esta capa se encuentran aplicaciones de red que permiten explotar los recursos de otros nodos. Dicha explotación se hace, por ejemplo, a través de emulación de terminales que trabajan en un nodo remoto, interpretando una gran variedad de secuencias de caracteres de control que permiten desplegar en el terminal local los resultados, aún cuando éstos sean gráficos. Una situación similar se da cuando se transmiten archivos de un computador que almacena sus archivos en un formato dado a otro, que usa un formato distinto. Es posible que el programa de transferencia realice las conversiones necesarias de manera que el archivo puede usarse inmediatamente bajo alguna aplicación.

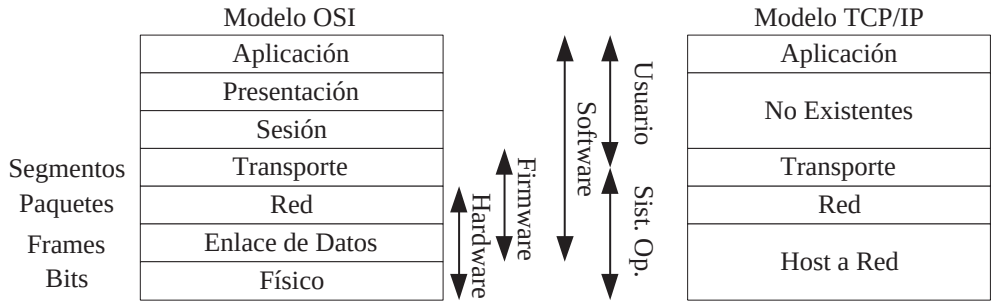

Figura 2: Comparación Entre los Modelos OSI y TCP/IP

1.3.2. El Modelo TCP/IP

El departamento de defensa de USA definió un conjunto de reglas que establecieron cómo conectar computadoras entre sí para lograr el intercambio de información, soportando incluso desastres mayores en la subred. Fue así como se definió el conjunto de protocolos de TCP/IP. Para los años 80 una gran cantidad de instituciones estaban interesados en conectarse a esta red que se expandió por todo EE.UU. El modelo TCP/IP consta solamente de 4 capas.

Capa Host a Red. La capa inferior, se relaciona con la capa física respecto del modelo OSI, y contiene varios estándares del IEEE como el 802.3 llamado *Ethernet* que establece las reglas para enviar datos por cable coaxial delgado (10Base2), cable coaxial grueso (10Base5), par trenzado (10Base-T), fibra óptica (10Base-F) y su propio método de acceso al medio físico. El 802.4 llamado *Token Bus* que puede usar estos mismos medios pero con un método de acceso diferente y otros estándares denominados genéricamente como 802.X.

Capa de Red. Esta capa cumple, junto con la anterior, los niveles 1, 2 y 3 del modelo OSI. En este nivel se definió el protocolo IP cuya responsabilidad es entregar paquetes en los destinos indicados, realizando las operaciones apropiadas de ruteo y la solución de problemas como congestión o caídas de enlaces.

Capa de Transporte. Está formada por dos protocolos: *TCP* y *UDP*. El primero es un protocolo confiable y orientado a conexión, lo que significa que ofrece un medio libre de errores para enviar paquetes. El segundo es un protocolo no orientado a conexión y no es confiable.

Capa de Aplicación. En la última capa se encuentran decenas de aplicaciones ampliamente conocidas actualmente. Las más populares son los protocolos WWW, FTP, telnet, DNS, el servicio de correo electrónico (SMTP), etc.

1.4. Protocolos, Interfaces, Servicios y Tipos de Servicios

Protocolo de comunicación: es un conjunto de reglas que indican cómo se debe llevar a cabo un intercambio de datos o información. Para que dos o más nodos en una red puedan intercambiar información es necesario que manejen el mismo conjunto de reglas, es decir, un mismo protocolo de comunicaciones.

Interfaz: corresponde a la separación o división entre dos capas de un modelo de comunicación, y es la encargada de definir las operaciones básicas y los servicios que el nivel inferior ofrece a la capa superior del modelo.

Servicios: son un conjunto de operaciones que un nivel provee al nivel superior. en otras palabras, define que operaciones puede ejecutar la capa, pero no especificar cómo son implementadas estas operaciones.

Entidades: son los elementos activos en cada nivel del modelo. Una entidad puede ser un software (un proceso) o hardware (un chip).

Cada capa tiene un conjunto de operaciones que realizar y un conjunto de servicios que usa de otra capa. De esta manera, se identifica como usuario de servicio a la capa que solicita un servicio y como proveedor a quien la da. Cuando una entidad se comunica con otra ubicada en la misma capa pero en diferentes nodos se dice que se establece comunicación entre entidades pares.

Cada capa tiene un conjunto de servicio que ofrecer, el punto exacto donde se puede pedir el servicio se llama punto de acceso al servicio o SAP. En cada capa, la entidad activa recibe un bloque de datos consistente de un encabezado que tiene significado para el protocolo de esa

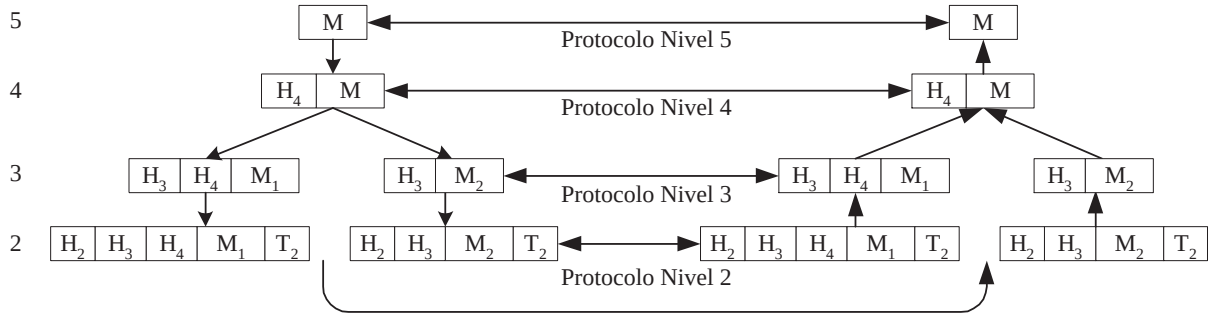

Figura 3: Flujo de Información en una Comunicación

capa y un cuerpo que contiene datos para ser procesados por esa entidad o que van dirigidos a otra capa.

Las capas ofrecen servicios de dos tipos: *orientadas a la conexión* y *no orientadas a la conexión*. Además, cada uno de estos servicios puede ser caracterizados por la cierta calidad de servicio que ofrecen. Así, se pueden tener *servicios confiables* y *servicios no confiables*.

Servicios orientados a la conexión. Es un tipo de servicio en el que obligatoriamente debe establecerse una conexión o camino, entre el origen y el destino antes de que cualquier dato pueda transmitirse. Los servicios orientados a conexión se caracterizan porque cumplen tres etapas en su tiempo de vida: negociación del establecimiento de la conexión (etapa 1), sesión de intercambio de datos (etapa 2) y negociación del fin de la conexión (etapa 3). Los servicios orientados a la conexión pueden ser considerados como "alambrados", es decir, que existe un conexión alambrada entre los dos interlocutores durante el tiempo de vida de la conexión.

Servicios no orientados a conexión. Los servicios no orientados a conexión carecen de las tres etapas antes descritas y en este caso, los interlocutores envían todos paquetes de datos que componen una parte del diálogo, por separado, pudiendo éstos llegar a su destino en desorden y por diferentes rutas. Es responsabilidad del destinatario ensamblar los paquetes, pedir retransmisiones de paquetes que se dañaron y darle coherencia al flujo recibido.

Servicio confiable. Un servicio es confiable si ofrece una transmisión de datos libre de errores. Para cumplir este requisito, el protocolo debe incluir mecanismos para detectar y/o corregir errores. La corrección de errores puede hacerse con información que está incluida en un paquete dañado o pidiendo su retransmisión al interlocutor. También es común que incluya mecanismos para enviar acuses de recibo cuando los paquetes llegan correctamente.

Servicio no confiable. Un servicio es no confiable si el protocolo no asegura que la transmisión está libre de errores y es responsabilidad del protocolo de una capa superior (o de

la aplicación) la detección y corrección de errores si esto es pertinente o estadísticamente justificable.

A un servicio que es a la vez no orientado a la conexión y no confiable se le conoce como *servicio de datagramas*. Un servicio que es no orientado a la conexión pero que incluye acuse de recibo se conoce como *servicio de datagramas con acuse de recibo*. Un tercer tipo de servicio se le llama con *solicitud de respuesta* si consiste de un servicio no orientado a conexión y por cada envío de datos se espera una respuesta inmediata antes de enviar el siguiente bloque de datos.

1.5. Ejemplos de una Red de Datos

1.5.1. Ejemplo de LAN: Red del DIE

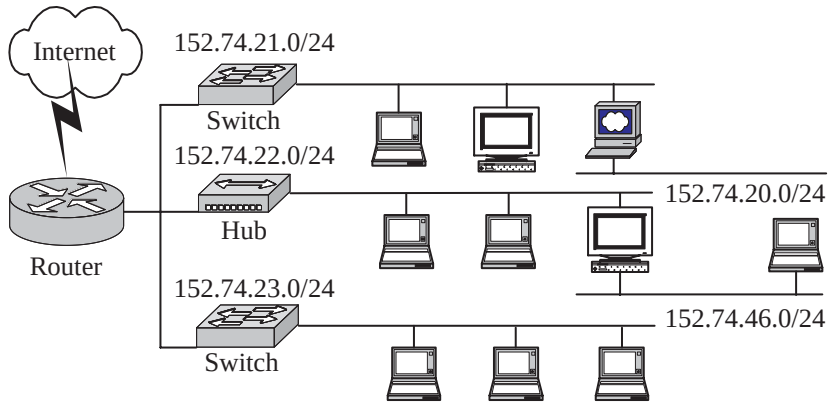

Figura 4: Esquema Lógico de la Red LAN del DIE.

El Departamento de Ingeniería Eléctrica consta de una red de datos formada por cinco subredes lógicas, de las cuales cuatro están funcionando y la otra está reservada para su uso futuro. La subred 152.74.21.0, ubicada en el segundo piso del edificio Tecnológico Mecánico está generada a partir de un switch, del cual cuelgan hubs que son los encargados de distribuir los puntos de red a los distintos clientes. El cableado está trazado con UTP categoría 5 principalmente, salvo por un par de laboratorios que, a partir de un hub, cablean su espacio utilizando coaxial delgado. El switch que genera la subred se conecta al router principal a través de fibra óptica. En esta subred se encuentran los servidores de correo, web, ftp, DNS, etc. además de los computadores presentes en cada uno de los laboratorios existentes en el edificio. Cabe mencionar que a partir de una máquina SUN usada como router, se genera la subred 152.74.20.0, la que actualmente no está siendo utilizada. La velocidad de la red es 10 Mbps.

La subred 152.74.22.0 es generada por un hub de cable coaxial, el que alimenta usando este cable una serie de clientes PCs y otros hubs de cable UTP categoría 5, a los cuales se

conectan más computadores. Al igual que en el caso de la otra subred, la conexión al router de salida está hecha usando fibra óptica, y la subred se encuentra ubicada físicamente en el segundo piso del mismo edificio. Las máquinas conectadas a esta subred son principalmente PCs y estaciones SUN SPARC. Nuevamente, al igual que en el caso anterior, usando una estación SUN como router se genera la subred 152.74.46.0, la que sirve para realizar pruebas de conexión y de servicios. La velocidad de la red es 10 Mbps.

La subred 152.74.23.0 está ubicada físicamente en el edificio nuevo de la Facultad de Ingeniería, y se genera a partir de un switch, que alimenta un hub y la serie de clientes PC que pertenecen a los profesores del departamento. El cableado de la red es estructurado, UTP categoría 5, y la conexión del switch con el router es a través del mismo tipo de cable. La velocidad de la red es, nuevamente, 10 Mbps.

El ruteo es simple, pues como las subredes se generan a partir de un mismo router las rutas son directas. No es el caso de las subredes 152.74.20.0 y 152.74.46.0, pues se utiliza una ruta estática para cada subred, y están indicada en el router de salida. Este router tiene routeo estático, usando protocolo RIP, y una ruta por defecto para la conexión del equipo con el resto de la red del campus y finalmente con Internet.

1.5.2. Ejemplo de WAN: Red Reuna 2

Reuna 2, es una Red de Tecnología ATM basada en una red de Transporte SDH. Está compuesta por un Backbone o troncal de una velocidad de 155 Mbps, cuyos nodos centrales están distribuídos a lo largo del país entre Arica y Puerto Montt. A esta troncal se conectan las Universidades miembros del Consorcio REUNA, mediante enlaces de fibra óptica, cuyas velocidades de acceso son de 155 Mbps.

La red opera de la siguiente forma cuando un cliente de la red trata de conectarse a otro punto: *Sí* la dirección de red IP destino *pertenece* a las direcciones de red IP de Reuna, *entonces*, por medio de un enlace dedicado, establecido hasta la Universidad destino, desde la Universidad origen, se debe dar paso a la transferencia de datos. Esto es llamado *tráfico nacional o Intranet*. *Si* la dirección de red IP destino *no pertenece* a las direcciones de red IP de Reuna, *entonces* se debe utilizar otro enlace dedicado, que es exclusivo para cada universidad, hasta el router central, que encaminará la petición de conexión donde corresponda por el enlace internacional o hacia otro router, según sus tablas de rutas. Esto es llamado *tráfico internacional o Internet*.

El diseño de la red contempla que para el tráfico nacional o intranet se aplica un ancho de banda dedicado de 10 Mbps, y utiliza el protocolo de ruteo dinámico OSPF. La conectividad a Internet de cada Universidad está dada por un circuito dedicado exclusivo, cuyo ancho de banda es el contratado para salida Internacional. El ruteo en este circuito es estático, es decir, cada router de acceso tiene configurada la ruta por defecto hacia el router central por su circuito exclusivo y el router central tiene rutas estáticas para llegar a cada red interna de las Universidades por el circuito que corresponda.

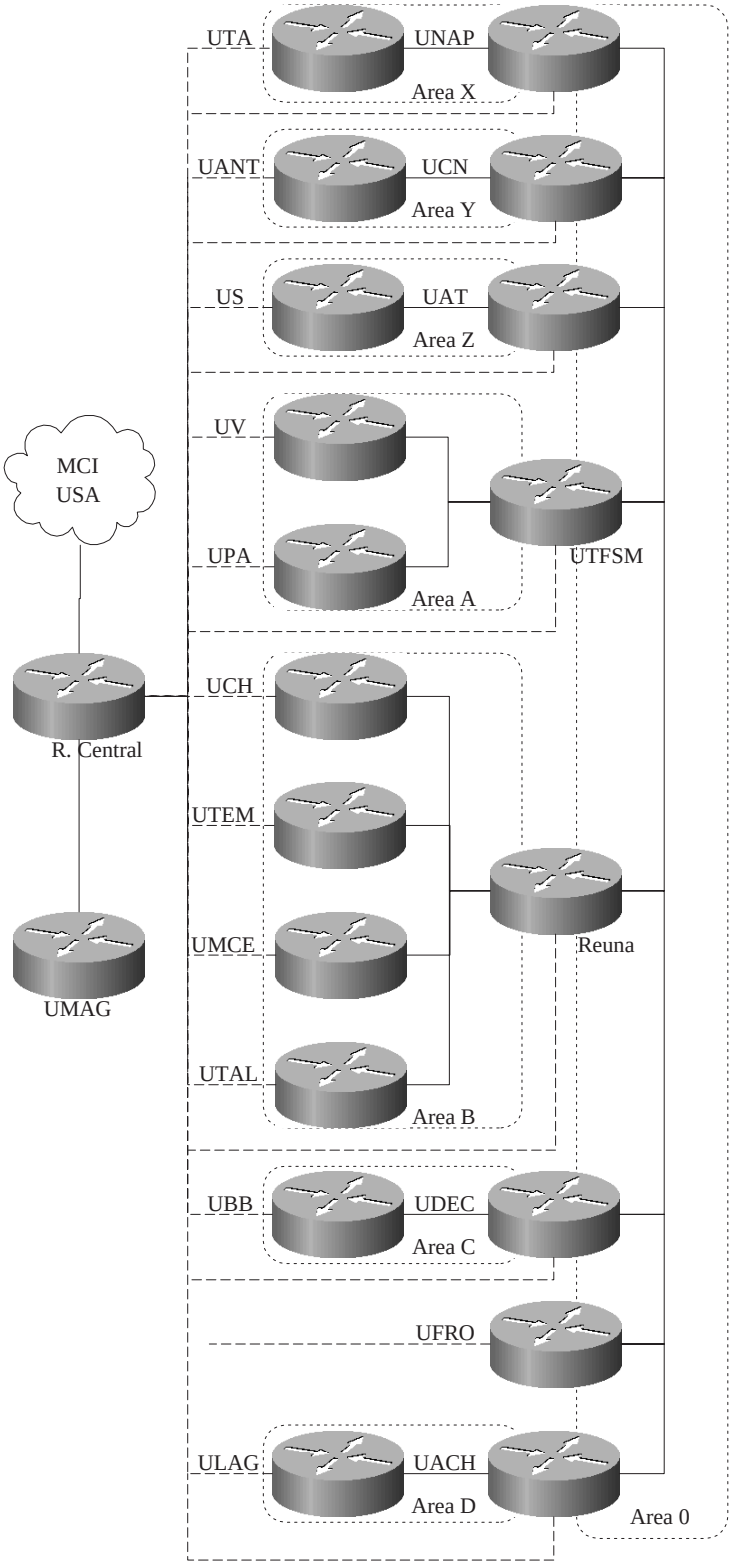

Figura 5: Esquema a Nivel IP de la Red Reuna 2.

2. Protocolos LAN

2.1. Introducción

Una LAN es una red de datos de alta velocidad, tolerante a fallas, que cubre un área geográfica relativamente pequeña. Generalmente conectan estaciones de trabajo, impresoras, PCs, etc. permitiendo el acceso compartido a dispositivos y aplicaciones, intercambio de archivos, etc.

Las redes LAN podemos dividirlas en:

- LAN tradicionales entre las que están los estándares IEEE 802.3, IEEE 802.4 y IEEE 802.5.
- LAN rápidas entre las que cuentan Fast Ethernet, 100VGAnyLAN, FDDI, ATM y Gigabit Ethernet.
- LAN inalámbricas.

2.2. Protocolos LAN y Modelo OSI

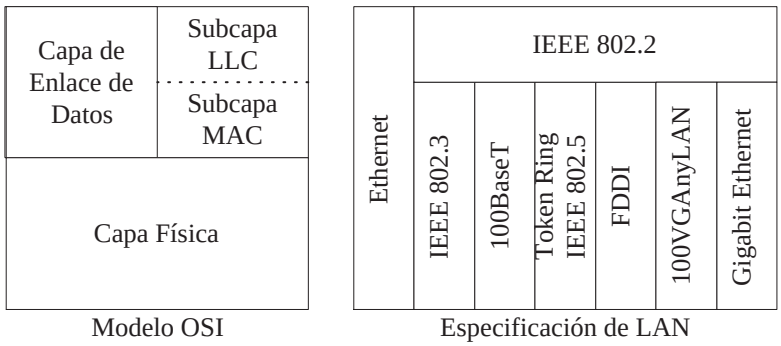

Figura 6: Relación Entre el Modelo OSI y los Protocolos LAN.

El término Ethernet se refiere a la familia de implementaciones LAN que incluyen tres categorías Ethernet e IEEE 802.3, Ethernet a 100 Mbps y Ethernet a 1000 Mbps. De lo anterior se desprende que existe una diferencia entre Ethernet e IEEE 802.3, ya que el primero especifica las capas 1 y 2 del modelo OSI, en cambio, IEEE 802.3 especifica la capa física y la subcapa MAC no definiendo la subcapa LLC (estándar IEEE 802.2), que es común para IEEE 802.5, 100BaseT, etc. Estas diferencias se aprecian en la figura 6.

2.3. Topologías

Las topologías típicas de una LAN se pueden observar en la figura 7. Las topologías mostradas son topologías lógicas, por lo que no necesariamente deben ser topología físicas de

conexión.

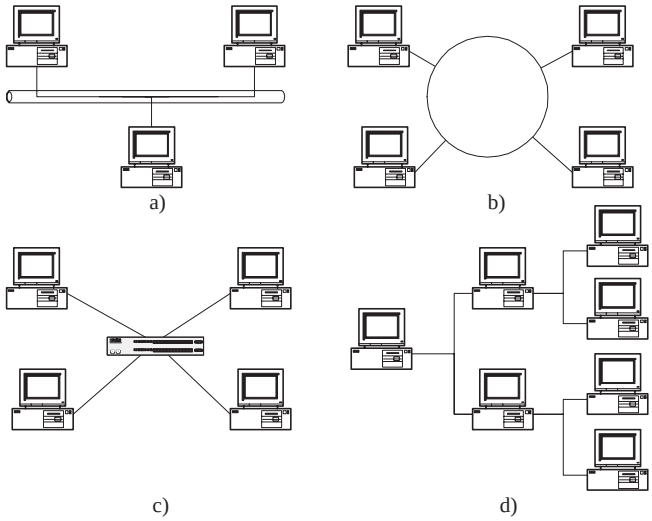

Figura 7: Topologías Típicas de una LAN. a) Bus b) Anillo c) Estrella d) Árbol.

2.4. El Medio Físico

Los medios de transmisión más utilizados en una LAN son el cable coaxial grueso y delgado, par trenzado y fibra óptica. Estos medios de transmisión son llamados *guiados*, a diferencia de los *no guiados* como los enlaces de radio, de microondas o satelitales.

2.4.1. Cable Coaxial

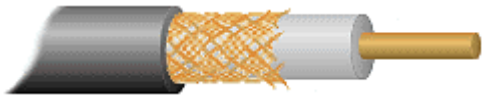

Figura 8: Cable Coaxial.

Consiste en un cable conductor interno cilíndrico separado de otro cable conductor externo por anillos aislantes o por un aislante macizo. Esto se recubre por otra capa aislante que es la funda del cable. Este medio físico, es más caro que el par trenzado, pero se puede utilizar a más larga distancia, con velocidades de transmisión superiores, menos interferencias y permite conectar más estaciones.

Se suele utilizar para televisión, telefonía a larga distancia, LAN, conexión de periféricos a corta distancia, etc. Se utiliza para transmitir señales analógicas o digitales. Sus inconvenientes

principales son: atenuación, ruido térmico, ruido de intermodulación. Para señales analógicas, se necesita un amplificador cada pocos kilómetros y para señales digitales un repetidor cada kilómetro.

2.4.2. Par Trenzado

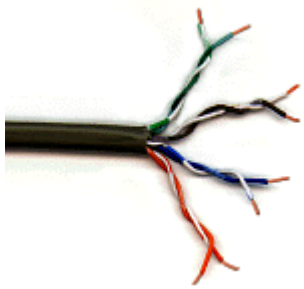

Figura 9: Cable Par Trenzado.

Se trata de dos hilos de cobre aislados y trenzados entre sí, y envueltos por una cubierta protectora. Los hilos están trenzados para reducir las interferencias electromagnéticas con respecto a los pares cercanos que se encuentran a su alrededor (dos pares paralelos constituyen una antena simple, en tanto que un par trenzado no). Se pueden utilizar tanto para transmisión analógica como digital, y su ancho de banda depende de la sección de cobre utilizado y de la distancia que tenga que recorrer.

Se trata del cableado más económico y la mayoría del cableado telefónico es de este tipo. Presenta una velocidad de transmisión que depende del tipo de cable de par trenzado que se esté utilizando. Está dividido en categorías por el EIA/TIA :

Categoría 1 Hilo telefónico trenzado de calidad de voz no adecuado para las transmisiones de datos. Velocidad de transmisión inferior a 1 Mbps.

Categoría 2 Cable de par trenzado sin apantallar. Su velocidad de transmisión es de hasta 4 Mbps.

Categoría 3 Velocidad de transmisión de 10 Mbps. Con este tipo de cables se implementa las redes Ethernet 10BaseT.

Categoría 4 La velocidad de transmisión llega a 16 Mbps.

Categoría 5 Puede transmitir datos hasta 100 Mbps.

Tiene una longitud máxima limitada y, a pesar de los aspectos negativos, es una opción a tener en cuenta debido a que ya se encuentra instalado en muchos edificios como cable

telefónico y esto permite utilizarlo sin necesidad de cambiar el cableado. Además, resulta fácil de combinar con otros tipos de cables para la extensión de redes.

Existen dos tipos de pares trenzados, los *apantallados o STP* y los *sin apantallar o UTP*. Los pares sin apantallar son los más baratos aunque menos resistentes a interferencias. A velocidades de transmisión bajas, los pares apantallados son menos susceptibles a interferencias, aunque son más caros y más difíciles de instalar.

2.4.3. Fibra Óptica

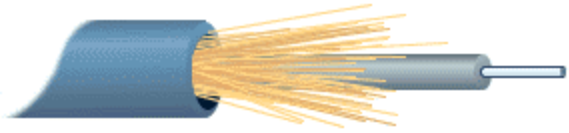

Figura 10: Fibra Óptica.

Se trata de un medio muy flexible y muy fino que conduce energía de naturaleza óptica. Su forma es cilíndrica con tres secciones radiales: núcleo, revestimiento y cubierta. El núcleo está formado por una o varias fibras muy finas de cristal o plástico. Cada fibra está rodeada por su propio revestimiento que es un cristal o plástico con diferentes propiedades ópticas distintas a las del núcleo . Alrededor de esto está la cubierta, constituida de material plástico o similar, que se encarga de aislar el contenido de aplastamientos, abrasiones, humedad, etc.

Sus beneficios frente a cables coaxiales y pares trenzados son:

- Permite mayor ancho de banda.
- Menor tamaño y peso.
- Menor atenuación.
- Aislamiento electromagnético.
- Mayor separación entre repetidores.

Su rango de frecuencias es todo el espectro visible y parte del infrarrojo. El método de transmisión es el siguiente: los rayos de luz inciden con una gama de ángulos diferentes posibles en el núcleo del cable, entonces sólo una gama de ángulos conseguirán reflejarse en la capa que recubre el núcleo. Son precisamente esos rayos que inciden en un cierto rango de ángulos los que irán rebotando a lo largo del cable hasta llegar a su destino. A este tipo de propagación se le llama *multimodal*. Si se reduce el radio del núcleo, el rango de ángulos disminuye hasta que sólo sea posible la transmisión de un rayo, el rayo axial, y a este método de transmisión se le llama *monomodal*.

Los inconvenientes del modo multimodal es que debido a que dependiendo al ángulo de incidencia de los rayos, estos tomarán caminos diferentes y tardarán más o menos tiempo en

llegar al destino, con lo que se puede producir una distorsión (rayos que salen antes pueden llegar después). Debido a esto, se limita la velocidad de transmisión posible.

Hay un tercer modo de transmisión que es un paso intermedio entre los anteriormente comentados y que consiste en cambiar el índice de refracción del núcleo. A este modo se le llama *multimodo de índice gradual*.

Los emisores de luz utilizados son: LED (de bajo costo, con utilización en un amplio rango de temperaturas y con larga vida media) e ILD (más caro, pero más eficaz y permite una mayor velocidad de transmisión).

2.4.4. Enlaces de Radio

Las ondas de radio tienen como principales características que son fáciles generar, pueden viajar distancias largas, y penetran edificios fácilmente. Además, son omnidireccionales, lo que significa que ellas viajan en todas las direcciones desde la fuente, para que el transmisor y receptor no tengan que estar físicamente alineados con cuidado.

Las propiedades de ondas son dependientes de la frecuencia. A frecuencias bajas, atraviesan bien obstáculos, pero el poder baja grandemente cuando se aleja de la fuente. A frecuencias altas, las ondas tienden a viajar en líneas rectas y rebotar cuando consiguen obstáculos. Ellas también son absorbidas por la lluvia. A cualquier frecuencia, las ondas están sujetas a interferencia de los motores y otros equipos eléctricos. El problema principal que se presenta al usar estas bandas para comunicación de datos es el ancho de banda relativamente bajo que ellas ofrecen.

Debido a la habilidad de radio de viajar grandes distancias, la interferencia entre los usuarios es un problema. Por esta razón, todos los gobiernos licencian al usuario de transmisores de radio.

2.4.5. Enlaces de Microondas

Por encima de los 100 MHz, las ondas viajan en líneas rectas y pueden por consiguiente enfocarse estrechamente. Concentrando toda la energía en una haz pequeño usando una antena parabólica se obtiene una razón señal a ruido bastante alta, permitiendo la comunicación, pero las antenas transmisoras y receptoras deben alinearse con precisión entre sí. Además, esta direccionalidad permite que múltiples transmisores sean alineados seguidamente para comunicarse con múltiples receptores seguidos sin interferencia.

Puesto que las microondas viajan en una línea recta, si las torres están demasiado separadas, la Tierra estará en el camino (recordar la curvatura del palneta). Por consiguiente, se necesitan repetidoras periódicamente. Mientras más altas sean las torres, más distantes pueden estar. La distancia entre las repetidoras sube muy bruscamente con la raíz cuadrada de la altura de la torre. Para torres con altura de 100 metros, las repetidoras pueden estar separadas entre sí unos 80 kms. Este hecho las hace ser relativamente baratas.

A diferencia de las ondas a bajas frecuencias, las microondas no atraviesan bien edificios. Más aún, aunque el haz pueda enfocarse bien al transmisor, hay todavía alguna divergencia en el espacio. Algunas ondas pueden refractarse por capas atmosféricas bajas y pueden tomar

ligeramente más tiempo en llegar que las ondas directas. Las ondas retrasadas pueden llegar fuera de fase con la onda directa y por lo tanto cancelar la señal.

La comunicación por microondas se usa ampliamente para la comunicación de teléfono a larga distancia, teléfonos celulares y distribución de la televisión.

2.4.6. Infrarrojos y Ondas Milimétricas

Estos medios de transmisión son ampliamente usados en la comunicación de corto rango, por ejemplo, controles remotos de televisores, VCRs, etc. Son relativamente direccionales, baratos, y fáciles de construir, pero su mayor inconveniente es que no atraviesan objetos sólidos. Por otro lado, el hecho que las ondas infrarrojas no atraviesen paredes sólidas también es una ventaja. Significa que un sistema infrarrojo en un cuarto de un edificio no interferirá con un sistema similar en oficinas adyacentes. Además, la seguridad de sistemas infrarrojos contra escuchar detrás de las puertas es mejor que el de sistemas de radio precisamente por esta razón. Por esto, ninguna licencia gubernamental se necesita para operar un sistema infrarrojo, en contraste con sistemas de radio que deben ser autorizados.

Estas propiedades han hecho del infrarrojo un candidato interesante para LANs inalámbricas interiores. Por ejemplo, pueden equiparse computadores y oficinas en un edificio con transmisores y receptores infrarrojos sin necesidad de enfocar.

2.4.7. Enlaces Satelitales

Un satélite de comunicación puede ser pensado como un repetidor de microondas en el cielo. Contiene diversos transponders, cada uno de los cuales escucha alguna porción del espectro, amplifica la señal entrante, y hace una difusión de vuelta en otra frecuencia para evitar interferencia con la señal que entra. Los rayos que bajan son anchos o angostos, pudiendo cubrir grandes o pequeñas superficies de la tierra, respectivamente.

Los enlaces satelitales se diferencian de los enlaces punto a punto terrestres en que los retardos producto de las distancia involucradas son considerables, típicamente 270 mseg. Esto es bastante en comparación con los 3 μseg/km de los enlaces de microondas y los 5 μseg/km del coaxial o la fibra. Otra diferencia es que los satélites son por naturaleza elementos de difusión, lo que es útil en algunos casos, pero en otros, como la seguridad, no lo es. Otras características son que el costo de una transmisión es independiente de la distancia y que tienen una tasa de error bajísima.

2.5. Capa de Enlace de Datos

El objetivo principal de esta capa son los métodos para la comunicación confiable y eficiente entre dos máquinas adyacentes. Los problemas típicos que debe resolver son: los errores en los circuitos de comunicación, sus velocidades finitas de transmisión y el tiempo de propagación. Además debe proveer cuatro servicios principales a la capa de red.

Transferencia de Datos la función principal del nivel es transferir los datos del nivel de red desde la fuente hasta el nivel de red del destino. Para esto, genera *servicios no*

orientados a la conexión sin acuse de recibo, donde la máquina de fuente envía frames al destino sin necesidad de saber si esta los recibió. Es apropiado si la frecuencia de errores del canal es muy baja o si el tráfico es de tiempo real. Otro servicio un poco más confiable es el *no orientado a la conexión con acuse de recibo* donde tampoco se requiere una conexión preestablecida, pero donde cada frame debe ser chequeado como recibo por el receptor. Este servicio es ideal en medios de transmisión no confiables como los inalámbricos. El servicio más confiable de implementar es el *orientado a la conexión con acuse de recibo* donde se debe establecer una conexión previo a la trasmisión de datos. Cada frame en este servicio es numerado y enviado al destino, donde llegan en el mismo orden de envío y luego son acusados como recibidos.

Creación de Frames la capa de enlace recibe de la física un flujo de bits, que puede o no estar libre de errores. Para asegurar un servicio libre de errores la capa crea frames, que son simplemente una cierta cantidad de bits recibidos desde el nivel inferior que incluyen un checksum y que tienen una cierta interpretación útil. La generación de frames presenta un gran problema de sincronismo, dónde empieza o termina un frame. Una forma de solucionar esto es mediante la *cuenta de caracteres* donde existe un campo que indica la cantidad de caracteres que contiene el frame. Esta forma presenta serios problemas si se corrompe el valor del campo que indica la cantidad de caracteres. Una forma alternativa de solución es utilizar *caracteres de inicio y término* como los caracteres ASCII DTE STX y DTE ETX, que indican el inicio y fin del texto transmitido. Otra alternativa, que resulta ser una mejora a la anterior es el *bit stuffing*, donde se utilizan patrones de bits que son las marcas de inicio y fin del mensaje, y si por algún motivo los datos presentan dentro de sí el pattern de marca, este es modificado agregando un bit que le receptor sabe de antemano que debe remover. La última forma de crear los frames es aplicable sólo a medios físicos que presentan redundancia, y consiste en crear *violaciones en la codificación del código.*

Control de Errores en los servicios confiables es necesario proveer mecanismos que permitan retransmitir de alguna forma los frames que han llegado erróneos. Para esto se utilizan acuses de recibo positivos (frame está OK) y negativos (frame llegó en mal estado y debe retransmitirse). Además de los acuses de recibo deben utilizarse timers y números de secuencia, pues puede darse el hecho de que se pierda algún acuse, lo que generaría que la transmisión de cuelgue, o bien que un frame llegue más de una vez en buen estado.

Control de Flujo se relaciona con el hecho de cómo regular el envío de información desde el emisor al receptor, que no tienen igual capacidad de procesamiento de datos. Para esto existen protocolos que no dejan transmitir datos al emisor sin previa autorización del emisor.

Detección y Corrección de Errores

Se han desarrollado dos estrategias fundamentales para el manejo de los errores que pueden

presentarse en la transmisión de datos. Una de ellas consiste en agregar información redundante en cada bloque de datos enviado, para que el receptor pueda deducir cuál fue el caracter que se envió. La otra estrategia consiste en incluir redundancia para que el receptor pueda detectar la existencia de un error y solicitar una retransmisión. En el primer caso se utilizan *códigos correctores de error* y en el segundo caso *códigos detectores de errores*. Los métodos más usados en la detección y corrección de errores son: chequeo de redundancia cíclica, bits de paridad, código de Hamming y los algoritmos de Checksum.

2.5.1. Protocolos de Transmisión Confiable

Un frame es una estructura que contiene varios campos: el tipo de frame, número de secuencia (seq), número de acuse de recibo (ack) y naturalmente el paquete recibido o enviado a la capa de red. También una serie de rutinas que permiten la detección de acontecimientos varios o el establecimiento de tiempos de expiración para determinadas situaciones de espera. Dado que en algunos casos interesa asociar un número de orden a cada frame, se contempla una variable k que se va a utilizar para identificar cada frame, hasta el valor máximo MAX-PKT a partir del cual se empieza otra vez desde cero.

Protocolos Elementales

1. Protocolo simplex no restringido
Se supone una comunicación perfecta, sin errores, donde el receptor está siempre disponible y preparado para recibir frames con un espacio de buffer infinito, por lo que no debe efectuarse control de flujo. El emisor está siempre preparado para transmitir cualquier cosa que reciba de la capa de red. En este caso el único evento posible es llegada de un frame.

2. Protocolo simplex stop and wait
En un caso más real, puede suceder que el receptor no siempre está disponible para recibir, por tener ocupado su buffer de entrada debido a que la capa de enlace no sea capaz de procesar los frames con suficiente rapidez o porque la capa de red del receptor no sea lo bastante rápida. En este caso, lo más sencillo es que el emisor espere una confirmación después de enviar cada frame, de forma que sólo después de recibir la confirmación envíe el siguiente. De esta manera se garantiza la no saturación del receptor. Esto es lo que se conoce como un protocolo stop and wait.

3. Protocolo simplex para un canal con ruido
Si el canal de comunicación no es perfecto, los frames pueden alterarse debido al ruido de la comunicación, o incluso perderse por completo. Gracias a la presencia del campo CRC el receptor podrá detectar la llegada de un frame defectuoso, en cuyo caso pedirá retransmisión. La posibilidad de que un frame se pierda por completo introduce una complicación adicional, ya que si esto le ocurre por ejemplo a un ack el emisor pasado un tiempo razonable enviará el frame de nuevo pensando que no ha llegado la primera vez, lo cual produciría un

frame duplicado que el receptor pasaría a la capa de red, situación inaceptable para cualquier protocolo.

Para poder reconocer cuando un frame llega duplicado lo más sencillo es numerarlo, en este caso, el emisor no enviará un frame hasta recibir el acuse de recibo del anterior, por lo que bastaría con numerarlos con un sólo bit. Los protocolos donde el emisor espera una confirmación o acuse de recibo para cada dato enviado se denominan protocolos *Positive Acknowledgement with Retransmission (PAR)* o *Automatic Repeat reQuest (ARQ)*.

En este protocolo el receptor no realiza la comprobación del campo CRC, para él todo frame que reciba de la capa física es correcto y se supone que éstos pueden llegar o perderse, pero no llegar de forma parcial o alterados. Se puede considerar que hay un nivel inferior que se ocupa de comprobar el CRC, y que descarta el frame en caso de detectar cualquier error. De esta forma el efecto sería equivalente a la suposición simplista de que los frames o no llegan o llegan perfectamente.

Protocolos de Ventana Deslizante

Los protocolos vistos transmiten datos en una sola dirección, y el canal de retorno es utilizado únicamente para enviar frames de acknowledge cuyo contenido es irrelevante. Si se tiene que transmitir datos en ambas direcciones podría utilizarse dos canales half dúplex con los protocolos anteriores, pero sería mas eficiente utilizar el canal half dúplex ya existente para enviar, en cada sentido, frames de datos y de ack. El campo kind permitirá diferenciar unos de otras.

Aun más eficiente, en vez de generar un frame ack de manera automática cada vez que se recibe algo, podría esperarse a enviarlo cuando haya información útil que enviar. En tal caso, el ack viajaría "gratis" y se ahorraría el envío de un frame. Esta técnica se conoce como *piggybacking o piggyback acknowledgement*. Para "montar" el ack en un frame de datos es preciso que éste llegue pronto, o de lo contrario el emisor reenviará el frame, lo que hecharía por tierra la idea. Como no es posible saber de antemano cuando va a llegar el siguiente paquete de la capa de red, generalmente se adopta una solución salomónica: se espera un determinado tiempo y si no llega ningún paquete en ese tiempo se genera un frame ack.

1. Protocolo full duplex con piggybacking

En el protocolo anterior los frames se numeraban con un bit. La numeración sólo era utilizada en el receptor para verificar el frame recibido, no para informar al emisor a que frame se aplicaba el ack. Esto produce problemas en el caso de un receptor lento y un frame que se pierde. Si el receptor informa en el ack del frame de datos recibido, el problema de que el emisor tome un ack como correspondiente al paquete equivocado desaparece, el protocolo se hace más robusto ya que tanto emisor como receptor llevan control del frame recibida.

Cada frame enviado contiene el número de secuencia correspondiente (seq), el número correspondiente al paquete recién recibido del otro lado (ack) y los datos a enviar. Para que el protocolo funcione correctamente es preciso convenir en que uno de los lados inicie la comunicación, y el otro le siga. De lo contrario podría ocurrir que ambos lados iniciaran la comunicación a la vez, y cada frame es enviado dos veces. También se enviarían duplicados en caso de tener los timers demasiado bajos, sin embargo, los duplicados serían correctamente

detectados a partir de la información de secuencia y la información transmitida a la capa de red sería correcta

2. Protocolo de retroceso n

Los protocolos ARQ son aquellos en que se espera confirmación para cada dato enviado. Se suelen distinguir dos tipos de ARQ, el RQ "ocioso" (iddle RQ), que corresponde a los protocolos stop and wait, o de ventana deslizante de un bit, y el RQ continuo (continuous RQ) que se utiliza en los protocolos de ventana deslizante de más de un bit.

Cuando se utiliza un protocolo de ventana deslizante de más de un bit el emisor no actúa de forma sincronizada con el receptor: cuando el receptor detecta un frame defectuoso hay varios posteriores ya en camino, que llegarán de todas formas a él, aún cuando reporte el problema inmediatamente. Existen dos posibles estrategias en este caso: el receptor ignora los frames recibidos a partir del erróneo (inclusive) y solicita al emisor la retransmisión de todas los frames siguientes. Esta técnica se denomina *retroceso n* y corresponde a una ventana deslizante de tamaño uno en el receptor. La segunda opción consiste en que el receptor descarte el frame erróneo y pide retransmisión de éste, pero acepte los frames posteriores que hayan llegado correctamente. Esto se conoce como *repetición selectiva* y corresponde a una ventana deslizante mayor de 1 en el receptor (normalmente de igual tamaño que la ventana del emisor). En el caso de retroceso n el receptor se asegura que los frames se procesarán en secuencia, por lo que no tiene que reservar espacio en el buffer para más de un frame. En el caso de repetición selectiva el receptor ha de disponer de espacio en el buffer para almacenar todos los frames de la ventana, ya que en caso de pedir retransmisión tendrá que intercalar en su sitio el frame retransmitido antes de pasar los siguientes a la capa de red (recordar que la capa de red debe recibir los paquetes estrictamente en orden).

En cualquiera de los dos casos el emisor deberá almacenar en su buffer todos los frames que se encuentren dentro de la ventana, ya que en cualquier momento el receptor puede solicitar la retransmisión de alguno de ellos.

3. Protocolo con repetición selectiva

La repetición selectiva consiste en aprovechar aquellos frames correctos que lleguen después del erróneo, y pedir al emisor que retransmita únicamente este frame. Su funcionamiento corresponde al de una ventana deslizante de igual tamaño en el emisor que en el receptor. Como a la capa de red se le requiere que transfiera los paquetes en orden, para que esta estrategia funcione correctamente el receptor deberá mantener en su buffer todos los frames posteriores hasta que reciba correctamente el frame en cuestión (esto supone tener un buffer lo suficientemente grande para almacenar un número de frames igual al tamaño de ventana que se esté utilizando).

La posibilidad de una recepción no secuencial de frames plantea nuevos problemas. Por ejemplo, suponiendo que el emisor envía los frames 0 a 6, los que son recibidos correctamente. Entonces el receptor realiza las siguientes acciones: los transmite a la capa de red, libera los buffers correspondientes avanza la ventana para poder recibir siete frames más, cuyos números de secuencia podrán ser 7,0,1,2,3,4,5 y envía un ack para los frames 0 a 6 recibidas. Si el ack no llega al emisor, éste supondrá que ninguno de ellos ha llegado, por lo que reenviará los frames

0 a 6 de nuevo. De éstos, los frames 0 a 5 se encuentran dentro de la ventana del receptor. En un procesamiento secuencial, el receptor no aceptaría estos frames si no recibe antes el frame 7 pendiente, pero con retransmisión selectiva se aceptarían y se pediría retransmisión del 7. Una vez recibido, éste se pasaría a la capa de red seguido de los frames 0 a 5 antes recibidos, que serían duplicados de los anteriores. En este caso el receptor pasará frames duplicadas al nivel de red.

La solución a este conflicto está en evitar que un mismo número de secuencia pueda aparecer en dos ventanas consecutivas. Por ejemplo, si el tamaño de ventana es de 7 el número de secuencia podría ser de 4 bits y la ventana del receptor sería 0-6, 7-13, 14-4, etc. El valor máximo de la ventana para un protocolo de repetición selectiva en el caso general sería (MAX_SEQ+1)/2.

Aunque el número de secuencia se ha duplicado respecto al caso anterior, el número de frames que hay que mantener en el buffer no necesita ser superior al tamaño de ventana, ya que este será el número máximo de frames que habrá que manejar en cualquier circunstancia. La técnica de repetición selectiva da lugar a protocolos más complejos que la de retroceso n, y requiere mayor espacio de buffers en el receptor. Sin embargo, cuando las líneas de transmisión tienen una tasa de errores elevada da un mejor rendimiento, ya que permite aprovechar todos los frames correctamente transmitidos.

2.5.2. Protocolo Punto a Punto

PPP fue desarrollado por el IETF en 1990 y está especificado en los RFC 1661, 1662 y 1663. PPP fue diseñado para ser flexible, por ello incluye un protocolo especial, denominado LCP (Link Control Protocol), que se ocupa de negociar una serie de parámetros en el momento de establecer la conexión con el sistema remoto.

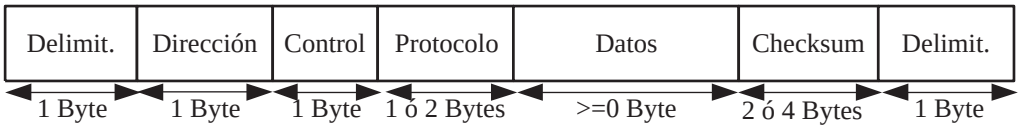

Figura 11: Estructura de un Frame PPP.

La estructura de un frame PPP (ver figura 11) se basa en el de HDLC, pero a diferencia de éste, PPP es un protocolo orientado a carácter, lo que implica que la longitud del frame ha de ser un número entero de bytes. En función de las características del medio físico se aplica relleno de bits o relleno de bytes (por ejemplo para transmisión por medios asíncronos). La descripción de cada uno de los campos del frames es la siguiente:

Delimitador el frame tiene siempre la secuencia 01111110 como delimitador.

Dirección este campo no se utiliza y siempre vale 11111111.

Control tiene por defecto el valor 00000011, que corresponde a un servicio no confiable y no orientado a conexión. De todas formas, en el momento de establecer la conexión LCP puede negociar una transmisión fiable.

Dirección por defecto, corresponde a la secuencia 1111111100000011, a menos que se negocie una transmisión confiable. Para no transmitir estos dos bytes de información inútil en todos los frames, generalmente LCP negocia la supresión de estos dos bytes al inicio de la sesión (salvo que se pida transmisión fiable).

Protocolo establece a que tipo de protocolo pertenece el paquete recibido de la capa de red. Así, PPP permite establecer una comunicación multiprotocolo, es decir, puede utilizarse para transmitir paquetes pertenecientes a diferentes protocolos del nivel de red. Entre las posibilidades se encuentra IP, IPX, Appletalk, DECNET, OSI y otros.

Datos es de una longitud variable hasta un máximo que negocia LCP al establecer la conexión. Por defecto el tamaño máximo del frame es de 1500 bytes.

Checksum es normalmente de 2 bytes, pero puede ser de 4 si se negocia.

PPP puede utilizarse sobre medios físicos muy diversos, por ejemplo, conexiones mediante módem, ISDN, líneas dedicadas, o incluso por conexiones SONET/SDH de alta velocidad.

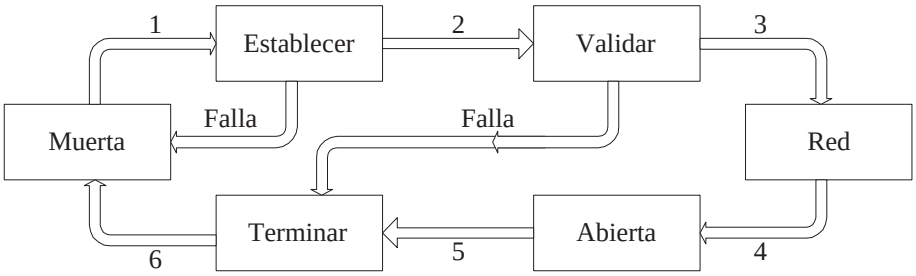

Figura 12: Fases de la Negociación PPP.

La negociación entre dos LCPs puede dar lugar a que todos o sólo algunos de los valores propuestos sean aceptados. El protocolo establece mecanismos que permiten a los LCPs dialogar para llegar a un consenso en caso de discrepancia. Existe otro componente de PPP que es el NCP (Network Control Protocol) que se encarga de negociar los parámetros específicos para cada protocolo utilizado. Por ejemplo, en el caso de una conexión IP desde un usuario conectado vía módem le asigna dinámicamente una dirección IP, lo que es útil cuando el número de direcciones IP disponibles es menor que el número de usuarios del servicio. Además, LCP permite utilizar diversos protocolos de autentificación, como CHAP, PAP, Kerberos, etc. Las fases de una conexión PPP (ver figura 12) son las siguientes:

1. Cuando se detecta la portadora es porque se ha realizado una conexión a nivel de capa física y la conexión está en la fase *Establecer*. Hasta entonces la línea estaba en reposo o *Muerta*, ya que no existía conexión.

2. Se negocian las opciones LPC y si se llega a un acuerdo se pasa a la fase de *Validar* que consiste en la verificación de identidad del usuario.

3. Al entrar en la fase de *Red* se invoca al protocolo NCP apropiado para configurar la capa de red.

4. Una vez configurada se pasa a la fase *Abierta*, y comienza el transporte de datos.

5. Finalmente la conexión pasa a fase de *Terminar* cuando ya no existen más datos para transmitir y se desea liberar la conexión.

6. Una vez finalizada la conexión se pasa a la etapa de reposo o *Muerta*.

2.5.3. El Problema de la Asignación del Canal

De acuerdo a la clasificación tecnológica de las redes éstas pueden ser redes broadcast o redes formadas por enlaces punto a punto. En el caso de las punto a punto, la información se envía a la máquina situada al otro lado del enlace, que está claramente identificada y el medio de transmisión normalmente está siempre disponible. Los protocolos de nivel de enlace vistos hasta ahora tienen estas suposiciones.

En las redes broadcast existe una nueva dificultad. Como el canal es compartido, es necesario habilitar mecanismos que permitan a cada host utilizar el medio para enviar frames a la máquina destino. El hecho de compartir el canal generará conflictos o incluso pérdida de frames en algunos casos, por lo que los protocolos deberán establecer mecanismos adecuados para resolver estos conflictos y permitir que los nodos retransmitan los frames que no hayan podido ser enviadas correctamente.

Debido a esta mayor complejidad es que en las redes broadcast se suele dividir la capa de enlace dos subcapas: una inferior, que se ocupa de controlar la función de acceso al medio de transmisión, denominada *subcapa MAC* y la superior, llamada *subcapa de control de enlace lógico o LLC* que corresponde a las funciones de la capa de enlace comunes a todo tipo de redes.

Existen dos soluciones principales para resolver la problemática de cómo asignar el canal o medio físico compartido a los nodos de la red, una es la *asignación estática del canal* y la otra es la *asignación dinámica*.

Las formas de asignación estática más utilizadas son *Multiplexación por División de Frecuencias o FDM* y *Multiplexación por División de Tiempo o TDM*. La idea es repartir el canal entre las estaciones que lo requieren, y una vez asignado un ancho de banda o un slot de tiempo a una estación ésta lo tendrá reservado para su uso mientras no lo libere, independientemente de que lo use o no. Estos métodos no son aplicables a las redes broadcast debido a que la naturaleza del tráfico es impulsivo o de ráfagas, lo que hace que que se pierdan las frecuencias o slots asignados cuando un host no desea transmitir, haciendo el rendimiento muy pobre cuando el número de estaciones es mucho menor que el número de asignaciones de frecuencia o tiempo. Por otro lado, si el número de hosts es mayor que el de asignaciones habrán estaciones que no podrán transmitir ni recibir datos.

La asignación dinámica del canal consiste en reservar el medio para transmitir sólo cuando sea necesario. Como el medio de comunicación es único las estrategias de competencia para acceder a él son variadas y asumen la ocurrencia de colisiones (que corresponde a la transmisión simultánea de dos frames, los que se traslapan y producen una señal distorsionada en el medio), sin embargo, existen esquemas que evitan las colisiones. Otras estrategias utilizan el tiempo continuo para transmitir, es decir, transmiten cuando lo necesitan, y otras usan slots de tiempo o transmiten por intervalos. Existen métodos que introducen mejoras que permiten al adaptador de red detectar colisión y esperar un tiempo para retransmitir el frame. Los que no detectan colisión deben esperar la respuesta de si la transmisión fue exitosa o no para reenviar el frame si es necesario.

En resumen, las reglas que especifican como se resuelve la situación de accesar el medio compartido se llaman *protocolos de acceso al medio*. La asignación estática tiene la desventaja de que se puede subutilizar el canal cuando las transmisiones son impulsivas o de ráfagas de datos. La asignación dinámica se realiza sólo cuando sea necesario transmitir, pero pueden ocurrir colisiones producto de la contienda por el canal y se necesitan definir estrategias para llegar a un acuerdo entre los nodos en cuanto a la utilización del canal.

2.5.4. Protocolos de Acceso Múltiple Sin Detección de Portadora

ALOHA

Cuando un host desea transmitir, simplemente emite un frame, sin preocuparse en ningún momento del estado del canal. Una vez finalizado, queda en espera de recibir la confirmación de que la información ha sido recibida correctamente por el destinatario, quien verifica esto usando el campo CRC del frame. Si pasado un tiempo no se recibe confirmación, el emisor supone que ha ocurrido una colisión y espera un tiempo aleatorio y a continuación reenvía el frame. El problema principal del protocolo es que el envío de frames por parte de los nodos se hace en forma caótica y basta que dos frames colisionen o se solapen solamente en un bit para que ambos sean inútiles y deban retransmitirse, puesto que los nodos sólo se percatarán del problema después de haber terminado la transmisión. Por otro lado, el segundo frame podría colisionar con un tercero, y así sucesivamente, de ahí que en una red ALOHA cuando el tráfico crece las colisiones aumentan de manera no lineal y el rendimiento decae rápidamente. El rendimiento máximo de ALOHA es de 18,4 %, que se consigue con una utilización del canal del 50 %.

ALOHA Ranurado

Es una mejora a ALOHA que consiste en dividir el tiempo para la emisión de frames en intervalos de duración constante del tamaño de un frame. Además, los nodos deben sincronizarse para saber cuando empieza cada intervalo. Esto reduce la probabilidad de colisión, ya que limita el efecto de colisión a un intervalo concreto, y no se pueden encadenar colisiones. En ALOHA ranurado, la eficiencia máxima es de 36,8 % y se consigue con una utilización del 100 % del canal.

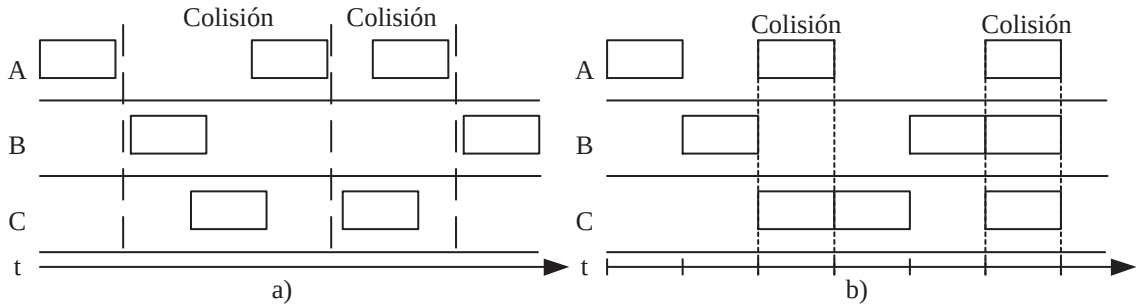

Figura 13: Tiempos Involucrados en las Colisiones a) ALOHA b) ALOHA Ranurado.

2.5.5. Protocolos de Acceso Múltiple Con Detección de Portadora

Estos protocolos antes de transmitir observan si alguien ya está transmitiendo, lo que permite hacer un uso más eficiente del canal ya que no se interrumpe la transmisión que está en proceso. El nombre genérico de estos protocolos es *de acceso múltiple con detección de portadora o CSMA*.

CSMA 1-persistente

El protocolo CSMA 1-persistente funciona de la siguiente forma: cuando tiene que transmitir un frame, primero escucha el canal y si está libre envía el frame, caso contrario, espera a que se libere y en ese momento lo envía. Se denomina CSMA 1-persistente porque existe la probabilidad 1, es decir, certeza de que el frame se transmitirá cuando el canal esté libre.

En una situación real con alto tráfico es muy posible que cuando un nodo termine de transmitir existan varios esperando enviar sus datos, y con CSMA 1-persistente todas los frames serán emitidos a la vez y colisionarán, pudiéndose repetir el proceso varias veces con la consiguiente degradación del rendimiento. Cabe señalar que una colisión ocurrirá aunque no empiecen a transmitir exactamente a la vez, basta simplemente con que dos nodos empiecen a transmitir con una diferencia de tiempos menor que la distancia que los separa, ya que en tal caso ambos detectarán el canal libre en el momento de iniciar la transmisión. Se deduce entonces, que en este tipo de redes el retardo de propagación de la señal puede tener un efecto importante en el rendimiento. El rendimiento obtenido con este protocolo puede llegar al 55 % con un grado de ocupación del 100 %.

CSMA no persistente

Corresponde a una modificación del protocolo anterior que funciona de la siguiente manera: antes de enviar se escucha el canal, si el canal está libre se transmite el frame. Si está ocupado, en vez de quedar escuchando, se espera un tiempo aleatorio después del cual se repite el proceso. El protocolo tiene una menor eficiencia que CSMA 1-persistente para tráficos moderados, pues introduce una mayor latencia; sin embargo se comporta mejor en situaciones de tráfico intenso ya que evita las colisiones producidas por las estaciones que se encuentran a la espera

de que termine la transmisión de un frame en un momento dado.

CSMA p-persistente

Este protocolo utiliza intervalos de tiempo y funciona de la siguiente manera: cuando el nodo tiene algo que enviar primero escucha el canal, si está ocupado espera un tiempo aleatorio. Cuando el canal está libre se selecciona un número aleatorio con distribución uniforme entre 0 y 1, si el número seleccionado es menor que p el frame es transmitido, si se espera el siguiente slot de tiempo para transmitir y repite el algoritmo hasta que el frame es transmitido o bien otro nodo utiliza el canal, en cuyo caso se espera un tiempo aleatorio y empieza de nuevo el proceso desde el principio. La eficiencia del protocolo es, en general, superior a la de CSMA 1-persistente y CSMA no persistente.

CSMA con Detección de Colisión

Un problema con los protocolos anteriores es que una vez se ha empezado a transmitir un frame se sigue transmitiendo aún cuando se detecte una colisión. Como es más eficiente parar de transmitir y esperar un tiempo aleatorio para volver a hacerlo, ya que el frame será erróneo de todas formas, los protocolos de acceso múltiple detección de portadora con detección de colisiones o CSMA/CD implementan esta mejora.

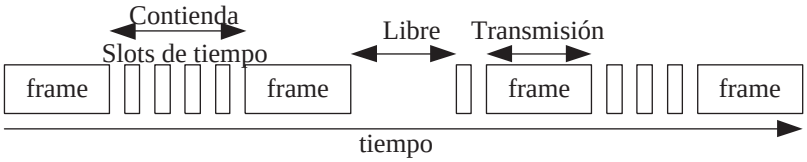

Figura 14: Estados de una Red CSMA/CD.

En una red CSMA/CD la única circunstancia en la que puede producirse una colisión es cuando dos hosts empiezan a transmitir a la vez, o con una diferencia de tiempo lo bastante pequeña como para que la señal de uno no haya podido llegar al otro antes de que éste empiece a transmitir. Suponiendo que se tienen los nodos de los extremos de la red, llamados A y B y que la señal tarda un tiempo t en propagarse de uno a otro extremo a otro, cabría pensar que si A empieza a transmitir pasado ese tiempo t puede estar seguro de que no detectará colisiones, ya que su señal ha llegado al otro extremo de la red. En el peor caso, B podría haber empezado a transmitir justo en el instante t-e, es decir, inmediatamente antes de que le haya llegado el frame de A, por lo que sólo después de un tiempo $2t$, A puede estar seguro de no colisionar con ninguna otra estación. A este período de tiempo se le llama *período de contienda* y corresponde a uno de los tres posibles estados que tiene una red CSMA/CD, los otros estados son los de transmisión y el estado libre (figura 14).

Protocolos Libre de Colisiones El problema de las colisiones es que en la práctica producen una disminución del rendimiento debido a que las transmisiones que se producen du-

rante la colisión son inútiles, el problema se acrecenta al aumentar el tráfico en la red. En cambio, los protocolos que por su funcionamiento no tienen colisiones, suelen tener una mayor eficiencia cuando la carga de la red es elevada.

Protocolo Bit-Map

Suponiendo que la red tiene N nodos, numerados de *0* a *N-1*. Para empezar a funcionar, se establece una ronda exploratoria de N intervalos en la que por turno cada host, empezando por el *0*, tiene la posibilidad de enviar un bit con el valor 1 ó 0 para indicar si tiene algún frame que transmitir. Pasados los N intervalos todos los hosts han manifestado su situación y todos saben quien tiene datos para transmitir. Luego, ordenadamente cada host que tenía datos para transmitir lo hace, una vez finalizado esto se vuelve ha establecer una nueva ronda exploratoria. Si a algún host le surge la necesidad de transmitir justo después de haber dejado pasar su turno, tendrá que esperar a la siguiente vuelta.

Considerando el rendimiento, este protocolo genera un frame adicional de N bits. Si la red no tiene tráfico, se genera un frame bitmap que está continuamente dando vueltas por la red. Si la carga en la red es baja (un frame transmitido por vuelta) la eficiencia es $\frac{d}{N+d}$, con d el tamaño del frame de información transmitida y N el número de nodos. Si la red está saturada existirá un frame por host que enviar y la eficiencia será $\frac{d}{d+1}$. Esto muestra que el rendimiento de este protocolo aumenta a medida que lo hace el tráfico en la red.

Un problema con este protocolo es que la calidad de servicio que ofrece a cada nodo no es la misma. En situaciones de poco tráfico el protocolo bitmap no da un trato equitativo, sino que favorece a los nodos con dirección elevada. En cambio en situaciones de saturación este efecto desaparece, ya que si todos los hosts tienen frames que enviar cada uno podrá transmitir una frame a la vez. En resumen, el protocolo bitmap es más eficiente y más homogéneo en su comportamiento a medida que la carga de la red aumenta.

Protocolo de Cuenta Regresiva Binaria

El usar un bit para reservar implica una sobrecarga muy grande si en número de nodos es alto. En cambio, el protocolo de cuenta regresiva binaria usa direcciones binarias de igual largo. El protocolo funciona de la siguiente forma: los nodos que desean transmitir envían a la red el bit más significativo de su dirección, el medio de transmisión está diseñado de tal forma que retransmite el OR de todos los bits transmitidos. Con este resultado los nodos que desean transmitir y que tengan el bit menor al obtenido en el medio se retiran de la competencia. Los nodos restantes envían el siguiente bit de dirección hasta que quede sólo un nodo (el de mayor dirección) que será el que transmita. El proceso se repite después con los nodos que aún no han transmitido. La eficiencia de este protocolo supera al bitmap para tráficos reducidos.

2.5.6. Protocolos de Contención Limitada

Los protocolos con colisiones son ideales cuando los niveles de tráfico son bajos, ya que tienen retardos pequeños y no introducen overhead. En cambio, cuando el tráfico aumenta, es preferible perder una parte de la capacidad del canal en habilitar mecanismos que habiliten

turnos de transmisión, ya que de lo contrario no es posible utilizar el canal al máximo de sus posibilidades.

Cuando la red tiene poco tráfico, los protocolos de contención limitada se comportarán según alguno de los protocolos con colisiones ya vistos. Cuando se superan determinados niveles de utilización, el protocolo dividirá el canal en intervalos asignando uno a cada host.

En la práctica suelen ser unos pocos nodos los que generan la mayor parte del tráfico, por lo que lo ideal es identificarlos y aislarlos en intervalos propios, independientes del resto de los hosts. De esta forma, esos nodos con tráfico elevado consiguen un buen rendimiento sin perjudicar a la mayoría que es la que no transmite tanto. La rápida identificación de nodos con alto tráfico es la clave del funcionamiento de estos protocolos. Los hosts no necesariamente han de ser identificados individualmente, es suficiente detectar un grupo con tráfico elevado y aislarlo del resto para que el protocolo funcione de buena forma.

2.5.7. Protocolos de Redes Inalámbricas

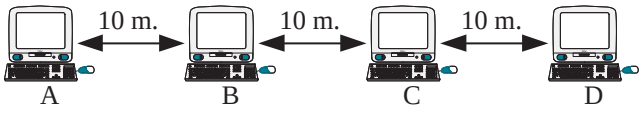

Figura 15: Red LAN Inalámbrica.

Actualmente han aparecido redes locales basadas en ondas de radio e infrarrojos. Los últimos, por sus características tienen un alcance reducido y requieren línea de vista entre emisor y receptor. Los de radio solo pueden transmitir a muy baja potencia, por restricciones legales, por lo que su alcance es también reducido, aunque no tanto como los infrarrojos.

Típicamente una LAN inalámbrica está formada por un conjunto de estaciones base, unidas entre sí por algún tipo de cable, y una serie de estaciones móviles que comunican con la estación base más próxima. El conjunto de estaciones base forma en realidad un sistema celular en miniatura.

Suponiendo lo siguiente: existen cuatro nodos A, B, C y D situados en línea y separados 10 metros (figura 15), el alcance máximo de cada uno de ellos es de 12 metros, se implementa un protocolo CSMA para la comunicación. La secuencia de sucesos para transmitir un frame podrían ser la siguiente: A desea transmitir datos a B, al detectar el medio lo encuentra libre y comienza la transmisión. A está transmitiendo a B y C también desea transmitir datos hacia B, detecta el medio y lo encuentra libre (C no escucha a A pues esta a 20 m de distancia), por lo tanto, C empieza a transmitir. El resultado es una colisión en el receptor B que no es detectada ni por A ni por C. Esto se conoce como el problema de la estación oculta.

Si ahora, con la misma distribución de nodos, ocurre lo siguiente: B desea transmitir datos hacia A, detecta el medio libre e inicia la transmisión. A continuación, C desea transmitir datos hacia D, y como detecta que B está transmitiendo espera a que termine para evitar

una colisión. El resultado es que una transmisión que en principio podría haberse hecho sin interferencias (ya que A no puede escuchar a C y D no puede escuchar a B) no se lleva a cabo, reduciendo así la eficiencia del sistema. Esto se conoce como el problema de la estación expuesta.

MACA

Multiple Access with Collision Avoidance o MACA es un protocolo MAC resuelve los dos problemas antes mencionados de la siguiente forma: cuando una estación tiene un frame que transmitir, antes de enviarlo envía un frame pequeño de aviso denominada RTS (Request To Send). El frame contiene información sobre la longitud del frame que se pretende transmitir y el nodo de destino. El nodo destino, al recibir el frame RTS, y si está en condiciones de recibir la transmisión, responde con otro frame denominado CTS (Clear To Send). El CTS también indica la longitud del frame que se va a recibir. Esto ocurre en el funcionamiento normal del sistema.

En el caso de la estación oculta ocurre lo siguiente: A transmite un RTS a B indicando la longitud del frame que desea enviar. B responde con un CTS que también especifica la longitud del frame a recibir. En este momento C capta la respuesta de B, por lo que se percata de que va a tener lugar una transmisión en la que B actuará de receptor y sabe que deberá permanecer en silencio durante el tiempo que dure la transmisión (C sabe lo que durará pues conoce la longitud del frame y la velocidad de la red). Con esto, A envía a B los datos correspondientes.

En el caso de la estación expuesta ocurre lo siguiente: B transmite a A un RTS indicando que quiere enviarle datos. En ese momento C se entera de las intenciones de B. A devuelve a B un CTS. Mientras tanto, C, que ha captado el RTS pero no el correspondiente CTS, comprende que aunque detecta que B está transmitiendo el destinatario está fuera de su alcance, por lo que puede comunicar con D cuando quiera, sin esperar a que B termine.

2.5.8. Protocolos Token Passing

Estos protocolos se pueden considerar como un conjunto de líneas punto a punto simplex que interconectan nodos en un anillo, que puede ser lógico y/o físico. Los frames se transmiten en un determinado sentido dentro del anillo y dan la vuelta completa, lo que para efectos prácticos implica que la red funciona como un medio broadcast.

Cada estación de la red puede funcionar en uno de los dos modos siguientes:

Modo escucha: cada frame que se recibe del nodo anterior se transmite al siguiente.

Modo transmisión: el nodo emite un frame hacia el siguiente nodo, y paralelamente, recibe y procesa los bits que le llegan del nodo anterior en el anillo.

En un determinado momento, sólo un nodo de la red puede estar en modo transmisión, y los demás deben estar a la escucha. Si no hay tráfico en la red todos los nodos están escuchando.

Un protocolo token passing funciona de la siguiente manera:

- Cuando ningún host desea transmitir, todos están en modo escucha y se envía por el anillo un frame especial denominado *token*. El token va pasando de un host a otro indefinidamente.
- Cuando algún nodo desea transmitir debe esperar a que pase por él el token. En ese momento, se apodera de éste, típicamente convirtiendo el token en el delimitador de inicio del frame. A partir de ese momento, el nodo pasa a modo transmisión y envía el frame al siguiente nodo.
- Todos los demás hosts del anillo, incluido el destino, siguen en modo escucha, retransmitiendo el frame recibido hacia el siguiente nodo. El host destino, además de retransmitirlo, retiene una copia del frame que pasará al nivel de red para su proceso.
- Al finalizar la vuelta, el emisor empieza a recibir su propio frame. Éste puede optar por descartarlo o compararlo con el frame enviado para verificar si la transmisión ha sido correcta.
- Cuando el nodo ha terminado de transmitir el último bit del frame pueden ocurrir dos cosas: que restaure el token en el anillo inmediatamente, o que espere hasta recibir, de la estación anterior, su frame, y sólo entonces restaure el token. El primer modo de funcionamiento se conoce *Early Token Release*.

Si el emisor tiene varios frames listos para emitir puede enviarlos sin liberar el token, hasta consumir el tiempo máximo permitido, denominado *token-holding time*. Una vez agotados los frames que hubiera en el buffer, o el tiempo permitido el nodo restaura el token en el anillo. Bajo ninguna circunstancia un host debe estar en modo transmisión durante un tiempo superior al token-holding time.

Este protocolo genera problemas nuevos: ¿qué pasa si se pierde un frame? ¿ qué pasa si el nodo encargado de regenerar el token falla?. En toda red token passing existe una estación monitora que se ocupa de resolver estas situaciones y garantizar el normal funcionamiento del protocolo. En caso de problemas restaurará un token en el anillo para que el tráfico pueda seguir circulando normalmente. Cualquier estación de una red token passing está capacitada para actuar como monitor en caso necesario.

Cuando un nodo se añade a la red queda a la escucha en busca de tokens o datos. Si no detecta actividad, emite un frame de control especial denominado *claim token*. Si existe ya un monitor éste responderá con un token a la petición. Si no, el recién incorporado recibirá su propio claim token, momento en el cual pasará a constituirse en monitor.

Existe también un mecanismo de prioridades, el que funciona de la siguiente manera: existen bits en el frame que permiten establecer la prioridad de un nodo, por lo que nodos de mayor prioridad podrán tomar el control del token aunque algún host, pero de menor prioridad, esté transmitiendo. Una vez finalizada la transferencia, se debe devolver la prioridad que tenía al token.

2.6. Estandarización de Redes LAN

La mayoría de las LANs han sido estandarizadas por el IEEE, en el comité denominado 802. Los estándares desarrollados por este comité están enfocados a las capas 1 y 2 del modelo OSI. Este comité se divide en subcomités, cuyo nombre oficial es Grupos de Trabajo, que se identifican por un número decimal (ver tabla 1).

Los grupos de trabajo 802 continuamente están planteando nuevas técnicas y protocolos para su estandarización, nuevos medios físicos, etc. Al surgir una propuesta, el grupo correspondiente nombra un grupo de estudio que la analiza, y si el informe es favorable se crea un subgrupo que eventualmente propone un adendum al estándar para su aprobación. Los proyectos se identifican por letras añadidas al grupo de trabajo del que provienen. Por ejemplo:

- 802.1D: puentes transparentes
- 802.1G: puentes remotos
- 802.1p: Filtrado por clase de tráfico (Calidad de Servicio)
- 802.1Q: Redes locales virtuales (VLANs)
- 802.3u: Fast Ethernet
- 802.3x. Ethernet Full dúplex y control de flujo
- 802.3z: Gigabit Ethernet
- 802.3ab: Gigabit Ethernet en cable UTP-5

2.7. Tecnologías Ethernet

Ethernet se refiere a la familia de implementaciones LAN que usan CSMA/CD como protocolo MAC, y se incluyen tres categorías principales:

Ethernet Original: que es el sistema más utilizado actualmente, transmite frames a 10 Mbps y está especificado por el estándar IEEE 802.3 y el Ethernet.

Fast Ethernet: es un sistema con un ancho de banda de 100 Mbps. Uno de los aspectos importantes de Fast Ethernet, es que conserva el formato del frame Ethernet y la cantidad de datos que pueden ser transmitidos en él, lo que lo hace ser compatible con la versión anterior.

Gigabit Ethernet: que es una extensión más del estándar de Ethernet. Este sistema ofrece un ancho de banda de 1000 Mbps, manteniendo absoluta compatibilidad con los nodos Ethernet ya existentes.

Cuadro 1: Grupos de Trabajo del Comité 802 de IEEE.

802.1	Aspectos comunes: puentes, gestión, redes locales virtuales, etc.
802.2	Logical Link Control (LLC). En hibernación e inactivo
802.3	Redes CSMA/CD (Ethernet)
802.4	Redes Token-Passing Bus. En hibernación e inactivo
802.5	Redes Token Ring
802.6	Redes MAN DQDB (Distributed Queue Dual Bus). En hibernación e inactivo
802.7	Grupo asesor en redes de banda ancha. En hibernación e inactivo.
802.8	Grupo asesor en tecnologías de fibra óptica
802.9	Redes de servicios Integrados (Iso-Ethernet). En hibernación e inactivo
802.10	Seguridad en estándares IEEE 802. En hibernación e inactivo.
802.11	WLAN (Wireless LANs)
802.12	Redes Demand Priority (100VG-AnyLAN). En hibernación e inactivo
802.14	Redes de TV por cable, pendiente de ratificación. Disuelto.
802.15	WPAN (Wireless Personal Area Network)
802.16	BWA (Broadband Wireless Access)

2.7.1. Especificación IEEE 802.3 y Ethernet

Medio Físico

La tabla 2 muestra los tipos de medios físicos posibles de utilizar en la especificación. Actualmente casi todo el cable de cobre utilizado en redes Ethernet es el de UTP categorías 3 y 5 preferentemente. Rara vez se emplea STP o cable coaxial.

Cuadro 2: Medios Físicos más Utilizados Especificados en IEEE 802.3.

	10Base5	10Base2	10Base-T	10Base-FL
Cable	Coaxial grueso	Coaxial delgado	UTP Cat 3/5	Fibra 62,5/125 micras
Pares	1	1	2/2	2
Full dúp	No	No	Sí/Sí	Sí
Tipo Conector	N	BNC	RJ-45/RJ-45	ST
Topología	Bus	Bus	Estrella/Estrella	Estrella
Dist. Seg.	500, máx 2500 m	185, máx 925 m	100, máx 500 m 150, máx 750 m	2 km
Nº Nodos/seg.	100	30	1024/1024	1024

En Ethernet, como en todas las redes locales, la transmisión es realiza de asincrónica. Por esto, se utiliza un sincronismo implícito en los datos mediante el uso de códigos que incorporan cierto nivel de redundancia. Ethernet usa el código Manchester, que utiliza dos

voltajes e identifica el bit 0 como una transición alto-bajo y el 1 como una transición bajo-alto. El código Manchester es poco eficiente, pero resulta sencillo y barato de implementar. Su mayor inconveniente resulta ser la elevada frecuencia de la señal, lo que complicó bastante las cosas cuando se adaptó Ethernet para UTP.

Como medida de confiabilidad del medio físico, el estándar 802.3 establecía inicialmente una BER máxima de 10^{-8}, pero las nuevas especificaciones de medios físicos han ido fijado requerimientos superiores. Una buena instalación Ethernet actual en un entorno de oficina puede dar sin problemas una BER inferior a 10^{-12}. Transmitiendo a 10 Mbps ininterrumpidamente esto representa menos de un error por día, lo que implica que los errores de CRC en una red Ethernet funcionando correctamente deberían ser casi nulos, salvo los originados por la conexión y desconexión de equipos. Debido a la elevada confiabilidad del medio físico, el protocolo MAC de Ethernet no realiza ningún tipo de verificación, ya que la probabilidad de que un frame no llegue a su destino es tan baja que esto sería perjudicial para el rendimiento de la red.

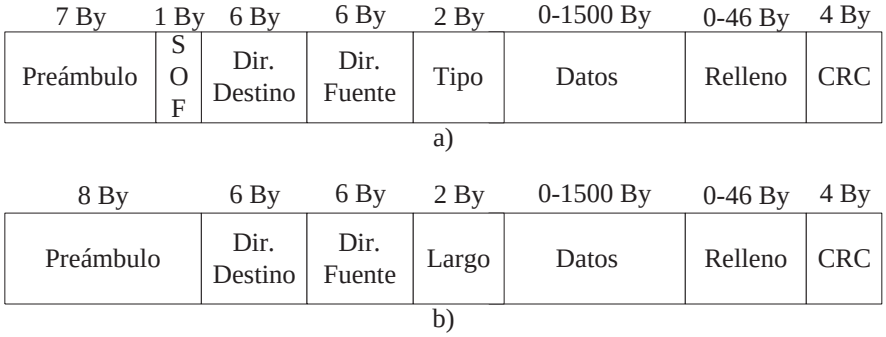

Figura 16: Formato de un frame a) Ethernet b) IEEE 802.3.

Subcapa MAC

Ethernet e IEEE 802.3 utilizan el protocolo CSMA/CD, y el formato de un frame de datos es el mostrado en la figura 16. El detalle de los campos es el siguiente:

Preámbulo: es un delimitador consistente en la secuencia 10101010 repetida 7 u 8 veces según corresponda. El frame Ethernet incluye un byte adicional que es equivalente al campo SOF del frame IEEE 802.3.

SOF: (sólo en IEEE) byte delimitador de IEEE 802.3 terminado con dos bits en 1 consecutivos (10101011), que sirven para sincronizar las porciones de recepción de frames de los nodos.

Direcciones origen y destino: las direcciones corresponden a los identificadores de los nodos de una LAN. Los primeros 3 bytes son asignados por el IEEE a cada fabricante,

y los tres últimos son especificados por el fabricante. La dirección de origen siempre es unicast, la destino puede ser uni, multi o broadcast.

Longitud: (sólo en IEEE) especifica el número de bytes de datos que siguen a continuación.

Tipo: (sólo en Ethernet) identifica el protocolo de la capa superior que recibirá los datos una vez recibido el frame.

Datos: corresponde a los datos provenientes de las capas superiores. Un frame debe tener un tamaño mínimo de 64 bytes (a partir del campo dirección destino), por lo tanto, se utiliza un relleno si es que la situación lo requiere.

CRC: es una suma de verificación para asegurar que el frame llegó en buen estado.

Ethernet asigna direcciones globalmente únicas a cada dispositivo Ethernet. Para ello utiliza una dirección de seis bytes, de los cuales los tres primeros correspondían al fabricante y los tres últimos al dispositivo. El estándar permite direcciones de 2 ó 6 bytes, aunque en la práctica sólo se utilizan las de 6 bytes. Además de utilizarse en otras redes 802 las direcciones MAC IEEE se emplean en redes locales no IEEE, como FDDI y Fibre Channel.

Los dos primeros bits de los 48 que componen una dirección MAC IEEE tienen un significado especial:

- El primer bit indica el ámbito del envío. Se contemplan tres posibilidades: envío unicast, multicast o broadcast. Si el primer bit está a 0 se trata de un envío unicast, si está en 1 es un envío multicast o broadcast. En caso de que toda la dirección esté a 1 será un envío broadcast, que deberá ser atendido por todos los nodos. Si es un frame multicast, tendrá en 1 el primer bit, viniendo especificado por el resto de la dirección el grupo multicast al que va dirigido. En una frame unicast el primer bit será 0, en cuyo caso el frame sólo deberá ser interpretado por el nodo al que va dirigido.

- El segundo bit se utiliza para indicar si se trata de una dirección global (grabada por el fabricante en el hardware de la tarjeta) o si se trata de una dirección local, asignada por software a ese equipo. Las direcciones locales sólo pueden ser utilizadas dentro de la red, ya que en otras redes podrían estar duplicadas. En cambio las globales, dado que son únicas en todo el mundo, podrían utilizarse para enviar frames a cualquier nodo existente.

Para que CSMA/CD funcione bien, es decir, detecte las colisiones, se requiere que el tiempo de ida y vuelta entre dos estaciones cualquiera no supere el tiempo de transmisión mínimo, que corresponde al tiempo que tarda en emitirse el frame de tamaño mínimo permitido. El tiempo de transmisión mínimo depende exclusivamente de la velocidad de operación de la red, y el tiempo máximo de ida y vuelta o *round trip time* fija las distancias máximas entre los nodos. Estos cuatro parámetros: velocidad de la red, tamaño de frame mínimo, round trip time y distancia máxima están relacionados entre sí.

Se producirá una colisión cuando dos o más nodos empiecen a transmitir simultáneamente, o con una separación de tiempo menor que el tiempo de propagación que las separa. En

Ethernet se producirá una colisión siempre que dos nodos transmitan con una separación en el tiempo menor de 25.6 μseg., que corresponde a la mitad del tiempo de transmisión del frame mínimo permitido. Si la separación es mayor que 25.6 μseg. no se producirá colisión ya que el segundo nodo detectará el medio ocupado cuando vaya a transmitir. En ese caso, esperará a que el primero termine y transmitirá a continuación, respetando el tiempo entre frames que debe existir, y que para Ethernet es de 9.6 μseg. A pesar de que transcurridos los 25.6 μseg. ya no puede ocurrir colisión, para el emisor no existe garantía de no colisión sino sólo hasta pasados 2*25.6=51.2 μseg., ya que si otra estación empieza a transmitir justo antes de que el frame alcance el otro extremo de la red se producirá una colisión en el lugar más distante, de la que el emisor se informará solo cuando vuelva el frame, tendrán que haber transcurrido otros en total los 51.2 μseg.

En caso de producirse una colisión, los nodos Ethernet utilizan el algoritmo de *de retroceso exponencial binario* para iniciar la retransmisión. Al detectar la colisión, los involucrados dejan de transmitir y a partir de ese momento dividen el tiempo en intervalos de 51.2 μseg. y esperan 0 ó 1 intervalos para reintentar. La elección entre 0 y 1 la hace cada uno independientemente de forma aleatoria, por lo que la probabilidad de colisión es de 0.5. Si se produce una segunda colisión, cada nodo espera aleatoriamente 0, 1, 2 ó 3 intervalos para reintentar, bajando la probabilidad de colisión a 0.25. Si siguen colisionando el número de intervalos se duplica en cada intento sucesivo, con lo que la probabilidad de colisión decrece exponencialmente, hasta que los nodos eligen intervalos distintos. Así, quien eligió el intervalo más bajo transmite primero y así sucesivamente. El algoritmo tiene un tope máximo de intentos, al final del cual produce un error de transmisión.

El Rendimiento de Ethernet

Probablemente el factor que más influye en el rendimiento de Ethernet es el *tamaño del frame* utilizado. Debido a que una colisión sólo puede suceder durante los primeros 512 bits del frame, se puede decir que cuando ésta tiene 512 bits de longitud el riesgo de colisión es permanente, mientras que si tiene el máximo, es decir, 1518 bytes la colisión sólo puede producirse durante los primeros 512 bits, es decir el 4.2 % del tiempo. Por lo tanto, dado un nivel de ocupación constante, el número de colisiones se reduce, y el rendimiento aumenta, si aumenta el tamaño de los frames.

Otro factor que influye en la eficiencia, es el *número de estaciones emisoras*. Esto se puede comprender fácilmente dándose cuenta de que la probabilidad de generar colisiones aumenta en la medida que aumenta el número de posibles transmisores en la red, debido a la competencia por el medio físico.

Un tercer parámetro que influye en el rendimiento es el *round trip time*. Como es sabido, las colisiones siempre ocurren dentro de los primeros 512 bits del frame, es más, ocurrirán sólo en el bit 512 cuando se esté transmitiendo a la distancia máxima de la red. Por otro lado, si la separación entre nodos es menor, entonces se reduce el round trip time entre ellos, con lo que se reduce el tiempo de colisión entre ellas, lo que implica una menor probabilidad de colisión. A la inversa, dada una misma topología de red, tamaño de frame, número de estaciones y nivel de ocupación relativa, la probabilidad de colisión disminuye a medida que aumenta la velocidad de operación de la red, ya que el valor de round trip time disminuye.

2.7.2. Especificación IEEE 802.3u Fast Ethernet

A causa de la importancia que han cobrado las redes, fácilmente caen en congestión debido a su propio crecimiento. Esto, unido al crecimiento en el número de aplicaciones hace cada vez mayor la necesidad de velocidad. En respuesta a ello, han surgido tecnologías de redes de amplio ancho de banda entre las que se incluyen ATM, FDDI, 100VG-AnyLAN y Fast Ethernet.

Fast Ethernet provee muchas ventajas, entre las que se cuentan: estar basado en el estándar IEEE 802.3, una velocidad de 100 Mbps y la maximización del uso de administración, equipo y cableado existente, manteniendo la esencia de Ethernet y encargándose sólo de hacerla más rápida bajo la misma estructura. Esto permite la fácil migración de Ethernet a Fast Ethernet. Por otro lado, como ambas emplean CSMA/CD, los datos pueden pasar de una a otra sin necesitar ningún tipo de protocolo de traducción. Esta capacidad permite instalar una red por fases, ya que se puede integrar una 100Base-T a una 10Base-T con sólo usar un puente 10/100.

Cuadro 3: Medios Físicos Especificados en IEEE 802.3u.

	100Base-TX	100Base-T4	100Base-FX
Cable	UTP Cat 5	UTP Cat 3/5	Fibra 62,5/125 micras
Pares	2	4	2
Full dúp	Sí	No	Sí
Tipo Conector	RJ-45	RJ-45	SC
Topología	Estrella	Estrella	Estrella
Dist. Seg.	100, máx 200 m	100, máx 200 m	400 m

Medio Físico

Fast Ethernet puede correr a través de la misma variedad de medios que 10BaseT: UTP, STP y fibra, pero no soporta cable coaxial. La especificación define tres tipos de medios con una subcapa física separada para cada tipo de medio: 100Base-TX, 100Base-T4, 100Base-FX. Ver tabla 3.

100Base-TX define la especificación a través de dos pares de categoría 5 de cable UTP o dos pares de tipo 1 de cable STP. 100Base-TX adopta el mecanismo de señalización full-duplex de FDDI (ANSI X3T9.5) para trabajar con la Ethernet MAC. Un par de cables es utilizado para transmitir, con codificación 4B/5B, y el otro par es utilizado para detectar colisiones y recibir datos.

La especificación 100Base-T4 utiliza pares de categoria 3, 4 o 5 UTP. 100Base-T4 es half-duplex y usa tres pares para transmisión 100 Mbps y el cuarto par para detección de colisiones. A diferencia del anterior, utiliza el código ternario 8B6T.

La capa física 100Base-FX define la especificación a través de dos hilos de fibra de 62.5/125 micras. Utiliza una fibra para la transmisión y la otra fibra para detección de colisiones y recepción.

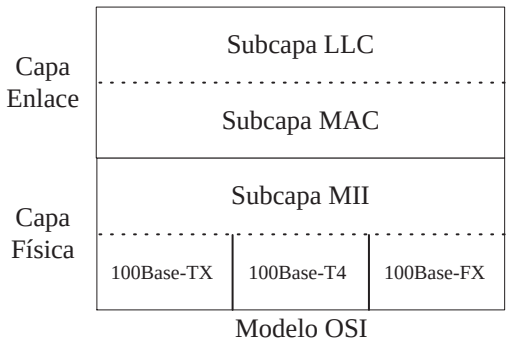

Figura 17: Ubicación de la Subcapa MII en el Modelo OSI.

Media Independent Interface

El MII es una especificación nueva que define una interface estándar entre la subcapa MAC y cualquiera de las tres capas físicas: 100Base-TX, 100Base-T4, y 100Base-FX (figura 17). Su función principal es ayudar a la subcapa de convergencia a hacer uso del rango de bits más alto y diferentes tipos de medio transparente a la subcapa MAC. Es capaz de soportar 10 Mbps y 100 Mbps. Puede ser implementada en un dispositivo de red tanto interna como externamente. Internamente conecta la subcapa MAC directamente a la capa física.

MII también define un conector de 40 pines que puede soportar transceivers externos. Un uso del transceiver adecuado puede conectar estaciones de trabajo a cualquier tipo de cables instalados, muy parecido a un conector AUI para 10 Mbps Ethernet.

Subcapa MAC

La subcapa MAC está basada en el protocolo CSMA/CD al igual que Ethernet. CSMA/CD tiene un round trip time máximo de 51.2 μseg. y un tamaño mínimo de frame de 512 bits y para Fast Ethernet la velocidad de operación de la red es de 100 Mbps. Con esto, Fast Ethernet reduce el tiempo de duración de cada bit que es transmitido en un factor de 10, permitiendo que la velocidad del frame se incremente. El formato y longitud del frame se mantuvieron, por lo que el round trip time y intervalo entre frames se reducen también en un factor de 10. Todo esto hace que no se requiera traducción de protocolo para moverse entre Ethernet y Fast Ethernet.

Auto-Negociación

La especificación IEEE 802.3u describe un proceso de negociación que permite a los dispositivos de red intercambiar información automáticamente sobre sus capacidades y desarrollar la configuración necesaria para operar juntos en su nivel común máximo. La auto-negociación es desarrollada utilizando Fast Link Pulse (FLP) Burst para identificar la tecnología más avanzada de capa física que puede ser utilizada por ambos dispositivos, como 10Base-T, 100Base-TX o 100Base-T4.

Provee también una función de detección paralela que permite reconocer capas físicas half y full-duplex 10Base-T, half y full-duplex 100Base-TX, y 100Base-T4, aún si uno de

los dispositivos conectados no ofrece capacidades de auto-negociación. El control de flujo puede ser implementado en base a enlaces o punto a punto y permite a todos los dispositivos presentes en el camino reducir la cantidad de datos que reciben.

2.7.3. Gigabit Ethernet

Las redes Fast Ethernet se extendieron con una rapidez mayor que las expectativas más optimistas. Como consecuencia de esto los precios bajaron y su uso se popularizó hasta el punto de que se utiliza Fast Ethernet incluso en la conexión del usuario final. Para mantener un diseño coherente y equilibrado de una red se requieren velocidades superiores en el backbone. Este hecho junto con la experiencia positiva de Fast Ethernet animó al subcomité 802.3 a iniciar en 1995 otro grupo de trabajo que estudiara el aumento de velocidad de nuevo en un factor diez, creando Gigabit Ethernet, que el 29 de junio de 1998 produjo la aprobación del suplemento 802.3z.

De forma análoga a lo hecho con Fast Ethernet,se pretendía poder utilizar los mismos medios físicos que en Fiber Channel: emisores láser con fibra óptica multimodo y monomodo, cable de pares trenzados apantallado y además cable UTP categoría 5. Se puede comentar también que siguiendo con la tradición ya establecida de aumentar cada vez la velocidad en un factor diez, el IEEE aprobó en enero del 2000 la creación de un grupo de estudio de alta velocidad para la eventual estandarización de una Ethernet de 10 Gigabits. Las decisiones sobre como se implementará el nivel físico de esta red se encuentran todavía en fases muy preliminares.

Cuadro 4: Medios Físicos Especificados en IEEE 802.3z.

	1000Base-T	1000Base-CX	1000Base-SX	1000Base-LX
Cable	UTP Cat 5	STP	Fibra óptica	Fibra óptica
Pares	4	2	2	2
Full dúp	Sí	Sí	Sí	Sí
Tipo Conector	RJ-45	9 pin D sub	SC	SC
Topología	Estrella	Estrella	Estrella	Estrella
Dist. Seg.	100 m	25 m	275, máx 500 m	550, máx 5000 m

Medio Físico

En Gigabit Ethernet existen tres especificaciones de medios físicos: 1000BASE-SX, 1000BASE-LX y 1000BASE-CX. Estos emplean código 8B/10B que ya se utilizaba en Fibre Channel, de donde deriva toda la capa física de 1000BASE-X. La transmisión de Gigabit Ethernet por cable UTP categoría 5 1000BASE-T se realiza de forma muy similar a 100BASE-T2, se utilizan 4 canales de 250 Mbps y se envían los datos en paralelo por los cuatro pares.

En las anteriores especificaciones, el alcance de la fibra óptica viene limitado por la atenuación de la señal, pero en Gigabit Ethernet el alcance está limitado fundamentalmente por

el efecto del retardo en modo diferencial. Este fenómeno consiste en que cuando el haz láser llega a la fibra, al ser ésta apreciablemente más ancha que el haz, éste genera haces de luz secundarios que van rebotando por las paredes al avanzar por la fibra. Este rebote no ocurre exactamente igual para todos los rayos, por lo que unos realizan un trayecto un poco más largo que otros, con lo que el pulso de luz se ensancha ligeramente. El ensanchamiento es mayor cuanto mayor es la distancia recorrida.

Subcapa MAC

La longitud mínima de un frame Ethernet fija el diámetro de la red, debido al funcionamiento de CSMA/CD. De haber mantenido el frame mínimo de 64 bytes en Gigabit Ethernet el diámetro máximo habría sido de unos 45 m, inaceptable en la mayoría de situaciones. Para evitar esto, el frame Gigabit Ethernet incorpora un segundo relleno denominado *extensión de portadora* que se añade al final del frame para garantizar que la longitud mínima nunca sea inferior a 512 bytes. De esta forma, el round trip time máximo es de 4.096 ms y el diámetro puede ser de unos 330 m. Este segundo relleno no es formalmente parte del frame Ethernet, por lo que solo existirá mientras viaje por Gigabit Ethernet. De esta forma, se respetará la compatibilidad con los tipos anteriores de Ethernet. En el caso de que un frame con extensión de portadora sea transmitida a una red de 100 o 10 Mbps, ésta se eliminará. Inversamente, si un frame menor de 512 bytes llega a una red Gigabit Ethernet desde otra, el switch correspondiente añadirá la extensión de portadora necesaria para que la longitud sea de 512 bytes.

2.8. Token Bus/IEEE 802.4

El problema principal que las empresas interesadas en automatización vieron con Ethernet era que tenían serias dudas sobre su aplicación a sistemas en tiempo real. La razones principales eran: el comportamiento no determinista de Ethernet, donde cabía la probabilidad de que dos nodos no pudieran comunicarse debido al exceso de tráfico y que no era posible reservar capacidad o establecer prioridades. Token Ring resolvía muchos de estos problemas, pero seguía presentando dos serios problemas: el papel de la estación monitor resultaba demasiado importante y una topología en bus era mas adecuada que un anillo para una línea de producción de una fábrica.

General Motors promovió entonces el desarrollo del estándar 802.4 o Token Bus, que es una red que se utiliza en algunas fábricas para el control de la maquinas. Cabe señalar que su uso es mucho más restringido que Ethernet o Token Ring, y de una manera muy simplista se puede decir que Token Bus es una mezcla entre Ethernet y Token Ring.

Medio Físico

Token Bus utiliza cable coaxial de 75 Ω idéntico al utilizado en TV. Se permite un bus single o doble con o sin terminadores. Se definen además tres esquemas distintos de modulación análoga: dos son tipo FSK y la restante es PSK. Las velocidades de transmisión son de 1, 5 o 10 Mbps. Cabe señalar que el medio físico es totalmente incompatible y mucho más complicado que 802.3.

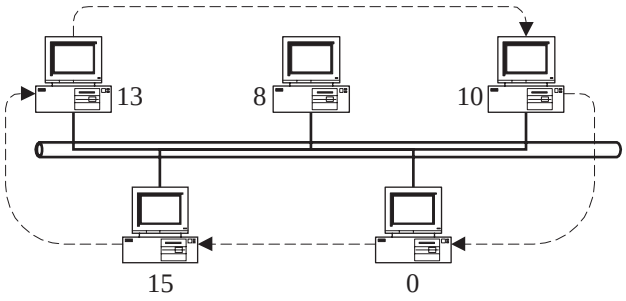

Figura 18: Organización Física y Lógica de una Red Token Bus.

Subcapa MAC

El funcionamiento básico del protocolo de nivel MAC es el siguiente: cada estación conoce la dirección de las estaciones que están a la izquierda y derecha suya. Al inicializarse el anillo el nodo con la dirección más alta puede enviar el primer frame, enviando el token a la estación vecina que tenga la dirección de nodo siguiente. Luego, el token pasa de nodo en nodo desde las direcciones altas a las bajas y el funcionamiento es similar al del protocolo token passing ya visto. Cabe señalar que el orden físico que tengan la estaciones no importa para nada en el funcionamiento y que estaciones físicamente conectadas al bus no necesariamente deben estar conectadas al anillo lógico (ver figura 18).

El protocolo define además clases de prioridades de tráfico: 0, 2, 4 y 6, siendo la 0 la más baja. Al llegar el token a la estación, se verifica si la prioridad 6 tiene datos para transmitir, si es así los envía y pasa después a la prioridad 4, sino pasa inmediatamente a la siguiente prioridad, situación que sigue hasta llegar al nivel 0 o bien hasta que haya expirado el token holding time. El protocolo además provee mecanismos para hacer uso del tiempo por parte de las prioridades inferiores si es que las superiores no tienen nada que transmitir. Además, se puede permitir reservar una cierta fracción del tiempo al tráfico de prioridad 6, lo que implica una reservación de ancho de banda para este tipo de tráfico, que permitirá el mejor tratamiento de tráfico en tiempo real.

1 B	1 B	1 B	2-6 B	2-6 B	0-8182 B	4 B	1 B
Pre	DI	FC	Dir. Destino	Dir. Fuente	Datos	CRC	DT

Figura 19: Formato del Frame Token Bus.

El formato del frame token bus se observa en la figura 19 y el detalle de los campos es el siguiente:

Preámbulo: un byte que es utilizado para sincronizar el clock del receptor.

Delimitador de Inicio: un byte utilizado para demarcar el inicio del frame.

Frame de Control: un byte que distingue al frame entre un frame de datos o de control. Si es de datos lleva el nivel de prioridad y puede llevar la solicitud de acknowledge positivo o negativo para el destino. Caso contrario indica el tipo de frame de control que representa.

Dirección Destino: indica la dirección del nodo destino.

Dirección Origen: indica la dirección del nodo fuente.

Datos: capo que encapsula los datos del nivel superior.

CRC: cuatro bytes que corresponden a una suma de verificación para asegurar que el frame llegó en buen estado.

Delimitador de Término: un byte utilizado para demarcar el final del frame.

Mantenimiento del Anillo

Una vez que el anillo entró en funcionamiento cada nodo sabe quién es su predecesor y su sucesor. Periódicamente, el nodo que tiene el token envía un frame de control, especial llamado *Solicit Successor* que permite agregar nuevas estaciones al anillo, siempre que se encuentren en el rango del poseedor del token. Si sólo un nodo desea entrar, hace ingreso al anillo y será el siguiente poseedor del token. Si existe más de uno, entonces se producirá una colisión, y el poseedor del token enviará un frame *Resolve Contention* que permitirá arbitrar la colisión de una manera similar a como funciona el protocolo de cuenta regresiva binaria. Se debe notar que no existe garantía de cuál será el tiempo máximo que debe esperar un nodo para poder hacer ingreso a la red, pero en la práctica no debiera ser más que algunos segundos.

Si un nodo desea dejar el anillo envía un frame *Set Successor* a su predecesor con la dirección de su sucesor y entonces queda fuera del anillo.

Para inicializar el anillo el primer nodo que se enciende envía el frame de control *Claim Token*, de no recibir respuesta, crea el token y el anillo, enviando periódicamente frames *Solicit Successor*. Nuevamente, si más de un nodo envía un frame *Claim Token* se produce una colisión y el problema se resuelve de la misma forma que antes.

El problema de que un nodo falle cuando deba enviar el token se soluciona haciendo que el predecesor quede escuchando si su sucesor envío algún frame, de no ser así, vuelve a enviar el token. Si el problema persiste entonces envía un frame *Who Follows* con la dirección de su sucesor. El frame, al ser visto por el sucesor de la estación que falló, envía un *Set Successor* y el anillo se reestablece. Si fallara también el nodo que sigue a la estación que originalmente falló, entonces se envía un frame *Solicit Successor 2* para ver si algún nodo más está operativo. Esto puede producir colisiones, las que se resolverán de la manera tradicional, reestableciendo el anillo. Si el problema se presenta con el nodo que tiene el token y éste se pierde, entonces pasado un cierto time out, los nodos restantes utilizan el algoritmo de inicialización del anillo.

2.9. Token Ring/IEEE 802.5

Después de la propuesta de Ethernet y de Token Bus, el comité IEEE 802.3 recibió otra propuesta, esta vez de IBM que presentó una red con topología de anillo y protocolo MAC token passing que denominaban Token Ring. El comité, viendo que no sería posible elaborar un único estándar y considerando que el apoyo de la industria a cada propuesta era demasiado importante como para descartar cualquiera de ellas, optó por aceptar las tres y crear un subcomité para cada una de ellas: 802.3 para CSMA/CD (Ethernet), 802.4 para Token Bus y 802.5 para Token Ring.

Medio Físico

El estándar define dos velocidades de operación: 4 y 16 Mbps. El cableado utilizado es STP o UTP categoría 3 o superior para 4 Mbps, y STP para 16 Mbps. La señal se representa usando codificación Manchester diferencial, que emplea la presencia o ausencia de transición entre dos voltajes para indicar un 0 o un 1, respectivamente. Requiere un equipo mas caro y complejo que la codificación Manchester, pero es más inmune al ruido y está mejor adaptada al uso de cable pares, ya que no tiene problemas de polaridad invertida.

El gran problema de la topología anillo es que la rotura de éste en un punto impide la comunicación. Para evitar el problema, lo que se hace es colapsar el anillo en un hub o concentrador, también llamado centro de cableado, al cual se conectan los cables de entrada y salida de cada estación. El cableado sigue siendo lógicamente un anillo, aún cuando físicamente sea una estrella. En el concentrador se instalan relés de bypass alimentados por el nodo correspondiente, de forma que si la conexión de ésta falla el relé cortocircuita el enlace correspondiente restaurando el anillo. También es posible constituir anillos dobles para obtener mayor confiabilidad, ya que en caso de corte en un punto, el doble anillo puede cerrarse sobre sí mismo solucionando el problema.

Cada nodo que se agrega a la red añade una cierta cantidad de jitter en la retransmisión de la información, situación que limita el número máximo de estaciones que pueden estar presentes en una red Token Ring. En redes de 4 Mbps con cable UTP el máximo es de 72, mientras que en las de 16 Mbps con cable STP el máximo es de 250 estaciones.

Subcapa MAC

Las redes token ring funcionan básicamente de la siguiente forma: si ningún host desea transmitir, todos están en modo escucha y el token, que es un frame de 3 bytes, va pasando de un host a otro indefinidamente. Cuando algún nodo desea transmitir debe esperar a que pase por él el token. En ese momento, se apodera de éste, pasa a modo transmisión y envía el frame al siguiente nodo. Todos los demás hosts del anillo, incluido el destino, siguen en modo escucha, retransmitiendo el frame recibido hacia el siguiente nodo. El host destino, además de retransmitirlo, retiene una copia del frame. El nodo transmisor puede que restaure el token en el anillo inmediatamente, o puede que espere hasta recibir su frame para restaurar el token.

El formato del frame token bus se observa en la figura 20 y el detalle de los campos es el siguiente:

Delimitador de Inicio: un byte utilizado para demarcar el inicio del frame.

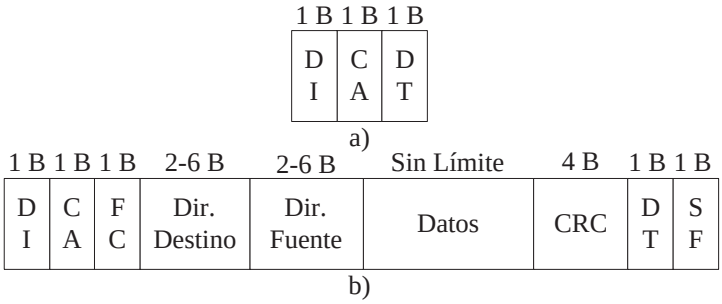

Figura 20: Formato del a) Token de Token Ring b) Frame Token Ring.

Control de Acceso: un byte que contiene bits especiales, el del token, el monitor, tres de prioridad y tres de reserva.

Frame de Control: un byte que distingue al frame entre un frame de datos o de control.

Dirección Destino: indica la dirección del nodo destino.

Dirección Origen: indica la dirección del nodo fuente.

Datos: capo que encapsula los datos del nivel superior.

CRC: cuatro bytes que corresponden a una suma de verificación para asegurar que el frame llegó en buen estado.

Delimitador de Término: marca el final del frame. Los seis primeros bits forman una secuencia inválida en la codificación Manchester diferencial. El séptimo se utiliza para indicar el último frame cuando lo que se transmite es una secuencia de frames. El octavo bit indica si se ha producido un error en la transmisión del frame entre dos nodos. Si algún nodo detecta un error en el frame pondrá en 1 este bit. Esto integra un mecanismo intrínseco de detección de errores en la transmisión.

Status del Frame: un byte que contiene dos bits especiales, el A y el C. Al llegar un frame al destino, éste coloca el bit A en uno y si el nodo copia el frame coloca el bit C en uno. Con esto, el nodo emisor al recibir su frame tiene las siguientes opciones en los bits AC: 00 destino no presente, 10 destino presente y frame no aceptado, 11 destino presente y frame copiado. Con esto, el protocolo incorpora un mecanismo automático de acuse de recibo.

La estructura de un token es una versión simplificada de un frame. Contiene únicamente los campos DI, CA y DT. En el campo CA el bit de token está siempre puesto a 0. En el campo DT los dos últimos bits están siempre a 0.

El campo CA dispone de tres bits de prioridad que funcionan de la siguiente manera: cuando un host desea transmitir un frame con prioridad n debe esperar a que pase por él un

token de prioridad menor o igual que *n*. Además, los hosts pueden aprovechar un frame en tránsito para solicitar al emisor un token de una determinada prioridad. Un host sólo puede utilizar los bits de reserva si éstos no contienen ya una petición de mayor prioridad. Cuando el frame de datos vuelve a su emisor, éste emitirá un token de la prioridad solicitada, que será la más alta que hubiera pendiente en el anillo. En el caso de funcionar con Early Token Release este mecanismo de prioridad queda parcialmente deshabilitado debido a que el emisor ha de restaurar el token antes de haber recibido las solicitudes de reserva.

Mantenimiento del Anillo

En toda red Token Ring existe un nodo denominado monitor que se ocupa de resolver los problemas y garantizar el normal funcionamiento del protocolo. En caso de problemas, el monitor restaurará un token en el anillo para que el tráfico pueda seguir circulando normalmente. Además, cualquier host de la red está capacitado para actuar como monitor en caso necesario.

Cuando un nodo se une a la red, escucha la red en busca de tokens o frames de datos. Si no detecta actividad emite un frame *Claim Token*. Si existe ya un monitor, responderá con un token a la petición. Si no, el nodo recién incorporado a la red recibirá su propio claim token, momento en el cual pasará a constituirse en monitor.

Un frame huérfano es un frame enviado por una estación que posteriormente falla y no lo saca del anillo. El monitor es el encargado de darse cuenta de la situación y eliminar el frame, utilizando para ello el bit monitor, pues si un frame pasa dos veces por el monitor con el bit seteado quiere decir que corresponde a un frame huérfano. Además, el monitor tiene un timer asociado que permite determinar la falta de tokens en la red.

El monitor se ocupa también de facilitar los buffers necesarios para garantizar que en todo momento la red puede albergar los 24 bits del token, pues para que el protocolo pueda funcionar es necesario que el tamaño de la red permita mantener el token circulando en todo momento.

Una función que no es realizada por el monitor es la de detectar rupturas del anillo. Si un nodo detecta que uno o más de sus vecinos está abajo, transmite una señal *Beacon* con la dirección de los nodos que fallan. Una vez que esta se propaga, los nodos se sacan del anillo con los relés de bypass.

2.10. 100VGAnyLAN/IEEE 802.12

100VGAnyLAN es un estándar LAN que pretende ofrecer una alta velocidad con un medio compartido sustituyendo los protocolos más lentos, pero utilizando los medios existentes y siendo compatible con ellos.

100VGAnyLAN está formada principalmente por nodos finales y repetidores. Los nodos finales son normalmente computadores, bridges, routers, switches o servidores. Ellos son los que transmiten y reciben datos a través del repetidor. Los repetidores son el centro conceptual y físico de la red. Sirven de controladores centrales y manejan el acceso a la red realizando continuamente tests. Cada repetidor tiene un puerto de conexión especial a través del cual se puede conectar con otros repetidores en cascada. Un repetidor debe ser configurado para

manejar formatos de frame token ring o IEEE 802.3. Todos los repetidores de un mismo segmento deben usar el mismo formato de frame.

Medio Físico

La topologia de 100VGAnyLAN es estrella, pero no es necesario que los nodos finales estén conectados directamente al nodo central, sino que pueden conectarse a través de nodos intermedios, como un árbol.

La especificación estable cable UTP categoría 3 o 5, STP y fibra óptica. La velocidad de operación es de 100 Mbps y la codificación es 5B6B.

Subcapa MAC

La transmisión de frames consiste en una secuencia en la que el emisor realiza una petición y la contraparte contesta la petición. La secuencia de transmisión sigue los siguientes pasos:

1. Si un nodo tiene un frames para transmitir emite una señal de control *Request* que puede tener una prioridad normal o alta. Si no es el caso, el nodo transmite la señal de control *IdleUp*.

2. El repetidor sondea todos los puertos para determinar que hosts han pedido enviar un frame y con que prioridad se ha realizado la petición.

3. El repetidor selecciona el nodo con petición de prioridad alta pendiente, de acuerdo al puerto al que está conectado. Lo mismo se hace para las prioridades bajas. La selección causa que el puerto elegido reciba la señal *Grant* y la transmisión del frame comienza cuando el nodo detecta la señal de permiso.

4. El repetidor envía una señal *Incoming* a todos los otros nodos finales, avisándoles de la posible llegada de un frame. El repetidor decodifica la dirección de destino del frame transmitido en cuanto la recibe.

5. Cuando un nodo final recibe la señal de llegada, se prepara para recibir un frame interrumpiendo la transmisión de peticiones y escuchando en el medio.

6. Una vez que el repetidor ha decodificado la dirección destino, el dato es enviado al nodo o nodos finales correspondientes. Los nodos que no reciben el frame, reciben la señal de control *IdleDown* del repetidor para que vuelvan a lo que estuvieran haciendo.

7. Cuando un nodo final recibe un frame de datos, vuelve a su estado anterior a la recepción, enviando una señal *IdleUp* o haciendo una petición para enviar un frame.

100VGAnyLAN está diseñada para operar de forma compatible con los formatos de frame Ethernet y token ring. Esto significa que el software y los protocolos sobre el nivel de enlace sólo necesitan saber que están operando en una red Ethernet o Token Ring. La descripción del frame es la siguiente (ver figura 21):

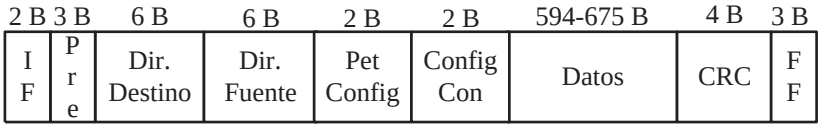

Figura 21: Formato del Frame 100VGAnyLAN.

Inicio del Frame: delimitador de inicio de frame que permite detectar cuando se esté enviando un frame.

Preámbulo: permite detectar donde empiezan los datos

Dirección Destino: indica la dirección del nodo destino.

Dirección Origen: indica la dirección del nodo fuente.

Petición de Configuración: permite al nodo informar al repetidor sobre sí mismo y pedir una configuración de su puerto. El bit repetidor de este campo informa al repetidor si se trata de un nodo final o de otro repetidor.

Configuración Concedida: permite al repetidor responder a la petición de configuración con la configuración asignada. Este campo se pone a cero por el nodo final y el repetidor asigna los valores apropiados.

Datos: capo que encapsula los datos del nivel superior.

CRC: cuatro bytes que corresponden a una suma de verificación para asegurar que el frame llegó en buen estado.

Final del Frame: permite finalizar la recepción del frame, y enviar los datos al nivel superior. La detección en el delimitador de una secuencia *Invalid Packet Marker* informa de un error.

Prueba del Enlace

La prueba de enlace tiene diferentes propósitos, como la verificación de la calidad del cable para la transmisión de datos, ayudar al receptor a adaptarse al enlace estableciendo la dirección MAC del nodo final en la tabla del repetidor y establecer la configuración del enlace para frames Ethernet o token ring y determinar si se trata de un nodo final o repetidor. La prueba de enlace se efectúa cada vez que se establece un enlace, como al encender el equipo y al conectar el cable o cuando ocurren algunos errores.

La prueba de enlace siempre es iniciada por la entidad inferior al repetidor, que es quien desea conectarse a la red. La prueba incluye la transmisión de frames de prueba entre la entidad inferior y el repetidor.

Una vez que la entidad inferior se ha conectado a la red, el repetidor añade su dirección su tabla interna. Si la entidad inferior es también un repetidor, todas los frames que reciba el repetidor superior serán transmitidos a esa entidad inferior. Si el repetidor superior tiene repetidores conectados o esta conectado a un repetidor superior, todos los frames de la entidad inferior son enviadas a esos repetidores. Si el repetidor tiene la dirección destino en su tabla interna, el frame es dirigido al nodo especifico. En caso de que el repetidor reciba un frame de su repetidor superior, no dirigida a ninguno de sus nodos finales, el frame se descarta.

2.11. Interfaces FDDI

FDDI funciona a 100 Mbps y su estándar fue definido inicialmente por ANSI y más tarde adoptado por ISO. Análogamente a los otros estándares, el documento describe la capa física y la subcapa MAC. Para la parte LLC se utiliza el estándar IEEE 802.2.

Las características de velocidad, distancia y confiabilidad de FDDI la convirtieron durante algún tiempo en la red ideal para ser utilizada como backbone que concentre las LANs de una gran organización.

FDDI tiene muchos elementos comunes con Token Ring, y en cierta medida puede considerarse una evolución de aquella tecnología. La topología es de doble anillo para aumentar la seguridad, no la velocidad. En condiciones normales un token gira por cada anillo en sentido opuesto. Las estaciones pueden ser SAS (Single Attach Station) si están enchufadas a un anillo únicamente, o DAS (Dual Attach Station) si lo están a ambos. Si se produce un corte en los anillos las estaciones DAS más próximas a cada lado del corte unen entre sí ambos anillos, con lo que se crea un anillo de mayor longitud que permite mantener conectados todos los hosts. En el caso de producirse un segundo corte en otro punto del anillo se crean dos anillos aislados, cada uno de los cuales puede seguir funcionando. Las interfaces DAS son más caras que las SAS, pero dan una mayor tolerancia a fallas al permitir cerrar el anillo.

Hasta 1996 FDDI era la principal tecnología de red de alta velocidad. Sin embargo, su grado de implantación siempre ha sido escaso. La principal razón de esto ha sido su elevado costo comparado con las LANs tradicionales, como Ethernet o Token Ring. Hoy en día ha sido completamente superada por Fast Ethernet y Gigabit Ethernet.

Medio Físico

El medio físico de transmisión es fibra óptica multimodo o UTP Categoría 5. En este último caso se utiliza una topología física similar a la de 10Base-T, con los nodos conectados a hubs. La distancia máxima del nodo al hub es de 100 metros. También es posible utilizar hubs para la conexión de nodos por fibra óptica. FDDI puede considerarse una versión evolucionada de Token Ring en muchos de sus aspectos. No se utiliza codificación Manchester sino 4B5B.

Subcapa MAC

La estructura de un frame y token FDDI es muy similar a la de token ring (ver figura). La longitud máxima del campo datos puede ser de hasta 4500 bytes.

El protocolo MAC de FDDI es también muy parecido al de Token Ring. La diferencia más notable es que en FDDI siempre se funciona con el principio de Early Token Release. Existe

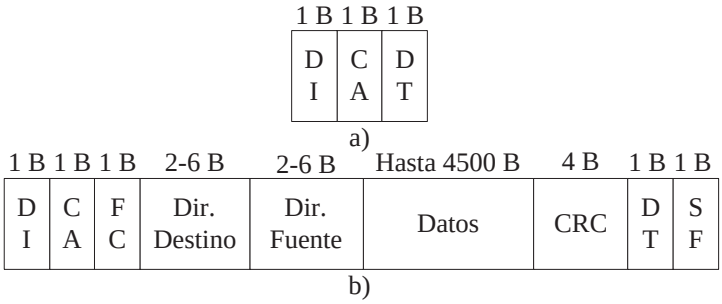

Figura 22: Formato del a) Token FDDI b) Frame FDDI.

también un token-holding timer que establece el tiempo máximo que una estación puede transmitir de una vez. Este parámetro tiene importantes consecuencias en el rendimiento de la red. El valor por defecto de 4 ms es adecuado en la mayoría de las situaciones, excepto en las redes muy grandes (más de 20 Kms) en que es conveniente aumentarlo.

2.12. LAN Emulation: ATM y su Interconexión LAN

La clave para el éxito de ATM reside en la capacidad que tenga de operar con las redes de datos que actualmente se encuentran en operación. Para ello, la interconexión debe ser usando los mismos protocolos de nivel de red que están en uso, como IP e IPX.

El protocolo LAN Emulation o LANE lo que hace es definir una interfaz de servicios para los protocolos del nivel superior, que es idéntica a la existente en las LANs y que los datos enviados a través de la red ATM son encapsulados en el formato apropiado de paquetes LAN MAC. Esto no significa que se intente emular el método de control de acceso al medio (CSMA/CD para Ethernet y token passing para Token Ring) en la LAN emulada. En otras palabras, los protocolos LANE hacen que una red ATM se vea y comporte como una LAN Token Ring o Ethernet, pero operando a mucho mayor velocidad. La razón de todo esto es que así no se requieren modificaciones sobre los protocolos de nivel 3 existentes. Es así como el funcionamiento básico de LANE es la resolución de direcciones MAC en direcciones ATM.

Está considerado que LANE será desarrollado sobre tipos de equipos ATM:

1. **ATM Network Interface Cards (NIC)**: que implementará LANE y serán la interfaz hacia la red ATM, pero ofreciendo una interfaz de servicio corriente a los drivers de los protocolos de nivel superior. La comunicación será igual que si estuvieran en una LAN. Sin embargo, serán capaces de usar un ancho de banda mucho mayor.

2. **Equipos de Red y Switches LAN**: corresponde a switches LAN y routers asociados a equipos ATM. Los switches LAN serán usados para proporcionar servicios de LAN virtuales. Los equipos de red como los routers también permitirán la implementación de LANs virtuales a nivel de red.

La figura (23) muestra la arquitectura del protocolo LANE.

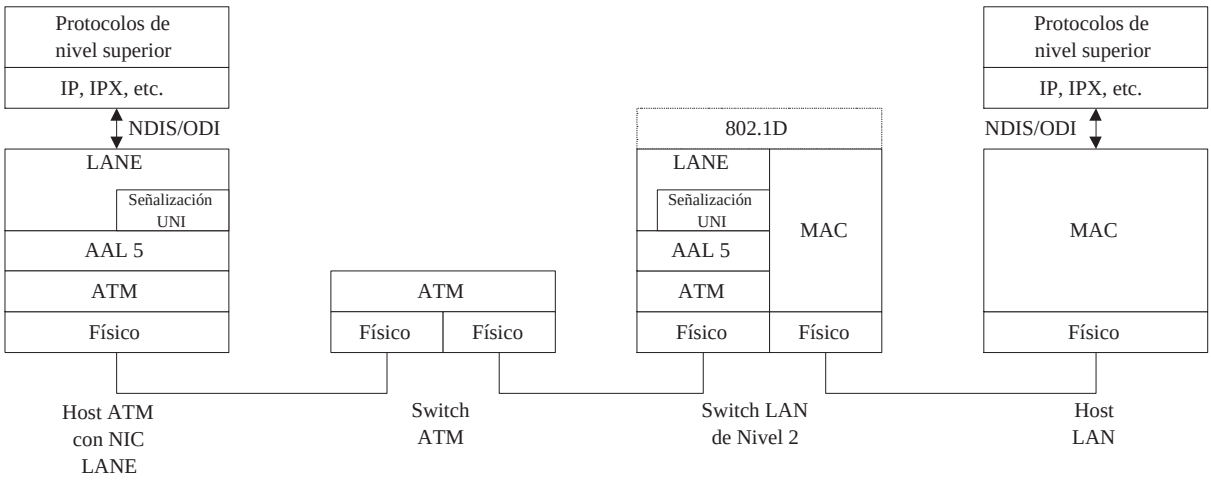

Figura 23: Arquitectura del Protocolo LANE.

Componentes y tipos de Conexión LANE

Una LAN emulada (ELAN), ya sea Ethernet o Token Ring, consiste de las siguientes entidades (ver también figura (24)):

Cliente LANE o LEC: es la entidad que en un sistema final, ejecuta el envío de datos, resolución de direcciones y otras funciones de control para un sistema final dentro de una ELAN. También provee una interfaz estándar para las entidades de nivel superior. Una NIC ATM o un switch LAN que es interfaz de una ELAN soporta un sólo LEC por cada ELAN a la que esté conectado. Los sistemas finales que se conecten a múltiples ELANs tendrán un LEC por ELAN. Cada LEC es identificado por una dirección ATM única, y es asociado con una o más direcciones MAC accesibles a través de la dirección ATM.

Servidor LANE o LES: implementa la función de control para una ELAN en particular. Existe un sólo LES lógico por ELAN, y para pertenecer a una ELAN en particular se requiere establecer alguna relación de control con el LES asociado a la ELAN. Cada LES es identificado con una dirección ATM única.

Servidor de Broadcast y Desconocido o BUS: es un servidor de multicast usado para inundar la red con el tráfico de direcciones desconocidas y enviar tráfico multicast y broadcast a los clientes de la ELAN. Cada LEC está asociado a un sólo BUS por ELAN, pero existen múltiples BUS dentro de una ELAN. El BUS al que se conecta el LEC es identificado por una dirección ATM única, que es asociada en el LES con la dirección de broadcast.

Servidor de Configuración LANE o LECS: es una entidad que asigna LECs a las ELANs, dirigiéndolos a un LES en particular. Existe un LECS lógico por dominio administrativos

y sirve a todas las ELANs dentro del dominio.

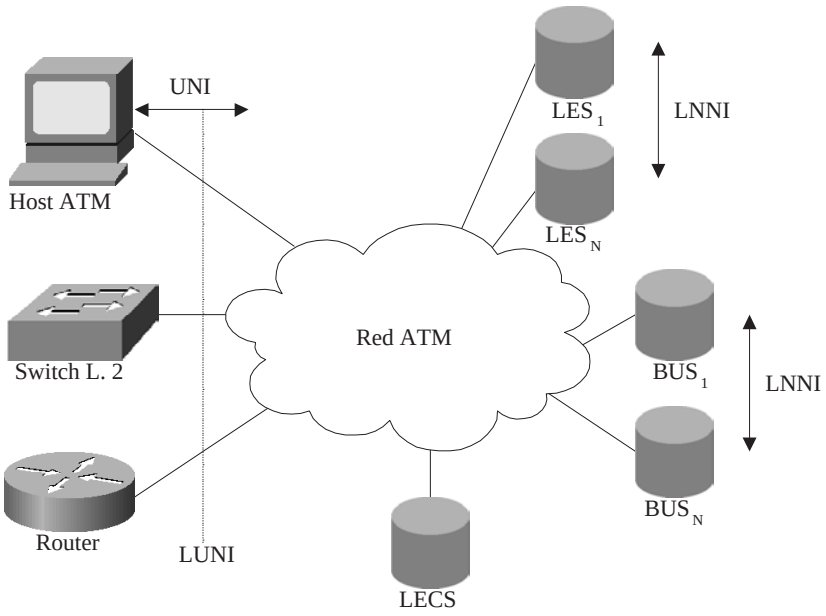

Figura 24: Interfaces del Protocolo LANE.

El protocolo LANE especifica sólo la operación de la interfaz LANE usuario a red (LUNI) entre el LEC y la red que provee el servicio. Esto puede ser contrastado con con la interfaz LANE red a red (LNNI) que opera entre los servidores dentro de una ELAN.

Las entidades definidas anteriormente se comunican entre sí usando dos tipos de conexiones, una para control de tráfico y otra para transmisión de datos.

Las conexiones de control son las siguientes (ver figura (25)):

VCC de Configuración Directo: VCC bidireccional punto-a-punto desde LEC al LECS.

VCC de Control Directo: VCC bidireccional punto-a-punto desde LEC al LES.

VCC de Control Distribuido: VCC unidireccional desde LES al LEC, es típicamente punto-a-multipunto.

Las conexiones de datos son las siguientes (ver figura (26)):

VCC Directo de Datos: VCC bidireccional entre LECs que desean intercambiar datos. Típicamente son conexiones ABR o UBR y no ofrecen soporte de QoS.

VCC de Envío de Multicast: VCC bidireccional punto-a-punto desde LEC al BUS.

VCC de Reenvío de Multicast: VCC unidireccional desde BUS al LEC. Típicamente es una conexión punto-a-multipunto con cada LEC como hoja.

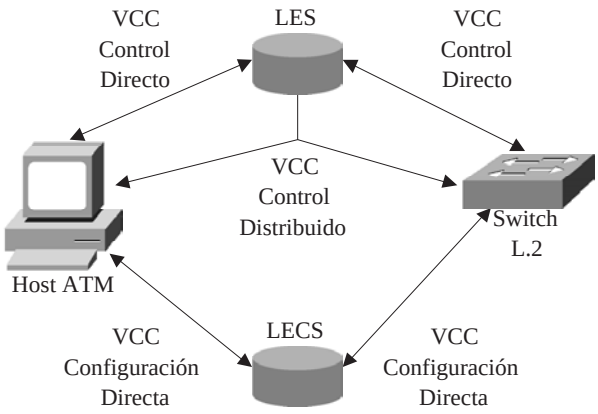

Figura 25: Conexiones de Control LANE.

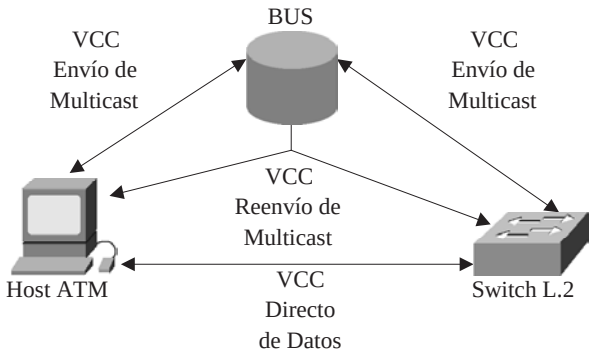

Figura 26: Conexiones de Datos LANE.

Operación de LANE

Inicialización y Configuración

En la inicialización el LEC debe obtener su dirección ATM. Para esto el LEC establece una conexión de configuración directa, ubicando la direccción del LECS de una de las siguientes tres maneras: usando un determinado procedimiento ILMI, usando una dirección de LECS conocida o usando una conexión a LECS conocida.

Una vez encontrado el LECS, el LEC establece la conexión de configuración directa y un protocolo de configuración se usa para indicar al LEC de la información necesaria que requiere para conectarse a una ELAN (dirección ATM del LES, tipo de LAN emulada, tamaño máximo de los paquetes y nombre de la ELAN).

Unión y Registro

Una vez que el LEC obtiene la dirección del LES, se debe establecer una conexión de control directa al LES, para que éste asigne un identificador LEC (LECID). El LEC registra sus direcciones MAC y ATM.

El LES establece un enlace de control distribuido. Éste último y el enlace de control directo pueden ser usados por el LEC para implementar la resolución de direcciones con el procedimiento LANE ARP (LE_ARP). Esto se lleva a cabo de la siguiente manera: el LEC solicita con un LE_ARP una dirección al LES, que consulta su tabla y la devuelve por el VCC de control directo si la conoce, o por el de control distribuido si no la conoce, para que el que la sepa responda al LE_ARP.

Para completar el mecanismo de inicialización, el LEC usa el LE_ARP para determinar la dirección ATM del BUS, enviando al LES la dirección de broadcast MAC, a la que el LES responde con la dirección ATM del BUS. El LEC establece entonces la conexión de envío de multicast con el BUS, a la que el BUS responde con la conexión de reenvío de multicast, añadiendo al LEC como una hoja nueva.

Transferencia de Datos

Durante la transferencia de datos el LEC recibe un paquete del nivel de red desde un protocolo de nivel superior o un paquete MAC a enviar a través de un puerto, en el caso de los switches LAN. En primera instancia el LEC no conoce la dirección ATM del LEC destino. Para resolver esto envía al LES un LE_ARP.

Mientras espera respuesta al LE_ARP, el LEC envía el paquete al BUS, usando alguna encapsulación definida. El BUS enviará el paquete a todos los LECs. Si se recibe la respuesta al LE_ARP, el LEC establece un VCC directo de datos con el nodo destino, y utiliza esa conexión para la transferencia de datos en vez de la trayectoria del BUS. Si ya existía una conexión directa con el LEC, en la misma ELAN, opcionalmente puede reutilizarse la conexión antigua, conservando recursos y evitando latencia de conexión.

Si no se recibe respuesta al LE_ARP, el LEC continuará enviando paquetes al BUS, pero regularmente enviará LE_ARPs hasta recibir respuesta. Un LEC construirá un cache de las direcciones MAC a ATM, para luego consultar sus propias tablas y así aumentar el desempeño del sistema, disminuyendo el tráfico.

El BUS también es utilizado por el LEC para enviar paquetes broadcast y multicast. Estos paquetes son enviados al BUS, que los reenvía a todos los LECs. Debido a que algunos protocolos de nivel superior no toleran que se reciba una copia de un paquete propio, la encapsulación requiere que todos los paquetes MAC contengan un LECID que permita a los LECs filtrar sus propios paquetes.

2.13. Subcapa de Control de Enlace Lógico

La subcapa MAC forma la mitad inferior de la capa de enlace en las redes broadcast. Sobre ella se encuentra la subcapa LLC que corresponde en funciones a la capa de enlace de las líneas punto a punto, esto es, realiza la comunicación punto a punto entre los dos hosts que interactúan.

El IEEE ha desarrollado el estándar 802.2 para especificar el protocolo de esta subcapa. Éste es compatible con todos los protocolos de nivel MAC de la serie 802, de forma que todas las redes locales 802 presentan una interfaz común a la capa de red independientemente de cual sea el medio físico y el protocolo MAC que se esté utilizando. El protocolo LLC está basado en HDLC y suministra tres tipos de servicio:

LLC Tipo 1: datagramas sin acuse de recibo. Este es el más utilizado, es un servicio similar al ofrecido por PPP dónde no existe control de flujo, pues no hay realimentación del receptor al emisor. A diferencia de PPP aquí no se realiza verificación de errores pues ésta ya ha sido efectuada por la subcapa MAC.

LLC Tipo 2: servicio confiable orientado a la conexión, similar al ofrecido por HDLC. Se realiza control de flujo y solicitud de retransmisión si detecta error en el checksum.

LLC Tipo 3: es un intermedio de los dos anteriores. El emisor envía datagramas y solicita acuse de recibo, pero éstos son enviados también como datagramas, no hay un proceso explícito de establecimiento de la conexión como ocurre en el tipo 2.

La mayoría de los protocolos de red utilizados, como IP, requieren únicamente el LLC de tipo 1, por lo que las funciones de la subcapa LLC son casi inexistentes.

La principal función que desempeña la subcapa LLC es suministrar el soporte multiprotocolo, es decir multiplexar adecuadamente los frames recibidos de los diferentes protocolos posibles en el nivel de red antes de pasarlos a la subcapa MAC. Esto se hace mediante campos especiales del frame LLC. En el caso de redes Ethernet con frames en formato Ethernet la capa LLC es totalmente inexistente ya que esta información se suministra en el Ethertype.

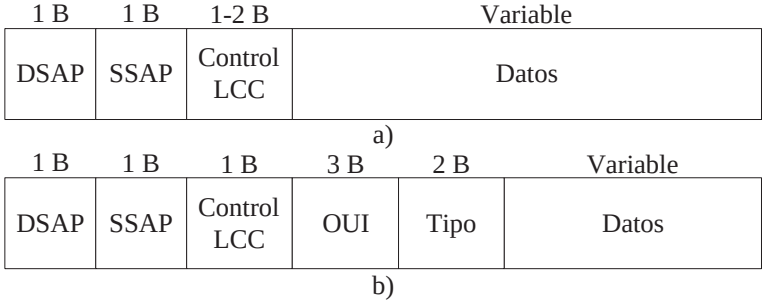

Figura 27: Formato del a) Frame IEEE 802.2 LLC b) Frame IEEE 802.2 LLC-SNAP.

El campo Control LLC especifica el tipo de servicio utilizado. En LLC tipo 2 se utilizan los mismos tipos de frame y comandos que en HDLC. En LLC tipo 1 el campo control siempre vale 00000011 que significa frames no numerados.

Los campos DSAP y SSAP tienen la finalidad de permitir identificar a que protocolo de red pertenece el frame LLC. Aunque se reserva un byte para especificar el protocolo los dos primeros bits del DSAP y el SSAP están reservados, ya que tienen significados de

grupo/individual y local/global, igual como en las direcciones MAC del IEEE. Con sólo 64 posibles valores el campo DSAP/SSAP se mostró rápidamente insuficiente.

La solución al problema fue reservar un valor en el DSAP y el SSAP (11111111) para indicar la existencia de un campo adicional denominado SNAP, inmediatamente a continuación del campo Control LLC y antes de los datos, que permite especificar cualquier protocolo. El campo SNAP se divide en dos partes: los primeros tres bytes forman lo que se denomina el OUI (Organizationally Unique Identifier) que identifica al fabricante que registra el protocolo ante el IEEE, mientras que los dos últimos identifican el protocolo dentro de ese fabricante.

Un frame LLC es la manera normal de enviar los datos en cualquier LAN, excepto en Ethernet, donde existen dos posibilidades:

- Usar el campo longitud en el frame MAC y poner en el campo datos un frame LLC que contiene el tipo de protocolo utilizado a nivel de red. En este caso, normalmente se utilizará un frame LLC-SNAP, por lo que la longitud máxima del paquete de nivel de red será de 1492 bytes. Esta es la aproximación empleada por Appletalk fase 2, NetBIOS y algunas implementaciones de IPX.
- Usar el campo tipo en el frame MAC y poner directamente en el campo datos el paquete de nivel de red. En este caso, la longitud máxima del paquete a nivel de red podrá ser de 1500 bytes. Este formato es empleado por TCP/IP, DECNET fase 4, LAT y algunas implementaciones de IPX.

2.14. Dispositivos LAN

2.14.1. Repetidores

Son dispositivos activos de solamente dos puertas que permiten interconectar dos medios de comunicación con el objeto de amplificar y reformar los pulsos constituyentes de la señal. Usualmente se utilizan para extender la longitud de los cables en una LAN o conectar medios de tipo diferente, generando una LAN única más extensa. Los repetidores interconectan las LAN al nivel del nivel ISO más bajo, el nivel físico. Esto significa, que los repetidores pueden sólo conectar LAN idénticas, tales como Ethernet/802.3 a Ethernet/802.3 o Token Ring a Token Ring.

Los repetidores Ethernet/802.3 dejan pasar todos los frames, para así asegurar que todos los hosts respeten el método de detección de colisión. Una LAN puede contener múltiples segmentos de cable y múltiples repetidores, en IEEE 802.3 se permiten máximo 4. Los repetidores Ethernet/802.3 pueden tener sus puertas idénticas o incluir combinaciones de los diferentes tipos permitidos por la norma IEEE 802.3. Con esto, permiten interconectar segmentos diferentes.

Cuando una colisión es detectada, el repetidor también coloca la señal de jamming para asegurar que todos los otros dispositivos se percaten que ha ocurrido una colisión. El detector de jam cuenta el número de colisiones consecutivas, y sí ésta excede un valor predefinido, entonces el repetidor desactiva el segmento. Sí se recibe un frame desde ese segmento de red, el repetidor lo reactiva automáticamente. De esta forma, segmentos con problemas son desconectados y segmentos válidos reconectados en forma dinámica.

2.14.2. Hubs

Un hub permite derivar desde un segmento único varios segmentos del mismo u otro tipo, y así estructurar una LAN en mejor forma. Los hubs pueden ser pasivos o activos. En los activos se incluyen las funciones básicas de un repetidor. En una red Ethernet/IEEE 802.3 los hub típicamente permiten crear derivaciones desde una red 10Base5 a múltiples segmentos 10Base2 (hub de coaxial), para implementar conexiones multipunto, o bien crean conexiones desde una red 10Base2 a múltiples segmentos 10BaseT (hub UTP). Se puede decir entonces, que los hubs 10BaseT son realmente repetidores multipuerta.

2.14.3. Bridges

Existen muchas circunstancias en las que no se quiere o no se puede tener una sola LAN. Por ejemplo:

1. Se dispone de un par de LANs diferentes (una Token Ring y una Ethernet) y desean conectarse.
2. Se necesita cubrir una distancia mayor que la que puede cubrirse con una LAN.
3. Se quiere conectar más nodos que los que se permiten en una LAN.
4. Se desea evitar que un problema en un nodo pueda colapsar toda la red, pues por ejemplo en Ethernet, una tarjeta en mal estado puede inutilizar toda la red.

En estos casos, es posible tener múltiples redes locales interconectadas mediante el uso de dispositivos llamados bridges o puentes. Los bridges se encargan de capturar los frames de una LAN, y reenviarlos selectivamente a otra LAN. Para esto, analizan la dirección de origen y destino del frame a nivel MAC. De acuerdo a la función que desempeñan, los bridges se pueden clasificar en: puentes transparentes, puentes remotos, puentes traductores o puentes con encaminamiento desde el origen.

Puentes Transparentes

La idea tras estos bridges es que puedan utilizarse sin alterar el protocolo o la configuración de los nodos. Normalmente estos equipos no necesitan ningún tipo de configuración previa, actuando como dispositivos plug and play.

Un puente transparente funciona de la siguiente forma: se supone que un bridge que une LAN1 y LAN2. Éste tendrá dos interfaces físicas, cada una conectada a una de las dos LANs. Al encender el bridge, éste empieza reenviando todas los frames que recibe por LAN1 a LAN2, y viceversa. En todo momento, el bridge actúa en modo promiscuo, es decir, capturando todos los frames que se envían por cada una de las redes a las que está conectado, independiente de cual sea la dirección de destino.

Además de reenviar los frames, el bridge extrae de cada uno de ellos la dirección de origen y la dirección de destino. La de origen la anota en una tabla hash correspondiente a la LAN por la que ha llegado el frame, y la de destino la busca en la misma tabla. Suponiendo

que el bridge recibe por la interfaz LAN1 un frame que lleva la dirección de origen A y la dirección de destino B. En primer lugar, el bridge actualizará su tabla de direcciones de LAN1 añadiendo A y después buscará en su tabla si en la columna LAN1 aparece B. Si es así sencillamente descartará el frame, ya que sabe que A y B están ambos en LAN1 y no hay ninguna necesidad de reenviar ese frame. Por el contrario, si B no aparece en la tabla de LAN1 el puente reenviará el frame a LAN2. Es posible que B esté en LAN1 y el bridge no lo sepa, porque B no haya enviado aún ningún frame, pero ante la duda, el bridge reenvía el frame por la otra interfaz. Esta estrategia de tirar por elevación enviando la información en caso de duda se denomina inundación (flooding).

Este mecanismo de aprendizaje tiene algunas consecuencias que vele la pena destacar: un nodo que no emita ningún frame, no puede ser localizado, por lo que un bridge enviará por todas sus interfaces los frames dirigidos a dicho nodo. Los frames enviados a direcciones multicast o broadcast siempre son retransmitidos por todas las interfaces, ya que en principio puede haber destinatarios en cualquier parte. Se deduce entonces que los bridges no almacenan direcciones multicast en sus tablas.

Como una forma de adaptarse a cambios de topología en la red, las entradas en las tablas de direcciones son eliminadas cuando han pasado varios minutos sin que la dirección correspondiente haya enviado datos.

Existen bridges multipuerta, es decir, con múltiples interfaces, que permiten interconectar varias LANs en un mismo equipo. El algoritmo en estos casos es similar, salvo que se mantiene una tabla de direcciones para cada interfaz. Las tablas se van llenando con las direcciones escuchadas en cada interfaz. Cuando se recibe un frame en cualquiera de las interfaces se busca la dirección de destino en la columna de dicha interfaz, y si el destinatario se encuentra allí, el frame simplemente se descarta, si no se busca en las columnas correspondientes a las demás interfaces. Si se encuentra en alguna columna se manda a la interfaz correspondiente. Por último, si no se encuentra en ninguna de las tablas se envía a todas las interfaces excepto aquella por la que llegó (flooding).

Los bridges han de mantener una tabla de direcciones para cada una de sus puertas; la cantidad de memoria destinada a dichas tablas es limitada, y en redes grandes puede llegar a agotarse. Los fabricantes suelen especificar el número máximo de direcciones MAC que sus puentes son capaces de soportar.

En algunas situaciones es interesante unir dos LANs con más de un bridge. Si se tienen LAN1 y LAN2, unidas por los puentes B1 y B2, y un nodo en LAN1 emite un frame que lleva una dirección de destino F aún no registrada. Al no aparecer en sus tablas B1 reenvía el frame a LAN2, y lo mismo hace B2. A continuación, B1 detecta en LAN2 el frame generado por B2 y, al no encontrar en sus tablas la dirección de destino, ya que aún no se conoce la ubicación de F, lo reenvía a LAN1. Análogamente, B2 detecta en LAN2 el frame de B1 y al no estar F en su tabla de direcciones, lo reenvía a LAN1. LAN1 recibe dos copias de un frame que ya tenía, y se repite el proceso: B1 recoge el frame de B2 y lo retransmite. B2 hace lo mismo con el de B1, y así sucesivamente. El ciclo no tiene fin, por lo que un sólo frame es suficiente para saturar ambas redes.

Para evitar este problema existe un mecanismo que permite a los bridges comunicarse entre sí, pasándose información sobre la topología de las conexiones existentes. Una vez cono-

cida la topología los bridges desactivarán las conexiones redundantes para garantizar que haya un único camino uniendo todas las redes, de forma que se evite la creación de loops. Las conexiones que lógicamente se pongan fuera de servicio quedarán listas para entrar en funcionamiento si las conexiones activas fallan por algún motivo. El algoritmo se repite cada cierto tiempo, por lo que si alguno de los enlaces queda fuera de funcionamiento por algún motivo en la siguiente ronda se habilitará algún camino alternativo que lo sustituya. El protocolo que permite esto se conoce como Spanning Tree o como Spanning Tree Learning Bridge Protocol, y forma parte de la especificación IEEE 802.1D.

Puentes Remotos

En ocasiones existe la necesidad de conectar entre sí dos LANs remotas como si fueran la misma LAN. Para esto se usa un tipo de bridge denominados puentes remoto. El mecanismo básico de funcionamiento es el mismo que para los puentes locales, salvo que el puente está constituido por dos 'medios puentesínterconectados por una línea dedicada cuya velocidad típicamente suele estar entre 64 Kbps y 2048 Mbps. También se pueden unir los puentes remotos por redes X.25, Frame Relay o incluso radioenlaces.

El protocolo spanning tree también se utiliza en bridges remotos. Para representar topológicamente un bridge remoto el enlace punto a punto se debe considerar como una LAN con un bridge en cada extremo.

Dado que generalmente los bridges remotos se conectan mediante líneas de menor velocidad que las redes a las que enlazan, es frecuente que dicha conexión sea el factor limitante del desempeño de la red. Esto es especialmente crítico cuando se utilizan líneas de baja velocidad y más aún cuando se trata de LANs grandes en las que el tráfico broadcast/multicast es importante.

Puentes Traductores

Un puente traductor es aquel que interconecta dos redes que utilizan diferente protocolo MAC, por ejemplo Ethernet y Token Ring. La utilización de puentes traductores tiene diversos problemas entre los que se destacan los siguientes:

Reformateo del frame: la estructura del frame en ambas redes es diferente. Además de acomodar los datos a la nueva estructura es necesario recalcular el CRC.

Campos inexistentes: cuando se pasa de Token Ring a Ethernet los campos prioridad y reserva se pierden, y en el sentido inverso, al pasar de Ethernet a Token Ring es preciso asignar un valor arbitrario a los campos prioridad y reserva sin disponer de ninguna información al respecto.

Acuse de recibo: los bits A y C del campo Frame Status en Token Ring, que permiten indicar un acuse de recibo al nodo emisor, plantean un problema cuando el frame atraviesa el bridge. Si el bridge no acusa recibo es muy probable que el emisor reintente varias veces, hasta abandonar. Por el contrario, si el puente acusa recibo está mintiendo, ya que podría presentarse algún problema más tarde en el envío del frame a su destino y el nodo que envía el frame creerá que todo es correcto.

Formato de las direcciones MAC: aunque tanto Ethernet como Token Ring soportan las direcciones IEEE de 48 bits en Ethernet las direcciones MAC se transmiten enviando primero el bit menos significativo de cada byte, mientras que en Token Ring se transmite primero el bit más significativo. Los puentes entre ambas redes han de tener esto en cuenta para invertir el orden de los bits de cada byte cuando se transmite un frame. El tema se complica cuando aparecen direcciones MAC en la parte de datos del frame, como en los paquetes ARP y RARP utilizados en IP para la resolución de direcciones.

Diferente tamaño de frame máximo: el campo datos en el frame Ethernet tiene un tamaño máximo de 1500 bytes. En cambio en Token Ring está limitado únicamente por el token-holding time. Con el valor por defecto de 10 ms esto supone 5000 bytes a 4 Mbps y 20000 bytes a 16 Mbps.

De todos estos problemas el último es el más grave, y existen tres posibles soluciones, de las cuales normalmente se adopta la tercera.

1. Cada nodo de la red Token Ring limita a 1500 bytes el tamaño máximo de frame. Esto reduce el grado de transparencia de la red.

2. Los bridges segmentan los frames Token Ring superiores a 1500 bytes en frames menores, cada una con la misma dirección origen y dirección destino que el frame original. Esta función no forma parte del estándar 802.1D y no está soportada normalmente por los bridges, además aumenta el tiempo de proceso.

3. Las redes se unen mediante un router, en vez de un bridge. El router actúa al nivel de red y dispone de los recursos necesarios para extraer el paquete del frame, fragmentarlo si es preciso, y generar uno o varios frames en la otra red. El problema en este caso es que el router no puede ser transparente al tipo de protocolo utilizado en el nivel de red, que era una de las grandes ventajas de los puentes.

Puentes con Encaminamiento Desde el Origen

Aunque podrían utilizarse en cualquier LAN, los bridges con encaminamiento desde el origen se utilizan únicamente en Token Ring, y su estandarización está recogida en el IEEE 802.5. La idea consiste en que el host que genera el frame disponga de suficiente información sobre la topología de la red como para que pueda indicar la ruta que debe seguir el frame en todo su recorrido.

Para poder indicar la ruta, el emisor incluye la información pertinente en un campo adicional del frame, ubicado detrás del campo dirección origen. Se utiliza para ello una versión modificada del frame normal Token Ring. La presencia de información de routing se indica poniendo a 1 el primer bit de la dirección origen. La información de routing está formada por una secuencia de números de: bridge, LAN, bridge, LAN, etc. hasta llegar a la LAN de destino. Para esto, en la red Token Ring las LANs se numeran con direcciones de 12 bits únicas en toda la red, y los bridges con direcciones de 4 bits únicas en el contexto de las LANs que interconectan.

Un bridge con encaminamiento desde el origen descarta automáticamente todos los frames que no tienen en 1 el primer bit de la dirección origen, ya que se supone que estas no deben ser encaminadas. Para el resto, el bridge analiza la información de routing y busca del número de la LAN por la que le ha llegado. Si este número de LAN es seguido por su propio número de bridge, entonces reenvía el frame a la LAN que le sigue en la secuencia. Si no es así, entiende que el frame no va dirigido a él y lo descarta.

Para poder incluir la información de routing los hosts deben obtener información sobre la topología de la red. Para ello, cuando un nodo desea saber la ruta por la que puede llegar a otro cuyo destino le es desconocido envía una frame especial, llamado discovery frame, en todas direcciones, de forma que sea retransmitido por todos los bridges en todas las LANs. Eventualmente el frame llega a su destino por uno o más de los posibles caminos, y el nodo destino responde con un frame de acuse de recibo que viaja en orden inverso, pero esta vez cada puente por el que pasa anota en el frame de respuesta su propio número y el número de la LAN por el que lo emite. Al final del proceso la estación origen recibe uno o varios frames que le indican todas las rutas posibles hacia el destino especificado y elige la que considera óptima y la incluye en su tabla de rutas para poder utilizarla en posteriores envíos a dicha estación.

2.15. LAN Conmutadas

En la actualidad los bridges presentan un elevado número de interfaces, habiendo modelos que pueden llegar a tenr más de 500. Estos equipos suelen ir equipados con chips VLSI diseñados específicamente para este tipo de tareas lo que aumenta su desempeño. A estos bridges multipuerta de alta velocidad se les llama switches LAN, ya que gracias al redireccionamiento inteligente que realizan mediante sus tablas de direcciones actúan conmutando frames entre sus múltiples puertas. Aquellas LANs basadas en switches se las suele llamar LAN conmutadas.

En los switches Ethernet se suele decir que cada puerto constituye un *dominio de colisiones independiente*, ya que las colisiones que se producen en un puerto no afectan a los demás. Los switches Token Ring pueden funcionar según el principio de bridge con encaminamiento desde el origen o como bridge transparente.

Con switches LAN de elevado número de puertas es posible incrementar de forma notable la capacidad de una red local con una modificación mínima de la misma. Por ejemplo, si en una red se sustituye un hub Ethernet de 24 puertas por un switch LAN de 24 puertas, el rendimiento máximo teórico se ha multiplicado por 24 sin haber tenido que modificar el cableado o las tarjetas de red de los hosts.

En el caso de redes Ethernet el uso de switches tiene un efecto adicional en la mejora del rendimiento. Se debe recordar que en redes CSMA/CD la eficiencia disminuye a medida que aumenta el número de nodos, por lo que si se divide una red en varias mediante un switch se consigue un mejor rendimiento en cada una de ellas al soportar un número más reducido de equipos.

Un problema que se presenta con los switches LAN es que se pueden producir situaciones de congestión, para las que no disponen de muchos mecanismos de control pues funcionan

únicamente a nivel de enlace. Por ejemplo, si en un switch de puertas a 10 Mbps diferentes puertos quieren enviar tráfico a la vez a un mismo puerto a 10 Mbps cada uno y ésta situación se mantiene durante bastante tiempo el switch puede agotar el espacio en buffers disponible, perdiéndose frames a partir de ese momento. En el caso de Ethernet, este problema se resuelve mediante el control de flujo.

Al igual que los bridges, los switches LAN han de mantener en memoria las tablas de direcciones MAC en cada una de sus puertas. Las especificaciones de un switch LAN indican normalmente el número máximo de direcciones que puede soportar. Si el número de equipos activos en la red es superior a este número máximo el rendimiento de la red se ve afectado ya que las entradas en la tabla de direcciones expiran con demasiada rapidez, provocando tráfico adicional en la red debido al mecanismo de flooding.

2.15.1. Store and Forward vs Cut-Through

Dado que los switches LAN son esencialmente bridges con muchas puertas, su funcionamiento normal requiere que antes de retransmitir un frame lo reciban en su totalidad para que puedan comprobar el CRC y descartarlo en caso de que éste sea erróneo. Esto obliga a que los switches funcionen como dispositivos de almacenamiento y reenvío (Store and Forward). En el caso de frames grandes el requerimiento de comprobación del CRC introduce un retardo notable en la propagación del frame, a menudo superior al que introduce el propio proceso de conmutación. Además, si el frame ha de atravesar varios switches el almacenamiento y reenvío se ha de realizar en cada uno de ellos. Si no se hiciera la comprobación del CRC la velocidad de conmutación podría aumentarse notablemente y el retardo reducirse, ya que el switch podría empezar a enviar los bits tan pronto hubiera recibido la dirección MAC a la que va dirigido el frame. A este tipo de funcionamiento alternativo se le conoce como funcionamiento en modo Cut-Through. El problema del Cut-Through es que se pueden estar produciendo errores de CRC que pasen inadvertidos, y en este caso los frames erróneas serán descartadas por el host de destino, por lo que no hay riesgo de que se interpreten como correctos datos erróneos, pero aun así la situación es perjudicial para la red puesto que se esta ocupando ancho de banda con tráfico inútil.

Para evitar este problema cuando los switches funcionan en modo Cut-Through siguen comprobando el CRC, y cuando se produce un error no pueden descartar el frame, puesto que ya ha sido transmitido, pero sí pueden poner al nodo emisor bajo sospecha y a partir de ese momento pasar a funcionar en modo almacenamiento y reenvío de forma selectiva, únicamente para los frames que tengan como dirección de origen la del nodo sospechoso. Si se comprueba más tarde que el error detectado fue algo esporádico se volverá al modo Cut-Through, de lo contrario se mantendrá el estado de vigilia.

2.15.2. LANs virtuales o VLANs

Una de las grandes virtudes de bridges y switches es su sencillez de manejo. Debido a su funcionamiento transparente es posible realizar una compleja red, incluso con enlaces WAN si se utilizan bridges remotos, sin tener que configurar ningún router. A fines de los ochenta se

puso de moda la idea de desarrollar grandes redes, incluso a nivel de redes nacionales, basadas únicamente en el uso de puentes transparentes.

Sin embargo pronto se vio que esta estrategia tenía dos inconvenientes serios:

- Los bridges propagan el tráfico broadcast y multicast. Generalmente los protocolos orientados a redes locales hacen un uso exhaustivo de este tipo de frames, especialmente las broadcast, para anunciar todo tipo de servicios. Incluso IP, emplea broadcasting para la resolución de direcciones. La proliferación de tráfico broadcast en una red es especialmente grave más que por el ancho de banda desperdiciado por el consumo de ciclos de CPU que se produce en todos los nodos de la red. Este no es el caso con los frames multicast, ya que cuando un frame multicast no incumbe a una estación es descartado por la interfaz.

- La transparencia de los bridges hace difícil establecer mecanismos de control, protección y filtrado de tráfico, por lo que las redes muy grandes basadas en bridges se hacen inmanejables. Además, en los casos en que se requieren controles o mecanismos de administración se han de utilizar direcciones MAC que no tienen ningún prefijo común que permita referirse o identificar una parte de la red, ya que la asignación no ha seguido ningún criterio geográfico ni corresponde con la topología de la red.

Como consecuencia de esto la creación de grandes redes locales está desaconsejada y es práctica habitual en estos casos separar mediante routers las diversas partes de la red. Este es el caso en un campus o gran edificio. Los routers, al actuar a nivel de red, aíslan los frames broadcast y multicast y facilitan la administración al agregar las direcciones de nivel de red.

Dividir una red local con criterios geográficos resulta relativamente sencillo, ya que normalmente la topología del cableado permite realizar esa división de manera directa. Por ejemplo, si se quiere dividir en varias una red local que abarca el campus de una universidad, se puede crear una LAN por edificio con switches en cada edificio e interconectar cada edificio a una interfaz diferente de un router que interconecte todo el campus. Sin embargo, a menudo se requiere realizar una división lógica de acuerdo a criterios funcionales, que no siempre coinciden con la ubicación física. Por ejemplo, en el caso de una universidad se podría pensar por razones de eficiencia y seguridad en crear una red para investigación, otra para docencia y otra para tareas administrativas. Normalmente habrá varios edificios en los que habrá que dotar una serie de puestos de cada una de las tres redes mencionadas, en cuyo caso habría que instalar en los correspondientes armarios de cableado switches independientes e interconectarlos entre sí por separado. Esto provoca una red compleja y muy cara, ya que en muchos casos habrá equipos infrautilizados.

La solución a este problema es la creación de redes locales virtuales, o VLANs. Las VLANs son una forma de realizar una partición lógica de un switch en otros más pequeños, de forma que aunque se trata de un solo equipo, se dividen los puertos en grupos que son completamente independientes entre sí. Esta funcionalidad está disponible hoy en día en la mayoría de los switches del mercado.

Suponiendo el caso anterior, que se ha decidido dividir la red de campus en tres VLANs: I (de investigación), D (de docencia) y A (de administración). Se tiene un switch de 16 puertas

en un closet de cableado y se plantea la necesidad de suministrar servicio a 4 equipos de la VLAN I, 4 de la D y 4 de la A. Se podría asignar, por ejemplo, los puertos 1 a 4 a la VLAN I, 5 a 8 a la VLAN D y 9 a 12 a la VLAN A, dejando los puertos 13 a 16 libres para futuras ampliaciones. A partir de ese momento, el switch se comportará como cuatro switches virtuales de 4 puertos cada uno, los correspondientes a las tres VLANs y un cuatro correspondiente a los puertos no asignados. De esta forma, se puede asignar puertos a una u otra VLAN de forma flexible en función de las necesidades.

Queda por resolver aún la conexión de las tres VLANs con el resto de la red. Una posibilidad sería asignar los puertos 13, 14 y 15 a cada una de las tres VLANs y conectarlos con a tres puertos del switch principal del edificio, asignados a las tres VLANs. Siguiendo este sistema, se llegaría al router del campus donde un switch con puertos en las tres VLANs se conectaría a tres interfaces físicas diferentes del router. Aunque físicamente las tres VLANs comparten los switches, sigue habiendo tres redes separadas en el cableado, ya que nunca viajan por un mismo cable frames de VLANs diferentes.

Cabre pensar en un nivel adicional de optimización en el que se compartiera un mismo cable para diferentes VLANs. Esto permitiría un ahorro considerable en el número de puertos consumidos en los enlaces troncales, especialmente cuando se manejan muchas VLANs. Por ejemplo, se podríamos emplear solo un puerto, digamos el 13, para conectar las tres VLANs, liberando así los puertos 14 y 15 para otros usos. Esto se denomina configurar un enlace trunk o troncal. Como es lógico los enlaces Trunk suelen ser de mayor capacidad que los puertos normales del switch ya que soportan un tráfico más elevado. Por ejemplo, en un switch de puertos a 10 Mbps el enlace trunk típicamente será de 100 Mbps y en uno con puertos de 100 Mbps será de Gigabit Ethernet.

Los enlaces Trunk suponen un cambio importante en el funcionamiento de los conmutadores, ya que al mezclar frames de diferentes VLANs por el mismo cable es preciso marcarlas o etiquetarlas de alguna manera a fin de poder entregarlas a la VLAN adecuada en el otro extremo. El marcado se hace añadiendo un campo nuevo en el header del frame MAC, lo que hace que el tamaño del frame Ethernet supere ligeramente la longitud máxima de 1500 bytes en algunos casos, ya que un switch puede recibir un frame de 1500 bytes y si lo ha de enviar por un enlace trunk tendrá que incorporarle la etiqueta correspondiente, pues en ningún caso está permitido fragmentar el frame original. Hoy en día existe un formato estándar para colocar las etiquetas de VLAN que es el conocido como IEEE 802.1q que es el que utilizan prácticamente la totalidad de los equipos actuales. De esta forma es posible diseñar complejas redes con VLANs utilizando equipos de diferentes fabricantes

Una propiedad interesante de las VLANs es la posibilidad de configurar interfaces virtuales en los hosts. Suponiendo que en el caso analizado con tres VLANs, I, D y A, se tiene un servidor que se desea esté accesible de forma directa en las tres VLANs, de forma que cualquier host de cualquiera de las VLANs pueda acceder a él sin necesidad de pasar por un router. Una posible solución sería conectar al servidor mediante tres interfaces de red y conectar cada una de ellas a un puerto del switch asignado a cada una de las VLANs. Cada interfaz recibiría una dirección de red correspondiente a la VLAN en la que se encuentra. Sin embargo, esta solución se hace inmanejable si aumenta el número de VLANs. Otra posibilidad, más interesante, sería configurar una interfaz de red del servidor como tres interfaces virtuales y conectarla a un

puerto trunk del switch. Para esto se necesita disponer de drivers con soporte de IEEE 802.1q para la interfaz de red y el sistema operativo que se esté utilizando.

3. Nivel de Red

3.1. Introducción

El principal objetivo de la capa de red es rutear, encaminar o dirigir los paquetes desde el origen al destino. Ésta es la única capa que "ve" y conoce la topología de la red, y está formada por dos tipos de nodos:

Nodos terminales: generan o reciben paquetes de otros nodos, nunca rutean paquetes dirigidos a terceros.

Nodos intermedios o de ruteo: se utilizan para rutear paquetes entre los nodos terminales. Suelen ser arquitecturas dedicadas y diseñadas específicamente para esa función, con sistemas operativos en tiempo real, aunque en ocasiones también se utilizan para desempeñar esta función computadores normales.

Cuadro 5: Denominación de los tipos de nodos en diferentes redes

Tipos de nodos	Internet (IP)	X.25	ATM	ISO
Nodo terminal	Host	DTE	Host	End System (ES)
Nodo intermedio o de ruteo	Router	DCE	Switches	Intermediate System (IS)

La terminología de los dos tipos de nodos es muy diversa y varía según el tipo de red y la "cultura" de que se trate. Aunque la terminología no se puede dividir de forma estricta la tabla 5 refleja las denominaciones más características en algunos de los casos más habituales.

Dado que la función de los nodos intermedios es interconectar redes, normalmente tienen varias interfaces físicas, y los nodos terminales normalmente una. En cada interfaz física de un nodo funciona una instancia independiente del nivel físico y del nivel de enlace, y por el contrario, el nivel de red es normalmente global para todo el nodo. En IP cada interfaz física tiene normalmente una dirección de red al menos, pudiendo tener varias.

En una LAN Ethernet, todos los nodos terminales se comunican directamente entre sí, sin necesidad de nodos intermedios, por lo que la capa de red es prácticamente inexistente. Esto incluye el caso en que la LAN incluya bridges o switches de cualquier tipo. Debido a esto, el nivel de enlace tiene una complejidad mayor. Los bridges MAC, en especial los de encaminamiento desde el origen, desempeñan una función hasta cierto punto equivalente a la de un router.

Los servicios que ofrece el nivel de red deberán en lo posible aislar al nivel de transporte de detalles tales como tipo de tecnología física utilizada (LAN, WAN, broadcast, etc.), número y topología de las subredes, etc. Las direcciones de red deberán tener un formato homogéneo, cualquiera sea el medio físico o subred utilizados.

Los servicios de red pueden ser orientados a conexión o no orientados a conexión. Ejemplos de servicios no orientados son el protocolo IP y el ISO CLNS, elaborado a imagen y semejanza del IP. Ejemplos de servicios orientados a la conexión son ATM, X.25 o Frame Relay. En una

red no orientada a la conexión, el nivel de transporte es normalmente más complejo pues ha de desempeñar más funciones que en una red orientada a la conexión.

3.2. Algoritmos de Ruteo

La función fundamental de la capa de red es averiguar por que interfaz se han de enviar los paquetes recibidos. Con redes basadas en datagramas esta decisión se toma para cada paquete y el nodo que la realiza se denomina *router*. Con redes orientadas a conexión, que son las basadas en circuitos virtuales, la decisión se toma en el momento de establecer el circuito virtual, y a partir de entonces sólo conmutan paquetes.

El mecanismo que permite elegir la ruta a utilizar es lo que se denomina *algoritmo de ruteo*, el que debiera ser óptimo y justo. Estos conceptos a veces se contraponen, ya que el algoritmo que permite un aprovechamiento óptimo de los recursos no siempre es el que ofrece el reparto más equitativo.

Los algoritmos de ruteo se dividen en dos grupos: estáticos y dinámicos. Los algoritmos estáticos toman las decisiones utilizando información previamente recopilada sobre el estado de la red, en cambio los algoritmos dinámicos utilizan información recopilada en tiempo real sobre el estado de la red que se actualiza constantemente mediante paquetes que intercambian los routers a través de la misma red.

En el ruteo estático las rutas se fijan en función de la capacidad de la línea, el tráfico medio u otros criterios similares. En cada router se cargan las tablas de rutas de forma estática, por lo que no necesita intercambiar información con sus vecinos, y por tanto, no se requiere un protocolo de ruteo. Con el ruteo estático no es posible responder a situaciones cambiantes, por ejemplo, saturación, exceso de tráfico o fallo de una línea. Al realizar los cálculos de las rutas óptimas en diferido es posible aplicar algoritmos sofisticados, aun cuando requieran gran cantidad de recursos de cálculo o de memoria.

En el ruteo dinámico, las rutas óptimas se recalculan continuamente en función de la información que los routers reciben en tiempo real sobre el estado de la red. Se utilizan algoritmos autoadaptativos y es preciso utilizar un protocolo de routing que permita a los routers intercambiar continuamente esa información. Los algoritmos no pueden ser demasiado complejos pues han de implementarse en los routers y ejecutarse en tiempo real con los recursos de CPU y memoria de que el router dispone.

3.2.1. El Principio de Optimalidad

```
Si B está en la ruta óptima de A a C, entonces el camino óptimo
de B a C está incluido en dicha ruta.
```

Una consecuencia importante de este principio es que todas las rutas óptimas para llegar a un punto determinado forman un árbol con raíz en el punto de destino. Si el árbol no contiene loops, se dice que es un spanning tree y siempre es posible llegar al punto de destino en un número finito de saltos o hops.

3.2.2. Ruteo por el Camino Más Corto y Métricas

Existen diversos algoritmos que permiten calcular el camino más corto entre dos nodos de un grafo. Uno de los más conocidos es el *algoritmo de Dijkstra*, y se utiliza tanto en routing estático como dinámico.

Para saber elegir el camino más corto primero se debe definir que se entiende por distancia. En los casos más simples la distancia se mide como el número de saltos o hops, donde, a mayor número de saltos mayor distancia. Evidentemente esto es satisfactorio únicamente en casos muy simples en que todos los enlaces tiene la misma capacidad. Normalmente la distancia se mide como una combinación de los siguientes factores:

- El inverso de la capacidad del enlace (información estática).
- Tráfico medio (puede ser información estática o dinámica).
- Retardo (información dinámica medida a partir de los paquetes enviados).
- El inverso de la confiabilidad (información dinámica medida a partir de los paquetes enviados).

El peso relativo que se da a cada uno de los factores que intervienen en el cálculo de la distancia en una red se denomina métrica. La métrica puede ser fijada o modificada al configurar los router, aunque los parámetros que entran en juego y la fórmula que se utiliza para calcularla suelen estar muy relacionados con el algoritmo y el protocolo de routing utilizados.

Cuando un trayecto está compuesto por varios tramos la longitud o distancia del trayecto será igual a la suma de las distancias de los tramos que lo componen.

Cuando el parámetro utilizado para el cálculo de la distancia es invariante en el tiempo, por ejemplo velocidad del enlace, puede aplicarse a un algoritmo de ruteo estático. Se podría cargar en un computador toda la información sobre la topología de la red y calcular las rutas óptimas para cada caso, y una vez obtenidas éstas se cargarían en todos los routers de la red.

Si se emplean además parámetros dinámicos como retardo o confiabilidad, puede utilizarse un algoritmo dinámico. En este caso la información se propaga en toda la red y los cálculos se hacen de manera descentralizada entre todos los routers.

3.2.3. Ruteo Basado en el Flujo

Este algoritmo toma en cuenta la cantidad de tráfico medio que soportan las líneas, y en base a esta información intenta optimizar el conjunto de las rutas para utilizar el camino menos congestionado en cada caso.

Para aplicarlo se ha de conocer bastante bien el tráfico, y éste ha de ser muy regular. Se pueden aplicar algoritmos relativamente sofisticados ya que el cálculo de rutas se hace offline y se carga en el router después. Este algoritmo sólo se aplica en algunos casos de routing estático. Puede ser útil para diseñar la topología de una red. Por ejemplo, si se conectan una serie de oficinas y se dispone de la matriz de tráfico previsto entre cada par de oficinas se pueden plantear diversas topologías y estudiar cuál es la más adecuada.

3.2.4. Flooding

La inundación o flooding consiste en enviar cada paquete por todas las interfaces, excepto por la que se ha recibido. Ésta técnica se aplica en los frames de descubrimiento en los bridges por encaminamiento desde el origen y los transparentes cuando la dirección de destino era desconocida.

La inundación puede multiplicar el tráfico si existen loops en la topología, ya que en ese caso se envían paquetes duplicados. Para limitar este problema se fija un número máximo de saltos, que suele ser igual al número de saltos que hay entre los dos puntos más alejados de la red. Otra posibilidad es identificar los paquetes, por ejemplo numerándolos, para que cada router mantenga una lista de los paquetes enviados y así puede evitar reenviarlos de nuevo. También puede usarse flooding selectivo en el que el paquete se envía sólo por las líneas que aproximadamente apuntan en la dirección correcta. Flooding se utiliza en algunos algoritmos de routing multicast.

3.2.5. Ruteo por Vector de Distancia

Este algoritmo se aplica en diversos protocolos de routing. También se conoce como algoritmo de Bellman-Ford o Ford-Fulkerson, que fueron los autores de la idea. Fue el algoritmo original de ARPANET, se utilizó en DECNET, IPX y Appletalk. Se usa en el protocolo RIP, que hasta 1988 era el único protocolo de ruteo utilizado en Internet. También se utiliza en los protocolos propietarios IGRP y EIGRP de Cisco.

En el ruteo por vector distancia cada router mantiene una tabla o vector que le indica la distancia mínima conocida hacia cada posible destino y que línea o interfaz debe utilizar para llegar a él. La tabla se actualiza regularmente con información obtenida de los routers vecinos. Cada router envía la tabla completa de distancias a todos sus vecinos, y sólo a ellos. Con la información que tiene y la recibida de sus vecinos cada router recalcula continuamente su tabla de distancias.

La métrica puede ser número de saltos, retardo, paquetes encolados, etc. o una combinación de estos u otros parámetros. Para medir el retardo el router envía paquetes de prueba que deben ser respondidos por el router remoto. Cada router sólo mide el retardo con sus vecinos, los retardos y distancias a routers más lejanos los calcula como suma de la distancia a sus vecinos más la información que éstos le facilitan.

3.2.6. Ruteo por Estado del Enlace

El ruteo basado en el estado del enlace apareció como un intento de resolver los problemas que planteaba el ruteo por vector distancia, fundamentalmente el problema de la cuenta a infinito. Se trata de un algoritmo más sofisticado y robusto, compuesto por cuatro fases:

1. Descubrir los routers vecinos y averiguar sus direcciones. Esto se hace mediante el envío de paquetes HELLO por todas sus interfaces. Los paquetes HELLO son respondidos con mensajes que identifican a los routers que los reciben.

2. Medir el retardo o costo de llegar a cada vecino. Para esto, se envían paquetes de ECHO que son respondidos por el router remoto y miden el tiempo de ida y vuelta.

3. Construir un paquete que resuma toda esta información, y enviarlo a todos los routers de la red. Se utiliza flooding, y los paquetes se numeran para detectar y descartar duplicados, e ignorar paquetes obsoletos. Además cada paquete tiene una vida limitada, al cabo de la cual es descartado.

4. Calcular el camino más corto a cada router. Con toda la información obtenida el router construye el árbol de expansión de las rutas óptimas a cada destino de la red aplicando el algoritmo de Dijkstra. De esta forma conoce la topología de la red.

Entre los protocolos de routing que utilizan algoritmos basados en el estado del enlace destaca OSPF (Open Shortest Path First) que es el protocolo de ruteo estándar de Internet. Otro protocolo de estado del enlace también utilizado en Internet y que proviene de OSI es IS-IS (Intermediate System-Intermediate System). IS-IS es multiprotocolo, es decir, soporta múltiples protocolos de red por encima. OSPF esta basado en IS-IS, pero no es multiprotocolo.

En el routing por el vector distancia cada router envía información sólo a sus vecinos, pero esta información incluye a todos los nodos de la red. En cambio en el routing por el estado del enlace cada router envía su paquete de información a toda la red, pero éste solo contiene la relativa a sus vecinos más próximos. En el estado del enlace cada router puede, a partir de la información obtenida, conocer su árbol de expansión completo, mientras que esto no es posible con routing por el vector distancia.

3.2.7. Ruteo Jerárquico

A medida que una red crece la cantidad información de routing aumenta de forma exponencial, ya que cada router ha de calcular las rutas óptimas a todos los demás. Esto incrementa el tráfico, la memoria en los routers, y la complejidad de los cálculos necesarios para obtener las rutas óptimas. Como consecuencia de esto los algoritmos de routing no son escalables.

Para reducir este problema las redes se organizan en niveles jerárquicos. Se divide la red en regiones o sistemas autónomos, y sólo un número reducido de routers de cada región se puede comunicar con el exterior. Las rutas quizá no sean tan óptimas, pero se simplifica la administración y mantención de las tablas y se reduce el tráfico de la red.

3.2.8. Ruteo Broadcast

En algunos casos se necesita enviar un paquete a todos los destinos posibles en una red, es decir se quiere hacer un envío broadcast. Esto puede hacerse por flooding, pero se pueden producir loops en la red. Para evitarlo se suele poner al paquete un contador de saltos con un límite igual al diámetro de la red.

Otro método es el ruteo multidestino, que consiste en enviar un único paquete con todas las direcciones de destino. El paquete es replicado en cada router por las interfaces por donde debe enviarse, es decir, las que son parte de la mejor ruta para alguno de los destinos indicados.

Otro algoritmo es construir el árbol de expansión o spanning tree con raíz en el origen y seguirlo, replicando el paquete donde haya una bifurcación. El sistema es óptimo, ya que se asegura que la distribución se hará generando el número mínimo de paquetes y sin envíos duplicados, pero esto requiere que cada router conozca cuales de sus interfaces forman parte del spanning tree para el router origen y cuales no, es decir, los routers han de conocer en detalle la topología de la red.

Por último, el algoritmo de ruteo por el camino inverso es una aproximación al spanning tree cuando la información sobre la topología no está disponible. El mecanismo es el siguiente: el router examina la dirección origen del paquete recibido, y la interfaz por la que le ha llegado. Si esa interfaz es el camino más corto para llegar a esa dirección es bastante probable que el paquete no sea un duplicado, por lo que lo reenviará por todas las interfaces excepto por aquella por la que vino. Si no lo es, el paquete se descarta pues es muy probable que sea un duplicado. Esta técnica evita que se produzcan loops y consigue una eficiencia bastante buena, aunque no es óptima ya que se generan algunos envíos duplicados.

3.2.9. Ruteo Multicast

Para el envío de paquetes multicast primero hay que crear el grupo multicast. Una vez creado el grupo, los usuarios pueden unirse al él o abandonarlo. Cuando un usuario se une a un grupo multicast debe informar a su router y éste a sus vecinos.

En una LAN el envío de paquetes multicast no plantea ningún problema desde el punto de vista del routing, ya que simplemente se envían los paquetes a la red y serán captados por aquellas estaciones que pertenezcan al grupo. En una WAN, cada router que quiera enviar paquetes multicast ha de construir el spanning tree de toda la subred, colocándose él como raíz. A continuación, deja sólo las ramas necesarias para hacer llegar los paquetes a los routers que forman parte del grupo multicast.

Si se utiliza ruteo de estado de enlace cada router conoce la topología de la red, por lo que puede generar el árbol sin problemas, de abajo hacia arriba. Con vector de distancia se puede utilizar ruteo por camino inverso.

Existen dos estrategias posibles para construir el árbol de expansión de una distribución multicast. La primera, conocida como *modo denso* la emisión multicast se envía en principio a todas las ramas del árbol y ésta se va limitando a algunas ramas sólo a medida que los routers lo solicitan. La segunda, denominada *modo disperso* consiste en que la emisión solo se realiza por aquellas ramas cuyos routers lo han solicitado. La segunda estrategia es más conveniente cuando el número de miembros es reducido en proporción a las ramas del árbol. Para que el modo disperso pueda funcionar correctamente es preciso que haya un servicio de anuncio de sesiones disponibles de forma que los que no reciben la emisión multicast sepan de su existencia.

3.3. Algoritmos de Control de Congestión

Congestión: situación en la que el rendimiento de la red, o una parte de ella, se degrada debido a la presencia de tráfico excesivo.

No se debe confundir la congestión con el control de flujo. A diferencia de la congestión, el control de flujo es una circunstancia que sólo puede darse en conexiones punto a punto, es decir, a nivel de enlace o a nivel de transporte. Una de las posibles consecuencias del control de congestión es ejercer control de flujo sobre él o los nodos que están produciendo la congestión.

La congestión es generalmente un problema más complejo que el control de flujo, ya que generalmente el emisor del tráfico es un router, es decir un intermediario que lo más que puede hacer es reenviar el mensaje de control de congestión hacia atrás. Generalmente, cuando el mensaje llega al verdadero causante de la congestión el tráfico ya ha cesado y resulta inútil tomar medidas. Este problema se acentúa especialmente en redes de alta velocidad y elevado retardo (gran distancia).

3.3.1. Principios Generales del Control de Congestión

Para el control de la congestión caben dos planteamientos:

- Diseñar las cosas desde el principio para que la congestión no pueda llegar a ocurrir.
- Tomar medidas que permitan detectar la congestión y adoptar medidas correctoras en su caso.

La primera técnica es más segura, pero puede provocar ineficiencias si se aplican las limitaciones con demasiada severidad. La segunda permite aprovechar mejor la red, pero en caso de congestión puede ser difícil controlar la situación.

Entre los parámetros que permiten detectar la presencia de congestión, a nivel de red, se encuentran:

- Porcentaje de paquetes descartados.
- Longitud media de las colas en las interfaces de los routers.

En cambio, a nivel de transporte se tiene:

- Retardo medio por TPDU (Transport Protocol Data Unit).
- Desviación media del retardo por TPDU (jitter).
- Número de TPDUs que se pierden o llegan al timeout y se retransmiten (se supone que esto no se debe a errores).

Para informar sobre situaciones de congestión el receptor puede utilizar paquetes de alerta y el emisor enviar paquetes de sondeo para averiguar el estado de la red.

Para resolver la congestión solo hay dos posibles medidas:

- Reducir el tráfico solicitando al emisor que pare de transmitir, o que busque rutas alternativas.
- Aumentar la capacidad.

3.3.2. Factores que Influyen en la Congestión

Entre los factores, a nivel de enlace, que pueden influir en la congestión se encuentran:

- El intervalo de timeout, ya que si es pequeño originará retransmisiones innecesarias.
- El tamaño de ventana. Si es grande es más fácil que se produzca congestión.
- El uso de retroceso n o repetición selectiva. El retroceso n genera más tráfico.
- El uso o no de ACK piggybacked. Si no se usa se genera más tráfico.

En el nivel de red, los factores que influyen en la congestión son los siguientes:

- Uso de circuitos virtuales o datagramas. Existe mejor control de congestión cuando se trata de circuitos virtuales.
- Uso de mecanismos de encolamiento y criterios de selección o prioridades.
- Uso de mecanismos de descarte de paquetes.
- Algoritmo de ruteo.
- Vida media de los paquetes o timeout. Si es muy pequeña, tendrán que ser retransmitidos, si es excesiva terminarán siendo inútiles.

En el nivel de transporte se dan básicamente los mismos factores que en el nivel de enlace, la principal diferencia es que la estimación del timeout adecuado es mucho más difícil al no ser una comunicación entre entidades vecinas.

3.3.3. Traffic Shaping y Traffic Policing

El tráfico de ráfagas o bursty es la principal causa de congestión. Si todos los nodos transmitieran siempre un flujo constante sería muy fácil evitar las congestiones.

En el traffic shaping, se establece márgenes máximos al tráfico a ráfagas. Suelen utilizarse para fijar una calidad de servicio o QoS entre el operador y el usuario. Mientras el usuario respete lo establecido, el operador se compromete a no descartar paquetes. El perfil de tráfico actúa entonces como una especie de contrato entre las partes.

El traffic policing corresponde a una labor de monitoreo o seguimiento del tráfico introducido por el usuario en la red para verificar que no excede el perfil pactado.

Uno de los sistemas mas utilizados para establecer perfiles de tráfico es el algoritmo de *leaky bucket*. El host puede enviar ráfagas que son almacenadas en un buffer de la interfaz, la que envía a la red un flujo constante de salida. Si la ráfaga es de tal intensidad o duración que el buffer se llena, los paquetes excedentes son descartados, o bien son enviados a la red con una marca especial que les identifica como de segunda clase. Estos paquetes serán los primeros candidatos a descarte en caso de congestión. Esta técnica se utiliza en ATM y en Frame Relay y se está proponiendo su introducción en IP. Para definir el algoritmo se utilizan dos parámetros, el flujo r con que sale el flujo a la red, y la capacidad del buffer C.

Un nodo que esté continuamente transmitiendo con un flujo r mantendrá vacío su buffer y a la hora de enviar una ráfaga estará en igualdad de condiciones respecto a otro nodo que no haya transmitido nada durante cierto tiempo. El algoritmo *token bucket* es una versión mejorada del anterior que compensa al host que alterna intervalos de tráfico con otros de inactividad frente al que esta siempre transmitiendo. El mecanismo que sigue para ello es el siguiente: cuando el host no envía datos el pozal va sumando créditos o tokens hasta un máximo igual a la capacidad del buffer. Los créditos acumulados pueden utilizarse después para enviar ráfagas con un flujo M mayor de lo normal. Cuando se agotan los créditos, el flujo vuelve a su valor normal r y el algoritmo funciona como un leaky bucket. Los parámetros que definen este algoritmo son el flujo normal r, la capacidad del buffer C y el flujo máximo M que normalmente igual a la velocidad de la interfaz física.

3.3.4. Control de Admisión

El control de admisión se aplica únicamente a las redes orientadas a conexión y consiste en evitar el establecimiento de nuevos circuitos virtuales que pasen por una zona de la red que se considera congestionada. Si se conoce la capacidad máxima que necesita cada circuito virtual en principio es posible controlar el acceso de forma que nunca se produzca congestión. Generalmente un control de admisión estricto no usa los recursos de manera eficiente ya que al reservar para cada circuito la capacidad máxima la red se encuentra subutilizada la mayor parte del tiempo.

3.3.5. Choke Packets

Los choke packets o paquetes de asfixia se puede aplicar tanto en redes de circuitos virtuales como de datagramas. En esta técnica el router comprueba regularmente cada una de sus líneas, monitoreando, por ejemplo, el grado de utilización, la longitud de la cola o la ocupación del buffer correspondiente. Cuando el parámetro inspeccionado supera un determinado valor considerado umbral de peligro se envía un choke packet al host considerado culpable para que reduzca el ritmo.

Los paquetes de asfixia tienen el riesgo de producir comportamientos oscilatorios, ya que cuando se percibe una situación peligrosa se envían avisos que pueden bloquear a todos los emisores. Al detectar que el problema está resuelto se pide reanudar los envíos, con lo que hay riesgo de caer nuevamente en la situación de alerta, y así sucesivamente.

Normalmente los paquetes de asfixia se envían a los hosts que generan el tráfico, ya que son éstos y no los routers los verdaderos causantes de la congestión. Los hosts cuando reciben estos paquetes suelen reducir, típicamente a la mitad, la velocidad con la que envían datos a la red.

3.3.6. Descarte de Paquetes

El último recurso para resolver un problema de congestión es descartar paquetes. En ocasiones los paquetes llevan alguna indicación de su grado de importancia, en cuyo caso los

routers intentan descartar los menos importantes primero. Por ejemplo, sería bastante grave si un router para resolver una situación de congestión descartara paquetes de asfixia.

A veces el nivel de aplicación puede dar información sobre la prioridad de descarte de los paquetes. Por ejemplo, en aplicaciones de tiempo real suele ser preferible descartar el paquete viejo al nuevo ya que el viejo seguramente es inútil. Por el contrario, en la transferencia de archivos el receptor necesita recibirlos todos y el más antiguo causará antes la retransmisión por timeout.

En algunos casos el paquete transmitido por la red es parte de una secuencia correspondiente a otro paquete de mayor tamaño que viaja fragmentado. En estos casos si se descarta un paquete cualquiera de una secuencia se tendrá que reenviar todo el grupo, por lo que al descartar uno es conveniente descartar todos los demás ya que son tráfico inútil.

3.4. El Protocolo IP

Internet es un conjunto de redes diferentes que comparten una pila de protocolos comunes. Cada una de estas redes es administrada por una entidad diferente: universidades, redes académicas nacionales, ISPs (Internet Service Providers), operadores, empresas multinacionales, etc. Como consecuencia de esto las políticas de uso son muy variadas.

Técnicamente a nivel de red Internet puede definirse como un conjunto de redes o sistemas autónomos conectados entre sí que utilizan el protocolo de red IP. IP es una red de *datagramas*, no orientada a conexión, con servicio "best effort", es decir, no ofrece QoS. La entrega de los paquetes no está garantizada ya que en momentos de congestión éstos pueden ser descartados sin previo aviso por los routers que se encuentren en el trayecto.

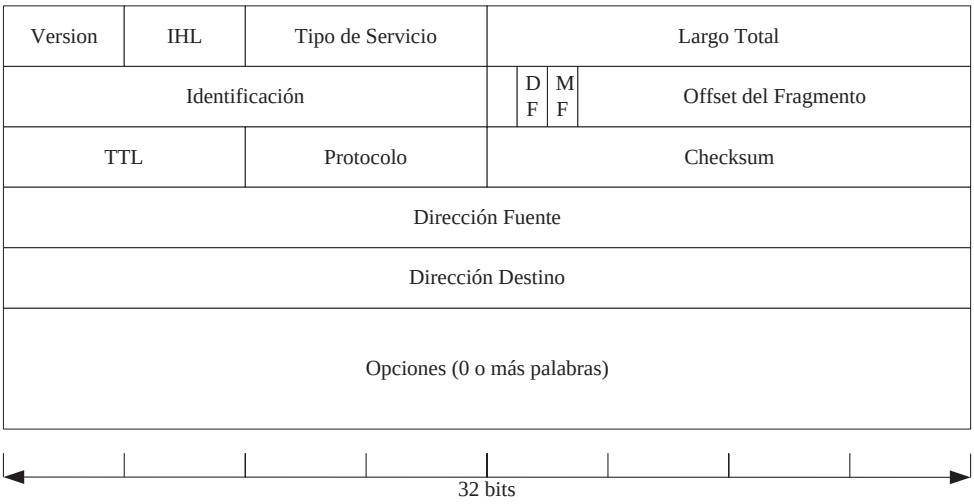

Figura 28: Encabezado del Paquete IP.

En una red IP toda la información viaja en paquetes o datagramas IP. Esto es: cualquier información de control que tenga que intercambiarse (routing dinámico, mensajes de error,

etc.) y los datos de nivel superior.

El paquete IP (ver figura 28) tiene dos partes: encabezado y texto. El encabezado tiene una parte fija de 20 bytes y una opcional de entre 0 y 40 bytes, pero siempre es un múltiplo de 4. Los campos son los siguientes:

Versión: 4 bits que permiten coificar los valores de las distintas versiones de IP. La versión actualmente es la 4, pero se empezó ya a extender el uso de la versión 6 con una estructura de paquete diferente a la de la figura.

IHL: 4 bits que especifican la longitud del encabezado, pues éste puede variar debido a la presencia de campos opcionales. Se codifica en palabras de 32 bits, donde la longitud mínima es 5 y la máxima 15, que equivale a 40 bytes de información opcional. La longitud del encabezado siempre es un múltiplo de 32 bits, por lo que se puede añadir un relleno al final del encabezado.

Tipo de Servicio: 8 bits. Permite establecer que calidad de servicio requiere el paquete. Se pueden establecer varios tipos de confiabilidad y velocidad (ej. rapidez en vez de confiabilidad para aplicaciones como audio o video, confiabilidad para transferencia de archivos, etc.). Este campo tiene subcampos de importancia: *P* (precedencia) son 3 bits de prioridad, un *Flag* tres bits D, T y R que permite especificar que importa más Retardo, Throughput o Confiabilidad. Recientemente, el campo Differentiated Services ha sustituido a este campo. Su finalidad es implementar QoS en redes IP mediante la arquitectura denominada Servicios Diferenciados o Diffserv.

Largo Total: 16 bits que especifican la longitud del paquete completo, encabezado incluido, en bytes.

Identificación: 16 bits. Este campo permite al destino determinar a que paquete pertenece el fragmento que recientemete ha llegado a él. Está relacionado con la fragmentación de paquetes.

DF: 1 bit, que indica no fragmentar el paquete.

MF: 1 bit, indica que vienen más fragmentos. Todos los fragmentos del paquete, salvo el último, tienen este bit en 1.

Offset del Fragmento: 13 bits para indicar a que parte del paquete total pertenece el fragmento que se está recibiendo.

TTL: 8 bits que permiten descartar un paquete una vez que ha dado un número excesivo de saltos o ha pasado un tiempo excesivo viajando por la red. Es un contador regresivo que indica el tiempo de vida restante del datagrama medido en segundos, de forma que si su valor llega a cero el paquete debe ser descartado. Esto permite evitar que se produzcan loops y un paquete pueda permanecer "flotando" indefinidamente en la red. Como no es trivial calcular con precisión el tiempo que un paquete emplea en el tránsito entre dos routers, en la práctica lo que se hace es restar el TTL en uno por

cada salto, y en el caso de que el datagrama se encuentre durante más de un segundo esperando en un router, se resta uno por cada segundo de espera. Como los paquetes casi nunca están más de un segundo en un router, en la práctica este parámetro funciona como un contador de saltos. En el caso de producirse fragmentación, el host receptor puede retener datagramas durante varios segundos, mientras espera a recibir todos los fragmentos. En este caso, el host restará uno del TTL por cada segundo de espera, pudiendo llegar a descartar paquetes por este motivo. Los valores de TTL típicos están entre 40 y 64.

Protocolo: 8 bits que especifican a que protocolo de nivel de transporte corresponde el paquete. La tabla de protocolos válidos y sus correspondientes números son controlados por el IANA (Internet Assigned Number Authority)la tabla 6 muestra algunos de los posibles valores de este campo. Llama la atención el valor 4 de la tabla 6 que está reservado para el uso de IP para transportar IP, es decir, al encapsulado de un paquete IP dentro de otro. Esta técnica permite realizar ruteo desde el origen de los paquetes encapsulando el paquete en otro dirigido al nodo intermedio por el que se quiere pasar.

Checksum: 16 bits que sirven para detectar errores en el encabezado del paquete. El checksum permite evitar al paquete de una alteración en alguno de los campos del encabezado que pudiera producirse, por ejemplo, por un problema de hardware en un router. El checksum sólo cubre el encabezado del paquete, no los datos. El campo checksum se ha de recalcular en cada salto, ya que al menos el TTL cambia. Notar que en routers con alto tráfico, el recálculo del checksum supone un inconveniente desde el punto de vista de rendimiento.

Dirección Fuente: 32 bits que corresponden a la dirección IP origen.

Dirección Destino: 32 bits que corresponden a la dirección IP destino.

Opciones: campo de longitud variable que no siempre está soportado en los routers y se utiliza muy raramente. Fue diseñado para permitir expansiones al protocolo, experimentos,etc. Las opciones son de tamaño variable, comenzando siempre por un yte de codificación, y siempre son rellenadas a múltiplos de 4 bytes. Entre las opciones destacables están *Record Route* que pide a cada router por el que pasa el paquete que anote en el encabezado su dirección, obteniéndose un trazado de la ruta seguida (debido a la limitación a un máximo de 40 bytes en la parte opcional del encabezado, como máximo pueden registrarse 9 direcciones). *Timestamp* actúa de manera similar a record route, pero además de anotar la dirección IP de cada router atravesado se anota en otro campo de 32 bits el instante en que el paquete pasa por dicho router. El uso de dos campos de 32 bits aumenta el problema antes mencionado del poco espacio disponible para grabar esta información. *Source Routing* permite al emisor especificar la ruta que debe seguir el paquete hasta llegar a su destino. Existen dos variantes: strict source routing que especifica la ruta exacta salto a salto, de modo que si en algún caso la ruta marcada no es factible por algún motivo se producirá un error. La segunda es loose source routing donde se dicen los routers por los que debe pasar el paquete, pero se da libertad a la red

para que use otros routers cuando lo considere conveniente. La limitación en la longitud de las opciones impone un límite máximo en el número de saltos que pueden especificarse. El uso de los campos opcionales del encabezado IP tiene generalmente problemas de rendimiento, ya que las implementaciones de los routers optimizan el código para las situaciones normales, es decir, para paquetes sin campos opcionales. Las opciones están implementadas y funcionan, pero lo hacen generalmente de forma poco eficiente ya que en el diseño del software no se ha hecho énfasis en su optimización.

Cuadro 6: Algunos Valores y Significados del Campo Protocolo en un Paquete IP

Valor	Protocolo	Descripción
0	Reservado	
1	ICMP	Internet Control Message Protocol
2	IGMP	Internet Group Management Protocol
3	GGP	Gateway-to-Gateway Protocol
4	IP	IP en IP (encapsulado)
5	ST	Stream
6	TCP	Transmission Control Protocol
8	EGP	Exterior Gateway Protocol
17	UDP	User Datagram Protocol
29	ISO-TP4	ISO Transport Protocol Clase 4
38	IDRP-CMTP	IDRP Control Message Transport Protocol
80	ISO-IP	ISO Internet Protocol (CLNP)
88	IGRP	Internet Gateway Routing Protocol (Cisco)
89	OSPF	Open Shortest Path First
255	Reservado	

3.4.1. Fragmentación

El tamaño de un paquete IP se especifica en un campo de dos bytes, por lo que su valor máximo es de 65535 bytes. Sin embargo, muy pocos protocolos o tecnologías a nivel de enlace admiten enviar frames de semejante tamaño. Normalmente el nivel de enlace no fragmenta, por lo que tendrá que ser IP el que adapte el tamaño de los paquetes para que quepan en los frames del nivel de enlace. Por lo tanto, en la práctica el tamaño máximo del paquete viene determinado por el tamaño máximo del frame característico de la red utilizada. Este tamaño máximo de paquete se conoce como **MTU o Maximum Transfer Unit**. La tabla 7 muestra algunos valores característicos de MTU de redes típicas.

Existen dos situaciones en que se produce fragmentación. La primera, denominada *fragmentación en ruta*, se produce cuando un paquete es creado por un host en una red con un valor determinado de MTU y en su camino hacia el host de destino ha de pasar por otra red

Cuadro 7: Valor de MTU Para Protocolos Comunes de Nivel de Enlace.

Protocolo a nivel de enlace	MTU (bytes)
PPP (valor por defecto)	1500
PPP (bajo retardo)	296
SLIP	1006 (límite original)
X.25	1600 (RFC 1356)
Frame Relay	1600 (depende de la red)
SMDS	9235
Ethernet DIX	1500
Ethernet LLC-SNAP	1492
IEEE 802.4/802.2	8166
Token Ring 16 Mb/s	17940 (token holding time 8 ms)
Token Ring 4 Mb/s	4440 (token holding time 8 ms)
FDDI	4352
Hyperchannel	65535
Classical IP over ATM	9180

con una MTU menor. En estos casos, el router que hace la transición a la red de MTU menor ha de fragmentar los paquetes para que no excedan el tamaño de la nueva red. La segunda, llamada *fragmentación en origen*, se produce como consecuencia del diseño de la aplicación. Por ejemplo, muchas implementaciones de NFS generan paquetes de 8 Kbytes de datos (8212 bytes con el encabezado IP). Un host en una red Ethernet que utilice NFS tendrá que fragmentar cada paquete en seis fragmentos antes de enviarlo, aún cuando el host de origen y destino se encuentren ambos en el mismo segmento Ethernet.

La fragmentación se realiza cortando la parte de datos del paquete en trozos del tamaño máximo permitido por la nueva red. Todos los campos del encabezado del paquete original se repiten en los fragmentos, excepto aquellos que se emplean para distinguirlos entre sí. Una vez fragmentado, un paquete no se reensambla hasta que llegue al host de destino, aún cuando en el trayecto pase a través de redes que admitan una MTU mayor. Los estándares Internet recomiendan que todas las redes que soporten TCP/IP tengan una MTU de al menos 576 bytes, condición que cumplen la mayoría de las redes. La MTU mínima imprescindible para funcionar en TCP/IP es de 68 bytes, valor que corresponde a 60 bytes de encabezado (el máximo con todos los campos opcionales) y 8 bytes de datos, que es el fragmento mínimo de datos que puede hacerse.

El campo identificación del encabezado IP es usado por el emisor para marcar cada paquete emitido. De esta forma, en caso de que se produzca fragmentación, el receptor podrá reconocer las partes que corresponden al mismo paquete, ya que todas irán acompañadas de la misma identificación. El bit DF cuando está a 1 indica a los routers que este paquete no debe fragmentarse. Normalmente esto se hace por uno de los dos motivos siguientes:

1. El receptor no está capacitado para reensamblar los fragmentos.

2. Cuando se aplica la técnica de descubrimiento de MTU del trayecto o "path MTU discovery" que permite averiguar el MTU de una ruta. Esta técnica consiste en que el host de origen envía un paquete del tamaño máximo al host de destino con el bit DF en 1. Si el paquete no puede pasar en algún punto del trayecto el router correspondiente genera un mensaje de error que es devuelto al host emisor. Entonces, éste envía otro paquete más pequeño, también con el bit DF en 1. Así, usando prueba y error, se consigue que algún paquete pase sin fragmentar al host destino. Para acelerar el proceso, algunos routers incorporan en los mensajes de error información sobre la MTU máximo que puede admitir la red que ha provocado el rechazo.

El Offset del Fragmento sirve para indicar, en el caso de que el paquete sea un fragmento, en que posición del original se sitúan los datos que contiene el fragmento actual. Los cortes siempre se realizan en múltiplo de 8 bytes, que es la unidad elemental de fragmentación, por lo que este campo cuenta los bytes en grupos de 8.

Como los fragmentos de un paquete pueden llegar desordenados a su destino, el receptor podrá identificarlos gracias al campo Identificación. La longitud total del paquete puede calcularla cuando recibe el último fragmento, que está identificado por el bit MF en 0. A partir de los campos Longitud y Offset del Fragmento la longitud será: $Fragment\_Offset * 8 + Longitud$.

Cuando se fragmenta un paquete, el host receptor retiene en su buffer los fragmentos y los reensambla cuando los ha recibido todos. Mientras mantiene retenido un fragmento, el host va restando cada segundo una unidad al campo TTL. Cuando el valor de TTL es igual a cero, descarta el fragmento. Si alguno de los fragmentos de un paquete se pierde, el resto terminarán desapareciendo a medida que agoten su TTL. No existe ningún mecanismo en IP que contemple el reenvío de paquetes o de fragmentos. Si el protocolo utilizado a nivel superior contempla reenvío de datos perdidos, por ejemplo TCP a nivel de transporte, se provocará el reenvío del paquete correspondiente. Normalmente, el segundo envío se verá sometido a la misma fragmentación que el primero, pero el segundo no podrá en ningún caso aprovechar fragmentos residuales que pudiera haber en el host receptor correspondientes al primer envío, ya que desde el punto de vista del nivel IP se trata de dos paquetes distintos e independientes que reciben identificaciones diferentes.

3.4.2. Direcciones IP

Cada interfaz de red de cada host o router en una red IP se identifica mediante, al menos una, dirección única de 32 bits. Las direcciones IP se suelen representar por cuatro números decimales separados por puntos, que equivalen al valor de cada uno de los cuatro bytes que componen la dirección. Por ejemplo una dirección IP válida sería 152.74.21.3. Si un nodo dispone de varias interfaces físicas, por ejemplo un router, cada una de ellas deberá tener necesariamente una dirección IP distinta si se desea que sea accesible de forma diferenciada para este protocolo. Es posible también y en algunas situaciones resulta útil, definir varias direcciones IP asociadas a una misma interfaz física.

Como ocurre en la mayoría de las redes, las direcciones IP tienen una estructura jerárquica. Una parte de la dirección corresponde a la red, y la otra al host dentro de la red. Cuando un router recibe un paquete por una de sus interfaces compara la parte de red de la dirección

con las entradas contenidas en sus tablas y reenvía el paquete por la interfaz correspondiente, situación denominada ruteo.

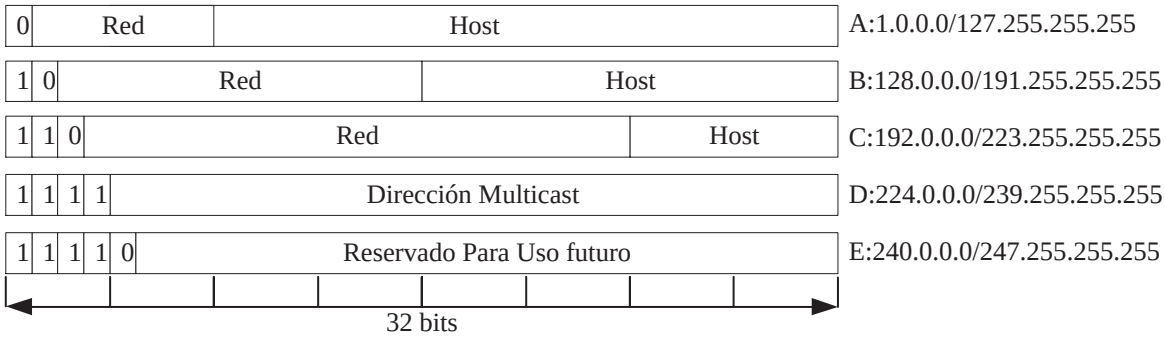

Figura 29: Formato de las Direcciones IP.

En el diseño inicial de Internet se reservaron los ocho primeros bits para la red, dejando los 24 restantes para el host, pues se creía que con 254 redes habría suficiente para la red experimental del DoD. Pronto se vio que esto resultaba insuficiente, por lo que se reorganizó el espacio de direcciones reservando unos rangos para definir redes más pequeñas. El resultado de esa reorganización es lo que hoy se conoce como redes clase A, B y C:

- Una red de clase A se caracteriza por tener en 0 el primer bit de dirección. El campo red ocupa los 7 bits siguientes y el campo host los últimos 24. Puede haber hasta 128 redes de clase A con 16777216 direcciones cada una.
- Una red de clase B tiene el primer bit en 1 y el segundo en 0. El campo red ocupa los 14 bits siguientes, y el campo host los 16 últimos. Puede haber 16384 redes clase B con 65536 direcciones cada una.
- Una red clase C tiene los primeros tres bits en 110. El campo red ocupa los siguientes 21 bits, y el campo host los 8 últimos. Puede haber hasta 2097152 redes clase C con 256 direcciones cada una.

Para indicar qué parte de la dirección corresponde a la red y qué parte al host, se suele utilizar una notación denominada máscara, consistente en poner a 1 los bits que corresponden a la parte de red y a 0 los que corresponden a la parte host. Así por ejemplo, una red clase A tiene una máscara 255.0.0.0, lo que equivale a decir que los ocho primeros bits especifican la red y los 24 restantes el host. Análogamente, una red clase B tiene una máscara 255.255.0.0 y una clase C una máscara 255.255.255.0.

Las máscaras permiten extraer de forma sencilla la parte de red o de host de una dirección. Por ejemplo, un router que ha de enviar un paquete puede realizar un AND entre la dirección de destino y la máscara correspondiente, con lo que extraerá la parte de red de la dirección.

Existen además direcciones clase D cuyos primeros cuatro bits valen 1110. Las direcciones clase D se utilizan para definir grupos multicast. El grupo queda definido por los 28 bits

siguientes. Puede haber hasta 268435456 direcciones multicast en Internet. Las direcciones clase D nunca puede aparecer como direcciones de origen de un paquete. Finalmente, está la clase E, que corresponde al valor 1111 en los primeros cuatro bits, no se utiliza de momento y está reservada para usos futuros.

De los valores de los primeros bits de cada una de las clases antes mencionadas se puede deducir el rango de direcciones que corresponde a cada una de ellas. Así pues, en la práctica es fácil saber a que clase pertenece una dirección determinada sin más que observar el primer byte de su dirección. La figura 29 muestra el esquema de direccionamiento IP.

Existen algunas reglas y convenios que asignan significados especiales a determinadas direcciones IP:

1. La dirección 255.255.255.255 se utiliza para indicar broadcast en la propia red. Por ejemplo, podría se utilizada como dirección de destino por un host que está booteando desde la red en una LAN y que para averiguar la red en la que se encuentra y su propia dirección IP necesita localizar un servidor que le de los parámetros de configuración básicos. Sólo se puede utilizar como dirección de destino, nunca como dirección de origen.

2. La dirección 0.0.0.0 identifica al host actual. En el caso anterior, la utilizaría el host como dirección de origen de sus paquetes. Sólo se puede utilizar como dirección de origen, no de destino.

3. Las direcciones con el campo host en cero identifican redes y por tanto no se utilizan para ningún host. Se emplean para especificar rutas y nunca deberían aparecer como direcciones de origen o destino de un paquete. Por ejemplo, la dirección 152.74.21.0 identifica la una red clase B que pertenece al Departamento de Ingeniería Eléctrica de la Universidad de Concepción.

4. Una dirección con todos los bits del campo host en uno se utiliza como dirección de broadcast dentro de la red, por lo tanto, no se utiliza para ningún host y sólo puede ser una dirección de destino. Por ejemplo, para enviar un mensaje broadcast a la red anterior se debe utilizar la dirección 152.74.21.255.

5. Una dirección con el campo red con todos los bits en ceros identifica a un host en la propia red, cualquiera que sea la red. Por ejemplo, si se desea enviar un paquete al primer host (1.1) de una red clase B, se puede utilizar la dirección 0.0.1.1. Esto permite enviar un paquete a un host en una red sin saber el número de ésta, aunque es preciso conocer si es clase A, B o C para saber que tamaño tiene la parte red de la dirección.

6. La dirección 127.0.0.1 se utiliza para pruebas de loopback. Todas las implementaciones de IP devuelven a la dirección de origen los paquetes enviados a esta dirección sin intentar enviarlos a ninguna parte.

7. Las redes 127.0.0.0, 128.0.0.0, 191.255.0.0, 192.0.0.0 y el rango de 240.0.0.0 en adelante (clase E) están reservados y no deben utilizarse.

8. Las redes 10.0.0.0 (clase A), 172.16.0.0 a 172.31.0.0 (clase B) y 192.168.0.0 a 192.168.255.0 (clase C) están reservadas para redes privadas o intranets por el RFC 1918. Estos números no se asignan a ninguna dirección válida en Internet. Por lo tanto, pueden utilizarse para construir redes, por ejemplo, detrás de un firewall, sin riesgo de entrar en conflicto de acceso con redes válidas de la Internet.

Como consecuencia de las reglas 3 y 4 antes mencionadas siempre hay dos direcciones inútiles en una red, la primera y la última. Por ejemplo, en la red 152.74.21.0 (clase B) se tiene que reservar la dirección 152.74.21.0 para denotar la red, y la dirección 152.74.21.255 para envíos broadcast a toda la red. Por lo tanto, se dispone de 254 direcciones para hosts, no de 256.

3.4.3. División en Subredes

Suponiendo que una empresa dispone de varias oficinas, cada una con una LAN, todas ellas interconectadas entre sí, y que desea unirlas mediante el protocolo TCP/IP y una de las oficinas dispone además de una conexión a Internet y suponiendo también que cada oficina tiene suficiente con 254 direcciones de hosts. Entonces, en principio sería posible asignar una red clase C diferente para cada oficina, pero esto supone solicitar al NIC una red para cada oficina que se conecte, y al ser cada una independiente de las demás la administración se complica. Por ejemplo, sería preciso anunciar en Internet la ruta para cada nueva red para que la oficina correspondiente fuera accesible. Dado que cada red sería en principio independiente de las demás no habría una forma sencilla de agrupar las redes de la organización.

Existe un mecanismo que permite dividir una red IP en trozos o subredes, de forma que esta empresa podría solicitar una clase B y asignar fragmentos de dicha red a cada oficina a medida que se fueran incorporando a la red. Esto equivale a crear un nivel jerárquico intermedio entre la red y el host, permitiendo así grados variables de agrupación según el nivel en que se encuentre. Suponiendo que a la empresa se le asigna una red clase B, la 152.74.0.0. De los 16 bits que en principio corresponden al host podría reservar los primeros 8 para subred y dejar los 8 siguientes para host, con lo que dispondrá de 256 subredes de 256 direcciones cada una. Desde fuera, la red de la empresa seguirá siendo 152.74.0.0, ya que la estructura de subred no será visible.

Para dividir la red en subredes se define una nueva máscara. Como siempre los bits en 1 de la máscara identifican la parte de red, en este caso, la parte de red y subred, y los bits en cero corresponden al host. Por ejemplo, la máscara 255.255.255.0 aplicada sobre una red clase B la divide en 256 subredes de 256 direcciones cada una, pues tiene puestos en 1 los primeros 24 bits. En cierto modo, se puede decir que esta máscara convierte una red clase B en 256 subredes clase C. Se pueden hacer divisiones que no correspondan a bytes enteros, por ejemplo la máscara 255.255.252.0 hace subredes más grandes, reserva los primeros 6 bits para la subred y deja 10 para el host, con lo que podría haber hasta 64 redes con 1024 direcciones cada una.

Cuando se crean subredes hay dos direcciones en cada subred que quedan automáticamente reservadas: las que corresponden al campo host todos en ceros y todo en uno. Éstas se emplean para designar la subred y para el broadcast dentro de la subred respectivamente. Así, si

la red 152.74.0.0 se subdivide con la máscara 255.255.255.0 se crean 256 subredes del tipo 152.74.subred.host, cada una con 256 direcciones. En cada subred existen 254 direcciones aprovechables para hosts, ya que la primera dirección (152.74.subred.0) identifica a la subred y la última (152.74.subred.255) es la dirección broadcast de esa subred.

Del mismo modo que los valores de todos los bits en cero o en uno del campo host están reservados con un significado especial, los valores de los bits todos en cero y todo en uno del campo subred también son especiales. El valor todos cero se utiliza para representar la subred misma. Por ejemplo, si a la red 152.74.0.0 se le aplica la máscara 255.255.255.0 la primera subred (campo subred todo a ceros) no debería utilizarse, pues resultaría ambiguo el significado de la dirección 156.134.0.0, que representaría tanto a dicha subred como a la red entera. Análogamente, la última subred (campo subred todo a unos) tampoco debería utilizarse porque entonces la dirección 156.134.255.255 significaría tanto broadcast en dicha subred como en la red entera.

Mientras que la restricción de las direcciones todos los bits en cero o todos en uno en el campo host se ha de respetar siempre, existen muchas situaciones en las que interesa aprovechar la subred todos en cero o todos en uno, violando la norma antes mencionada. Esta violación, permitida por muchas implementaciones, se conoce como subnet-zero y se adopta para aprovechar mejor el espacio de direcciones disponible. Con subnet-zero es posible, por ejemplo, dividir una red clase B por la mitad en dos subredes mediante la máscara 255.255.128.0, cosa que no sería posible si no se permitiera esta excepción a la regla.

La tabla 8 resume todas las posibles subredes y máscaras que se pueden utilizar con una red clase B, y al tabla 9 cubre el caso de una red clase C.

La división en subredes no necesariamente debe hacerse de forma homogénea en todo el espacio de direcciones, como se ha hecho hasta ahora. Por ejemplo, podría partirse la red 152.74.0.0 en subredes de diferentes tamaños y asignar a cada oficina una subred adecuada a sus necesidades. Así en el ejemplo anterior se podría dividir la red 152.74.0.0 de la siguiente manera: 16 subredes de 256 direcciones (subredes desde 152.74.0.0 a 156.74.15.0 con máscara 255.255.255.0), 16 subredes de 1024 direcciones (subredes desde 152.74.16.0 a 156.74.76.0 con máscara 255.255.252.0), 3 subredes de 4096 direcciones (subredes desde 152.74.80.0 a 156.74.112.0 con máscara 255.255.240.0) y una subred de 32768 direcciones (subred desde 152.74.128.0 con máscara 255.255.128.0). La técnica anterior que permite dividir una red en subredes de diferentes tamaños se conoce como *máscaras de tamaño variable*.

3.4.4. Protocolos de Control IP

Normalmente, los paquetes IP transportan TPDUs (Transport Protocol Data Unit) de TCP o UDP, que son los dos protocolos de transporte utilizados en TCP/IP. Sin embargo, existen otros posibles contenidos para un paquete IP, en el que los datos que pueden transportarse son mensajes de los distintos protocolos de control de IP.

ICMP (Internet Control Message Protocol)

En las redes pueden producirse problemas, por lo tanto debe generarse mecanismos que permitan devolver un mensaje al host emisor indicándole lo sucedido. El mecanismo para

Cuadro 8: Subredes y Máscaras que Pueden Definirse en una Red Clase B.

Bits subred	Nº de subredes	Nº subredes (subred cero)	Bits host	Nº hosts	Máscara
0	0	0	16	65534	255.255.0.0
1	0	2	15	32766	255.255.128.0
2	2	4	14	16382	255.255.192.0
3	6	8	13	8190	255.255.224.0
4	14	16	12	4094	255.255.240.0
5	30	32	11	2046	255.255.248.0
6	62	64	10	1022	255.255.252.0
7	126	128	9	510	255.255.254.0
8	254	256	8	254	255.255.255.0
9	510	512	7	126	255.255.255.128
10	1022	1024	6	62	255.255.255.192
11	2046	2048	5	30	255.255.255.224
12	4094	4096	4	14	255.255.255.240
13	8190	8192	3	6	255.255.255.248
14	16382	16384	2	2	255.255.255.252
15	32766	32768	1	0	255.255.255.254
16	65534	65536	0	0	255.255.255.255

Cuadro 9: Subredes y Máscaras que Pueden Definirse en una Red Clase C.

Bits subred	Nº de subredes	Nº subredes (subred cero)	Bits host	Nº hosts	Máscara
0	0	0	8	254	255.255.255.0
1	0	2	7	126	255.255.255.128
2	2	4	6	62	255.255.255.192
3	6	8	5	30	255.255.255.224
4	14	16	4	14	255.255.255.240
5	30	32	3	6	255.255.255.248
6	62	64	2	2	255.255.255.252
7	126	128	1	0	255.255.255.254
8	254	256	0	0	255.255.255.255

reportar problemas en IP es el protocolo ICMP especificado en el RFC 792.

Los mensajes ICMP viajan por la red como datagramas IP, con el valor 1 en el campo protocolo, y están sujetos a las mismas reglas que cualquier otro paquete al llegar a un router. Los mensajes ICMP son generados por el host o router que detecta el problema o situación

extraordinaria y son dirigidos al host o router que aparece en como dirección origen del paquete que causó el problema. Para facilitar la identificación del paquete por parte del host emisor la mayoría de los mensajes ICMP incluyen, además del código de error correspondiente, el encabezado y los primeros ocho bytes de datos del paquete original.

Los mensajes ICMP más importantes son:

DESTINATION UNREACHABE. Se produce cuando no se puede entregar el paquete a su destino, las que pueden ser: cuando un router se encuentra con un paquete que tiene un 1 el bit DF y que no cabe en la MTU de la red por la que ha de enviarlo, o bien cuando un router no encuentra en sus tablas ninguna ruta por la que pueda llegar a la dirección para la que va dirigido el paquete. Cuando un router tiene configurada ruta por defecto, nunca enviará mensajes Destination Unreachable.

SOURCE QUENCH. Este mensaje se creó para permitir a los routers solicitar una reducción en el tráfico generado por los hosts en caso de congestión. En la práctica, se ha observado que el uso de este tipo de mensajes agrava los problemas en caso de congestión, por lo que la recomendación actual es que los routers no deben generar paquetes SOURCE QUENCH en ningún caso.

ECHO REQUEST & ECHO REPLY. Permiten detectar si un destino determinado está operativo. Al recibir el mensaje ECHO REQUEST el destino debe responder con un ECHO REPLY. El programa ping utiliza estos mensajes para su funcionamiento.

TIME EXCEEDED . Este mensaje se envía al emisor cuando un paquete es descartado porque su TTL ha llegado a cero, lo que puede ser síntoma de que se ha producido algún loop en la red o que el valor del TTL utilizado es demasiado bajo para el diámetro de la red. El programa traceroute, que permite averiguar la ruta a un destino determinado, utiliza paquetes con TTL de 1, 2, 3, y así sucesivamente, y a partir de los mensajes TIME EXCEEDED recibidos puede deducir la ruta completa hasta el destino especificado.

REDIRECT . Utilizado para alertar al host emisor cuando se sospecha que un paquete se está ruteando incorrectamente. Este mensaje lo utilizan los routers cuando reciben paquetes de un host que van dirigidos a otro host que se encuentra en la misma LAN.

Resolución de Direcciones: ARP

Cuando utiliza una red multiacceso, por ejemplo una LAN, ISDN o ATM, la tecnología utilizada para enviar los paquetes de datos permite llegar por una misma interfaz física a más de un destinatario. En este caso, es necesario algún mecanismo que permita saber a cual de todos los destinos posibles se dirigen los paquetes. Todas las redes multiacceso disponen de un sistema de direccionamiento propio, en el caso de una LAN las direcciones MAC de las estaciones son el método de direccionamiento. En todos estos casos, el nivel de red es el encargado de realizar el mapeo entre la dirección de la tecnología multiacceso correspondiente y la dirección de red, situación que se conoce como *resolución de direcciones*.

Entre los múltiples mecanismos de resolución de direcciones se cuentan:

1. Por medio de una tabla estática, que es mantenida manualmente en cada nodo y que contiene la equivalencia completa de todas las direcciones. El principal problema que tiene es la necesidad de actualizar las tablas en todos los nodos de la red cada vez que se produce alguna modificación en la tabla de direcciones.

2. Por medio de una tabla dinámica mantenida de forma automática en un servidor que es conocido por todos los hosts. Cuando un nodo quiere enviar un mensaje a un destino determinado indica al servidor la dirección de red que busca y éste le devuelve la dirección correspondiente. Debe existir un proceso de registro en el servidor para que un nodo pueda adherirse a la red. Los principales problemas de esta solución son la necesidad del registro previo y que el servidor se convierte en un cuello de botella de la red, lo que puede llegar a limitar la confiabilidad y el desempeño.

3. Establecer un mecanismo previamente conocido por el que se pueda deducir la dirección de la red multiacceso, a partir de la dirección de red. De esta forma, cualquier nodo puede deducir la dirección de nivel dos, a partir de la dirección de red. Este mecanismo se emplea en DECNET que construye la dirección MAC añadiendo a la dirección de red un prefijo determinado y conocido por todos los nodos.

4. Utilizar un mensaje broadcast para lanzar a la red una pregunta solicitando la respuesta del nodo en la red multiacceso que posee la dirección de red buscada. Esta técnica da máxima flexibilidad ya que los equipos no necesitan registrarse y no hay un servidor centralizado del que dependa el funcionamiento de la red. Sin embargo, sólo es factible de realizar en redes de naturaleza broadcast, como las redes locales. Su principal desventaja es el uso de paquetes broadcast que produce una degradación del rendimiento de la red. Esta técnica es la utilizada por IP sobre redes locales de todo tipo.

El protocolo funciona en base a esta última alternativa, por ejemplo, si un nodo quiere iniciar conexión con otro host cuya dirección, a modo de simplificación, se encuentra en la misma red local, entonces el emisor genera un mensaje ARP con la pregunta "¿quién tiene la dirección de red A.B.C.D?" y lo envía como un frame Ethernet que tiene como dirección MAC destino la dirección de broadcast. El frame es recibido y procesado por todas los hosts activos, y eventualmente una máquina se reconoce propietaria de la dirección de red solicitada y responderá entonces al mensaje. La respuesta puede ser, de hecho, normalmente lo será, un frame unicast, puesto que el servidor ya conoce la dirección MAC del cliente que lanzó la pregunta, y la respuesta incluirá la dirección MAC solicitada, por lo que a partir de ese momento ambos hosts pueden comunicarse mediante frames unicast, reduciendo la generación de tráfico broadcast sólo al primer mensaje. Para optimizar el proceso, cada nodo mantiene en memoria una tabla denominada *cache ARP*, con los pares de direcciones MAC-IP utilizadas recientemente. Generalmente, cuando un host envía un ARP buscando a otro, todas los nodos, y no sólo el destinatario del mensaje, aprovechan para captar al emisor, anotándolo en su cache ARP, optimizando nuevamente el proceso. Las entradas de la tabla ARP expiran pasados unos minutos sin que haya tráfico con la máquina correspondiente.

El formato del paquete ARP se observa en la figura 30 y la descripción de los campos es la siguiente:

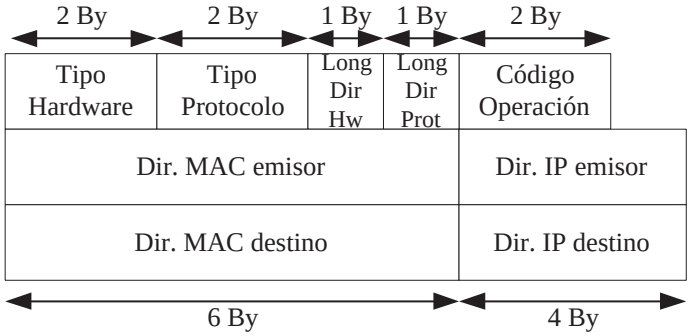

Figura 30: Estructura de un paquete ARP IP para redes 802.

Tipo de Hardware: especifica el tipo de red local. El código 1 identifica a Ethernet.

Tipo de Protocolo: especifica el protocolo de red utilizado. Se emplean los mismos códigos que en el Ethertype.

Longitud de Dirección de Hardware: el largo se especifica en bytes.

Longitud de Dirección de Red: el largo se especifica en bytes.

Código de Operación: vale 1 en el caso de una pregunta ARP y 2 en el de una respuesta.

Resolución Inversa de Direcciones

A veces se plantea el problema inverso a ARP, es decir, encontrar la dirección IP a partir de una determinada dirección LAN. Por ejemplo, cuando se bootea un nodo sin disco éste ha de cargar su sistema operativo desde otro host, que normalmente está situado en la misma red local, pero desconoce todo lo relativo a su configuración de red, incluida la dirección IP. Lo único que la estación conoce en principio es su dirección MAC, que se encuentra escrita en su tarjeta de red local. Para resolver esto existen las siguientes respuestas:

RARP Reverse Address Resolution Protocol funciona de la siguiente manera: el nodo envía un mensaje broadcast en el que indica su dirección MAC y solicita que alguien le informe cual es su dirección IP. En la red existe un servidor RARP encargado de atender este tipo de peticiones, que consultará sus tablas y devolverá la dirección IP correspondiente. RARP utiliza el mismo formato de mensaje que ARP, la única diferencia es el uso de los códigos de operación 3 y 4 para la pregunta y respuesta RARP, respectivamente. Como la consulta RARP se realiza mediante broadcast, el servidor RARP debe estar en la misma LAN que el cliente, ya que los mensajes broadcast a nivel MAC no atraviesan los routers. Otra limitación de RARP es el hecho de que sólo contiene el envío de la dirección IP, y sería interesante aprovechar el mensaje para informar al cliente de todo el conjunto de parámetros relacionados con la configuración de la red: máscara, gateway, etc.

BOOTP BOOTP (BOOTstrap Protocol) supera las limitaciones de RARP, cuando un host lanza una pregunta BOOTP lo hace con un paquete IP con la dirección de destino 255.255.255.255 y dirección de origen 0.0.0.0. De esta forma, el paquete es recibido por todos los hosts de la LAN, y si alguno de ellos es el servidor BOOTP responderá con los datos requeridos en otro paquete broadcast IP (dirección 255.255.255.255). Si no existe ningún servidor BOOTP en la red local, deberá existir algún router designado como relay BOOTP que se encarga de retransmitir los mensajes BOOTP que reciba por una de sus interfaces a través de la cual pueda acceder al servidor BOOTP. De esta forma el servidor BOOTP puede colocarse en una ubicación arbitrariamente remota respecto del cliente. Además de la dirección IP, BOOTP permite indicar el nombre del host, la máscara de subred, el gateway, servidor de nombres, etc.

DHCP RARP y BOOTP utilizan asignación estática y única entre direcciones MAC y direcciones IP. Existen situaciones en las que esto no es conveniente. Por ejemplo, puede darse el caso de que se disponga de una sala para la conexión a Internet de hosts portátiles, por lo tanto no se conocen las direcciones MAC de los clientes que se utilizarán el servicio, y tampoco se sabe de antemano cuantos serán, lo único que se sabe es que no habrá en ningún momento más nodos que la capacidad máxima que posee la sala.

El IETF, en 1993, diseño el protocolo DHCP (Dynamic Host Configuration Protocol), que es similar a BOOTP pero es más versátil en los mecanismos de asignación de direcciones IP. En DHCP los clientes pueden recibir sus direcciones por una de las siguientes formas:

1. Asignación indefinida y estática. En este caso la dirección es fija y similar a BOOTP.

2. Asignación automática. La asignación es también estática, pero la elección de la dirección IP correspondiente es tomada por el servidor DHCP la primera vez que el equipo le solicita su dirección.

3. Asignación dinámica. En este caso el cliente recibe la dirección IP del servidor durante un tiempo limitado. Pasado ese tiempo el cliente debe renovar su solicitud o de lo contrario la concesión expirará. De esta forma una misma dirección puede ser reutilizada por diferentes máquinas en momentos diferentes.

Las principales ventajas de DHCP son su mayor flexibilidad y la simplificación de las labores de administración. Un inconveniente de la asignación dinámica de direcciones es que si se desea rastrear un problema y sólo se dispone de la dirección IP, puede llegar a resultar imposible averiguar que nodo ha sido el causante del problema. Otro problema es la asociación de direcciones y nombres en el DNS, pues con la asignación dinámica diferentes máquinas pueden recibir el mismo nombre en diferentes momentos.

3.5. Conceptos de Ruteo

La Internet está formada por miles de redes interconectadas, pertenecientes a diversas empresas y organizaciones. Todas estas redes interconectadas comparten a nivel de red el

protocolo IP. Al margen de esta interoperabilidad existen dos aspectos fundamentales en los que las redes pueden diferir entre si:

- El protocolo de routing utilizado: existe una gran variedad de protocolos de ruteo. Aún utilizando el mismo algoritmo y protocolo de ruteo dos proveedores diferentes normalmente no querrán que sus routers intercambien entre sí la misma información de rutas que intercambian internamente.
- La política de intercambio de tráfico: un proveedor puede tener acuerdos bilaterales o multilaterales para intercambiar tráfico con otros, pero normalmente no estará dispuesto a ser utilizado como vía de tránsito para el tráfico entre dos proveedores si esto no está expresamente establecido en los acuerdos, aun cuando desde el punto de vista de topología de la red pueda ser ese el camino más corto entre ambas.

3.5.1. Sistema Autónomo

Un sistema autónomo o AS será la subred que es administrada por una autoridad común, que tiene un protocolo de ruteo homogéneo mediante el cual intercambia información en toda la subred y que posee una política común para el intercambio de tráfico con otras redes o sistemas autónomos. Normalmente cada ISP constituye su propio sistema autónomo, por ejemplo, REUNA 2 la red académica a la cual está suscrita la Universidad de Concepción, corresponde a un sistema autónomo propio.

Así pues, en Internet se dan, al menos, dos niveles jerárquicos de ruteo, el que se realiza dentro de un sistema autónomo (AS) y el que se efectúa entre sistemas autónomos. El primero es denominado *ruteo interno o intraáreas*, al segundo se le denomina *ruteo externo o interáreas*. Dado que los requerimientos en uno y otro caso son muy diferentes, se utilizan protocolos de ruteo distintos. Los protocolos de ruteo interior se denominan IGP (Interior Gateway Protocol) y los utilizados entre sistemas autónomos se llaman EGP (Exterior Gateway Protocol).

3.5.2. Protocolos de Ruteo Interior

Routing information Protocol (RIP)

El protocolo más simple y antiguo es RIP, que viene provisto por el demonio routed en Unix, y fue introducido en Berkeley para mantener tablas correctas en sus redes locales. Nunca fue concebido como un protocolo escalable de ruteo, a pesar que hoy se usa bastante en redes grandes.

La idea es mantener en la tabla de ruteo (junto con la red y el gateway) una métrica que cuente la distancia a la que se encuentra el host de esa red. De esta forma, al recibir otras posibles rutas a la misma red, puede elegir la más corta.

RIP es un protocolo de vector de distancias, donde cada router puede verse como un nodo en un grafo, y las distancias son el número de nodos por los que debe pasar para llegar a su destino. Cada router maneja su tabla de ruteo, donde figuran todos los nodos del grafo y la distancia asociada (así como el gateway). Cada cierto tiempo, los routers envían esa tabla completa a todos sus vecinos. Al recibir la tabla de otro router, aprende los caminos

a redes que no conocía (y los agrega a su tabla) y encuentra nuevos caminos a redes que ya conocía. Para elegir una ruta, compara las métricas (al recibir una tabla, le suma 1 a todas sus métricas, puesto que las redes están a un router más de distancia) y se queda con la más pequeña. En caso de igualdad, se queda con la ruta antigua, para evitar cambios permanentes en las rutas.

Además de las rutas aprendidas por RIP, típicamente se maneja una ruta default, y las rutas directas a las redes a las que está conectado el router, cuyas métricas son cero.

Para encontrar a los demás routers y poder intercambiar con ellos las tablas, RIP utiliza un esquema de broadcast. Un router que habla RIP, difunde vía broadcast a todas las redes a las que está conectado su tabla de rutas periódicamente. Al recibir un broadcast RIP, el router compara sus entradas con las recibidas y actualiza la tabla.

Sin embargo, para poder adaptarse a fallas o caídas de routers, hay que poder también borrar rutas. Como no se puede confiar que el router caído avise, se define un intevalo de tiempo fijo en RIP entre broadcasts (30 segundos). Al transcurrir varios intervalos sin escuchar nada de un router (180 segundos) todas las rutas que fueron recibidas desde él se invalidan.

RIP tiene varias ventajas, probablemente la principal es que funciona sólo, prácticamente sin configuración o ingeniería inicial. Basta habilitar RIP en el router, y aprende y difunde todas las rutas automáticamente. Esta misma sencillez es su principal defecto, puesto que satura la red con broadcasts innecesarios, utiliza métricas que no toman en cuenta capacidades de las distintas redes, etc.

El principal problema de RIP es un defecto fundamental de cualquier protocolo de vector de distancias: al manejar sólo distancias, no puedo detectar los ciclos en las rutas. Al cambiar las rutas, es fácil caer en ciclos infinitos. Para evitar el problema de los ciclos infinitos, en RIP se define que una métrica 16 es equivalente a infinito. Además, se implementan otras soluciones (como split horizon que no difunde por una interfaz las rutas aprendidas por esa misma). Sin embargo, estas soluciones siempre tienen efectos laterales negativos.

Interior Gateway Routing Protocol (IGRP)

Con la creación de IGRP a principios de los ochentas, Cisco Systems fue la primera compañía en resolver los problemas asociados con el uso de RIP para rutear paquetes entre routers interiores. IGRP determina la mejor ruta a través de una red examinando el ancho de banda y la demora de las redes entre los routers. IGRP converge más rápido que RIP, por lo tanto se evitan los ciclos de ruteo causados por el desacuerdo entre routers sobre cual es el próximo salto a ser tomado. Más aún, el IGRP no tiene limitación en cuanto a contador de saltos. Por lo anterior, el IGRP es utilizado en redes de gran tamaño, complejas y con diversidad de topologías.

Cisco lanzó también una nueva versión de IGRP para manipular redes de alto crecimiento y misión-crítica. Esta nueva versión es conocida como EIGRP (Enhanced IGRP) y combina la facilidad de uso de los protocolos de ruteo de vector de distancia tradicional con las capacidades de reruteo rápido de los protocolos estado del enlace.

El EIGRP consume mucho menos ancho de banda que el IGRP, porque éste es capaz de limitar el intercambio de información de ruteo para incluir solamente la información que ha cambiado. Además, es capaz de manipular información de ruteo de AppleTalk e IPX, además

de IP.

Intermediate System-Intermediate System (IS-IS)

El protocolo de ruteo IS-IS está basado en el algoritmo de estado de enlace. Además IS-IS permite hacer routing integrado, es decir, calcular las rutas una vez y aplicarlas para todos los protocolos utilizados, permitiendo así auténtico routing multiprotocolo. Admite además, hasta ocho niveles de jerarquía para reducir la cantidad de información de routing intercambiada. IS-IS fue diseñado para el protocolo DECNET de Digital y adoptado después por ISO como protocolo de routing para el protocolo de red CLNP. Una variante de IS-IS se utiliza en Netware de Novell. IS-IS no es un estándar Internet, pero se utiliza en algunos sistemas autónomos.

Open Shortest Path First (OSPF)

OSPF es una alternativa más reciente al RIP entre los protocolos internos, que corrige todas las limitaciones que tenía éste. OSPF fue desarrollado por el IETF (Internet Engineering Task Force) como el reemplazo de RIP. Este protocolo es soportado por todos los principales vendedores de equipos de ruteo IP. OSPF es un protocolo de ruteo del tipo estado de enlace, que soporta ruteo jerárquico dentro de un sistema autónomo. OSPF provee un muy rápido ruteo y soporta máscaras de subred de longitud variable. OSPF se derivó del protocolo de ruteo IS-IS de la OSI. Algunas características especiales de OSPF incluyen ruteo de múltiples caminos de costo y ruteo basado en un tipo de nivel superior de solicitudes del servicio (TOS Type-Of-Services). Por ejemplo, una aplicación puede especificar que ciertos datos son urgentes y si OSPF tiene enlaces de alta prioridad a su disposición, ellos pueden ser utilizados para transportar un paquete urgente. OSPF soporta uno o más métricas.

En OSPF, un router no intercambia distancias con sus vecinos. En vez de eso, cada router chequea el status de cada uno de sus enlaces con los routers adyacentes y envía a éstos la información recogida, la que se propaga de esta forma a través del sistema autónomo. Cada router captura esta información y construye su tabla de ruteo, y todos los routers involucrados tendrán la misma tabla de ruteo.

Desde un punto de vista práctico, la diferencia más importante es que un protocolo de estado del enlace converge con mayor rapidez que un protocolo de vector de distancia. Por convergencia se entiende que la estabilización después de cambios en la red, como caídas de router o de enlaces. OSPF se diferencia de RIP (y de otros muchos protocolos de ruteo) en que utiliza sólo IP, o sea, no es multiprotocolo.

Además de ser un protocolo de enlace en vez de distancia, OSPF tiene otras muchas características que lo hacen superior a RIP:

1. OSPF puede calcular un conjunto separado de rutas para cada tipo de servicio IP. Esto quiere decir que para un mismo destino puede haber varias entradas en la tabla de ruteo, una por cada tipo de servicio.

2. A cada interfaz se le asigna un costo. Este puede asignarse en función del ancho de banda de salida, seguridad, fiabilidad, etc. Pueden asignarse distintos costos para distintos servicios.

3. Cuando existen varias rutas a un mismo destino, con idénticos costos, OSPF distribuye el tráfico por ambas rutas de forma equitativa.

4. OSPF soporta subredes: una máscara de subred es asociada con cada ruta notificada. Esto permite que una única dirección IP de cualquier clase pueda ser dividida en múltiples subredes de varios tamaños. Las rutas a un host son notificadas mediante una máscara de subred con todos los bits a 1. Una ruta por defecto es notificada como una dirección IP de 0.0.0.0 con una máscara con todos los bits a 0.

5. Los enlaces punto a punto entre routers no necesitan una dirección IP a cada extremo. Es lo que se conoce como redes no numeradas. De esta forma se ahorran direcciones IP.

6. Es posible emplear un pequeño mecanismo de autentificación ya que es posible enviar un password.

7. OSPF emplea multicast en vez de broadcast, para reducir la carga en los sistemas que no emplean OSPF.

Si se considera que todos los routers poseen el mismo grafo representativo de la red, el protocolo se basa en el cálculo del árbol de distancias mínimas desde un nodo determinado. El árbol resultante dependerá del nodo desde el cual se realice el cálculo. Los enlaces que no están marcados se considera que tienen una distancia de 0. Una vez calculado el árbol, los paquetes se enviarán por la rama más corta que conduzca al destino. A partir de ahí, serán los siguientes routers los que decidan la ruta a seguir. La actualización de la tabla se puede realizar mediante protocolos externos como BGP, o puede modificarse de forma estática. También es posible añadir rutas por defecto.

Dominio de Ruteo OSPF y Áreas OSPF permite que se agrupen juntas colecciones de redes y hosts. Esta agrupación, junto con todos los routers que tienen interfaces a cualquiera de las redes incluidas es llamada un *área*. Cada área ejecuta una copia separada del algoritmo de ruteo básico SPF, lo que implica que cada área tiene su propia base de datos topológica.

La topología de un área es invisible para cualquier dispositivo que no pertenezca a ella. Es decir, los router internos de un área específica no saben nada de la topología externa al área. Esta aislación es la que permite introducir un bajo tráfico de ruteo en la red, en comparación a compartir toda la información del sistema autónomo. Los routers que están conectados a múltiples áreas son llamados *routers de borde de área (ABR)*. Es así como dos routers que pertenecen a una misma área tienen, para esa área, una base de datos idéntica.

El ruteo en un sistema autónomo tiene dos niveles, dependiendo de si la fuente y el destino están en una misma área o no. El *ruteo intraárea* pertenece al primer caso, los paquetes son ruteados con información exclusivamente del área en cuestión. Esto protege al ruteo de la inyección de información corrupta. En el *ruteo interárea*, se obtiene información del o las áreas exteriores involucradas.

Backbone OSPF Todo dominio de ruteo OSPF debe tener un backbone. El backbone es un área especial que tiene un identificador 0.0.0.0, o simplemente 0. Consiste de todas las redes que no son contenidas en ningún área específica, sus routers asociados y los routers que pertenecen a múltiples áreas. El backbone tiene como restricción que debe ser contiguo. Cada una de las interfaces que son configuradas en el área 0 deben ser alcanzables vía otros routers, donde cada interfaz en la trayectoria está configurada como si estuviera en el área 0. Sin embargo, es posible definir áreas en las que el backbone ya no sea contiguo, es decir, donde se rompa la continuidad entre routers. Esto es posible mediante la configuración de *enlaces virtuales*.

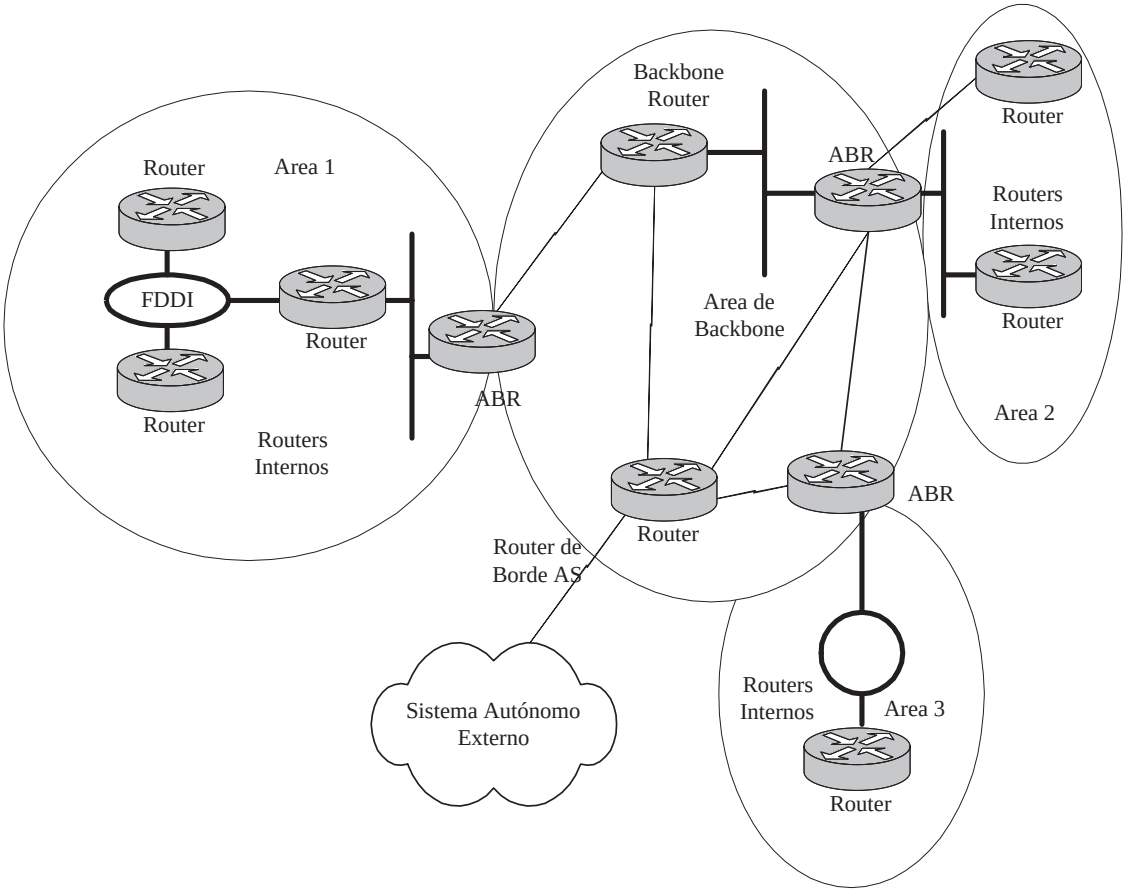

Figura 31: Descripción de los Tipos de Routers y Relaciones en OSPF.

Clasificación de los Routers OSPF Cuando un sistema autónomo se divide en áreas, los routers pueden clasificarse, de acuerdo a la función que cumplen, en cuatro clases que se traslapan entre sí.

Router Interno: tiene todas sus interfaces conectadas a redes que pertenecen a la misma

área. Los routers con interfaces sólo en el backbone también pertenecen a esta clase. Estos routers ejecutan una sola copia del algoritmo SPF.

Router de Borde de Área: se unen a múltiples áreas, ejecutan múltiples copias del algoritmo SPF, una por cada área a la que se asocian y una adicional para el backbone. Tienen la misión de condensar la información de sus áreas para su distribución por el backbone, que la distribuye a otras áreas.

Router de Backbone: tiene una interfaz conectada al backbone. Esto incluye todos los routers que se asocian a más de un área, esto no implica que sean routers de borde de área.

Router de Borde de AS: intercambia información de ruteo con otros routers pertenecientes a otros sistemas autónomos. La trayectoria hacia estos routers es conocida por cada uno de los routers del sistema autónomo. Esta clasificación es totalmente independiente de las anteriores, un router de borde de AS pude ser interno o de borde de área, y puede o no participar en el backbone.

La figura (31) muestra varios tipos de routers OSPF y la relación entre ellos y con todo el ambiente OSPF.

Vecinos y Adyacencias OSPF crea adyacencias entre routers vecinos para facilitar el intercambio de información. Los *routers vecinos* son dos routers que tienen interfaces a una red en común. En redes multiacceso, los vecinos son descubiertos, en forma dinámica, utilizando el protocolo de OSPF *Hello*. Una *adyacencia* es una relación formada entre los routers vecinos seleccionados con el propósito de intercambiar información de ruteo.

No todos los pares de routers vecinos llegan a ser adyacentes. En cambio, las adyacencias son establecidas con un subconjunto de los routers vecinos. Los routers que están conectados por enlaces punto-a-punto o mediante enlaces virtuales son siempre adyacentes. en las redes multiacceso, todos los routers llegan a ser adyacentes al router designado y al router designado de respaldo.

Router Designado y Router Designado de Respaldo Todas las redes multiacceso tiene un router designado y otro que servirá de respaldo en caso de que el primero falle. Sus dos funciones principales son:

- Originar los avisos de enlace de red de parte de la red. Este anuncio lista el conjunto de routers, incluyendo el designado, que actualmente están unidos a la red.
- Llegar a ser adyacente a todos los otros routers de la red. Debido a que las bases de datos de estado de enlace son sincronizadas a través de adyacencias (a través de la inicialización de la adyacencia y el proceso de inundación), el router designado juega un papel principal en el proceso de sincronización.

3.5.3. Protocolos de Ruteo Exterior

Protocolo de Gateway Exterior (EGP)

La autoridad administrativa asociada con cada sistema autónomo nomina uno o más routers para funcionar como router exterior para esos sistemas. Dentro de los sistemas autónomos, éstos se comunican con el otro gateway interior usando el IGP para ese sistema. De aquí en adelante cada gateway exterior, mediante su tabla de ruteo local, sabe sobre los net IDs dentro de ese sistema y sus distancias desde ese gateway. Cuando cada gateway exterior es inicializado, en primer lugar, se da la identidad única del sistema autónomo a la cual se accesa. También recibe el contenido de una tabla de ruta, conocida como la tabla de capacidad, que le permite comunicar con todas los otros gateways exteriores vía red central. El protocolo de gateway exterior (EGP) también envuelve cada gateway exterior haciendo contacto con gateways exteriores seleccionadas, como fue requerido, e intercambia información de ruteo con ellos. Esta información de ruta consiste de la lista de net IDs dentro del sistema autónomo correspondiente junto con sus distancias y routers desde el informe de gateway exterior. La última es usada para enviar un gateway y seleccionar el mejor gateway exterior para ser usado cuando se envían paquetes a un sistema autónomo particular.

Las tres funciones principales asociadas con el EGP son:

- Adquisición vecina.
- Capacidad vecina.
- Actualización de ruta.

Cada función opera usando un mensaje solicitud-respuesta. Desde cada sistema autónomo se administra y corre por una autoridad diferente, antes de que cualquier información de ruteo sea cambiada, dos routers exteriores asociados a sistemas diferentes deben primero ponerse de acuerdo para cambiar cada información. Este es el rol del procedimiento vecino de adquisición y terminación (neighbour acquisition and termination). Cuando dos routers acuerdan un intercambio, ellos se dicen vecinos.

Cuando un router primero puede cambiar información de ruteo, es enviado un mensaje de pedido de adquisición hacia el EGP en el router apropiado que devuelve también otro mensaje que confirma adquisición o, si esto no se puede aceptar la respuesta es un mensaje que rehusa la adquisición que incluye un código de error.

Una vez que una relación de vecino ha sido establecida entre 2 routers, y por lo tanto, sistemas autónomos, ellos periódicamente confirman su relación. Esto no está hecho para cambios específicos de mensajes o para agregar información de confirmación dentro del encabezado de mensajes de información de ruta normal.

El actual intercambio de información de ruta es llevado fuera por uno de los routers enviando un mensaje de solicitud de espera hacia el otro router preguntándole por la lista de redes (Net IDs) que son capaces vía ese router y otros distantes de él. La respuesta es un mensaje de actualización de ruta que contiene la información requerida. Finalmente, si cualquier mensaje requerido es incorrecto, un mensaje de error es retornado como respuesta con un apropiado código de error.

Al igual que con los otros protocolos IP, todos los mensajes asociados (PDUs) con el EGP son llevados en el campo de datos de usuario de un paquete IP. Todo mensaje EGP tiene el mismo encabezado fijo. El campo de versión define el número de versión del EGP. Los campos de tipo y código conjuntamente definen el tipo de mensaje mientras el campo de condición contiene información de condición de mensaje-dependiente. El checksum, es usado como un sistema de seguridad para mensajes erróneos, es utilizado del mismo modo como el IP. El número sistema autónomo es el número asignado de los sistemas autónomos al que el router remitente es asociado; el número de secuencia es usado para dar respuestas sincronizadas para los mensajes de pedido correspondientes.

Los mensajes de accesibilidad vecina sólo contienen un encabezado con un campo de tipo 5, un código de 0=hello y un 1=escucho. Los mensajes de accesibilidad vecina tienen un campo tipo 3; el número de código define el tipo de mensaje específico. El intervalo HELLO especifica la frecuencia con que los mensajes HELLO deberían ser enviados, el intervalo de espera ejecuta la misma función para mensajes de espera.

Un mensaje de espera tiene un campo de tipo 2. El campo código es usado para trasladar la información de accesibilidad del vecino: un código 0=hello y un código 1=escucho. La dirección IP de la red fuente en ambos, en la espera y los routers actualizados de mensajes indican la red que conecta los 2 routers exteriores. Esto permite a la red central estar conectada con redes múltiples.

El ruteo actualizado de mensajes contienen la lista de redes (Net IDs) que se conectan a través de cada router de los sistemas autónomos arreglados en orden de distancia desde el router exterior que responde. Esto permite que el router requerido sea el mejor router exterior, mediante el cual se envía un paquete para comunicarse dentro de un sistema autónomo.

Border Gateway Protocol (BGP)

BGP es un protocolo de gateway exterior para la comunicación entre routers en diferentes ASs. BGP es el reemplazo para el antiguo EGP que se empleaba en ARPANET. La última versión en desarrollo es la BGP Versión 4, desarrollada para soportar CIDR.

Un sistema BGP intercambia información como alcanzar redes con otros sistemas BGP. Esta información incluye el camino completo de los ASs que el tráfico debe recorrer para alcanzar dichas redes. Esta información es adecuada para construir un grafo de conectividad entre ASs. De esta forma es posible eliminar ciclos y tomar decisiones a la hora de rutear los paquetes.

En primer lugar, es necesario distinguir entre tráfico local y tráfico en tránsito. El primero se origina en el AS y termina en éste. El resto del tráfico se considera en tránsito. Uno de los objetivos de BGP es reducir el tráfico en tránsito.

Un AS puede englobarse en uno de lo siguientes tipos:

Terminal: tiene una única conexión con otro AS. Tiene tan sólo tráfico local.

Multihome: tiene conexión con varios ASs, pero rehusa transportar tráfico en tránsito.

De Tránsito: tiene conexión con más de un AS, y está destinado, bajo ciertas restricciones, a transportar tráfico tanto local como en tránsito.

La topología de Internet queda dividida, pues, en ASs terminales, multihome y de tránsito. Los dos primeros no requieren BGP, sino que pueden utilizar EGP para intercambiar información con otros ASs.

BGP permite realizar un ruteo basado en políticas. Ésta es fijada por el administrador del AS y especificada en los archivos de configuración de BGP. Esta política no forma parte del protocolo, pero las especificaciones de política permiten decidir entre distintos caminos cuando existen varias alternativas. También controla la forma en la que se transmite la información. La política vendrá especificada en función de requerimientos de fiabilidad, seguridad, etc.

BGP se diferencia de RIP en que este emplea TCP como protocolo de transporte. Dos sistemas que empleen BGP establecerán una conexión TCP e intercambiarán sus tablas BGP completas. En conexiones posteriores, se enviarán actualizaciones de dichas tablas.

BGP es un protocolo de vector de distancia, pero al contrario que RIP (que emplea como unidad de medida hops), BGP enumera las rutas a cada destino (la secuencia de ASs al destino). Asi se eliminan algunos de los problemas asociados con RIP. Cada AS tiene asociado un número de 16 bits.

BGP detecta el fallo de un enlace o un host mediante el envio de un mensaje keepalive a sus vecinos de forma regular (aproximadamente cada 30 segundos).

3.6. Classless Inter-Domain Routing: CIDR

¿Qué hacer cuando las direcciones IP asignables están a punto de agotarse y la siguiente versión del estándar simplemente no puede desplegarse a tiempo?. Solución parche: recuperar los millones ya asignadas pero que nunca se usarán.

CIDR (RFCs 1466, 1518 y 1519) ha mantenido el crecimiento de internet reorganizando las direcciones IP de las cinco clases originales en un sistema sin clases. Antes de CIDR, las direcciones IP se asignaban de acuerdo con el número de direcciones de "host" que una compañía u organización necesitaba. Tres de las cinco clases (A, B y C) proporcionaron más de 3 mil millones de hosts utilizables en más de 2 millones de redes, mientras que las restantes eran para multicasting y uso experimental. Sin embargo, a principios de los años 90, esas 2 millones de redes eran devoradas por los ISP que proporcionaban acceso a sus clientes y por compañías que querían conectarse a internet por cuenta propia. Todas las direcciones clase A se habían agotado, y las de clase B sólo se asignaban si se comprobaba su necesidad. Las de Clase C se asignaban a diario, acabándose con tal rapidez que se temía se agotaran en cuestión de meses. Por otro lado, el problema no era sólo la creciente necesidad de direcciones IP, sino que ya se habían asignado y no se utilizaban. Había 125 redes clase A, y todas se subutilizaban. Por ejemplo, America Online era la única compañía del planeta que podía necesitar tal número de direcciones, y sólo si todos sus usuarios estuvieran en línea al mismo tiempo.

Con tantas direcciones en manos de tan pocas organizaciones, era preciso hacer algo para liberar algunas y usarlas de modo más eficaz. CIDR utiliza las mismas máscaras de dirección que se emplean en la división en subredes para crear grupos de direcciones clase C, permitiendo la recuperación de porciones sustanciales de las antiguas redes clase A y B, con lo que se podrían formar más de 10 millones de redes clase C. La desventaja de esta reagrupación

es el mayor tamaño de las tablas de ruteo centrales debido al mayor número de redes que necesitan que se les identifiquen rutas. Además de esto, el tamaño de las tablas de ruteo se debe al mecanismo de asignación de direcciones que se ha seguido, que ha sido estrictamente cronológico y no existe correspondencia entre la ubicación geográfica de una organización o del ISP y su rango de direcciones, por lo que no es posible resumir las tablas de rutas. Por esto, la información se ha de incluir enumerando una a una todas las redes existentes. S i se siguiera una organización jerárquica de direcciones de acuerdo con criterios geográficos, como ocurre en el direccionamiento de la red telefónica, podría resolverse el problema.

Cuadro 10: Agrupación de Redes Clase C en Superredes Asignadas Geográficamente.

Multi regional	192.0.0.0 - 193.255.255.255
Europa	194.0.0.0 - 195.255.255.255
Otros	196.0.0.0 - 197.255.255.255
Noteamérica	198.0.0.0 - 199.255.255.255
Centro y Sudamérica	200.0.0.0 - 201.255.255.255
Anillo Pacífico	202.0.0.0 - 203.255.255.255
Otros	204.0.0.0 - 205.255.255.255
Otros	206.0.0.0 - 207.255.255.255

CIDR resuelve estos problemas de dos formas. La primera consiste en establecer una jerarquía en la asignación de direcciones, que en vez de utilizar un criterio puramente cronológico, que desde el punto de vista geográfico o de topología de la red equivale a una asignación aleatoria, los rangos se asignan por continentes. Inicialmente se ha realizado el reparto de una parte del rango de redes clase C de la manera mostrada en la tabla 10.

Con esta distribución es posible agrupar las entradas en las tablas de ruteo en forma geográfica. Por ejemplo, un router en Chile puede tener una sola entrada en su tabla indicando que todos los paquetes dirigidos a las redes 194.0.0.0 hasta 195.255.0.0 se envíen a la interfaz por la cual accede a Europa, evitando así las 131072 entradas que normalmente harían falta para este rango de direcciones. Sin embargo, este pequeño "arreglo" no es gratis, pues para que las rutas agrupadas sean posibles de rutear, es necesario modificar el software de los routers, ya que en principio no considera el rango 194.0.0.0-195.255.0.0 como una sola red sino como 131072 redes distintas. Por esto, se ha extendido el concepto de subred en sentido contrario, es decir la máscara no solo puede crecer hacia la derecha para dividir una red en subredes, sino que puede crecer hacia la izquierda para agrupar varias redes en una mayor, de ahí que a CIDR se le denomine también supernetting. Es decir, la parte de red de la dirección vendrá especificada por la longitud de la máscara únicamente, y la clasificación tradicional en clases no tiene ningún significado, sólo respetándose dicho significado en el caso de las clases D y E.

La segunda forma de solucionar el problema original, es una consecuencia de lo anterior, consiste en dar a cada organización la posibilidad de solicitar un rango de direcciones, pero que se ajuste a sus necesidades, dándole siempre un rango contiguo y que tenga una máscara

de red común. Por ejemplo, si una empresa requiere una cantidad de 2048 direcciones IP, puede asignársele un grupo de ocho redes clase C consecutivas comenzando en 234.170.168.0 y terminando en 234.170.175.255. Con esto, su dirección de red CIDR será 234.170.168.0 y su máscara 255.255.248.0. Recordar que la máscara por defecto de cada red clase C es 255.255.255.0, de aquí se observa que la máscara se ha corrido hacia la izquierda, perdiendo tres bits. Se debe recordar también que esta reorganización permite transformar una tabla de ruteo que tendrá 8 entradas para llegar al mismo punto en una que tendrá sólo una.

3.7. IPv6

El IETF empezó a trabajar en 1990 para resolver de mejor forma el problema de la falta de direcciones IP, para ello se planteó una nueva versión del protocolo IP llamada inicialmente IPng (Next Generation) y finalmente designada como IPv6 (versión 6) y especificada en los RFCs 1883 y 2460. Los objetivos de diseño planteados fueron los siguientes:

- Establecer un espacio de direcciones que no se agote en el futuro cercano.
- Reducir el tamaño de las tablas de ruteo y simplificar el protocolo para permitir a los routers procesar los paquetes más rápidamente.
- Ofrecer mecanismos que permitan incorporar fácilmente en el protocolo medidas de seguridad usando encriptación.
- Permitir un mejor manejo de los diferentes tipos de servicio, para poder ofrecer garantías de QoS y para permitir el uso de aplicaciones en tiempo real.
- Facilitar el uso de aplicaciones multicast.
- Permitir la movilidad de un host sin necesidad de cambiar su dirección.
- Contemplar un mecanismo que permita extender el protocolo en el futuro.
- Permitir la compatibilidad del protocolo nuevo con el viejo.

El tamaño de las direcciones IP a utilizar en IPv6 fue bastante discutido, y se plantearon alternativas que iban desde ocho hasta 20 bytes. Finalmente, la decisión final adoptó un protocolo con direcciones de 16 bytes.

IPv6, al igual que IPv4, ofrece un servicio de datagramas sin garantías, es decir, "best effort". Sin embargo algunas opciones que permiten ofrecer calidad de servicio. Por otro lado, se debe hacer notar que IPv6 no es realmente compatible con IPv4 pues utiliza un formato de encabezado diferente, sin embargo, con pequeñas modificaciones puede lograrse compatibilidad. La implantación del nuevo protocolo se realiza en forma gradual mediante la creación de redes isladas con IPv6. Para la interconexión de estas islas a través del backbone IPv4 se utiliza tunneling. La red experimental así formada se conoce como 6Bone (http://www.6bone.net) y empezó a funcionar en 1996.

Los protocolos de ruteo se han tenido que modificar para tener en cuenta las características propias y el nuevo formato de direcciones que utiliza IPv6; así se ha creado por ejemplo RIPv6 y OSPFv6.

Las principales características de IPv6 son:

- Direcciones de 16 bytes, suficiente para todo futuro uso previsible.
- Encabezado simplificado, pasando de 13 a 8 campos, lo que permite disminuir el procesamiento en los routers.
- Mejor soporte de los campos opcionales del encabezado.
- Se han considerado los aspectos de seguridad como parte fundamental del protocolo.
- Mayor facilidad para especificar el tipo de servicio.

3.7.1. Direcciones en IPv6

Las direcciones IPv6 están compuestas por 16 bytes. Los primeros bits identifican el tipo de dirección, de manera análoga a IPv4. Sin embargo, existen ahora muchas clases de direcciones, pero no todas tienen asignado el mismo rango, y la mayoría están reservadas para usos futuros. Además, se ha previsto un rango específico para las direcciones IPv4, de esta forma, cualquier dirección IPv4 puede incluirse en un datagrama IPv6.

Una parte del espacio de direcciones se reservó para distribución geográfica, de manera similar a como se hace actualmente con CIDR. Otra parte se reservó para repartir direcciones por proveedor. Se ha contemplado la posibilidad de que Internet evolucione hacia una red que interconecte las redes de los grandes proveedores a nivel mundial, siendo secundaria en este caso la ubicación geográfica. Para este caso se contempló una estructura de direcciones jerárquica con varios niveles.

Para las direcciones multicast se previó un rango específico, y en la dirección multicast se reservó un campo de 4 bits que permite especificar el ámbito que se pretende tenga la emisión. No se ha previsto ninguna dirección específica para broadcast, ya que esto se considera un caso particular de multicast. Además de envíos unicast, multicast y broadcast pueden hacerse envíos anycast, en los que un paquete se envía a un miembro cualquiera de un grupo, sin importar ni especificar a cual. Esto permite, por ejemplo, acceder a un servidor multihomed haciendo balance de carga entre varias interfaces, o por aquella que esté mas cerca del solicitante. También facilita configuraciones redundantes donde un determinado servicio puede ser entregado por más de un servidor.

También se consideró el caso de un rango de direcciones de significado local, equivalentes a las direcciones privadas, para casos en que por razones de seguridad se quiera estar completamente aislado del exterior.

La notación de las direcciones IPv6 es la siguiente: se escriben en ocho grupos de cuatro dígitos hexadecimales, separados por dos puntos. Por ejemplo:
`8000:0000:0000:0000:0123:4567:89AB:CDEF`. Para abreviar la gran cantidad de ceros que tenga una dirección se puede utilizar una notación abreviada, en la que los ceros a la izquierda

pueden omitirse, y además, si uno o más grupos tienen todos los dígitos a cero se pueden omitir poniendo en su lugar dobles dos puntos. Volviendo al caso anterior: `8000::123:4567:89AB:CDEF`. Para evitar ambigüedades, la notación abreviada :: sólo puede utilizarse una vez en una dirección.

Ya que el campo dirección en IPv6 es más largo, se puede reservar los seis últimos bytes de la dirección para incluir una parte local globalmente única en la dirección, que típicamente es una dirección MAC IEEE (esto es similar al direccionamiento usado en ATM), lo que permite la autoconfiguración de los nodos, pues éste fija la parte local de su dirección y a partir de la dirección contenida en su tarjeta de red y escucha por el cable para que el router le informe de la parte de red, configurando automáticamente al nodo y garantizando que la dirección es única.

3.7.2. Encabezado IPv6

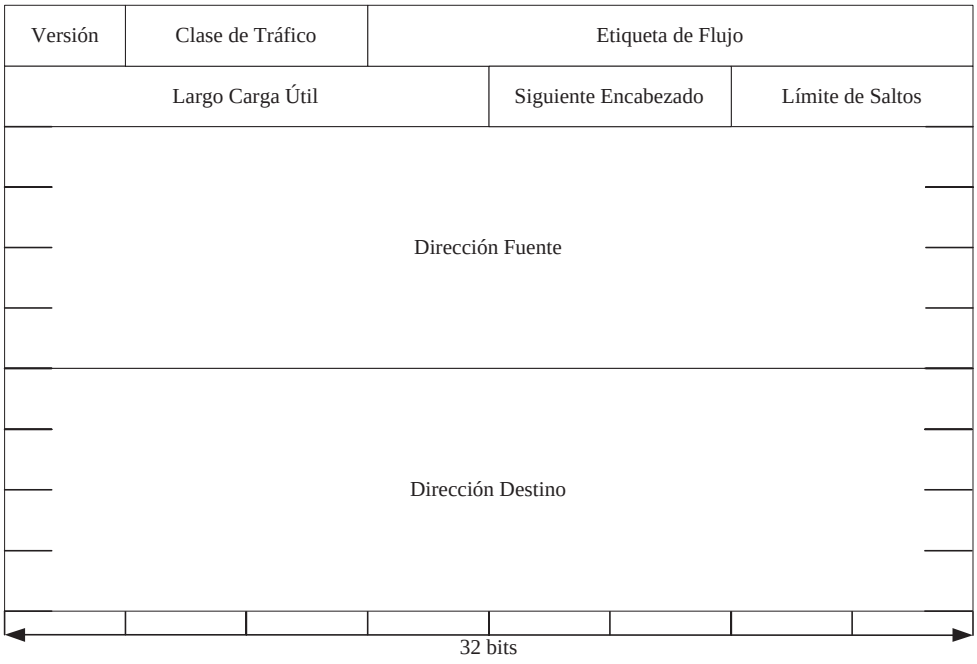

Figura 32: Encabezado del Paquete IPv6.

Los campos que tiene el encabezado IPv6 se observan en la figura 32 y su descripción es la siguiente:

Versión 4 bits y siempre vale 6. Este campo debería distinguir las versiones de IP, de forma que todas pudieran identificarse como un mismo protocolo a nivel de enlace con el mismo valor de Ethertype. Sin embargo, en la práctica muchas implementaciones de IPv4 no comprueban este campo sino que suponen que el datagrama es IPv4 cuando

el encabezado de nivel de enlace especifica protocolo IP. Por esto, a pesar de existir el campo versión es necesario asignar a IPv6 un valor propio en el nivel de enlace, como si se tratara de un protocolo diferente de IPv4.

Clase de Tráfico 8 bits utilizados para especificar parámetros de QoS de acuerdo a la especificación de la arquitectura Differenciated Services. Los valores del 0 al 7 indican poca sensibilidad al tiempo lo que permite encolar el tráfico. Los valores del 8 al 15 indican prioridad del tráfico fuera de flujo por lo que no se puede encolar este tipo de tráfico.

Etiqueta de Flujo 20 bits y permite identificar los paquetes que pertenecen a una sesión concreta entre dos hosts, usado típicamente para solicitar una determinada QoS.

Largo Carga Útil 16 bits que indican el tamaño del paquete en bytes, sin considerar los 40 bytes de encabezado. Como el valor máximo codificable es 65535, el paquete máximo será de 65575.

Siguiente Encabezado 8 bits y sirve para indicar si el encabezado está seguido por alguno de los encabezados opcionales. Si no existen opciones, este campo indica el protocolo de nivel de transporte al que pertenece el paquete, utilizando los mismos códigos que en IPv4 (tabla 6).

Límite Saltos 8 bits equivalentes al campo TTL de IPv4, donde el máximo número de saltos especificables es 255.

Dirección Fuente 128 bits para especificar la IPv6 del nodo fuente.

Dirección Destino 128 bits para especificar la IPv6 del nodo destino.

Los campos que han desaparecido son los siguientes: *IHL* pues el encabezado tiene largo fijo, *Protocolo* no es necesario debido a la función del campo Siguiente Encabezado, *Checksum* pues el cálculo del checksum en cada salto disminuye el rendimiento, además de que la posibilidad de que se produzca un error en el nivel de red es baja, y además el checksum protegía sólo el encabezado y no los datos. Todos los campos relativos a fragmentación han desaparecido porque en IPv6 la fragmentación se controla mediante encabezados opcionales. Además, en IPv6 todos los nodos han de aceptar paquetes de 576 bytes como mínimo y sólo se permite la fragmentación en el origen, es decir, el emisor debe generar datagramas suficientemente pequeños para que puedan llegar a su destino sin fragmentaciones adicionales. Normalmente el emisor realizará el Path MTU Discovery, situación habitual en muchas implementaciones de IPv4.

En IPv6 se ha habilitado un mecanismo más flexible y eficiente que en IPv4 para soportar los encabezados opcionales. Éstos aparecen como encabezados adicionales al final del header estándar, y su presencia queda indicada por el campo Siguiente Encabezado, que como ya se estableció, en caso de que no haya opciones indicará el protocolo de transporte. De esta forma, los campos opcionales en IPv6 pueden extenderse cuando se considere necesario. de esta forma, se entrega un mecanismo mediante el cual se puede indicar si las opciones deben

ser procesadas por todos los routers del trayecto o sólo por el último, lo que permite una reducción significativa del trabajo a desarrollar por los routers intermedios cuando se trata de una opción que sólo debe ser procesada por el nivel de red en el host de destino.

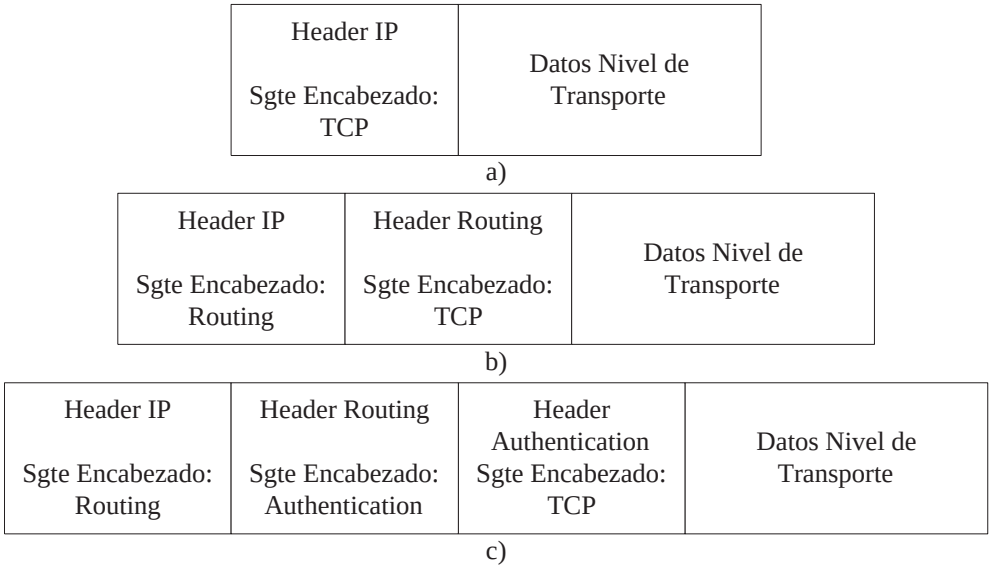

Figura 33: Encabezados Opcionales de un Paquete IPv6 a) Sin Opciones b) Una Opción c) Dos Opciones.

Cada paquete IPv6 incluye encabezados de extensión sólo para los requerimientos necesarios del datagrama. Cada encabezado de base y opcional contiene un campo Siguiente Encabezado para indicar el tipo del siguiente encabezado (ver figura 33). Para extraer toda la información del encabezado de un datagrama IPv6 se requiere entonces procesar secuencialmente todos los encabezados. Los ruteadores intermedios no necesariamente necesitan procesar todos los encabezados. Algunos tipos de encabezados posibles son los siguientes:

Salto-a-Salto entregará información que debe ser examinada por todos los routers por los que pase el datagrama. Hasta el momento se ha definido sólo una opción a este encabezado, y permite especificar datagramas de longitud superior a 64 KBytes, que pueden llegar a tener hasta 4 GBytes.

Routing realiza las funciones combinadas de Strict y Loose Source Routing de IPv4. El máximo número de direcciones que puede especificarse es de 24.

Fragment utilizada cuando se deba fragmentar un datagrama. El mecanismo utilizado es similar al de IPv4, con la diferencia de que en IPv6 sólo se permite la fragmentación en el origen. De esta forma, se simplifica notablemente la complejidad de proceso en los routers.

Authentication permite el uso de encriptación para incorporar un mecanismo de firma digital por el cual el receptor del paquete puede estar seguro de la autenticidad del emisor.

Encrypted Security Payload permite el envío de información encriptada que sólo pueda ser leída por el destinatario. La encriptación afecta sólo a los datos, ya que ésta ha de ser leída e interpretada por cada router por el que pasa.

3.8. IP Clásico sobre ATM

El transporte de cualquier protocolo de red en el modelo de sobrecapas de ATM envuelve dos aspectos básicos: encapsulación de paquetes y resolución de direcciones.

Encapsulación de Paquetes

Para poder permitir la reutilización de conexiones, debe existir un método para que un nodo que recibe un paquete de un nivel superior sepa que tipo de paquete ha recibido a través de la red ATM y a que aplicación debe pasarlo; por lo tanto, el paquete debe tener un prefijo con un campo de multiplexión. La figura (34)

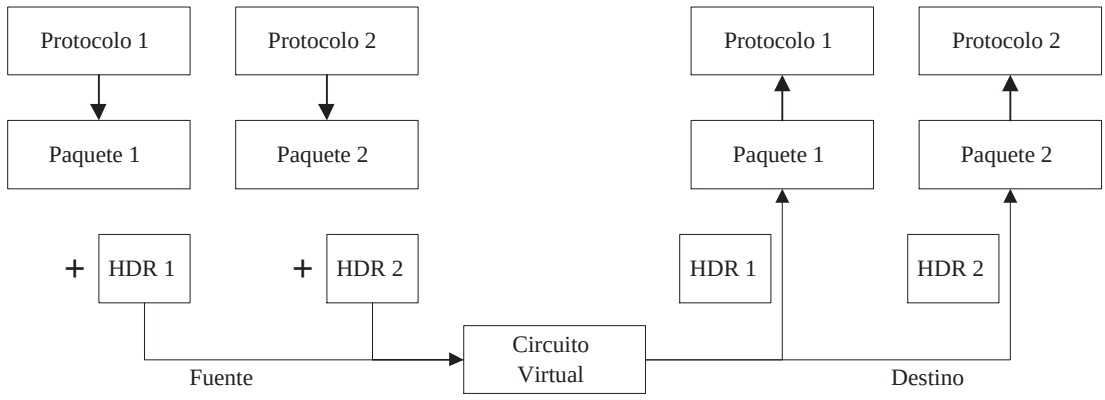

Figura 34: Conceptos de Encapsulación de Paquetes y Reutilización de Conexiones.

El RFC 1483 define dos métodos para hacer esto:

Encapsulación LLC/SNAP: en este método múltiples protocolos pueden ser transportados en una sola conexión con el tipo de identificador de paquete encapsulado en el header LLC/SNAP estándar. Todas las conexiones que usan estas encapsulaciones terminan en el nivel LLC dentro del sistema final, pues es allí donde ocurrió la multiplexación de paquetes.

Multiplexación de VC: en este método un sólo protocolo es transportado por la conexión ATM, con el tipo de protocolo implícitamente identificado en el establecimiento de la

conexión. No se necesita ni transporta un campo de multiplexación o de tipo de paquete. Este tipo de encapsulación es usada actualmente por LANE.

Resolución de Direcciones

Si se considera el caso de dos routers conectados por una red ATM, en el que uno de ellos recibe un paquete a través de una interfaz LAN. En primer lugar revisará su tabla de direcciones para averiguar el siguiente salto. Si este resulta ser por medio de la interfaz ATM, entonces necesitará consultar una tabla de resolución de direcciones para determinar la dirección ATM del siguiente router.

El protocolo "IP Clásico sobre ATM" ha sido definido en el RFC 1577 para que soporte la resolución dinámica de direcciones. Además introduce el concepto de subred lógica IP (LIS) que es un conjunto de nodos IP que se conectan a una red ATM y que comparten la misma subred IP.

Para ejecutar la resolución de direcciones dentro del LIS, cada LIS soporta un servidor ATMARP mientras que los nodos o clientes LIS son configurados con la dirección ATM del servidor ATMARP. Al inicializarse un nodo, en primer lugar se establece una conexión con el servidor ATMARP. Éste al detectar la conexión transmite una petición de ARP inversa y obtiene la dirección IP y ATM del nodo, que almacena en su tabla. Así, cada nodo que desee resolver alguna dirección enviará una petición ATMARP al servidor, que devolverá una respuesta ATMARP si conoce la dirección solicitada, caso contrario devuelve un ATM_NAK.

Una vez obtenida la dirección ATM el cliente LIS puede establecer la conexión con el nodo destino. La figura (35) muestra la situación expuesta.

Next Hop Routing Protocol (NHRP)

NHRP se construye sobre el modelo de IP clásico, sustituyendo el concepto de LIS por el de red Non-Broadcast Multiple-Access (NBMA), esto es, una red tecnológicamente similar a ATM, Frame Relay o X.25, que permite a múltiples dispositivos unirse a la red, pero que no permite fácilmente el mecanismo de broadcast de las LANs.

La red consiste de un conjunto de nodos que se asocian a la misma red NBMA y que no están física o administrativamente restringidos para comunicarse directamente entre sí. En lugar de los servidores ARP, NHRP usa servidores NHRP (NHS), que mantienen un cache de tablas de "resolución de siguiente salto", con las direcciones de los nodos asociados a un NHS en particular o para un conjunto de direcciones IP alcanzables a través de routers servidos por NHS. Los nodos son configurados con la dirección de su NHS y registran sus direcciones con él.

NHS puede ser operado en dos modos. El primero es el *modo servidor* en que cada NHS dentro de la red NBMA es configurado estáticamente con las direcciones de los destinos servidos por cada NHS en la red. El segundo es el *modo fábrica* donde en NHS conoce los destinos servidos por otros NHS a través del intercambio de información de ruteo intra e inter dominios. La forma de operación es, en cualquiera de los dos casos, transparente para el usuario.

La forma en que opera el protocolo es la siguiente: cuando un nodo necesita transmitir un paquete a través de la NBMA necesita resolver alguna dirección ATM en particular; para esto,

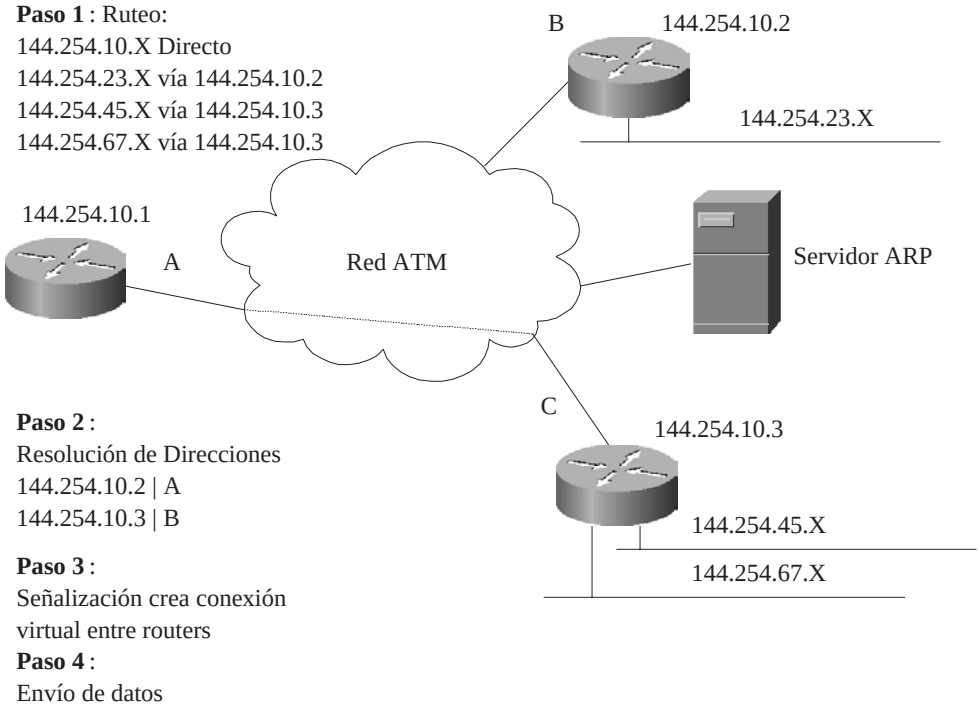

Figura 35: Ruteo A Través de ATM con el Modelo Clásico.

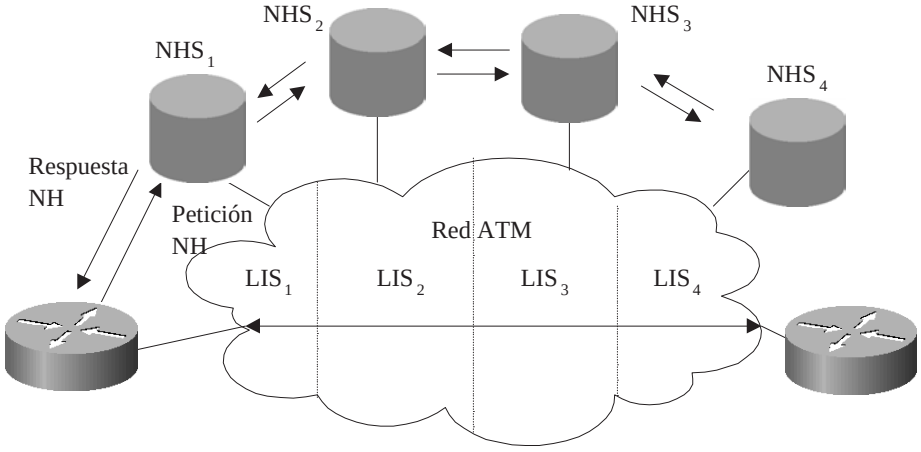

Figura 36: Modo de Operación de NHRP.

formula un petición NH a su NHS. Estos paquetes, al igual que todos los mensajes NHRP,

son enviados en paquetes IP.

Si la dirección de destino es servida por ese NHS, entonces retornará la dirección ATM solicitada, si no, el NHS consultará su tabla y determinará el siguiente NHS en la trayectoria y reenviará el paquete de petición. Este algoritmo es usado hasta que el NHS alcanzado conozca la dirección solicitada.

Este nodo retorna una respuesta NH, que, típicamente, viaja en orden inverso usando los mismos NHSs por los que llegó. Al llegar al nodo origen de la petición, éste puede establecer una conexión directa para transmitir datos. La razón para que el paquete de respuesta NH viaje por el mismo camino de ida que de vuelta es para que cada NHS intermedio aprenda la dirección solicitada, mejorando el desempeño y disminuyendo la latencia. La figura (36) muestra el modo de operación de NHRP en forma esquemática.

NHRP utiliza también un número de características opcionales, que incluyen grabación de rutas para detectar loops y retrasos dentro de la NBMA.

4. Tecnologías WAN

4.1. Introducción

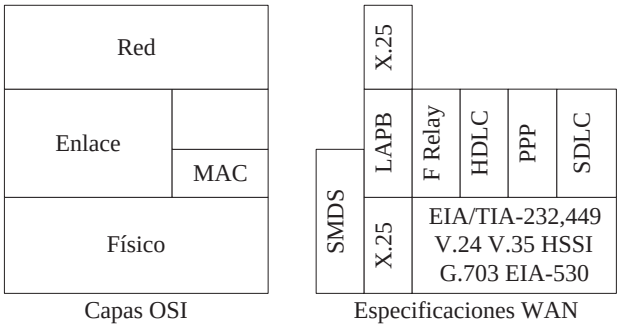

Figura 37: Tecnologías WAN y Su Relación con las Capas OSI

Una WAN es una red de comunicación de datos que tiene una cobertura geográfica relativamente grande y suele utilizar las instalaciones de transmisión que ofrecen compañías portadoras de servicios como las telefónicas. Las tecnologías WAN operan en las tres capas inferiores del modelo OSI. La figura 37 muestra algunas de las tecnologías WAN y el nivel OSI en el que operan.

4.2. Enlaces Punto-a-Punto

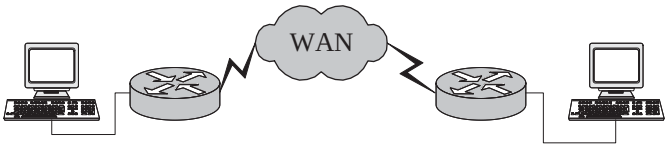

Figura 38: Esquema de Enlace Punto a Punto WAN

Un enlace punto a punto proporciona una única trayectoria entre dos nodos distantes, a través de una red de transporte que típicamente es provista por alguna empresa de servicios. A este tipo de conexión sea les llama también líneas privadas, debido a que la trayectoria establecida permanente y fija para cada red remota a la que se llega utilizando el enlace WAN. Las compañías que proveen el servicio reservan varios enlaces punto a punto para uso exclusivo del cliente, proporcionando dos tipos de conexiones: transmisión de datagramas y transmisión de ráfagas de datos. La figura 38 muestra un enlace punto a punto típico en una WAN.

4.3. Conmutación de Circuitos y de Paquetes

La conmutación de circuitos es un método de conmutación WAN en el que se establece, mantiene y termina un circuito físico dedicado a través de una red de transporte para cada sesión de comunicación. Al igual que los enlaces punto a punto, los circuitos conmutados manejan principalmente dos tipos de transmisiones: de datagramas y de ráfagas de datos. Este tipo de comunicación es bastante utilizada por las compañías de comunicaciones para la interconexión de enlaces, y su forma de operar es muy similar a la de una llamada telefónica normal. ISDN es un ejemplo simple y cotidiano de tecnología WAN de conmutación de circuitos. La figura 39 muestra un ejemplo de este tipo de tecnología.

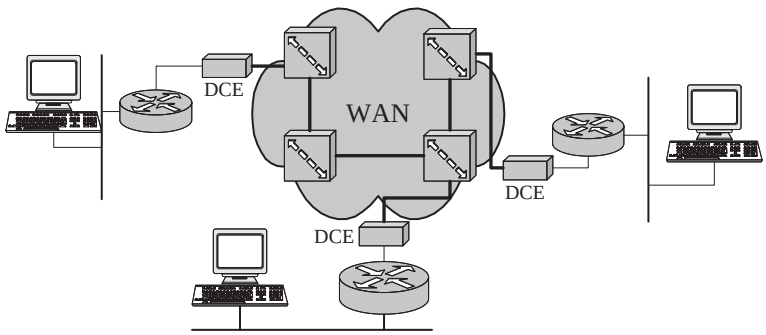

Figura 39: Conexión WAN de Circuitos Conmutados

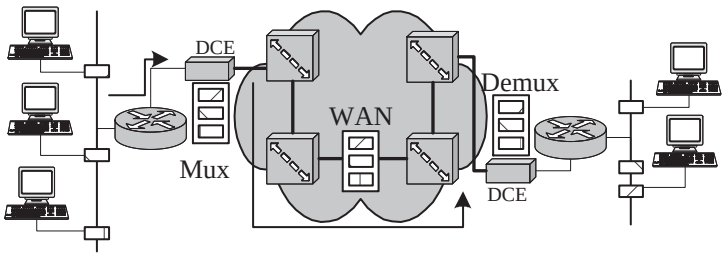

Figura 40: Conexión WAN de Paquetes Conmutados

La conmutación de paquetes es un método de conmutación WAN en el que los dispositivos de la red comparten un único enlace punto apunto para transferir los paquetes desde el origen hasta el destino a través de la red de transporte. Se utiliza multiplexaje estadístico para permitir que los dispositivos compartan los circuitos. ATM, Frame Relay y X.25 son ejemplos de tecnología WAN de conmutación de paquetes, y la figura 40 muestra la transferencia de datos en una red de este tipo.

4.4. Circuitos Virtuales WAN

Un circuito virtual es un circuito lógico creado para asegurar una comunicación confiable entre dos dispositivos de red. Existen dos tipos de circuitos virtuales: los conmutados o SVCs y los permanentes o PVCs. Los primeros se establecen de forma dinámica por demanda y se terminan al finalizar la transmisión. Debido a eso se tienen tres fases o etapas en la comunicación: el establecimiento del circuito (que implica la creación de un circuito virtual entre origen y destino), la transferencia de datos entre los nodos finales, utilizando en circuito virtual establecido, y la terminación del circuito que implica la desconexión. Por otro lado, los PVCs son establecidos de forma permanente, y sólo constan de la fase de transmisión de datos.

Los SVCs son utilizados en situaciones donde la transmisión de datos es esporádica, debido a que éstos incrementan demasiado el ancho de banda utilizado producto de las fases de establecimiento y terminación del circuito. Su principal ventaja es que disminuyen los costos asociados con la disponibilidad constante de un circuito virtual. Los PVCs son utilizados en

situaciones donde la transferencia de datos entre los dispositivos es constante. Con los PVCs s e disminuye el uso de ancho de banda asociado con el establecimiento y terminación de los circuitos virtuales, pero se incrementan los costos debido a la constante disponibilidad del circuito virtual.

4.5. Servicios de Marcado

Los servicios de marcados son un método de interconectividad WAN cuyas implementaciones más comunes son los servicios de ruteo de marcación en demanda y el de respaldo de marcación.

El ruteo de marcación por demanda o DDR es una técnica por medio de la cual un router puede iniciar y terminar de manera dinámica, una sesión de conmutación de circuitos a medida que las estaciones terminales de transmisión lo requieran. El router se configura para que considere un cierto tipo de tráfico como interesante (como el tráfico de algún protocolo en particular) y el resto como no interesante. Cuando el router recibe tráfico interesante destinado a la red remota, se establece un circuito y se transmite a destino en forma normal. Si se recibe tráfico no interesante, y ya estaba establecido el circuito en ese momento, ese tráfico también se rutea a destino. El router maneja un timer que se reinicia sólo cuando se recibe tráfico interesante. Sin embargo, el circuito se termina si el router recibe tráfico no interesante antes de que el timer expire. De la misma forma, si se recibe tráfico no interesante y no existe ningún circuito, el router elimina el tráfico. El DDR se utiliza como reemplazo para enlaces punto a punto y servicios WAN multiacceso conmutado.

La implementación de respaldo de marcación es un servicio que activa una línea serial de respaldo bajo determinadas condiciones. La línea serial secundaria puede actuar como un enlace de respaldo que se utiliza cuando el enlace principal falla o como una fuente que proporciona ancho de banda adicional cuando la carga en el enlace principal alcanza un cierto umbral. El respaldo de marcación proporciona protección contra la degradación del desempeño y el tiempo fuera de servicio de una WAN.

4.6. Dispositivos WAN

Switch WAN

Corresponde a un dispositivo multipuerto de interconectividad de redes que se utiliza en las redes de transporte. Por lo general, estos dispositivos conmutan tráfico como el de Frame Relay, X.25 y SMDS y operan en la capa de enlace de datos. La figura 41 muestra dos router ubicados en los extremos remotos de una WAN que se se encuentran conectados a través de switches WAN.

RAS

Un RAS o servidor de acceso remoto actúa como un punto de concentración para conexiones de marcación hacia adentro y hacia afuera. La figura 42 muestra una conexión RAS.

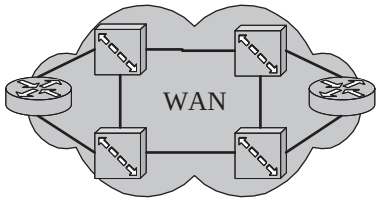

Figura 41: Switches WAN Interconectando Routers

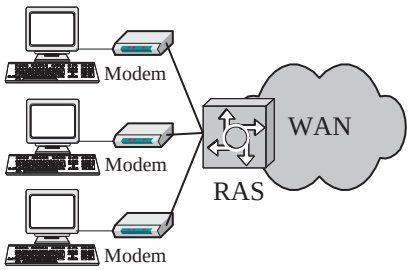

Figura 42: RAS Conectando Múltiples Clientes a una WAN

Módem

Un módem es un dispositivo que interpreta señales analógicas y digitales, permitiendo de

esta manera que los datos se transmitan a través de líneas telefónicas. En el punto de origen las señales digitales son convertidas a una forma apropiada para su transmisión a través de equipos de comunicación análogos. En el destino, las señales analógicas on convertidas de nuevo a su forma digital original.

CSU/DSU

Una CSU (Unidad de Servicio de Canal)/DSU (Unidad de Servicio de Datos) es un dispositivo de interfaz digital que adapta la interfaz física de un DTE, como un nodo final, a la interfaz del dispositivo DCE, como un switch, en una red conmutada de transporte. La CSU/DSU también proporciona la temporización de la señal para la comunicación entre los dispositivos.

Adaptador ISDN

Un adaptador de terminal ISDN es un dispositivo que se utiliza para conectar la BRI de ISDN con otras interfaces. Un adaptador de terminal es, en esencia, un módem ISDN.

4.7. Encapsulado y Tunneling

En las conexiones WAN se desea enviar paquetes de un protocolo determinado a través de una red de otro tipo, sabiendo que en el lado del receptor se dispone de una red del mismo protocolo que el emisor. Por ejemplo, al utilizar ATM como transporte de datos TCP/IP, lo que se hace es introducir los paquetes IP en el campo de datos de una celda ATM. La técnica descrita en el ejemplo anterior se denomina encapsulado o tunneling, ya que la unión puede verse como un túnel que permite intercambiar paquetes de un protocolo determinado de forma que no sean "vistos" por el protocolo intermedio. Los túneles se utilizan en Internet para interconectar las zonas con routing multicast constituyendo la red MBone. También se utilizan túneles para interconectar las zonas de Internet que funcionan con el protocolo IPv6, constituyendo la red 6Bone.

Recientemente se ha definido en Internet un estándar para la creación de túneles denominado L2TP (Layer 2 Tunneling Protocol, RFC 2661). Esto permite la creación de redes privadas virtuales (VPN, Virtual Private Network) a través de una red pública como Internet, mejorando notablemente las características de la comunicación desde el punto de vista de seguridad.

El encapsulado o tunneling supone una pérdida de rendimiento, ya que el paquete viaja con doble cantidad de encabezados. Sin embargo, puede ser una solución muy interesante debido a su sencillez cuando se trata de enviar poco tráfico, o para conexiones temporales.

4.7.1. Virtual Private Network (VPN)

Aunque el desempeño de Internet es, generalmente, una barrera para usarla como WAN para la mayoría de las aplicaciones críticas, Virtual Private Network permite enviar datos importantes a través de Internet en forma segura y eficaz.

En una configuración VPN, los clientes y servidores que componen la red virtual se conectan a Internet de manera tradicional: dial-up, ISDN, líneas dedicadas, cable-modem, etc. Cada nodo de la "red dentro de la red" encripta los datos que envía a otras ubicaciones de la red virtual. A medida que los datos encriptados atraviesan Internet, se ven como porciones de datos sin sentido y si algún intruso trataran de detectar estos datos, no podrá leer el contenido sin poseer las claves para desencriptar los datos.

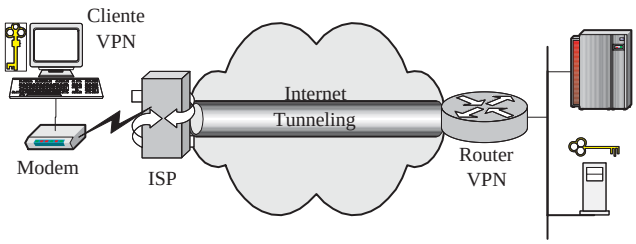

Figura 43: Conexión VPN Entre Cliente Conectado a ISP y Router VPN de una Compañía.

Esta solución es atractiva para las compañías, ya que el acceso a Internet es significativamente menos costoso que el uso tradicional de líneas dedicadas. Se pueden usar VPNs para comunicar sucursales o habilitar clientes móviles, o acceder a servidores desde cualquier ubicación.

Al unir dos sitios con una configuración VPN, cada punto de entrada de la debe tener un dispositivo de acceso a VPN, por ejemplo, un firewall o router que soporte VPN. Las claves de encriptación de VPN son compartidas por clientes y servidores. Estas claves permiten a los nodos en la VPN encriptar datos para que sólo puedan ser leídos por otros miembros de las mismas redes virtuales.

En VPN se utilizan distintos esquemas de encriptación. Uno de los más populares son L2TP (Layer 2 Tunneling Protocol) de Microsoft y Cisco, un standard que está siendo desarrollado por IETF (Internet Engineering Task Force). En este, como en cualquier otro esquema de encriptación por intercambio de claves, éstas necesitan ser distribuidas a los clientes remotos y sitios para permitir la interoperabilidad.

El flujo de datos de la encriptación y desencriptación en la VPN es una tarea muy intensiva con respecto a CPU. Cuando los datos llegan al nodo VPN, se debe controlar que los datos provienen de otro nodo de la red virtual. Si es así, el nodo receptor (router, firewall, o unidad VPN dedicada) debe desencriptar los datos antes de pasarlos a su destino en la red local. Por lo tanto, al diseñar una VPN, la interoperabilidad debe ser la consideración más importante y deben dimensionarse adecuadamente los dispositivos de entrada.

5. Nivel de Transporte

5.1. Introducción

El nivel de transporte es el nivel más importante desde el punto de vista de las comunicaciones. La tarea de este nivel es proporcionar un transporte de datos confiable y económico desde el origen al destino, independiente de la red o redes físicas en uso.

Esta capa proporciona un servicio a los procesos de la capa de aplicación, en el caso de internet y a los de sesión en el caso de OSI. Para lograr este objetivo la capa de transporte hace uso de los servicios de la capa de red.

El hardware o el software de la capa de transporte que se encarga del trabajo se llama entidad de trasporte. Esta entidad puede estar en el kernel del sistema operativo, en un proceso de usuario independiente, en un paquete de biblioteca que forma parte de las aplicaciones de la red o en la tarjeta de interfaz de la red. En algunos casos, la red portadora puede proporcionar servicio de transporte confiable, en cuyo caso la entidad de transporte reside en máquinas especiales de interfaz en la orilla de la subred a la que se conectan los hosts.

En el nivel de transporte existen dos tipos de servicios: orientados a la conexión y no orientado a la conexión. A pesar de esto, lo normal es trabajar con servicios orientados a la conexión donde se tienen, como siempre, tres fases: establecimiento, transferencia de datos y desconexión. Además, tanto el direccionamiento como el control de flujo son semejantes a los de las otras capas. Lo mismo ocurre con los servicios no orientados a la conexión.

Así como la unidad básica de intercambio de información a nivel de enlace se llama frame, la del nivel de red es llamada paquete, la unidad de transferencia del nivel de transporte se llama TPDU (Transport Protocol Data Unit) en la nomenclatura OSI, mensaje en el caso de usarse el protocolo UDP (servicio no orientado a conexión) y segmento en el caso de usarse el protocolo TCP (servicio orientado a conexión). De todas formas, se debe comentar que no existe una nomenclatura clara y general para la unidad mínima del nivel de transporte.

La necesidad de un nivel de transporte es debida a que los usuarios no tienen control sobre la subred, que es de lo que se encarga el nivel de red. Es por este motivo que no se pueden resolver los problemas de un mal servicio usando mejores routers o incluyendo una porción mayor del manejo de errores en la capa de enlace de datos. La única posibilidad es poner el nivel de transporte encima del nivel de red para mejorar la calidad del servicio. Con esto, si a la mitad de una transmisión se informa a una entidad de transporte que su conexión de red ha sido terminada abruptamente, sin indicación de lo sucedido a los datos actualmente en tránsito, la entidad de transporte puede establecer una nueva conexión de red con la entidad de transporte remota. Usando esta nueva conexión de red, la entidad puede enviar una solicitud a su igual preguntando qué datos llegaron y poder reiniciarse desde la interrupción la transmisión de los datos que no hayan llegado.

La existencia del nivel de transporte hace posible que el servicio de transporte sea más confiable que el de red. El nivel de transporte puede detectar y compensar paquetes perdidos y datos alterados. Es más, las primitivas del servicio de transporte pueden diseñarse de modo que sean independientes de las primitivas del servicio de red, que pueden variar según las redes, pudiendo escribir programas de aplicación usando un estándar de primitivas que puedan

trabajar en una variedad amplia de redes, sin tener que preocuparse por las interfaces de subred. Por lo tanto, esta capa cumple la función clave de aislar las capas superiores respecto de la tecnología, el diseño y las imperfecciones de la subred.

Entre las funciones que realiza la capa de transporte se pueden contar:

- Se encarga de la comunicación entre dos nodos, independizándola de como funciona la red.
- Lograr que la información llegue de la máquina A a la máquina B libre de errores y en orden(el nivel de red se encargaba de la comunicación entre el nodo A y el nodo B).
- Dividir o segmentar los datos que llegan desde el nivel superior.
- Multiplexar varias conexiones de transporte sobre una misma conexión de red para reducir el costo.
- Determinar el tipo de servicio que debe dar a la capa superior. Este se determina cuando se establece la conexión.
- Administrar varios enlaces simultáneos entre varias máquinas.
- En los host multiproceso, puede haber múltiples conexiones, en el header de este nivel se indica a qué conexión pertenece cada mensaje.
- Establecer y liberar las conexiones a través de la red y del control de flujo entre hosts.

5.1.1. Direccionamiento

Cuando un proceso de aplicación desea establecer una conexión con un proceso de aplicación remoto, debe especificar a cuál debe conectarse. El método que normalmente se emplea es definir direcciones de transporte en las que los procesos pueden estar a la escucha de solicitudes de conexión. Estas direcciones son los TSAP (Transport Service Access Point).

La dirección TSAP puede ser una dirección jerárquica en la que se pueden distinguir, como en un número telefónico, que la dirección consiste de una secuencia de campos usados para dividir en partes el espacio de direcciones. La alternativa al espacio jerárquico de direcciones es el espacio plano, donde se necesitaría un servidor de nombres que tomara direcciones de puerto como entrada y devolviera direcciones de red como salida.

El esquema de conexión más empleado es el utilizado por los hosts UNIX de Internet, y se llama protocolo inicial de conexión. Este protocolo cada máquina que desea ofrecer servicio a usuarios remotos tiene un servidor de procesos especial, el daemon *inetd*, que actúa como receptor de los servidores y escucha en un grupo de TSAP al mismo tiempo, esperando una solicitud de conexión TCP. Los usuarios potenciales de un servicio emiten una solicitud de conexión especificando la dirección TSAP, llamada *puerto TCP*, del servicio que desean. Tras obtener la solicitud entrante, el servidor de procesos genera el servidor solicitado, permitiéndole heredar la conexión con el usuario existente y el servidor de procesos retorna a escuchar solicitudes nuevas. Por ejemplo, cada vez que un proceso cliente desea conectarse a

un servidor para leer el correo electrónico, se hace la llamada a la dirección IP del servidor de correos, pero para poder acceder al servicio POP3, que es uno de los protocolos que permite leer el correo, se debe especificar la dirección TSAP que accesa al servicio. En este caso, la dirección TSAP o puerto TCP de conexión es el 110. Así, si se desea hacer una conexión a un servidor de correos POP3 puede ejecutarse el programa telnet de la siguiente forma: telnet A.B.C.D 110 donde A.B.C.D es la IP del servidor.

Muchas implementaciones TCP/IP disponen de una API (Application Programming Interfaces) para la programación de aplicaciones denominada *socket*. Un socket es una estructura de software que opera como un punto terminal de comunicaciones en un dispositivo de red. Los sockets son una interfaz multiprotocolo, es decir, soporta TCP, UDP y otros protocolos. Los sockets son la API más extendida en programación de aplicaciones TCP/IP y forman un estándar de facto. Existen implementaciones para muchos sistemas operativos. La filosofía básica de los sockets deriva directamente del sistema de entrada/salida de UNIX, con ampliaciones que permiten por ejemplo a un proceso servidor ponerse "a la escucha".

5.1.2. Primitivas de Servicio de Transporte

Las primitivas del servicio de transporte permiten a los usuarios del nivel el acceso al servicio de transporte. El servicio de transporte es parecido al servicio de red, pero existen algunas diferencias importantes. La principal es que la intención del servicio de red es modelar el servicio ofrecido por las redes reales, que pueden perder paquetes, por lo que generalmente no es un servicio confiable. El servicio de transporte orientado a la conexión, en cambio, sí es confiable, aunque las redes tengan errores el nivel de transporte proporciona un servicio confiable por una red no confiable.

Otra diferencia entre el servicio de red y el de transporte radica en los usuarios a los que se dirigen los servicio. El servicio de red lo usan sólo las entidades de transporte, pocos usuarios llegan a este nivel. Sin embargo, muchos programas pueden usar primitivas de transporte, con lo que el servicio de transporte debe ser cómodo y sencillo de usar.

Las primitivas de transporte son ejecutadas por la entidad de transporte enviando un paquete al servidor. Encapsulado en la carga útil de este paquete hay un mensaje del nivel de transporte para la entidad de transporte del servidor.

5.2. Elementos de Protocolos de Transporte

El nivel de transporte se parece al nivel de enlace en que debe ocuparse de la comunicación extremo a extremo. Por ejemplo, debe ocuparse del control de errores (incluyendo mensajes perdidos o duplicados) y el control de flujo. Aunque las técnicas que se aplican son parecidas, existen importantes diferencias entre los niveles, motivadas por el hecho de que en el nivel de enlace hay un sólo hilo físico (o su equivalente) entre las dos entidades comunicantes, mientras que en el nivel de transporte hay toda una red. Las mayores diferencias entre el nivel de transporte y el de enlace son las siguientes:

- El retardo que se observa en el nivel de transporte es normalmente mucho mayor y sobre todo más variable (mayor jitter) que en el de enlace.

- En el nivel de enlace el medio físico entre las dos entidades tiene una capacidad de almacenamiento de información normalmente muy reducida y siempre la misma. En el de transporte los routers intermedios pueden tener una capacidad considerable y esta puede variar con el estado de la red.
- En el nivel de enlace se asegura que los frames llegarán al receptor en el mismo orden que han salido del emisor (salvo que se pierdan, en cuyo caso no llegarán). En el nivel de transporte esto es cierto sólo cuando se utiliza un servicio orientado a conexión en el nivel de red. Si se utiliza un servicio no orientado a conexión el receptor podría recibir los datos en orden distinto al de emisión.
- En el nivel de enlace las dos entidades se ven directamente (suponiendo una comunicación dúplex) lo que permite que el emisor sepa en todo momento si el receptor está operativo, y el receptor sabe que los datos recibidos corresponden todos a una misma sesión del emisor. En el nivel de transporte la comunicación es indirecta, el emisor podría enviar datos, quedar fuera de servicio y más tarde entrar en funcionamiento otra vez. Si no se adoptan las medidas oportunas el receptor podría recibir todos esos datos sin siquiera percatarse de que corresponden a dos sesiones distintas del emisor o incluso podrían pertenecer a dos usuarios distintos.

Establecimiento de una Conexión

Para establecer una conexión el cliente emite un TPDU de petición de conexión, y el servidor responde con un TPDU de aceptación y a partir de ese momento puede empezar el intercambio de datos. Sin embargo, existen una serie de complicaciones a considerar.

En el nivel de transporte puede haber una gran fluctuación en el tiempo que tardan en llegar los TPDUs a su destino. Los TPDUs pueden perderse o llegar duplicadas, ya que si el emisor no recibe confirmación reenviará el mismo TPDU pasado el timeout. Por ejemplo, un cliente intercambia una serie de TPDUs con un servidor, y cuando ya ha terminado la transacción cierra la sesión. Segundos más tarde desde otro origen aparecen la misma secuencia de TPDUs del cliente duplicadas que llegan al servidor de nuevo. Éste realizaría la misma transacción otra vez.

Para evitar este tipo de problemas, se utiliza para establecer la conexión el mecanismo conocido como three-way handshake. La idea es que el servidor sólo aceptará la conexión después de haber pedido al cliente confirmación de que desea realizarla. En principio esto por sí solo no resuelve el problema, ya que cabría pensar que después del TPDU de petición inicial duplicada la red le entregue al servidor la TPDU de confirmación, también retrasada.

La solución a este problema es el siguiente: tanto el cliente como el servidor utilizan un protocolo de ventana deslizante para el envío de las TPDUs, para lo cual emplean un número de secuencia. A diferencia del número de secuencia del nivel de enlace, el del nivel de transporte emplea rangos mayores. Por ejemplo, en TCP el número de secuencia se almacena en un campo de 32 bits, con lo que es un número módulo de hasta 4 Gigas. Tanto el cliente como el servidor eligen de forma aleatoria o pseudoaleatoria el valor inicial del número de secuencia que van a utilizar, cada uno por separado para cada sentido de la comunicación. El cliente informa al servidor en su primer TPDU del número de secuencia elegido, y por su parte

el servidor le responde en otro TPDU con el número de secuencia elegido por él, incluyendo en ésta un ACK piggybacked del TPDU recibido. De esta forma, si el servidor recibe un TPDU de petición de conexión vieja responderá con un TPDU al cliente en la que pondrá en el campo ACK el número de secuencia recibido. Cuando la respuesta llegue al cliente éste verá que ese número no corresponde con ninguna conexión que él tuviera pendiente de confirmación, por lo que rechazará la conexión. El servidor por su parte esperará recibir en el campo ACK de la siguiente TPDU un valor que corresponda con el que él ha enviado en la anterior.

Esta técnica evita también el riesgo de que un proceso cliente que cae por algún motivo utilice la misma conexión cuando reaparece más tarde, ya que normalmente el nuevo proceso intentará utilizar un número de secuencia diferente. Esta es una medida de seguridad ya que el nuevo proceso cliente podría pertenecer a otro usuario.

Generalmente se establece una vida máxima para los TPDUs en la red, y de esta forma se reduce el riesgo de recibir duplicados retrasados. Cuando un nodo cae y vuelve a subir se recomienda esperar al menos el tiempo de vida de un TPDUs antes de activar el nivel de transporte. Así, es imposible que un TPDU de la sesión anterior pueda aparecer por alguna parte cuando se inicia la sesión nueva. En Internet el tiempo de vida máximo recomendado de las TPDUs es de 2 minutos, y se controla mediante el campo TTL en el datagrama IP.

Una vez establecidos los números de secuencia es posible utilizar para el intercambio de TPDUs cualquier protocolo de ventana deslizante. A diferencia del nivel de enlace, en el nivel de transporte se suelen numerar bytes no frames, ya que el tamaño de los TPDUs puede ser muy variable. Para las retransmisiones se puede utilizar tanto retroceso n como repetición selectiva.

Terminación de una Conexión

Una conexión puede terminarse de forma simétrica o asimétrica. La terminación asimétrica es unilateral, es decir uno de los dos hosts decide terminar y termina la conexión en ambos sentidos. En la terminación simétrica cada host corta la conexión únicamente en el sentido en el que emite datos. Se puede considerar entonces la terminación simétrica como dos circuitos simplex donde cada uno es controlado por el emisor.

La terminación asimétrica se considera anormal y puede provocar la pérdida de información, ya que cuando un host ha enviado un TPDU de desconexión ya no acepta más datos. Entretanto, el otro host podría haber enviado un TPDU de datos que no será aceptado.

En la terminación simétrica el host 1 invita al host 2 a desconectarse mediante un *DISCONNECT REQUEST*. El host 2 responde con otro *DISCONNECT REQUEST*, al cual el host 1 responde con *TPDU ACK* y cierra la conexión. Por su parte, el host 2 cerrará la conexión al recibir el *ACK*. Por este mecanismo se asegura que no se pierden TPDUs en ruta ya que ambos hosts tienen aviso previo de la desconexión y dan su conformidad explícitamente. Este mecanismo supone el intercambio de tres mensajes de forma análoga al proceso de conexión, por lo que también se denomina three-way handshake. No existe forma fiable de terminar la conexión en menos mensajes sin correr el riesgo de perder datos.

Si se pierde algún TPDUs de desconexión el mecanismo de handshake falla, pues los hosts se quedan esperando eternamente la respuesta. Para evitar esto se utiliza un mecanismo de timeouts que resuelve el problema reenviando el TPDU perdido si se trata de un *DISCON-*

NECT REQUEST, o cerrando la conexión por timeout cuando lo que se ha perdido es el *ACK*.

Existen muchas circunstancias que pueden provocar que una conexión se quede medio abierta. Por ejemplo, un host puede quedar fuera de servicio sin previo aviso y el otro, que tenía una conexión abierta con él, quedar a la espera sin saber que ha ocurrido. Para resolver estas situaciones se prevee normalmente un tiempo máximo durante el cual una conexión puede estar abierta sin tráfico, pasado ese tiempo los hosts se envían mensajes de prueba (denominados keep-alive en TCP) para comprobar que el otro lado aún responde. Los valores de timeout para el envío de mensajes keep-alive son grandes, por ejemplo, la documentación de TCP sugiere 2 horas como valor por defecto.

Control de Flujo y de Buffers

El control de flujo en el nivel de transporte es fundamental, ya que la velocidad con que los datos llegan al receptor puede ser muy variable al intervenir multitud de factores.

Si se utilizan protocolos de ventana deslizante, entonces la asignación de buffers estática para cada conexión no es apropiada, debido a que el número de conexiones simultáneas puede variar mucho al no haber una interfaz física asociada a cada conexión.

Por este motivo la asignación de espacio para buffers en el nivel de transporte tiene dos características: primero el lugar espacio de buffers es común y compartido por todas las conexiones, ya sean entrantes o salientes. Segundo, el reparto del espacio entre las conexiones activas se hace de forma dinámica de acuerdo con las necesidades. En todo momento cada conexión tiene asignado un espacio para emisión y uno para recepción. El de emisión está ocupado con TPDUs pendientes de ser enviados o de confirmación, y el de recepción tiene una parte ocupada con TPDUs recibidos pendientes de ser aceptadas por el nivel de aplicación, y otra libre reservada para TPDUs que puedan llegar del otro host.

Otra diferencia con respecto al nivel de enlace es que, mientras que el tamaño de los frames suele ser constante para una conexión física dada, el tamaño de los TPDUs puede ser muy variable. Para optimizar la utilización del espacio se asignan segmentos de buffer de longitud variable. Para una máxima flexibilidad en este sentido tanto los números de secuencia como los tamaños de ventana cuentan generalmente bytes, no TPDUs.

La parte de buffer que el receptor tiene reservada para TPDUs que puedan llegarle es anunciada al emisor regularmente, para que éste sepa que cantidad de datos esta dispuesto a aceptar el receptor. Este espacio puede fluctuar mucho con el tiempo en función de la actividad que tenga esa y el resto de conexiones que mantenga el host. Con este modo de funcionamiento el receptor realmente controla la situación, ya que si indica una ventana cero el emisor tendrá que esperar y no enviarle datos mientras el receptor no le anuncie una ventana mayor.

Multiplexación Generalmente el nivel de transporte es el encargado de multiplexar la diferentes conexiones solicitadas por el nivel de aplicación en una única conexión a nivel de red. Esto se conoce como multiplexación hacia arriba, ya que visto en el modelo de capas supone que varias direcciones del nivel de transporte confluyan en una única dirección del nivel de red. En redes no orientadas a conexión, como IP, el nivel de transporte suele ocuparse de

multiplexar el tráfico de las diferentes aplicaciones y usuarios en una única dirección a nivel de red.

5.3. Protocolos TCP y UDP

5.3.1. Protocolo TCP

TCP (Transmission Control Protocol) es el protocolo de transporte confiable utilizado en Internet en el nivel o capa de transporte. Este protocolo ha adquirido su popularidad gracias a las características que presenta:

- Protocolo orientado a conexión: es decir, las aplicaciones solicitan la conexión al destino y luego usan esta conexión para entregar y transferir los datos, garantizando que estos serán entregados sin problema.
- Punto a Punto: una conexión TCP tiene dos extremos, que son los entes que participan en la comunicación, es decir, emisor y receptor.
- Confiabilidad: TCP garantiza que los datos transferidos serán entregados sin ninguna pérdida, duplicación o errores de transmisión.
- Full duplex: los extremos que participan en una conexión TCP pueden intercambiar datos en ambas direcciones simultáneamente.
- Conexión de inicio confiable: el uso del three-way handshake garantiza una condición de inicio confiable y sincronizada entre los extremos de la conexión.
- Conexión de finalización aceptable: TCP garantiza la entrega de todos los datos antes de la finalización de la conexión.

Debido a que TCP, al igual que UDP, está en la capa de transporte, necesita de valerse de IP para el envío de sus segmentos o mensajes. De esta manera, IP trata al mensaje TCP como la información que debe entregar y en ningún momento intenta interpretar su contenido, como generalmente se hace al pasar un mensaje de una capa a otra inferior. Los extremos de la conexión son identificados por puertos, lo garantiza que se puedan establecer múltiples conexiones en cada host y que los puertos puedan estar asociados con una aplicación o un puerto directamente. De lo anterior se desprende que los routers o cualquier dispositivo de nivel tres sólo puede observar los encabezados IP (nivel 3) para el reenvío de los datagramas, y nunca interpretarán los datos de un nivel superior, pues esto supone violar el modelo de capas. Por lo tanto, TCP en la máquina destino, es el encargado de interpretar los mensajes TCP, después de recibirlos de la capa de red, quien previamente le ha quitado el encabezado IP.

TCP usa diversas técnicas para proveer la entrega confiable de datos. Estas técnicas permiten a TCP recobrarse de errores como paquetes perdidos, duplicados, retardo, diferentes velocidades de transmisión entre nodos y congestión.

Paquetes perdidos. TCP usa confirmación positiva con retrasmisión para lograr la entrega de datos confiable. De este modo, el receptor envía mensajes de control de confirmación (ACK) al emisor para verificar la recepción exitosa de la información. A su vez, el emisor inicializa un timer al transmitir la información. Si el timer expira antes que la confirmación llegue, el emisor debe retransmitir la información inicializando un nuevo timer.

Paquetes duplicados. Si el receptor recibe un paquete duplicado no lo toma en cuenta y procede a su descarte, ya que éste habrá sido tomado y marcado como recibido.

Retardo de paquetes. Si un paquete no es recibido y el siguiente sí, el receptor no mueve la ventana deslizante hasta que el segmento faltante sea recibido. De esta manera el receptor al no recibir el ACK correspondiente al paquete retrasado lo reenvía.

Diferentes velocidades de transmisión. Al establecer la conexión TCP, tanto el emisor como el receptor indican cual es su capacidad de almacenamiento intermedio (léase buffers) para acordar cual será la velocidad a la cual la transmisión se llevará a cabo.

Congestión. TCP implementa una política en la cual mantiene una ventana para medir la congestión, cada vez que un temporizador expira, ésta ventana es reducida. Para la decisión de envío de datos, el emisor toma en cuenta el tamaño de esta ventana para crear el tamaño de la ventana deslizante de datos.

Para proveer transparencia, cada aplicación entrega arbitrariamente toda la información como un flujo de datos, luego TCP se encarga de separar esta información en segmentos, cada uno de los cuales tiene a lo más el tamaño de un paquete IP. El flujo dado por la aplicación es numerado por la cantidad de bytes transferidos, y cada uno de estos segmentos contiene un número de secuencia de los bytes de información. Así, el receptor envía un segmento con el número de secuencia de la información confirmada, no de los segmentos. Los ACKs son acumulativos, de esta manera un ACK puede ser la confirmación de varios segmentos.

Para poder sintonizar el timeout de TCP, éste debe estar basado en el round trip time (RTT), ya que si es menor que éste se creará un tráfico innecesario y no habrá comunicación entre los extremos de la conexión. Sin embargo, existe un problema: el emisor no puede saber de antemano en RTT de ningún paquete antes de la transmisión. Debido a esto, el emisor usa un timeout de retransmisión (RTO) obtenido de RTTs previos. Esto es un método específico llamado algoritmo de retransmisión adaptativo

$$RTT_{actual} = \alpha RTT_{anterior} + (1 - \alpha) RTT_{medido} RTO = \beta RTT_{actual} \quad (1)$$

donde α debe estar entre 0.8 y 0.9 y β entre 0.1 y 0.2. El RTT es medido observando la diferencia entre el tiempo de transmisión y la llegada de una confirmación.

Debido a que el tráfico excesivo que pueda presentar una red es una de las causas de la pérdida de paquetes, algunos protocolos como TCP, proveen la retransmisión como mecanismo para garantizar la llegada de los mensajes. Esta solución más que una buena solución es un arma de doble filo, ya que la retransmisión excesiva puede contribuir a la congestión.

La pérdida de paquetes es interpretada por TCP como un indicador de congestión. El mecanismo de control de TCP es usado por el nodo emisor para detectar el nivel de congestión y si éste está sobre un cierto nivel umbral considerado el máximo aceptable, la retransmisión de paquetes es reducida. El mecanismo utilizado consiste en que un host envía un paquete, y si una confirmación llega sin perdida, el emisor envía dos paquetes y comienza a aumentar la ventana en potencias de dos. Cuando TCP envía un número de paquetes igual a la mitad del tamaño de una ventana, la tasa de incremento disminuye hasta recibir las confirmaciones de los paquetes enviados.

Cuadro 11: Algunos Puertos TCP/UDP Estándar.

Puerto	Aplicación	Descripción
9	Discard	Descarta todos los datos recibidos (para pruebas)
19	Chargen	Intercambio de strings (para pruebas)
20	FTP-Data	Transferencia de datos en FTP
21	FTP	Intercambio de información de control en FTP
23	Telnet	Sesión de remota en una máquina
25	SMTP	Envío de mails a través de servidor de correos
53	DNS	Consultas y transferencia de datos de servicio de nombres
110	POP3	Lectura de correo electrónico
139	NetBIOS	Intercambio de datos usando NetBIOS en redes locales con Windows

Puertos TCP

Un puerto es un número entero entre 0 y 65535 que especifica la dirección TSAP a la cual se dirige una conexión TCP o UDP. En un mismo host, un número de puerto puede ser utilizado simultáneamente por una aplicación para UDP y por otra para TCP, lo que no plantea ningún conflicto, ya que son TSAPs diferentes.

Por convenio los números 0 a 1023 están reservados para el uso de servicios estándar, por lo que se les denomina puertos bien conocidos (well-known ports). Cualquier número por encima de 1023 está disponible para ser utilizado libremente por los usuarios. En la tabla 11 se algunos algunos de los puertos más utilizados.

Así pues, una conexión de dos entidades usuarias del nivel de transporte se especifica por la combinación: $IP_{emisor} : Puerto_{emisor} \leftrightarrow IP_{receptor} : Puerto_{receptor}$. El programa `netstat` sirve como ayuda para conocer que puertos y con que nodos está conectada una máquina en particular. Por ejemplo:

```
C:\>netstat
Active Connections
  Proto  Local Address Foreign Address              State
  TCP    myPC:1479     DALI:netbios-ssn            ESTABLISHED
```

```
TCP    myPC:2897     lovecraft.die.udec.cl:22     ESTABLISHED
TCP    myPC:2903     lovecraft.die.udec.cl:telnet ESTABLISHED
TCP    myPC:2882     manet.die.udec.cl:smtp       ESTABLISHED
UDP    myPC:2888     vangogh.die.udec.cl:domain   ESTABLISHED
```

En este caso, el nodo local denominado `myPC` está conectado a la máquina `DALI` transfiriendo datos utilizando el Entorno de Red de Windows, el puerto local en el PC es el 1479, mientras que el servicio lo obtiene desde el puerto 139 (NetBIOS-ssn). Además, esta máquina está conectada a `lovecraft.die.udec.cl` mediante `ssh` y `telnet` (puertos 22 y 23) usando los puertos locales 2897 y 2903. También puede observarse una conexión de envío de mail, usando SMTP en el puerto 5 del servidor de correos. Las conexiones son todas TCP, salvo la última que es una consulta DNS hecha al puerto 53 del servidor de nombres usando UDP.

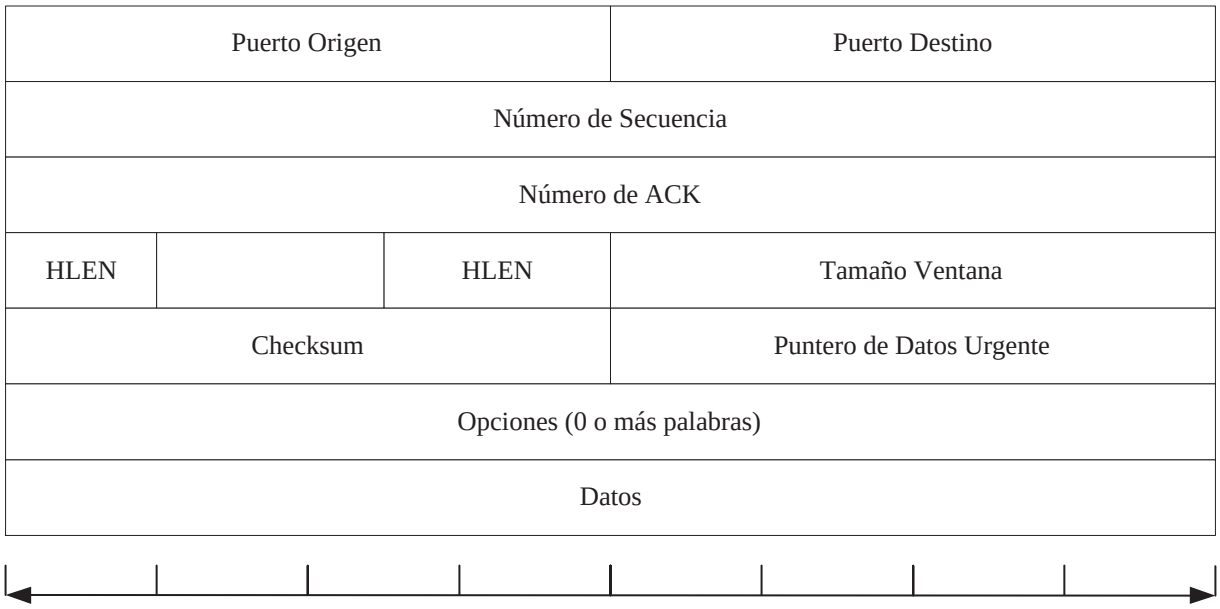

Figura 44: Encabezado de un Mensaje TCP.

Encabezado TCP

La figura 44 muestra el encabezado de un mensaje TCP, la descripción de sus campos es la siguiente:

Puerto Origen y Destino: 16 bits cada uno e identifican los puertos que se van a utilizar en cada host para comunicar con las aplicaciones que intercambian datos.

Número de Secuencia: 32 bits, indica el número de secuencia que corresponde en la conexión al primer byte que se envía en el campo datos de ese segmento.

Número de ACK: 32 bits que apuntan al número de secuencia del primer byte del próximo segmento que se espera recibir del otro lado.

Longitud de Encabezado TCP: 4 bits que especifican el largo del encabezado, en palabras de 32 bits. Este valor no incluye el campo datos, y el campo opciones hace que esta longitud pueda variar.

Bits de Codificación: 6 bits que se presentan a continuación de 6 bits no utilizados. Corresponden a bits flag, cuyo nombre y significado es el siguiente: URG (urgent, sirve para indicar que el segmento contiene datos urgentes, y el campo puntero de datos urgentes contiene la dirección donde terminan éstos), ACK (acknowledgement, indica que en este segmento el campo Número de ACK tiene el significado habitual , de lo contrario carece de significado; en la práctica, el bit ACK esta a 1 siempre, excepto en el primer segmento enviado por el host que inicia la conexión), PSH (push, indica que el segmento contiene datos PUSHed, es decir, que deben ser enviados rápidamente a la aplicación correspondiente sin esperar a acumular varios segmentos), RST (reset, usado para indicar que se debe abortar una conexión porque se ha detectado un error de cualquier tipo), SYN (synchronize, este bit indica que se está estableciendo la conexión y está puesto en 1 sólo en el primer mensaje enviado por cada uno de los dos hosts en el inicio de la conexión) y FIN (finish, indica que no se tienen más datos que enviar y que se quiere cerrar la conexión; se usa ya que para que una conexión se cierre de manera normal cada host ha de enviar un segmento con el bit FIN puesto en 1)

Tamaño de Ventana: 16 bits que indican la cantidad de bytes que se está dispuesto a aceptar del otro lado en cada momento. Mediante este parámetro el receptor establece un control de flujo sobre el flujo de datos que puede enviar el emisor.

Checksum: 16 bits y sirve para detectar errores en el segmento recibido. Estos podrían ser debidos a errores de transmisión no detectados, a fallos en los equipos o a problemas en el software.

Puntero de Datos Urgentes: 16 bits, indican el final de un fujo de datos de tipo urgente, ya que el segmento podría contener datos no urgentes. TCP no marca el principio de los datos urgentes, es responsabilidad de la aplicación averiguarlo.

Opciones: 0 o más palabras que habilitan un mecanismo por el cual es posible incluir extensiones al protocolo. Entre las más interesantes se encuentran las siguientes: tamaño máximo de segmento, uso de repetición selectiva (en vez de retroceso n), uso de NAK (acuse de recibo negativo en caso de no recepción de un segmento), uso de ventana mayor de 64 Kbytes mediante el empleo de un factor de escala.

5.3.2. Protocolo UDP

TCP tiene la robustez y funcionalidades propias de un protocolo de transporte orientado a conexión; sin embargo esa robustez y funcionalidad conllevan una cierta complejidad; por

ejemplo cualquier transmisión de información TCP requiere como mínimo el intercambio de seis mensajes para establecer la comunicación y terminarla; además mientras una conexión existe ocupa una serie de recursos en el host.

En determinadas oportunidades no se requiere toda la funcionalidad que TCP provee en las conexiones, más aún, cualquier transmisión de información TCP requiere como mínimo el intercambio de seis mensajes para establecer la comunicación y terminarla, or lo que este retardo puede llegar a ser significativo para alguna aplicación determinada. Por esto, en algunos casos se prefiere que el nivel de transporte preste un servicio más sencillo, no orientado a conexión y no confiable.

Algunos ejemplos de situaciones en las que es más conveniente un servicio no orientado a conexión son: aplicaciones tiempo real como audio o video, donde no se puede tolerar el retardo producido por los ACK; consultas a servidores en que se requiere el envío de uno o dos mensajes únicamente como es el caso del DNS, etc.

UDP (User Datagrama Protocol) es el protocolo no orientado a conexión de Internet y entre las aplicaciones que utilizan UDP se encuentran TFTP (Trivial File Transfer Protocol), DNS (Domain Name Server), SNMP (Simple Network Management Protocol), NTP (Network Time Protocol), NFS (Network File System) , etc.

Las TPDUs intercambiadas por UDP se denominan mensajes o datagramas UDP. Una de las características más interesantes de UDP es que puede ser utilizado por aplicaciones que necesitan soporte de tráfico multicast o broadcast. Con TCP esto no es posible debido a la naturaleza punto a punto, orientada a conexión del protocolo.

UDP no suministra ningún mecanismo de control de flujo o control de congestión. Cuando lo que se envía es únicamente un mensaje esto es innecesario, ya que presumiblemente un mensaje aislado no creará problemas de congestión y será siempre aceptado en destino. Si se desea enviar un flujo de mensajes, por ejemplo vídeo o audio en tiempo real, se deberán tomar las medidas adecuadas para asegurar la capacidad suficiente en la red y evitar la congestión no excediendo lo solicitado en el momento de hacer la reserva.

En caso de congestión en la red parte de los datagramas serán descartados por la red sin informar por ningún mecanismo al emisor, ni al receptor. En caso de saturación del receptor este sencillamente ignorará los datagramas que no pueda aceptar. En algunos se contemplan a nivel de aplicación mecanismos de control que permiten al receptor detectar si se producen pérdidas (por ejemplo, numerando los datagramas) informando al emisor para que baje el ritmo de emisión si se supera un umbral determinado.

Puertos y Encabezado UDP

De forma similar a los segmentos TCP, los mensajes UDP se dirigen a la aplicación adecuada mediante el puerto de destino, especificado en el encabezado. Análogamente a TCP los puertos UDP se identifican mediante un campo de 16 bits (números entre 0 y 65535). Aún en el caso de coincidir en número con un puerto TCP son TSAPs diferentes. Al igual que en TCP los valores por debajo de 1024 están reservados para los puertos bien conocidos, aunque su significado es diferente en la mayoría de los casos

La figura 45 muestra el encabezado de un mensaje UDP, la descripción de sus campos es la siguiente:

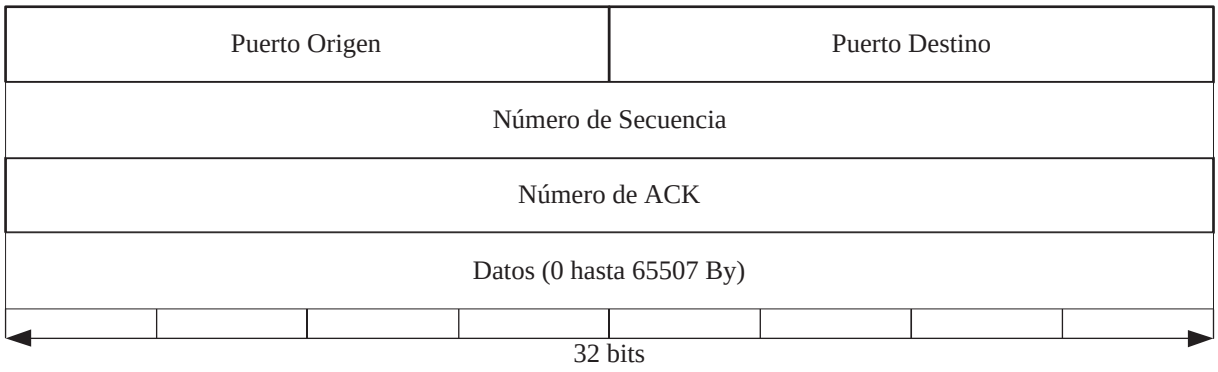

Figura 45: Encabezado de un Segmneto UDP.

Puerto Origen y Destino: 16 bits cada uno, que especifica el puerto de la aplicación que genera y recibe el mensaje. A diferencia de TCP, el campo origen valdrá normalmente cero, salvo que la aplicación solicite una respuesta.

Longitud: 16 bits e indica la longitud del mensaje, incluyendo los campos de encabezado.

Checksum: 16 bits. Su uso es opcional en IPv4 y obligatorio en IPv6, ya que en ese caso se ha suprimido el chechsum a nivel de red. Cuando se envía información en tiempo real su uso puede omitirse. Si la verificación del checksum en el receptor arroja un error, el mensaje es descartado sin notificarlo al nivel de aplicación ni al emisor.

Datos: contiene los datos a transmitir.

De la misma forma que un host o un router pueden tener que fragmentar un datagrama que contenga un segmento TCP, es posible que el host emisor o algún router intermedio tengan que fragmentar un mensaje UDP porque sea mayor que la MTU permitida en la red por la que ha de enviarse. Análogamente a los segmentos TCP la fragmentación ocurre de forma transparente a UDP y el encabezado del mensaje sólo aparecerá en el primer fragmento. En cambio, cada fragmento deberá incluir un nuevo encabezado IP.

Referencias

[1] **Computer Networks**. *Andrew Tanenbaum*. Prentice Hall PTR, Third Edition. 1996.

[2] **Tecnologías de Interconectividad de Redes**. *Merilee Ford, H. Kim Lew, Steve Spanier, Tim Stevenson*. Prentice Hall, Primera Edición. 1998.

[3] **Internetworking LANs and WANs**. *Gilbert Held*. Wiley Communications Technology, First Edition. 1995.

[4] **Comunicaciones y Redes de Computadores**. *William Stalling*. Prentice Hall, Sexta Edición. 2000.

[5] **Tecnologías Emergentes Para Redes de Computadores**. *Uyless Black*. Prentice Hall. 2000.

[6] MundoPC.NET - http://www.ciudadfutura.com/mundopc/redes/indred.htm

[7] **Apuntes Redes de Computadoras (Teoría y Taller)**, *Emilio Hernández*, Universidad Simón Bolivar, Venezuela.

[8] **Apuntes Redes de Datos**, *Eduardo Rivera*, Universidad de Concepción, Chile.

[9] **Apuntes Redes de Datos**, *Marcelo Iribarren*, Universidad de Concepción, Chile.

[10] **Apuntes Redes de Datos**, *Marcelo Maraboli*, Universidad Técnica Federico Santa María, Chile.

[11] **Interconexión de Sistemas Abiertos**, *José Nuñez*, Universidad del País Vasco, España.