CAPÍTULO 6

"Ningún problema verdadero tiene solución. En cada problema grande hay un problema pequeño que lucha por salir. Y… En cada problema pequeño hay un problema grande que lucha por salir."

Leyes de Smith–Hoare–Schainker (Leyes de Murphy)

COMUNICACIONES

Durante el siglo XX, la tecnología clave ha sido la recolección, procesamiento y distribución de información. Entre otros desarrollos, hemos asistido a la instalación de redes telefónicas en todo el mundo, a la invención de la radio y la televisión, al nacimiento y crecimiento sin precedente de la industria de las computadoras, así como a la puesta en orbita de los satélites de comunicación.

A medida que avanzábamos hacia los últimos años de este siglo, se ha dado una rápida convergencia de estas áreas, y además las diferencias entre la captura, transporte almacenamiento y procesamiento de información están desapareciendo con rapidez. Organizaciones con centenares de oficinas dispersas en una amplia área geográfica esperan tener la posibilidad de examinar en forma habitual el estado actual de todas ellas, simplemente oprimiendo un botón. Mientras crece nuestra habilidad para recolectar procesar y distribuir información, la demanda de procesos más sofisticados crece todavía con mayor rapidez.

La industria de informática ha mostrado un progreso espectacular en muy corto tiempo. El "viejo" modelo de tener una sola computadora para satisfacer todas las necesidades de cálculo de una organización se está reemplazando por otro que considera un número grande de computadoras separadas, pero interconectadas, que efectúan el mismo trabajo. Estos sistemas, se conocen con el nombre de redes. Se dice que los sistemas están interconectados, si son capaces de intercambiar información. Esta conexión puede realizarse a través de un alambre de cobre, fibra óptica, láser, microondas o satélites de comunicaciones.

6.1 OBJETIVOS DE LAS REDES

Las redes en general, consisten en "compartir recursos", y uno de sus objetivos es hacer que todos los programas, datos y equipos estén disponibles para cualquier usuario de la red que así lo solicite, sin importar la localización física del recurso y del usuario. En otras palabras, el hecho de que el usuario se encuentre a miles de kilómetros de distancia de los datos, no debe evitar que éste los pueda utilizar como si fueran originados localmente.

Un segundo objetivo consiste en proporcionar una alta fiabilidad, al contar con fuentes alternativas de suministro. La presencia de múltiples CPUs significa que si una de ellas deja de funcionar, las otras pueden ser capaces de encargarse de su trabajo, aunque el rendimiento global sea menor.

Otro objetivo es el ahorro económico. Las computadoras pequeñas tienen una mejor relación costo/rendimiento, comparada con la ofrecida por las máquinas grandes. Estas son, a grandes rasgos, diez veces más rápidas que el más rápido de los microprocesadores, pero su costo es miles de veces mayor. Este desequilibrio ha ocasionado que muchos diseñadores construyan sistemas constituidos por poderosos ordenadores personales, uno por usuario y con los datos guardados en una o más máquinas que funcionan como servidor de archivo compartido.

Este objetivo conduce al concepto de redes con varias computadoras en el mismo edificio. A este tipo de red se le denomina LAN, en contraste con lo extenso de una WAN.

Un punto muy relacionado es la capacidad para aumentar el rendimiento del sistema en forma gradual a medida que crece la carga, simplemente añadiendo mas procesadores.

Otro objetivo del establecimiento de una red, es que puede proporcionar un poderoso medio de comunicación entre personas que se encuentran muy alejadas entre sí.

Una forma que muestra el amplio potencial del uso de redes como medio de comunicación es Internet y el uso del correo electrónico (e–mail), que se envía a una persona situada en cualquier parte del mundo que disfrute de este servicio.

6.1.1 ESTRUCTURAS

Definir el concepto de redes implica diferenciar entre el concepto de redes físicas y redes de comunicación.

Respecto a la estructura física, los modos de conexión y los flujos de datos, etc.; una **Red** la constituyen dos o más computadoras que comparten determinados recursos, sea

hardware (impresoras, sistemas de almacenamiento, etc.) o software (aplicaciones, archivos, datos, etc.).

Desde una perspectiva más comunicativa y que expresa mejor lo que puede hacerse con las redes, podemos decir que existe una red cuando están involucrados un componente humano que comunica, un componente tecnológico (computadoras, telecomunicaciones) y un componente administrativo (institución que mantiene los servicios). Así, a una **Red** más que varias computadoras conectadas, la constituyen personas que solicitan, proporcionan e intercambian experiencias e informaciones a través de sistemas de comunicación.

Las redes deberían ser lo más transparentes posibles, de tal forma que el usuario final no requiera tener conocimiento de la tecnología (equipos y programas) utilizada para la comunicación.

6.1.1.1 TECNOLOGÍAS DE TRANSMISIÓN

Al crear una red, se toman en cuenta dos factores principales: el medio físico de transmisión y las reglas que rigen la transmisión de datos. El primer factor se llama nivel físico y el segundo protocolo.

En el nivel físico generalmente encontramos señales de voltaje que tienen un significado preconcebido. Estas señales se agrupan e interpretan para formar entidades llamadas paquetes de datos. La forma de acceder a estos paquetes la determina la tecnología de transmisión, aceptándose dos tipos:

1. Las redes de tipo **Broadcast** se caracterizan porque todos los miembros (nodos) pueden acceder a todos los paquetes que circulan por el medio de transmisión.
2. Las redes **Point–To–Point** sólo permiten que un nodo se conecte a otro en un momento dado.

6.1.1.2 MODELO CLIENTE/SERVIDOR

En vez de construir sistemas informáticos como elementos monolíticos, existe el acuerdo general de construirlos como sistemas Cliente/Servidor. El Cliente (un usuario de PC) solicita un servicio (por ejemplo imprimir) que un Servidor (un procesador conectado a la LAN) le proporciona. Este enfoque común de la estructura de los sistemas informáticos se traduce en una separación de las funciones que anteriormente formaban un todo. Los detalles de la realización van desde los planteamientos sencillos hasta la posibilidad real de manejar todas las PC´s de modo uniforme.

6.1.1.3 TECNOLOGÍA DE OBJETOS

Otro de los enfoques para la construcción de los sistemas parte de la hipótesis de que deberían estar compuestos por elementos perfectamente definidos, objetos cerrados y materializados haciendo de ellos agentes independientes. La adopción de los objetos como medios para la construcción de sistemas informáticos ha colaborado a la posibilidad de intercambiar los diferentes elementos.

6.1.1.4 SISTEMAS ABIERTOS

Esta definición alude a sistemas informáticos cuya arquitectura permite una interconexión y una distribución fácil. En la práctica, el concepto de sistema abierto se traduce en desvincular todos los componentes de un sistema y utilizar estructuras análogas en todos los demás. Esto conlleva una mezcla de normas (que indican a los fabricantes lo que deberían hacer) y de asociaciones (grupos de entidades afines que les ayudan a realizarlo). El efecto final es que sean capaces de "hablar" entre sí.

El objetivo último de todo el esfuerzo invertido en los sistemas abiertos consiste en que cualquiera pueda adquirir computadoras de diferentes fabricantes, las coloque donde quiera, utilice conexiones de banda ancha para enlazarlas entre sí y las haga funcionar como una máquina compuesta, capaz de sacar provecho de las conexiones de alta velocidad.

Parece lógico suponer que las computadoras podrán trabajar en conjunto cuando dispongan de una conexión entre ellas. Pero ¿Cómo conseguir, sin embargo, que computadoras de diferentes fabricantes en distintos países funcionen en común a través de todo el mundo?. Hasta hace poco, un equipo podía comunicarse con otro de su misma "familia", pero tenía grandes dificultades para hacerlo con un "extraño".

6.1.1.5 EL MODELO OSI

El modelo conceptual OSI (Open System Interconection) es utilizado por, prácticamente, la totalidad de las redes del mundo. Este modelo fue creado por el ISO (International Standard Organization), y consiste en siete niveles o capas donde cada una de ellas define las funciones que deben proporcionar los protocolos con el propósito de intercambiar información entre varios sistemas.

Esta clasificación permite que cada protocolo fuera desarrollado con una finalidad determinada, lo cual simplifica el proceso de implementación. Cada nivel depende de los que están por debajo de él, y a su vez proporciona alguna funcionalidad a los niveles superiores.

Los siete niveles del modelo OSI son los siguientes:

1. **Capa Física:** esta capa tiene que ver con el envío de bits en un medio físico de transmisión y asegura que si de un extremo del medio se envía un 1 (carga eléctrica) del otro lado se reciba ese 1. Brinda los medios eléctricos, mecánicos, de procedimiento y funcionales para activar y mantener el enlace físico entre los sistemas.

   Capa de Enlace: en esta capa se toman los bits que entrega la Capa Física y se agrupan para formar marcos de bits (Frames). Se realiza un chequeo de errores sobre cada frame. Si un marco se pierde o se daña en el medio físico este capa se encarga de retransmitirlo, aunque en ocasiones dicha operación provoca que un mismo marco se duplique en el destino. Dado el caso es obligación detectar tal anomalía y corregirla. También en esta capa se decide cómo acceder al medio físico.
2. **Capa de Red:** se encarga de controlar la operación de la subred (medios físicos y dispositivos de enrutado). Una tarea primordial es decidir cómo hacer que los paquetes lleguen a su destino desde su origen en el 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. A los efectos de la obtención de estadísticas, se registra el tipo y cantidad de paquetes que circulan.

3. **Capa de Transporte:** el objetivo de esta capa es el de 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 (MTU Maximum Transmission Unit) de la interfaz de red, por lo cual es necesario particionarlos y enviarlos en unidades más pequeñas, lo cual da origen a la fragmentación y ensamblado de paquetes cuyo control se realiza en esta capa.

   La última labor importante de la Capa de Transporte es ofrecer un mecanismo de nombrado que sirva para identificar y diferenciar las múltiples conexiones existentes, así como determinar en qué momento se inician y se terminan las "conversaciones"; es decir, en esta capa hay un mecanismo de control de flujo. Por ejemplo, si el usuario "a" en el nodo (A) quiere iniciar una sesión de trabajo remoto en un nodo (B), existirá una conexión que debe ser diferenciada de la conexión que el usuario "b" necesita para transferir un archivo del nodo (B) al nodo (A).
4. **Capa de Sesión:** esta capa ofrece el servicio de establecer sesiones de trabajo entre nodos diferentes de una red, sincroniza y establece 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 perderá la conexión. La solución es que se establezcan puntos de chequeo cada pocos minutos de manera que, si la conexión se rompe, más tarde se pueda reiniciar a partir del punto de chequeo, lo cual ahorra tiempo y permite la finalización de la transferencia.
5. **Capa de Presentación:** esta provee las facilidades para transmitir datos con la sintaxis propia de las aplicaciones o el nodo. En esta capa es posible convertir los datos a un formato independiente de los nodos que intervienen en la transmisión.
6. **Capa de Aplicación:** en esta capa se encuentran las aplicaciones de red que permiten explotar los recursos de otros nodos. Dicha explotación se hace, por ejemplo, a través de una emulación de una terminal que trabaja en un nodo remoto, interpretando una gran variedad de secuencias de caracteres de control que permiten desplegar en la terminal local los resultados, aún cuando éstos sean gráficos. Otra forma de explotación se da cuando se transmite desde una computadora origen que almacena sus archivos en un formato distinto al del destino. Es posible que el programa de transferencia realice las conversiones necesarias de manera que el archivo puede usarse inmediatamente bajo alguna aplicación.

Gráficamente:

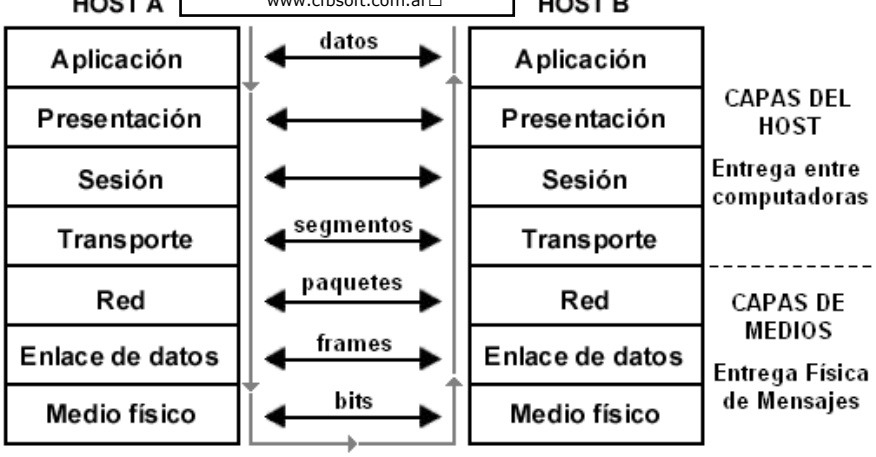

Gráfico 6.1 – Modelo OSI. Fuente: CISCO Networking Academies. Curriculum Online Versión 1.1.

6.1.1.5.1 Transmisión de Datos en el Modelo OSI

Un envío de datos típico bajo el modelo de referencia OSI comienza con una aplicación en un nodo cualquiera de la red. Esta Aplicación genera los datos que quiere enviar a su contraparte en otro nodo.

1. La Capa de Aplicación toma los datos y los encapsula añadiendo un encabezado que puede contener información de control o estar vacío. Envía el paquete resultante a la Capa de Presentación.
2. La Capa de Presentación recibe el paquete y no intenta decodificarlo o separar sus componentes, sino que lo toma como datos y le añade un encabezado con información de control de esta capa.
3. Las Capa de Sesión y de Transporte reciben el paquete, que también son sólo datos para ellas y le añaden un encabezado de control. El resultado es enviado a la capa inferior.
4. La Capa de Red se encarga de enrutar el paquete a su destino.
5. Las Capas de Red, Enlace de datos y Física toman, respectivamente, el paquete que les envía la capa superior y añaden a éste un encabezado definido por el protocolo que corresponde a cada capa y pasan el resultado a la capa inferior.
6. La Capa Física, por último, traducirá el último paquete a las señales apropiadas para que viajen por el medio físico hasta el nodo destino.
7. En el nodo destino comienza el camino inverso; es decir que cada capa quita su encabezado de control y envía el paquete a la capa superior hasta llegar a la de Aplicación en el nodo destino.

Como puede apreciarse, todas las capas, excepto la de Aplicación, procesan los paquetes realizando operaciones que sirven para verificar que el paquete de datos real esté íntegro, o para que éste llegue a su destino sin que los datos sufran alguna alteración.

6.2 PROTOCOLOS DE RED

En las redes, las computadoras deben comunicarse entre sí e intercambiar datos con sistemas operativos y hardware muy distintos.

En el nivel físico, esto se realiza a través de placas de redes, y una conexión entre las mismas. Lógicamente se debe establecer una comunicación "del mismo lenguaje" entre distintos sistemas operativos y placas. Este lenguaje es lo que se llama protocolo.

Algunos protocolos se encargan de transportar datos, mientras que otros se encargan de la comunicación entre computadoras, y otros de convertir correctamente los datos. Así, **Protocolo** es el conjunto de normas (lenguaje de reglas y símbolos) que rige cada tipo de comunicación entre dos computadoras (intercambio de información).

Actualmente existen protocolos para cualquier tipo de comunicación que nos imaginemos; muchos de ellos han caído en desuso y otros se encuentran en su plenitud de utilización. Esto es el producto de una sociedad cada vez más intercomunicada y relacionada, en donde lo importante es que la información llegue a su destino sí, pero también lo es que llegue en las mismas condiciones en que ha sido enviada y en el tiempo previsto.

Algunos de los protocolos mas conocidos y ampliamente difundidos son:

6.2.1 NETBIOS–NETBEUI–NWLINK–WINS

Network Basic Input Output System, es el protocolo más sencillo. Está compuesto por menos de 20 comandos que se ocupan del intercambio de datos. Se ha perfeccionado y ampliado recibiendo el nuevo nombre NetBEUI (NetBIOS Extended User Interface) pero continúa utilizando el juego de comandos del NetBIOS y luego para hacerlo compatible con otros protocolos (como IPX–SPX) se amplio nuevamente recibiendo el nombre de NWLink (NetWare Link).

NetBIOS toma el puerto 137–139 en computadoras que utiliza el sistema operativo Windows© de la empresa Microsoft®. Está considerado el protocolo más fácilmente vulnerable de los existentes, a punto tal que cualquier especialista de seguridad recomienda no utilizarlo.

6.2.2 TCP/IP

En los años 80 una gran cantidad de instituciones estaban interesadas en conectarse a una gran red que se expandía por todo EE.UU. (ahora Internet). Para esto definieron un conjunto de reglas que establecen cómo conectar computadoras entre sí para lograr el intercambio de información.

Actualmente TCP/IP se utiliza ampliamente en la versión 4 (IPv4) que no incluye la seguridad como parte de su construcción. Sin embargo se encuentra en desarrollo (IPv6 o IPSec) que dentro de sus estándares soporta autenticación, integridad y confidencialidad a nivel de datagramas

Basado en las capas del modelo OSI, se definió un conjunto de protocolos de TCP/IP, que consta de 4 capas principales y que se han convertido en un estándar a nivel mundial.

6.2.2.1 LAS CAPAS DEL MODELO TCP/IP

El Transmission Communication Protocol/Internet Protocol es actualmente el protocolo más ampliamente utilizado por su independencia del Sistema Operativo y hardware utilizado. Es un eficaz protocolo orientado por paquetes; es particularmente adecuado como plataforma para protocolos de los más distintos servicios y aplicaciones que se pueden conseguir a través de la red.

TCP/IP no es un único protocolo, sino que en realidad lo que se conoce con este nombre es un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto. Se diferencian cuatro capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI como se muestra en el gráfico 6.2.

Aplicación: Se corresponde con los niveles OSI de Aplicación, Presentación y Sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de archivos (FTP), conexión remota (Telnet) y otros más recientes como el protocolo HTTP (Hypertext Transfer Protocol).

Transporte: Coincide con el nivel de Transporte del modelo OSI. Esta capa está implantada por dos protocolos: el Transmission Control Protocol (TCP) y el User Datagram Protocol (UDP). El primero es un protocolo confiable (reliable) y orientado a conexiones, lo cual significa que ofrece un medio libre de errores para enviar paquetes. El segundo es un protocolo no orientado a conexiones (connectionless) y no es confiable (unreliable). El TCP se prefiere para la transmisión de datos a nivel red de área amplia y el UDP para redes de área local.

Internet: Es el nivel de Red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.

Interfaz de red: correspondiente al nivel de Enlace y Físico de la pila OSI. Los protocolos que pertenecen a este nivel son los encargados de la transmisión a través del medio físico al que se encuentra conectado cada Host, como puede ser una línea punto a punto o una red Ethernet.

La capa inferior, que podemos nombrar como Física respecto al modelo OSI, contiene varios estándares (conocidos con el nombre del IEEE 802.X) que establecen 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.

El TCP/IP necesita funcionar sobre algún tipo de red o de medio físico que proporcione sus propios protocolos para el nivel de enlace de Internet. Por este motivo hay que tener en cuenta que los protocolos utilizados en este nivel pueden ser muy diversos y no forman parte del conjunto TCP/IP. Sin embargo, esto no debe ser problemático puesto que una de las funciones y ventajas principales del TCP/IP es proporcionar una abstracción del medio; de forma que sea posible intercambiar información entre medios diferentes y tecnologías que inicialmente son incompatibles.

Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se transfieren. En TCP/IP cada una de estas unidades de información recibe el nombre de "Datagrama" (datagram), y son conjuntos de datos que se envían como mensajes independientes.

6.2.2.2 FUNCIONAMIENTO

Las aplicaciones de red presentan los datos a TCP. Este divide los datos en trozos (paquetes), y le otorga a cada uno un número. El conjunto de paquetes ordenados pueden representar imágenes, documentos, videos, o cualquier otra información que el usuario desee enviar.

Luego, TCP presenta los datos a IP, quien agrega su información de control (como ser dirección de origen y destino). Si por algún motivo IP no puede entregar algún paquete, TCP pedirá el reenvio de los faltantes. Por último TCP se encarga de reensamblar los paquetes en el orden correcto, basándose en los números asignados previamente.

6.2.2.3 COMPARACIÓN CON EL MODELO OSI

Si bien TCP/IP está basado en OSI, este último no tuvo éxito debido a causas como el momento de su introducción, la tecnología existente en ese momento, malas implementaciones y políticas por parte de los investigadores. Sin embargo OSI es un buen modelo y TCP/IP es un buen conjunto de protocolos y la combinación de ambos es la que permite contar con las comunicaciones que se tienen hoy.

El modelo TCP/IP no tiene bien divididas las Capas de Enlace de Datos, Presentación y Sesión y la experiencia ha demostrado que en la mayoría de los casos son de poca utilidad.

Los estándares 802.X junto con el protocolo IP realizan todas las funciones propuestas en el modelo OSI hasta la Capa de Red. Los protocolos TCP y UDP cumplen con la Capa de Transporte. Finalmente, las aplicaciones ya mencionadas son ejemplos prácticos y reales de la funcionalidad de la Capa de Aplicación.

Gráficamente pueden apreciarse las siete capas del modelo y su relación directa en su implementación sobre el protocolo TCP/IP.

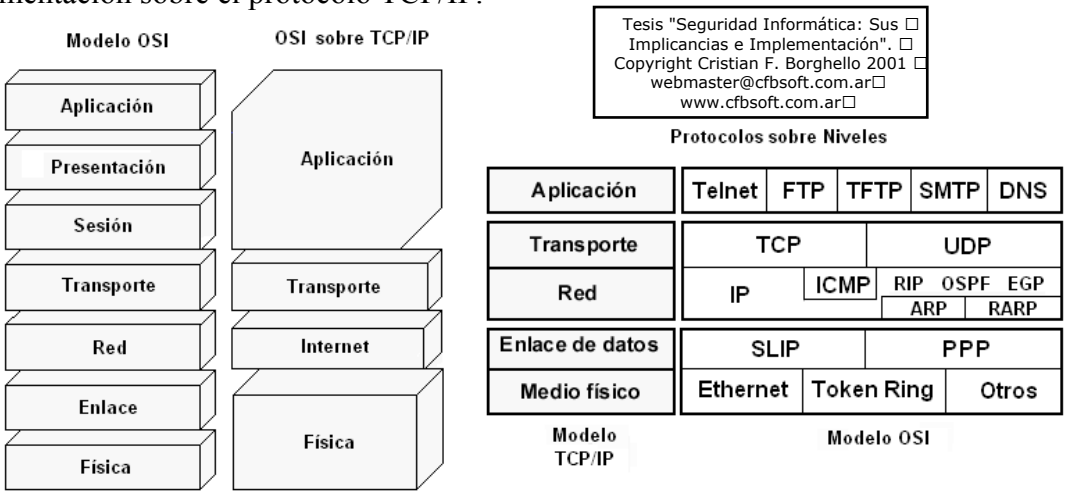

Gráfico 6.2 – Comparación Modelo OSI–TCP

6.2.3 NIVEL FÍSICO DEL MODELO TCP/IP

6.2.3.1 ARP

El Address Resolution Protocol no se dedica al transporte de datos sino a convertir las direcciones IP en direcciones de la red física.

El protocolo consigue la dirección mediante la difusión de un paquete de petición ARP que contiene la dirección IP del sistema destinatario. Todos los ordenadores de la red detectan estas difusiones y aquel que contenga la dirección IP solicitada, la transmitirá al sistema solicitante mediante una respuesta de paquete ARP. Luego el solicitante almacena estas direcciones en una tabla para su uso posterior; y esta tabla además servirá de referencia a otros equipos para evitar la búsqueda de las mimas direcciones.

6.2.3.2 RARP

El Reverse Address Resolution Protocol realiza el trabajo inverso de ARP. Es decir que obtiene la dirección IP a partir de una dirección física.

6.2.4 NIVEL DE DATOS DEL MODELO TCP/IP

6.2.4.1 SLIP

El Serial Link Internet Protocol/Point to Point Protocol brinda una conexión de velocidad aceptable, con al posibilidad de admitir varias conexiones simultaneas con un mismo modem. El mecanismo es sencillo: se llama al proveedor, quien oficia de puente entre su computadora y el resto de la red, y una vez establecida la comunicación se tiene acceso total a los servicios. Es un protocolo sencillo y pequeño, pensando en su fácil implementación; y en la baja velocidad de los enlaces telefónicos, por lo que ha caído en desuso.

Este protocolo apoya solamente IP, no provee detección de errores ni de autenticación y tienen la desventaja de que existen muchas implementaciones incompatibles entre ellas.

6.2.4.2 PPP

El Point to Point Protocol fue desarrollado por el IETF (Internet Engineering Task Force) en 1993 para mejorar algunas deficiencias de SLIP, y crear un estándar internacional.

PPP es un protocolo mucho más amplio, más potente y adaptable. Proporciona un método de enlace bidireccional full dúplex para transportar datagramas multiprotocolo sobre enlaces simples (conexión directa) de un equipo a otro (punto a punto), en cualquier situación sin importar el tipo de conexión, el hardware ni el sistema operativo.

Sus principales características son:

1. Es transparente a las capas superiores.
2. Transmite protocolos IP, IPX, Apple Talk, etc.

3. Es ampliable ya que no fue pensando para solucionar un problema en concreto.

PPP esta dividido en dos subprotocolos:

1. **LCP (Link Control Protocol):** es el encargado de comenzar una conexión (fase abierta), definir como se producirá el intercambio de datos (tamaño de los paquetes, identificación, tiempos de espera, etc.) y de finalizar la conexión (enlace muerto).
2. **NCP (Network Control Protocol):** se encarga de negociar y configura las opciones de los diferentes protocolos de red (IP, IPX, etc.) abriéndolos de a uno por vez. Una vez que un NCP ha alcanzado el estado abierto, PPP transportará los correspondientes paquetes. Cualquier paquete recibido mientras su NCP no esté en el estado abierto es descartado.

6.2.5 NIVEL DE RED DEL MODELO TCP/IP

6.2.5.1 IPX–SPX

El Internetwork Packet Exchange–Sequenced Packet Exchange es el protocolo de nivel de red propietario de NetWare (para su sistema operativo Novell) siendo utilizados en las redes tipo LAN.

6.2.5.2 IP

El Internet Protocol define la base de todas las comunicaciones en Internet. Es utilizado por los protocolos del nivel de transporte (como TCP) para encaminar los datos hacia su destino. IP tiene únicamente la misión de encaminar el datagrama, sin comprobar la integridad de la información que contiene. Para ello se utiliza una nueva cabecera que se antepone al datagrama que se está tratando.

La cabecera IP tiene un tamaño de 160 bits y está formada por varios campos de distinto significado entre los que se destaca el tipo de protocolo de transporte del datagrama, el número de paquete (para su posterior ensamble), la dirección de origen y la de destino, etc.

Es de notar que este protocolo no garantiza la llegada de los paquetes a destino (conexión sin garantía), ni su orden; tan solo garantiza la integridad del encabezado IP. La fiabilidad de los datos deben garantizarla los niveles superiores. También, se trata de una transmisión sin conexión porque cuando se envía el paquete, no se avisa al receptor para que esté preparado (no existe una conexión directa emisor–receptor). De hecho, muchas veces se mandan paquetes a un destino inexistente o que no se encuentra disponible.

El protocolo IP identifica a cada equipo que se encuentre conectado a la red mediante su correspondiente dirección. Esta dirección es un número de 32 bits que debe ser único para cada Host, y normalmente suele representarse como cuatro cifras de 8 bits separadas por puntos (por ejemplo: *205.025.076.223*)

En el nivel IP se definen los siguientes aspectos de intercambio de información:

1. Un mecanismo de direcciones que permite identificar de manera univoca al emisor y al receptor, sin considerar las ubicaciones ni las arquitecturas de las redes a las cuales pertenece cada uno. Este mecanismo permite la universalidad de la red.
2. Un concepto relativo al transporte de los paquetes de datos, para que el mismo llegue al receptor a través de los nodos de las redes involucradas. Dentro de cada red tendrá que haber al menos un receptor (Router) que esté conectado con otra computadora en otra red en el exterior. Los routers reconocen un paquete y comprueban que no sea para alguna maquina conectada a su red y entonces lo mandan a otra, más cercana al destino. Esto se hace sucesivas veces hasta que el paquete llega al router de la red donde se encuentra la computadora destinataria del mensaje.
3. Un formato para los paquetes (cabecera). Con esta, el Router podrá identificar al destinatario del mensaje, ya que como se explico, uno de los datos de la cabecera es el nombre de destino del mensaje.

La dirección IP se utiliza para identificar tanto a la computadora en concreto como la red a la que pertenece, de manera que sea posible distinguir a los ordenadores que se encuentran conectados a una misma red. Con este propósito, y teniendo en cuenta que en Internet se encuentran conectadas redes de tamaños muy diversos, se establecieron cuatro clases diferentes de direcciones, las cuales se representan mediante tres rangos de valores:

1. **Clase A:** son las que en su primer byte tienen un valor comprendido entre 1 y 126, incluyendo ambos valores. Estas direcciones utilizan únicamente este primer byte para identificar la red, quedando los otros tres bytes disponibles para cada uno de las computadoras (Hosts) que pertenezcan a esta misma red. Esto significa que podrán existir más de dieciséis millones de Hosts en cada una de las 126 redes de esta clase. Este tipo de direcciones es usado por redes muy extensas.
2. **Clase B:** estas direcciones utilizan en su primer byte un valor comprendido entre 128 y 191, incluyendo ambos. En este caso el identificador de la red se obtiene de los dos primeros bytes de la dirección, debiendo ser un valor entre *128.001* y *191.254* (no es posible utilizar los valores 0 y 255 por tener un significado especial). Los dos últimos bytes de la dirección constituyen el identificador de la computadora permitiendo, por consiguiente, un número máximo de 64.516 ordenadores en la misma red. Este tipo de direcciones tendría que ser suficiente para la gran mayoría de las organizaciones grandes.
3. **Clase C:** en este caso el valor del primer byte tendrá que estar comprendido entre 192 y 223, incluyendo ambos valores. Este tercer tipo de direcciones utiliza los tres primeros bytes para el número de la red, con un rango desde *192.001.001* hasta *223.254.254*. De esta manera queda libre un byte para el Host, lo que permite que se conecten un máximo de 254 computadoras en cada red.
4. **Clase D:** esta clase se usa con fines de multidifusión a más de un dispositivo. El rango es desde *224.0.0.0* hasta *239.255.235.255*.

Actualmente se planea la utilización de redes **Clase E** que comprenderían el rango desde *240.0.0.0* hasta *247.255.255.255*.

6.2.5.2.1 DNS – Nombres de Dominio

Ya que para el ser humano se hace difícil recordar direcciones IP como *209.89.67.156* se creó lo que dio en llamar DNS (Domain Name Server), el cual es el encargado de convertir la dirección IP en un nombre de dominio generalmente fácil de recordar y viceversa. Así *www.clarin.com* será entendida, merced al servicio de DNS como *110.56.12.106* o *\\Carlos* se convertirá en *10.0.0.33*.

6.2.5.2.2 Puertos

Para acceder desde el nivel de red al nivel de aplicaciones no sirve simplemente indicar la dirección IP; se necesitarán mas especificaciones para que el Host de destino pueda escoger la aplicación correcta. Estas especificaciones harán necesario la definición de **Puerto**. Un puerto se representa por un valor de 16 bits y hace la diferencia entre los posibles receptores de un mensaje.

La combinación Dirección IP + Puerto identifican una región de memoria única denominada **Socket**. Al indicar este Socket, se puede trasladar el paquete a la aplicación correcta (FTP, Telnet, WWW, etc.) y, si además recibe el puerto desde donde fue enviado el mensaje, se podrá suministrar una respuesta.

Actualmente existe miles de puertos ocupados de los 2^{16} = 65535 posibles, de los cuales apenas unos cuantos son los más utilizados y se encuentran divididos en tres rangos:

- Desde el puerto 0 hasta el 1023: son los puertos conocidos y usados por aplicaciones de servidor.
- Desde el 1024 hasta el 49151: son los registrados y asignados dinámicamente.
- Desde el 49152 hasta 65535: son los puertos privados.

Puerto	Aplicación	Protocolo	Descripción
20	FTP–Data	TCP/UDP	Transferencia archivos
21	FTP	TCP	Control Transferencia Archivos
23	TELNET	TCP/UDP	Servicio Remoto
25	SMTP	TCP/UDP	Envío de mails
43	Whois	TCP/UDP	
53	DNS	TCP/UDP	Servicio de Nombre de Dominios
70	Gopher	TCP/UDP	
79	Finger	TCP/UDP	
80	WWW–HTTP	TCP/UDP	World Wide Web
110	POP3 (PostOffice)	TCP/UDP	Recepción de mail
119	UseNet	TCP	Newsgropus de usuarios
137	NetBIOS	UDP	
194	IRC (Internet Relay Chat)	TCP/UDP	Chat
443	HTTPS	TCP	HTTP Seguro vía SSL
750	Kerberos	TCP/UDP	
6667	IRC (Internet Relay Chat)	TCP	Chat

Tabla 6.1 – **Fuente**: *http://www.isi.edu/in–notes/iana/assignments/port–numbers* según RFC 768, RFC 793 y RFC 1060

6.2.5.3 APPLETALK

Este protocolo (de nivel de red) está incluido en el Sistema Operativo de Apple Macintosh desde su aparición, permite interconectar computadoras y periféricos con gran sencillez para el usuario, ya que no requiere ningún tipo de configuración por su parte.

6.2.6 NIVEL DE TRANSPORTE DEL MODELO TCP/IP

6.2.6.1 TCP

El Protocolo de Control de Transmisión (TCP) nació principalmente por la necesidad de una comunicación "segura" entre el emisor y el destinatario del mensaje. Así, las aplicaciones pueden encargarse de su tarea sin preocuparse de la seguridad en la comunicación.

TCP divide el mensaje original en datagramas de menor tamaño (múltiplo de 32 bits), y por lo tanto, mucho más manejables. Los datagramas serán dirigidos a través del protocolo IP de forma individual. El protocolo TCP se encarga, además, de añadir cierta información necesaria al inicio de cada uno de los datagramas (cabecera). Luego, se ocupa de que los datos sean entregados y que los paquetes sean reensamblados correctamente asegurando así que lo que se recibe sea efectivamente lo enviado.

Si ocurriera algún error en la transmisión, TCP se encargará de reenviar los paquetes. TCP sabrá que hubo errores o que el paquete fue entregado correctamente gracias a un paquete de respuesta (acuse de recibo) que envía el destinatario al emisor (para que vuelva a realizar el envio) en donde indica si faltan paquetes, tamaños o datos erróneos, etc.

Las principales características de este protocolo son:

1. **Servicio orientado a conexión:** el destino recibe exactamente la misma secuencia de bytes que envía el origen.
2. **Conexión de circuito virtual:** durante la transferencia, el protocolo en las dos maquinas continua comunicándose para verificar que los datos se reciban correctamente.
3. **Transferencia con memoria intermedia:** la aplicación utiliza paquetes del tamaño que crea adecuado, pero el software de protocolo puede dividir el flujo en subpaquetes o armar uno con un grupo de ellos, independientemente de la aplicación. Esto se realiza para hacer eficiente el tráfico en la red. Así, si la aplicación genera piezas de un byte, el protocolo puede armar datagramas razonablemente más largos antes de hacer el envío, o bien, forzar la transferencia dividiendo el paquete de la aplicación en datagramas más pequeños.
4. **Flujo no estructurado:** se refiere a la posibilidad de envío de información de control de estado junto a los datos propiamente dichos.
5. **Conexión full duplex:** permite la transferencia concurrente en ambas direcciones, sin ninguna interacción. La ventaja es evidente: el protocolo puede enviar datagramas desde el origen al receptor e información de control en sentido opuesto, reduciendo el tráfico en la red.

El Grafico 6.3 detalla la constitución de cada datagrama del protocolo TCP (160–192 bits = 20–24 bytes). Comprender este diagrama es de especial interés para cualquiera que desee manipular datos en una comunicación actual.

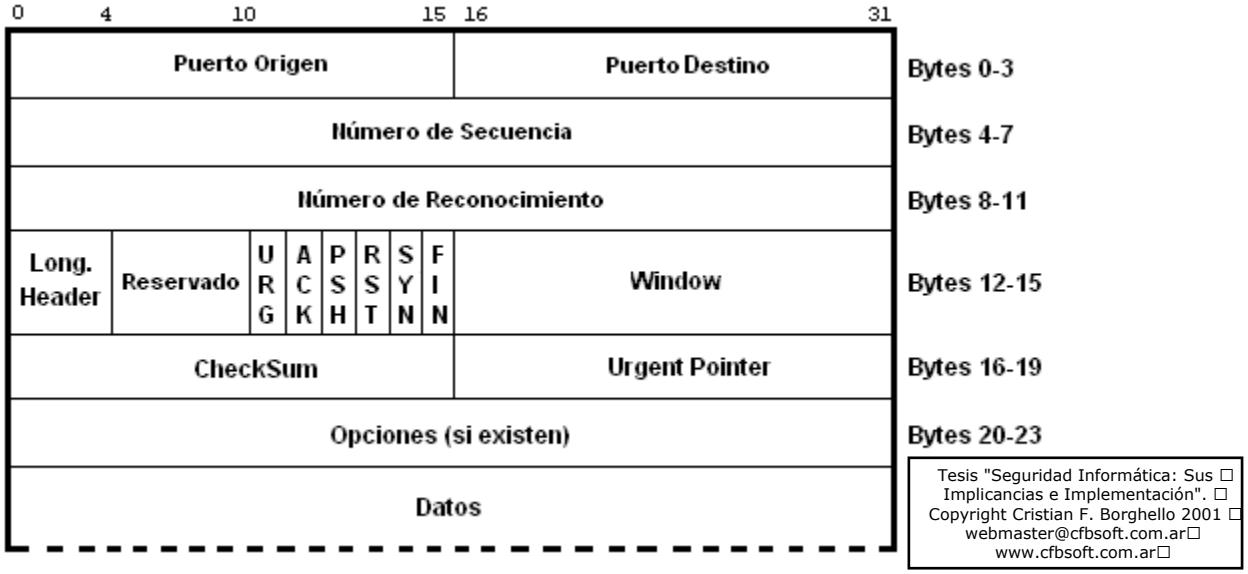

Grafico 6.3 – Constitución de un datagrama TCP

Los puertos proporcionan una manera de distinguir entre las distintas transferencias, ya que un mismo sistema puede estar utilizando varios servicios o transferencias simultáneamente, e incluso puede que por medio de usuarios distintos.

El **Puerto de Origen** (16 bits) contendrá un número cualquiera que sirva para realizar esta distinción. Además, el programa cliente que realiza la petición también debe conocer el número de puerto en el que se encuentra el servidor adecuado. Mientras que el programa del usuario utiliza números prácticamente aleatorios, el servidor deber tener asignado un número estándar para que pueda ser utilizado por el cliente (ver Tabla 6.1). Cuando es el servidor el que envía los datos, los números de puertos de origen y destino se intercambian.

El campo **Tamaño de la Cabecera** (4 bits) contiene la longitud total de la cabecera TCP expresada en el número de palabras de 32 bits que ocupa (el tamaño real divido 4). Esto permite determinar el lugar donde comienzan los **Datos**.

En la transmisión de datos a través del protocolo TCP la fiabilidad es un factor muy importante. Para poder detectar los errores y pérdida de información en los datagramas, es necesario que el cliente envíe de nuevo al servidor unas **Señales de Confirmación** (32 bits) una vez que se ha recibido y comprobado la información satisfactoriamente. Si el servidor no obtiene la señal de confirmación adecuada transcurrido un período de tiempo razonable, el datagrama completo se volverá a enviar.

Por razones de eficiencia los datagramas se envían continuamente sin esperar la confirmación, haciéndose necesaria la numeración, mediante los **Números de Secuencia** (32 bits), de los mismos para que puedan ser ensamblados en el orden correcto.

También puede ocurrir que la información del datagrama llegue con errores a su destino. Para poder detectarlos, cuando sucede esto, se incluye un **CheckSum** (16 bits), el cual contiene un valor calculado a partir de la información del datagrama completo. En el otro

extremo el receptor vuelve a calcular este valor, comprobando que es el mismo que el suministrado en la cabecera. Si el valor es distinto significará que el datagrama es incorrecto.

La forma en que TCP numera los datagramas es contando los bytes de datos que contiene cada uno de ellos y añadiendo esta información al campo correspondiente de la cabecera del datagrama siguiente.

De esta manera el primero empezará en cero; el segundo contendrá el tamaño de la parte de datos; el tercero contendrá la suma de ese número más el tamaño de los datos del segundo datagrama; y así sucesivamente. Por ejemplo, para un tamaño fijo de 500 bytes de datos en cada datagrama, la numeración sería la siguiente: 0 para el primero, 500 para el segundo, 1000 para el tercero, etc.

Existe otro factor más a tener en cuenta durante la transmisión de información, y es la potencia y velocidad con que cada computadora puede procesar los datos que le son enviados. Si esto no se tuviera en cuenta, el equipo de mayor potencia podría enviar la información demasiado rápido al receptor, de manera que éste no pueda procesarla. Este inconveniente se soluciona mediante un campo **Window** (16 bits), en el cual se introduce un valor indicando la cantidad de información que el receptor está preparado para procesar en un momento dado.

El campo **Opciones** (32 bits) permite que una aplicación negocie durante la configuración de la conexión, características como el tamaño máximo del segmento TCP. Si este campo tiene el primer octeto a cero, indica que no hay opciones, quedando un datagrama de 160 bits.

Por último cada datagrama tendrá un **Estado** que le indicará al servidor el contenido, motivo y la forma en que deberá ser atendido ese paquete. Este campo puede contiene los siguientes estados (Estado = 1 → Verdadero):

- Bit 5 (URGent): Identifica datos urgentes.
- Bit 4 (ACKnowledge): Indica que el campo de confirmación es válido.
- Bit 3 (PuSH): Aunque el buffer de datos no este lleno, se fuerza el envío.
- Bit 2 (ReSeT): Abortar la conexión. Todos los buffers asociados se vacían.
- Bit 1 (SYnchronize sequence Number): Sincronizar los números de secuencia.
- Bit 0 (FINish): Se solicita el cerrado de la conexión.

Todas estas características se traducen en un "protocolo pesado" por el envío de señales de confirmación y la velocidad se ve sacrificada en pos de la fiabilidad de los datos.

6.2.6.2 UDP

El User Datagram Protocol puede ser la alternativa al TCP en algunos casos en los que no sea necesario el gran nivel de complejidad proporcionado por el TCP. Puesto que UDP no admite numeración de los datagramas, este protocolo se utiliza principalmente cuando el orden en que se reciben los mismos no es un factor fundamental, cuando se quiere enviar información de poco tamaño que cabe en un único datagrama o si la fiabilidad de los datos no es un factor de relieve.

Cuando se utiliza UDP la garantía de que un paquete llegue a su destino es mucho menor que con TCP debido a que no se utilizan las señales de confirmación. Por todas estas

características la cabecera del UDP es bastante menor en tamaño que la de TCP. Esta simplificación resulta en una mayor eficiencia en determinadas ocasiones. Es utilizado en redes con muy buen cableado.

6.2.7 NIVEL DE APLICACIÓN DEL MODELO TCP/IP

6.2.7.1 ICMP

El Internet Control Message Protocol es de características similares al UDP, pero con un formato aún más simple. Su utilidad no está en el transporte de datos "de usuario", sino en los mensajes de error y de control necesarios para los sistemas de la red.

6.2.7.2 FTP

El File Transfer Protocol se incluye como parte del TCP/IP, estando destinado proporcionar el servicio de transferencia de archivos. El FTP depende del protocolo TCP para las funciones de transporte, y guarda alguna relación con Telnet (protocolo para la conexión remota).

FTP utiliza dos canales de conexión separados: uno es el canal de comandos que permanece abierto durante toda la sesión y el otro es el canal de transmisión de archivos.

Gráficamente:

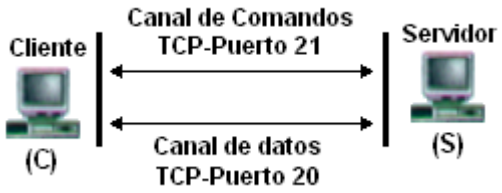

Gráfico 6.4 – Conexión FTP

El FTP permite acceder a algún servidor que disponga de este servicio y realizar tareas tales como moverse a través de su estructura de directorios, ver y descargar archivos al ordenador local, enviar o copiar archivos directamente de un servidor a otro de la red. Lógicamente y por motivos de seguridad se hace necesario contar con el permiso previo para poder realizar todas estas operaciones. El servidor FTP pedirá el nombre de usuario y clave de acceso al iniciar la sesión (login). Este debe ser suministrado correctamente para poder utilizar el servicio.

La manera de utilizar FTP es por medio de una serie de comandos, los cuales suelen variar dependiendo del sistema en que se esté ejecutando el programa, pero básicamente con la misma funcionalidad. Existen aplicaciones de FTP para prácticamente todos los sistemas operativos.

Existe una forma muy utilizada para acceder a fuentes de archivos de carácter público por medio de FTP y es el acceso FTP Anónimo, mediante el cual se pueden copiar archivos de uso público. Generalmente el acceso anónimo tendrá algunas limitaciones en los permisos,

siendo normal en estos casos que no se permita realizar acciones tales como añadir archivos o modificar los existentes.

El FTP proporciona dos modos de transferencia de archivos: ASCII y binario. El modo de transferencia ASCII se utiliza cuando se quiere transmitir archivos de texto puro. El binario se debe utilizar en cualquier otro caso (datos que no son texto plano).

6.2.7.3 HTTP

Este HiperText Tranfer Protocol es la base de toda comunicación desarrollada en la Web. Utilizado desde principios de lo 90 es un protocolo ASCII que se ocupa de establecer una comunicación TCP segura entre el cliente y el servidor a través del puerto 80.

HTTP es un protocolo de aplicación para sistemas de información distribuidos, e hipermediático. Es un protocolo genérico, sin estado, orientado a objetos, que se puede utilizar para muchas tareas, como servidores de nombres y sistemas de gestión de objetos distribuidos, por medio de la ampliación de sus métodos de petición o comandos.

Sus principales características son:

1. Protocolo de Aplicación: aunque generalmente se implementa sobre el TCP/IP, también es capaz de hacerlo sobre otros protocolos de capas más bajas. HTTP presupone únicamente un transporte fiable, así que puede utilizar cualquier protocolo que garantice este requisito mínimo.
2. Sistemas de información distribuidos, colaboradores, de hipermedios: HTTP soporta sistemas de información distribuidos es decir, sistemas esparcidos por múltiples servidores.
3. Genérico: HTTP no dicta el contenido de los datos que transfiere; simplemente actúa como un conducto para mover datos de aplicación, por lo que se puede transferir cualquier tipo de información por medio de HTTP.
4. Sin estado: HTTP no mantiene un estado. Cuando se solicita una transferencia a través de HTTP, se crea la conexión, se produce la transferencia y se termina la conexión. Esta es una de las debilidades de HTTP; sin información de estado, cada página Web está sola. Por ejemplo, es difícil desarrollar una aplicación basada en la Web que permita que un usuario se conecte en una página y que mantenga esta información de conexión durante todo el tiempo que el usuario esté accediendo activamente al destino. Cualquier documento transferido a través de HTTP no tiene ningún contexto y es completamente independiente de todos los documentos transferidos antes de él.
5. Orientado a objetos, escritura y negociación de la representación de los datos: HTTP no está orientado a objetos en el mismo sentido que un lenguaje de programación. Esta descripción significa simplemente que HTTP tiene etiquetas que indican el tipo de datos que se van a transferir por medio de la red, así como métodos, que son comandos que indican qué debe transferirse.
6. Sistema creado independientemente de los datos que se transfieren: Debido a que HTTP sólo mueve datos, no necesita tener información sobre cada uno de los tipos a transferir. Por ejemplo, un servidor Web no necesita un conocimiento específico

sobre el funcionamiento interno del formato de un archivo de vídeo para hacer el envío.

La comunicación que se establece en una conexión HTTP es de muy corta duración. El cliente establece la conexión con el servidor HTTP y le solicita un documento determinado. El servidor recibe la consulta, la evalúa y envía el documento solicitado (si existe) o un mensaje de error en caso contrario. Luego el servidor finaliza la conexión sin que existan otros estados intermedios.

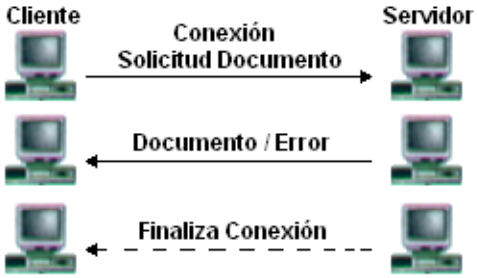

Gráfico 6.5 – Conexión HTTP

El protocolo HTTP en su estructura, divide el mensaje en encabezado (Header) y en cuerpo (Entity), separados entre sí por una línea en blanco.

6.2.7.4 SMTP

El servicio de correo electrónico se proporciona a través del protocolo Simple Mail Transfer Protocol, (empleando redes TCP/IP) y permite enviar mensajes a otros usuarios de la red. A través de estos mensajes no sólo se puede intercambiar texto, sino también archivos binarios de cualquier tipo.

Generalmente los mensajes de correo electrónico no se envían directamente a las computadoras personales de cada usuario, sino a un servidor de correo que actúa como almacén de los mensajes recibidos. Los mensajes permanecerán en este sistema hasta que el usuario los transfiera a su propio equipo para leerlos de forma local (vía POP).

El cliente de correo envía una solicitud a su e–mail Server (al puerto 25) para enviar un mensaje (y almacenarlo en dicho servidor). El Server establece una conexión SMTP donde emisor y receptor intercambian mensajes de identificación, errores y el cuerpo del mail. Luego de esto el emisor envía los comandos necesarios para la liberación de la conexión.

Gráfico 6.6 – Conexión SMTP

6.2.7.5 POP

El servidor POP (Post Office Protocol) fue diseñado para la recuperación de correo desde el e–mail Server hacia la computadora destinataria del mensaje.

Al igual que sucede con SMTP, inicialmente el proceso escucha el puerto del protocolo POP (el 110) y cuando el emisor solicita el mensaje se establece una conexión full duplex donde se intercambian los mensajes Emisor–Server para luego finalizar la conexión cuando se hallan enviado cada uno de los mails que se almacenaban en el servidor.

Actualmente el protocolo POP se encuentran en su tercera implementación por lo que generalmente se escuchará sobre POP3.

Gráficamente la relación entre el protocolo SMTP y el POP3 es la siguiente:

Gráfico 6.7 – Relación SMTP–POP

6.2.7.6 MIME

Multipurpose Internet Mail Extensions es una extensión del protocolo SMTP y se creo con el fin de soportar algunos juegos de caracteres extendidos (no US–ASCII) no soportados por este último (principalmente el francés y el alemán).

MIME especifica tres campos que se incluyen en la cabecera del mensaje, para hacer la conversión adecuada al sistema no US–ASCII utilizado:

1. MIME–Versión: especifica la versión de MIME utilizado para codificar el mensaje.
2. Content–Type: especifica el tipo y subtipo de los datos no ASCII.
3. Contenet–Tranfer–Encoding: especifica el tipo de codificación usado para traducir los datos en ASCII.

6.2.7.7 NNTP

El Network News Transfer Protocol fue diseñado para permitir la distribución, solicitud, recuperación y envío de noticias (News). NNTP está basado en las especificaciones de UseNet (tratado también en este capítulo) pero con algunas modificaciones en su estructura que le permiten ser adaptables a otros grupos de noticias no UseNet.

6.2.7.8 SNMP

El Simple Network Management Protocol se utiliza para monitorizar, controlar y administrar múltiples redes físicas de diferentes fabricantes, donde no existe un protocolo

común en la capa de Enlace. La estructura de este protocolo se basa en utilizar la capa de aplicación para evitar el contacto con la capa de enlace y, aunque es parte de la familia TCP/IP no depende del IP ya que fue diseñado para ser independiente y puede correr igual de fácil sobre, por ejemplo, IPX de Novell.

6.3 ESTRUCTURA BÁSICA DE LA WEB

La estructura básica de la World Wide Web (WWW) consiste en que el protocolo HTTP actúa como un transporte genérico que lleva varios tipos de información del servidor al cliente. Hoy en día las conexiones a servidores web son las más extendidas entre usuarios de Internet, hasta el punto tal de que muchas personas piensan que este servicio es el único existente, junto al IRC. Inicialmente se ideo para que unos cuantos físicos intercambiaran información entre universidades, hoy es uno de los pilares fundamentales de muchas empresas.

Cada entidad servidor se identifica de forma única con un Localizador de Recursos Universal (URL) que a su vez está relacionado unívocamente con una dirección IP.

El tipo más común de datos transportado a través de HTTP es HTML (HiperText Markup Language). Además de incluir directrices para la "compresión" de textos, también tiene directrices que proporcionan capacidades como las de enlaces de hipertexto y la carga de imágenes en línea. Los recursos hiperenlazados y los archivos de imágenes en línea están identificados con los URL intercalados dentro del documento HTML.

A pesar de que algunos servidores Web personales de gama baja sólo pueden enviar páginas estáticas, la mayoría de los servidores HTTP admiten la CGI (Common Gateway Interface). Con CGI se pueden escribir programas que se integran en la Web y que realizan tareas tales como el proceso de formularios y las búsquedas en bases de datos, las cuales HTML no puede ejecutar.

La principal limitación de CGI es que está restringida a programas en el lado del servidor. Por ejemplo, utilizando CGI, la única forma en la que se puede interactuar con los usuarios es suministrándoles formularios ha completar. Las tecnologías orientadas a objetos como Java afrontan esta limitación, permitiendo al servidor que envíe al cliente pequeños programas para ejecutarlos localmente.

6.3.1 SERVICIOS DE INTERNET

Como sabemos, Internet es en la actualidad, la red de computadoras más grande del mundo. Sin embargo la importancia de Internet no reside solamente en el número de máquinas interconectadas sino en los servicios que brinda.

Los servicios y recursos de Internet (Gopher, News, Archie, WWW, etc.) son accesibles de diversas formas, principalmente tres: por Telnet, por e–mail, y por un programa cliente.

A través de Telnet o e–mail, el servicio presenta una interface ANSI (sin gráficos), sólo con caracteres alfanuméricos. Con un programa cliente, la gestión es más sencilla, visual y

agradable, como sucede en la WWW donde se presentan cada una de las páginas en formato gráfico.

6.3.1.1 TELNET

Este protocolo fue diseñado para proporcionar el servicio de conexión remota (remote login). Forma parte del conjunto de protocolos TCP/IP y depende del protocolo TCP para el nivel de transporte.

El protocolo Telnet es un emulador de terminal que permite acceder a los recursos y ejecutar los programas de un equipo remoto en la red, de la misma forma que si se tratara de una terminal real directamente conectado al sistema remoto. Una vez establecida la conexión el usuario podrá iniciar la sesión con su clave de acceso. De la misma manera que ocurre con el protocolo FTP, existen servidores que permiten un acceso libre cuando se especifica "anonymous" como nombre de usuario.

El sistema local que utiliza el usuario se convierte en una terminal "no inteligente" donde todos los caracteres pulsados y las acciones que se realicen se envían al Host remoto, el cual devuelve el resultado de su trabajo. Para facilitar un poco la tarea a los usuarios, en algunos casos se encuentran desarrollados menús con las distintas opciones que se ofrecen.

Para utilizar Telnet se ejecuta un programa especial, llamado Telnet en el cliente. Este programa utiliza TCP para conectarse a un sistema específico (su servidor) en el puerto 23 (por defecto) . Una vez que se establece la conexión, Telnet actúa como un intermediario entre el cliente y el servidor.

La mayoría de las computadoras que permiten este tipo de acceso cuentan con los programas necesarios para diversos servicios de Internet, como Gopher, Wais, FTP o cualquier otro programa–cliente disponible en el Server. Estas conexiones suelen ser más económicas (pero más lentas) y están restringidas a los servicios que brinda el servidor.

6.3.1.2 IRC

El Internet Relay Chat es un sistema de coloquio en tiempo real entre personas localizadas en distintos puntos de la red. Es un servicio basado exclusivamente en texto por teclado. Fue desarrollado en 1988 en Finlandia y es sin duda, hoy, uno de los servicios más populares de Internet.

Su gran atractivo es que permite las conversaciones en vivo de múltiples usuarios la mayor parte desconocidos entre sí. El manejo del sistema es muy simple. El IRC está organizado por redes, cada una de las cuales está formada por servidores que se encargan, entre otras cosas, de ofrecer canales de conversación (existiendo miles de ellos) y transmitir los mensajes entre usuarios.

Para acceder a un servidor de este tipo es necesario disponer de un programa cliente, siendo los cuatro más populares el mIRC, Pirch, Ichat y Microsoft Chat.

Cada servidor IRC está conectado a los servidores más cercanos. De esta manera, todos los servidores IRC están conectados (al menos indirectamente) unos con otros.

IRC mantiene un número de diferentes “Canales”, teniendo que elegir al ingresar el canal de interés y pudiendo entrar y salir de los canales cuantas veces se desee y en cuantos canales se desee.

La mayoría de los nombres de canales empiezan con **“#”**. Algunos canales son para discutir de temas específicos y otros surgen en el momento. Hay canales públicos, privados, secretos e individuales.

Se pueden crear canales (obteniendo la condición de Operador) si el que se desea no existe y esperar que otras personas ingresen en él. Se suele entrar en las charlas con apodos (nick), sin dar el nombre real, de manera que los usuarios conserven el anonimato. Cuando la última persona abandona un canal, IRC lo elimina.

6.3.1.3 USENET

Una de las áreas más populares de Internet son los grupos de discusión o NewGroups**.** El término UseNet surge de USEr NETwork (red de usuarios) y se refiere al mecanismo que soportan los grupos de discusión.

Los grupos se forman mediante la publicación de mensajes enviados (“posteados”) a un grupo en particular (generalmente de un tema específico). El software original de News fue desarrollado para los sistemas Unix en 1979 por dos estudiantes graduados en la Universidad de Duke, como un mecanismo para la discusión técnica y conferencias.

Es una red que no se centra en un único servidor que distribuye los mensajes, sino en una cadena de servidores que se “pasan” los mensajes de los grupos que soporta ya que, normalmente, los servidores mantienen un grupo limitado de News.

Una vez creado un grupo*,* se puede enviar cualquier mensaje al mismo, y cualquiera dentro de Internet podrá leerlo, a menos que sea un grupo “moderado”, con lo cual nuestros mensajes pasan por la “censura” de un moderador.

Para hacer manejable toda la información que circula, se utiliza un sistema en el que los grupos de discusión se agrupan en categorías denominadas Jerarquías*.* Cada jerarquía tiene un nombre propio y se dedica a un área de interés particular.

Algunas de las jerarquías más relevantes son:

Tema	Descripción
alt (alternative)	Diferentes temas
Bionet	Biología
biz (business)	Negocios
comp (computer)	Computadoras e Informática
Ddn	Red de datos del departamento de defensa
News	Grupos sobre UseNet
rec (recreative)	Ocio
sci (science)	Ciencias
soc	Ciencias sociales
talk	Debates

Tabla 6.2 – Jerarquías más comunes en UseNet

La filosofía de las UseNet es la siguiente: Al dejar un mensaje, no sólo se queda en el grupo en cuestión, sino que también les llega a todos los usuarios suscritos al mismo, vía e–mail.

Tiene una gran utilidad práctica, ya que si un usuario determinado tiene algún comentario o duda acerca de un tema, puede acudir al grupo temático indicado, dejar un mensaje para pedir ayuda, y con seguridad recibirá la opinión de numerosas personas.

6.3.1.4 FINGER

La mayoría de las computadoras de Internet tienen una utilidad que permite buscar información sobre un usuario particular. Este servicio es conocido como Finger (dedo).

En Internet los usuarios se conocen por su identificador. Finger se puede utilizar para encontrar el nombre de un usuario si se conoce su identificador, ya que el objetivo de este servicio es obtener información sobre una persona en particular.

El servicio Finger es un sistema Cliente/Servidor que proporciona tres tipos principales de información:

1. Información pública sobre cualquier usuario.
2. Comprobación de si un usuario está utilizando actualmente un Host determinado en Internet, pudiendo ver un resumen de información para cada usuario que está conectado.
3. Conectar con determinado Host, que se han configurado para ofrecer otros tipos de información.

6.3.1.5 WHOIS

¿Quién es? es un servidor de directorio de páginas blancas que permite consultar una base de datos de nombres y direcciones de correo electrónico de un usuario (normalmente una empresa).

El servicio WhoIs contacta a un servidor que tiene información básica sobre las redes que comprenden Internet, y los nombres de las personas que dan mantenimiento. Por ejemplo la solicitud *whois Harvard* produce una lista de todas las redes de la Universidad de Harvard y las companías que tienen Harvard como parte de su nombre.

Originalmente, la información sobre los usuarios de Internet se almacenaba en una base de datos central. Hoy en día muchas organizaciones corren este servicio que proporciona información sobre los usuarios de una organización. Uno de los servidores WhoIs más conocido es *whois.internic.net* que contiene nombres y direcciones de Internet.

Es de remarcar que servicios como Telnet, Finger, Archie, Gopher, WhoIs y Ping están cayendo en desuso a favor de las cada vez más perfeccionadas herramientas basadas en la World Wide Web. De todas maneras (como se verá en capítulos posteriores) siguen brindando gran utilidad a administradores de sistemas y usuarios avanzados, sobre todo en temas de seguridad.