ETHERNET Y PROTOCOLOS TCP/IPv4

Las redes están integradas por diversos componentes que trabajan juntos para crear un sistema funcional. Los componentes de red son fabricados por lo general por varias compañías, por lo que es necesario que exista entendimiento y comunicación entre los fabricantes en relación con la manera en que cada componente trabaja e interactúa con los demás componentes de la red. Por esta razón se han creado estándares que definen la forma de conectar componentes de hardware y los protocolos de uso cuando se establecen comunicaciones.

Los tres estándares más populares que se utilizan son: Ethernet, ARCnet y *Token* Ring. Ethernet y *Token* Ring son estándares respaldados por IEEE (*Institute of Electrical and Electronic Engineers*); ARCnet es un estándar de ANSI (*American National Standards Institute*).

En términos de software, para la comunicación de computadoras también existen estándares; la tecnología ARPA (*Agencia de Proyectos de Investigación Avanzada*) incluye un grupo de estándares de red que especifican los detalles de cómo se comunican las computadoras, así como un grupo de reglas para interconectar redes y para rutear el tráfico de información, conocido de manera oficial como el grupo de protocolos Internet TCP/IP, pero llamado comúnmente TCP/IP.

Los protocolos TCP/IP se utilizan para establecer comunicación entre diferentes nodos en un entorno heterogéneo y definen los formatos y normas utilizados en la transmisión y recepción de información. En este capítulo hablaremos del estándar Ethernet y del conjunto de protocolos TCP/IP.

1.1 Ethernet

Ethernet, al que también se conoce como IEEE 802.3, es el estándar más popular para las LAN, usa el método de transmisión de datos llamado *Acceso múltiple con detección de portadora y detección de colisiones* (CSMA/CD) [4]. Antes de que un nodo envíe algún dato a través de una red Ethernet, primero escucha y se da cuenta si algún otro nodo está transfiriendo información; de no ser así, el nodo transferirá la información a través de la red. Todos los otros nodos escucharán y el nodo seleccionado recibirá la información. En caso de que dos nodos traten de enviar datos por la red al mismo tiempo, cada nodo se dará cuenta de la colisión y esperará una cantidad de tiempo aleatoria antes de volver a hacer el envío. Cada paquete enviado contiene la dirección de la estación destino, la dirección de la estación de envío y una secuencia variable de bits que representa el mensaje transmitido. El dato transmitido procede a 10 millones de bits por segundo y el paquete varia en una longitud de 64 a 1518 bytes, así el tiempo de transmisión de un paquete en la Ethernet esta en un rango de 50 a 1200 microsegundos dependiendo de su longitud. La dirección de la estación de destino normalmente es referida por una única interfaz

de red. Cada estación recibe una copia de cada paquete, pero ignora los paquetes que son dirigidos a otras computadoras y procesa solamente los que son dirigidos a ella.

Las velocidades de envío de paquetes utilizando la tecnología Ethernet son de 10 Mbps (Ethernet estándar), 100 Mbps (Fast Ethernet – 100BASEX) y de 1000 Mbps utilizando el Gigabit Ethernet cuya especificación se encuentra respaldada por la IEEE con número 802.3z, el cual cumple los siguientes objetivos [38]:

• Permite realizar operaciones de envío y recepción de datos a una velocidad de 1000 Mbps.
• Usa el formato de frame Ethernet 802.3.
• Usa el método de acceso CSMA/CD con soporte para un repetidor por dominio de colisión.
• Las direcciones de retorno son compatibles con las tecnologías 10BASE-T y 100Base-T.

Las redes Ethernet tienen un esquema de direccionamiento de 48 bits. A cada computadora conectada a una red Ethernet se le asigna un número único de 48 bits conocido como *dirección Ethernet*. Para asignar una dirección, los fabricantes de hardware de Ethernet adquieren bloques de direcciones Ethernet y las asignan en secuencia conforme fabrican el hardware de interfaz Ethernet, de esta manera no existen dos unidades de hardware de interfaz que tengan la misma dirección Ethernet. Por lo general, las direcciones Ethernet se colocan en el hardware de interfaz anfitrión de las máquinas de tal forma que se puedan leer. Debido a que el direccionamiento Ethernet se da entre dispositivos de hardware, a estos se les llama *direccionamientos* o *direcciones fisicas*.

La trama de Ethernet es de una longitud variable pero no es menor a 64 bytes ni rebasa los 1518 bytes (encabezado, datos y CRC), cada trama contiene un campo con la información de la dirección de destino. En la figura 1.1 se muestra una trama Ethernet. Además de la información que identifica la fuente y el destino, cada trama transmitida contiene un *preámbulo, un campo tipo, un campo de datos y un campo para verificación por redundancia cíclica* (CRC- C*yclic* R*edundancy* C*heck*). El preámbulo consiste en 64 bits que alternan ceros y unos para ayudar a la sincronización de los nodos de recepción. El CRC de 32 bits ayuda a la interfaz a detectar los errores de transmisión: el emisor calcula el CRC como una función de los datos en la trama y el receptor calcula de nuevo el CRC para verificar que el paquete se reciba intacto [2].

Preámbulo	Dirección destino	Dirección fuente	Tipo	Datos	CRC
8 bytes	6 bytes	6 bytes	2 bytes	46-1500 bytes	4 bytes

Figura 1.1 Formato de una trama (paquete) que viaja a través de Ethernet.

El campo de *tipo de trama* contiene un entero de 16 bits que identifica el tipo de dato que se está transfiriendo en la trama. Desde el punto de vista de Internet, este campo es esencial porque significa que las tramas se *autoidentifican*. Cuando una trama llega a una máquina dada, el sistema operativo utiliza el tipo de trama para determinar qué modulo de software de protocolos se utilizará para procesar la trama. La mayor ventaja de que las tramas se autoidentifiquen es que éstas permiten que múltiples protocolos se utilicen juntos en una sola máquina y sea posible entremezclar diferentes protocolos en una sola red física sin interferencia. Los protocolos TCP/IP utilizan tramas Ethernet autoidentificables para hacer una selección entre varios protocolos. Cuando se transmite un datagrama IP versión 4 el campo *tipo* de trama contiene el valor hexadecimal 0800 [77] y al transmitir un datagrama IP versión 6 el campo tiene el valor hexadecimal 86DD [80].

1.2 Protocolos TCP/IP

En forma general, el conjunto de protocolos TCP/IP tiene correspondencia con el modelo de comunicaciones de red definido por ISO (*International Organization for Standardization*), este modelo se denomina modelo de referencia de interconexión de sistemas abiertos (OSI). El modelo OSI describe un sistema ideal de redes que permite establecer una comunicación entre procesos de capas distintas y fáciles de identificar. En el *host*, las capas prestan servicios a capas superiores y reciben servicios de capas inferiores. La figura 1.3 muestra las siete capas del modelo de referencia OSI y su correspondencia general con las capas del conjunto de protocolos TCP/IP y en la tabla 1.1 se enumeran los protocolos más comunes del conjunto de protocolos TCP/IP y los servicios que proporcionan.

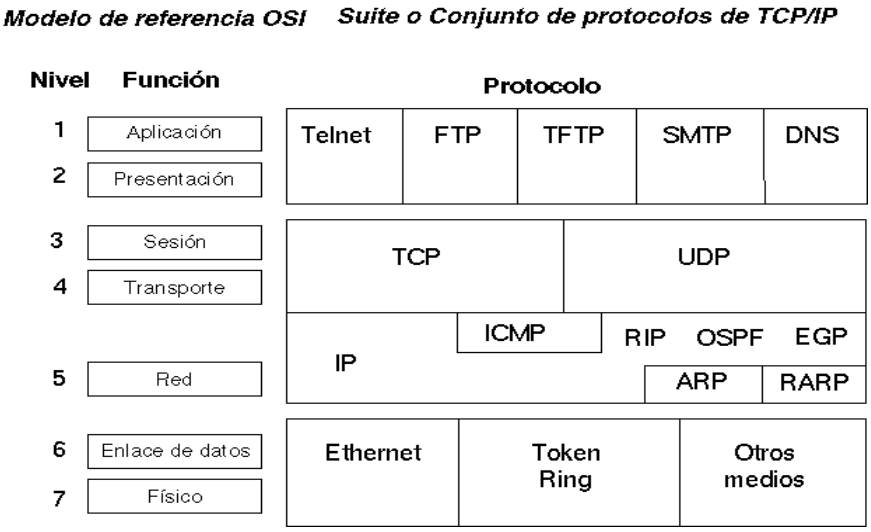

Figura 1.3. Modelo de referencia OSI y las capas de TCP/IP correspondientes.

Tabla 1.1 Protocolos más comunes de TCP/IP.

Protocolo	Servicio
Protocolo Internet (IP)	Proporciona servicios para la entrega de paquetes entre nodos.
Protocolo de control de mensajes de Internet (ICMP).	Regula la transmisión de mensajes de error y control entre los *hosts* y los *routers*.
Protocolo de resolución de direcciones (ARP).	Asigna direcciones Internet a direcciones físicas.
Protocolo de resolución de direcciones por réplica (RARP).	Asigna direcciones físicas a direcciones Internet.
Protocolo de control de transmisión (TCP).	Proporciona servicios de envío de flujos fiables entre los clientes.
Protocolo de *datagrama* de usuario (UDP).	Proporciona servicio de entrega de *datagramas* no fiable entre clientes.
Protocolo de transferencia de archivos (FTP).	Proporciona servicios de nivel de aplicación para la transferencia de archivos.
TELNET	Proporciona un método de emulación de terminal.
Protocolo de información de encaminamiento (RIP)	Permite el intercambio de información de rutas de vectores de distancia entre *routers*.
Protocolo abrir la vía más corta primero (OPSF)	Permite el intercambio de información de rutas de estado del enlace entre *routers*.
Protocolo Gateway Externo (EGP)	Permite el intercambio de información de rutas entre *routers* externos.

1.2.1 Encapsulado

Las aplicaciones que se desarrollan con TCP/IP, normalmente utilizan un conjunto de protocolos para llevar a cabo la comunicación. La suma de las capas de este conjunto de protocolos se conoce como *stack de protocolo*. De esta forma, cuando una aplicación envía datos usando el protocolo TCP, el dato es enviado hacia abajo del protocolo *stack*, a través de cada capa, hasta que este se envíe como un flujo de bits a través de la red. Cada capa coloca información adicional al dato en su encabezado (y algunos añaden información para rastreo) para que el dato sea recibido. En la figura 1.4 se muestra este proceso. Los números abajo de los encabezados y del CRC en la trama Ethernet representan los tamaños típicos en bytes. Una propiedad física de una trama Ethernet es que la MTU (*Maximum Transmisión Unit)* por default es del tamaño de 1500 bytes [77] [80], por lo cual los paquetes IPv4 e IPv6 no exceden este tamaño.

En la figura 1.4 no se colocó el encabezado UDP, dado que se supone que se está transportando un segmento TCP, pero si fuese el caso que se transportara un *datagrama* UDP, habrá que cambiar en la figura el encabezado TCP por el encabezado UDP, cuyo tamaño es de 8 bytes.

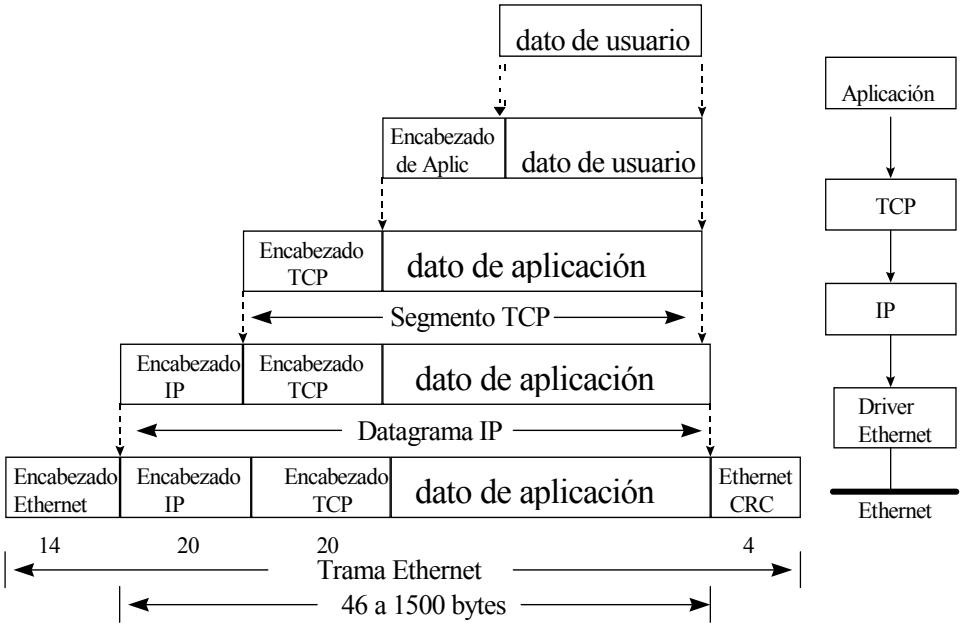

Figura 1.4 Encapsulado de un dato.

Como se puede notar en los párrafos anteriores cada capa del protocolo TCP/IP se refiere a los datos que transmite con términos diferentes, en la figura 1.5 se muestran estos términos. Las aplicaciones que usan TCP se refieren a los datos como *stream*, mientras que las aplicaciones que usan el protocolo de *datagrama* de usuario (UDP) se refieren a los datos como *mensajes.* TCP llama a estos datos *segmentos,* y UDP llama a estos datos *datagramas.* La capa de Internet ve a todos estos datos como bloques y les llama *datagramas* [6]. TCP/IP usa diferentes tipos de redes para mandar sus datos, cada una de las cuales tienen diferentes tipos de términos para los datos que transmiten, en nuestro caso ocuparemos el término que utiliza Ethernet, la cual llama a los datos *frame, trama o paquete.*

El proceso que utiliza una aplicación para transferir el contenido de un archivo es el siguiente:

1. La capa de la aplicación envía un flujo de bytes a la capa de transporte de la computadora de origen.
2. La capa de transporte divide el flujo en segmentos TCP, asigna un encabezado con un número de secuencia al segmento y transmite este segmento a la capa de Internet (IP), y se calcula la suma de comprobación.

3. La capa de IP crea un paquete con parte de los datos que contiene el segmento TCP. La capa de IP añade al paquete un encabezado que indica las direcciones IP de origen y de destino. Esta capa también determina la dirección física de la computadora destino o las computadoras que actúan como intermediarios hasta el *host* destino. Entonces, envía el paquete y la dirección física a la capa de enlace de datos y se vuelve a calcular la suma de comprobación.
4. La capa de enlace de datos transmite el paquete IP en la sección de datos de una trama de enlace de datos a la computadora destino. Si la computadora destino actúa como intermediario, el paso 3 volverá a repetirse hasta que se alcance el destino final.
5. Cuando se alcanza la computadora destino, la capa de enlace de datos descarta el encabezado del enlace y envía el paquete IP a la capa de IP.
6. La capa de IP verifica el encabezado del paquete. Si la suma de comprobación del encabezado no coincide con la calculada por dicha capa, el paquete se ignora.
7. Si las sumas coinciden, la capa IP descarta el encabezado y envía el segmento TCP a la capa TCP correspondiente. Esta capa comprueba el número de secuencia para determinar si el segmento, es el segmento correcto de la secuencia.
8. La capa TCP calcula una suma de comprobación para los datos y el encabezado TCP. Si la suma no coincide con la suma transmitida con el encabezado, la capa TCP descarta el segmento. Si la suma coincide y el segmento está en la secuencia correcta, la capa TCP envía un reconocimiento a la computadora destino.
9. La capa TCP descarta el encabezado TCP y transfiere los bytes del segmento que acaba de recibir a la aplicación.
10. La aplicación que se encuentra en la computadora destino recibe un flujo de bytes como si estuviera conectado directamente a la aplicación de la computadora origen.

En las secciones siguientes se describe con mayor detalle el funcionamiento y el formato de cada uno de los protocolos que realizan estas operaciones interactivas.

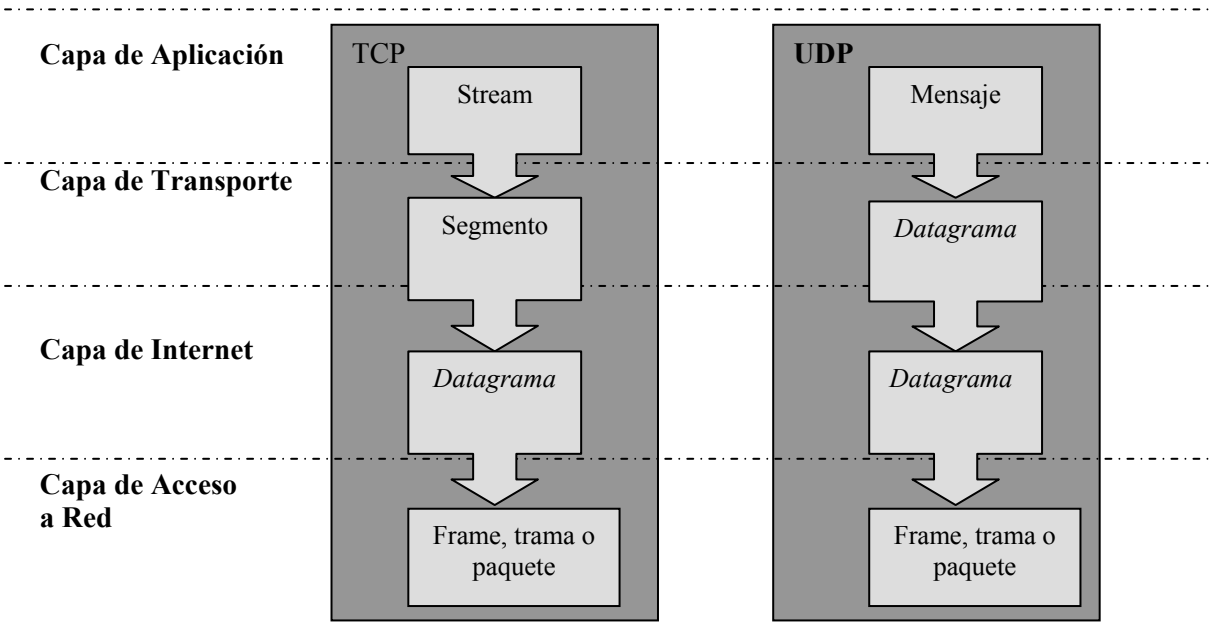

Figura 1.5 Estructura de datos.

1.2.2 Protocolo Internet (IP versión 4)

En el conjunto de protocolos TCP/IP, todos los paquetes se entregan mediante el servicio de entrega de *datagramas* IP, aunque este servicio no garantiza la entrega, ya que carece de

conexión por lo cual los paquetes se transmiten independientemente unos de otros y pueden dirigirse a lugares a los que no corresponden, duplicarse o perderse antes de llegar a su destino.

Las aplicaciones TCP/IP que utilizan este servicio de entrega de *datagramas* hacen un seguimiento del estado de la entrega esperando las respuestas desde el nodo destino o utilizando uno de los protocolos de capa de transporte del conjunto TCP/IP.

IP define el formato que los paquetes deben tener y el modo de utilizarlos durante el envío y la recepción. El formato que toma el paquete se denomina *datagrama* IP. Los *datagramas* IP son análogos a las tramas físicas que se transmiten en una red. Los *datagramas* tienen una sección de encabezado que incluye, entre otra información, las direcciones IP del receptor y emisor, y una sección de datos. El formato de un *datagrama* IPv4 [2] se muestra en la figura 1.6. El tamaño normal de un encabezado IP es de 20 bytes, a menos que presente el campo de opciones.

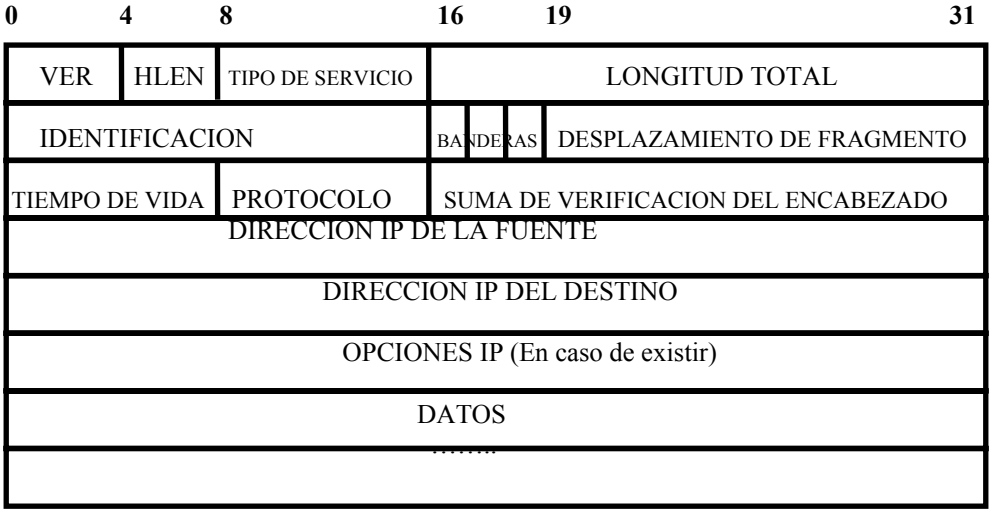

Figura 1.6 Formato de un *datagrama* IPv4.

Los campos del *datagrama* IP se describen a continuación: el campo VER es de 4 bits y contiene la versión del protocolo IP que se utilizó para crear el *datagrama*. HLEN es un campo de 4 bits, que proporciona la longitud del encabezado del *datagrama* medida en palabras de 32 bits. El encabezado común, que no contiene opciones ni rellenos, mide 20 bytes y tiene un campo de longitud igual a 5. El TIPO DE SERVICIO es un campo de 8 bits que está subdividido en 5 campos, tres bits para especificar la prioridad del *datagrama*, los siguientes tres D, T y R especifican el tipo de transporte deseado para el *datagrama*, y los dos últimos no se utilizan. El campo LONGITUD TOTAL proporciona la longitud del *datagrama* medido en bytes, incluyendo los bytes del encabezado y los datos. El campo IDENTIFICACION contiene un entero único para identificar el *datagrama*. BANDERAS es un campo de tres bits que controlan la fragmentación, el primer bit no se utiliza, y el segundo es llamado DF que quiere decir no fragmentación y el tercero MF que significa más fragmentos. El campo DESPLAZAMIENTO DE FRAGMENTO especifica el desplazamiento en el *datagrama* original de los datos que se están acarreando en el fragmento. El campo TIEMPO DE VIDA especifica la duración en segundos del tiempo que el *datagrama* tiene permitido permanecer en la red. El campo PROTOLOCO contiene un valor que especifica qué protocolo se utilizó para crear el mensaje que se está transportando en el área de datos. El campo SUMA DE VERIFICACIÓN DEL ENCABEZADO asegura la integridad de los valores del encabezado. Los campos DIRECCION IP DE LA FUENTE y DIRECCION IP DEL DESTINO contienen la dirección IP del emisor y del receptor respectivamente. El campo OPCIONES se incluye en principio para pruebas de red o depuración.

Cuando un *datagrama* IP es enviado por la red, se encapsula en la porción de datos de la trama de la red física. En una red Ethernet, las tramas que transportan *datagramas* IP tienen un campo de tipo cuyo valor hexadecimal es 0800.

Dado que la longitud de la trama de la red se define con independencia del protocolo IP, mediante requisitos técnicos de la red física, un *datagrama* IP puede no ajustarse a una trama de red. Además, durante el camino que recorre hasta su destino, un *datagrama* puede pasar a través de diferentes tipos de redes con diferentes longitudes de trama de red. Por lo tanto, puede suceder que un *router* reciba *datagramas* IP demasiado extensos para reenviarlos a la siguiente red. Para solucionar este aspecto de la transmisión de paquetes, IP especifica un método para romper los *datagramas* en fragmentos, estos fragmentos vuelven a unirse cuando llegan a su destino final para reconstruir el *datagrama* por completo.

El protocolo IP realiza también una clase de detección de error haciendo una validación del encabezado del paquete y verificando que la longitud del paquete coincida con el valor especificado en el encabezado, también asegura que el paquete no se encuentre indefinidamente ciclado en una red tratando de alcanzar su destino. Esto lo realiza decrementando un contador de *tiempo de vida* en el encabezado, cada vez que el paquete pasa por una máquina de ruteo, y descarta el paquete una vez que este contador ha llegado a cero. El protocolo IP manda un paquete especial de error a la fuente cada vez que se detecta alguno de estos errores y lo hace por medio del protocolo ICMP.

1.2.3 Protocolo de control de mensaje Internet (ICMP)

Otro elemento del conjunto de protocolos TCP/IP es el Protocolo de Control de Mensaje Internet (ICMP). Los paquetes ICMP contienen información sobre los errores originados en la red, tales como: nodos y *routers* fuera de servicio, congestión de paquetes en un *router*. El software IP, y no la aplicación, interpreta los mensajes ICMP y realiza la acción apropiada con cada mensaje. Dado que estos mensajes pueden viajar a través de varias redes para alcanzar su destino, se encapsulan en la sección de datos de un *datagrama* IP, por lo que podemos decir que requieren dos niveles de encapsulación, es decir, el ICMP se encapsula en un *datagrama* IP, y este, en una trama Ethernet.

Aunque cada mensaje ICMP tiene su propio formato, todos comienzan con los mismos tres campos como se muestra en la figura 1.7; un campo TIPO que se utiliza para identificar el tipo de mensaje cuya longitud es de 8 bits, un campo CODIGO de 8 bits, que proporciona información adicional sobre el tipo de mensaje; y un campo SUMA DE VERIFICACIÓN de 16 bits, que se utiliza para asegurar la integridad de la información.

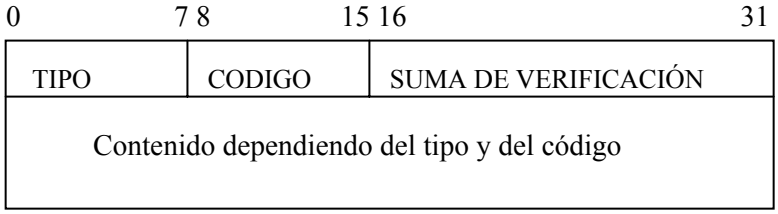

Figura 1.7 Mensaje ICMP.

En la tabla 1.2 se muestran los diferentes tipos de mensajes determinados por el campo tipo y el campo código [49]. Cuando un mensaje ICMP de error es enviado, el mensaje siempre esta contenido en el encabezado IP y los primeros 8 bits del *datagrama* IP muestran la causa del error.

Tabla 1.2 Tipos y códigos de mensajes ICMP.

TIPO	CÓDIGO	DESCRIPCIÓN
0	0	Respuesta de Eco
3		Destino no accesible
	0	Destino inaccesible
	1	*host* inaccesible
	2	Protocolo inaccesible
	3	Puerto inaccesible
	4	Requiere fragmentación pero no se encuentra el bit de fragmentación
	5	Falla en la ruta de la fuente
	6	No conoce la red destino
	7	No conoce el *host* destino
	8	*host* fuente incomunicado
	9	Red destino administrativamente prohibido
	10	*host* destino administrativamente prohibido
	11	Red inaccesible para el tipo de servicio (TOS)
	12	*host* inaccesible para el tipo de servicio (TOS)
	13	Comunicación administrativamente prohibida para filtrado
	14	Violación precedente del *host*
	15	Precedente salida realizada
4	0	Disminución de tasa fuente
5		Redireccionar (cambiar una ruta)
	0	Redireccionar *datagramas* para la red
	1	Redireccionar *datagramas* para el *host*
	2	Redireccionar *datagramas* para el tipo de servicio y la red
	3	Redireccionar *datagramas* para el tipo de servicio y el *host*
8	0	Solicitud de Eco
9	0	Respuesta de ruta
10	0	Solicitud de ruta
11		Tiempo excedido de un *datagrama*
	0	Tiempo de vida igual a 0 durante una transición
	1	Tiempo de vida igual a 0 durante un reemsamble
12		Problemas de parámetros en un *datagrama*
	0	Mal encabezado en IP
	1	Requiere opciones perdidas
13	0	Solicitud de *timestamp*
14	0	Respuesta de *timestamp*
15	0	Solicitud de información
16	0	Respuesta de información
17	0	Solicitud de máscara de dirección
18	0	Respuesta de máscara de dirección
19	0	Reservado (para seguridad)
20-29		Reservado para experimentos robustos
30		Trazado de ruta
31		Error de conversión de datagrama
32		Redireccionamiento de anfitrión móvil
33		IPv6. Dónde estás?
34		IPv6. Estoy aquí!
35		Solicitud de registro de movimiento
36		Respuesta de registro de movimiento
39		Rebote
40		Photuris

0	Reservado
1	Índice de parámetros de seguridad desconocidos
2	Parámetros de seguridad válidos, pero autenticación errónea
3	Parámetros de seguridad válidos, pero desencriptación errónea

1.2.4 Protocolos de nivel de transporte

El nivel de transporte del conjunto de protocolos TCP/IP consta de dos protocolos: el Protocolo de *datagramas* de usuario (UDP) y el Protocolo de control de transmisión (TCP). El protocolo UDP proporciona un servicio de entrega sin conexión y poco fiable para enviar y recibir mensajes y el protocolo TCP incorpora servicios de entrega fiable al servicio de entrega de *datagramas* IP.

1.2.4.1 Protocolo de *datagrama* de usuario (UDP)

UDP define un conjunto de destinos como los puertos del protocolo. Asimismo, el protocolo define dos tipos de puertos de protocolo: puertos conocidos y puertos asociados dinámicamente. En el caso de puertos conocidos, se reservan determinados números de puertos UDP para determinadas aplicaciones. Estos números se encuentran en el rango de 1 y 255, y se utilizan con aplicaciones específicas. Todas las aplicaciones de UDP hacen uso de dichos números de la misma manera. En el caso de los puertos asociados dinámicamente, las aplicaciones que solicitan servicios a un proceso deben consultar el nodo para determinar el puerto utilizado por el proceso, y después poder enviar los *datagramas* UDP al puerto.

En la figura 1.8 se muestra la estructura de un *datagrama* UDP, y como se puede observar el encabezado se divide en cuatro campos de 16 bits, que especifican el puerto desde el que se envió el mensaje, el puerto para el que se destina el mensaje, la longitud del mensaje y una suma de verificación UDP.

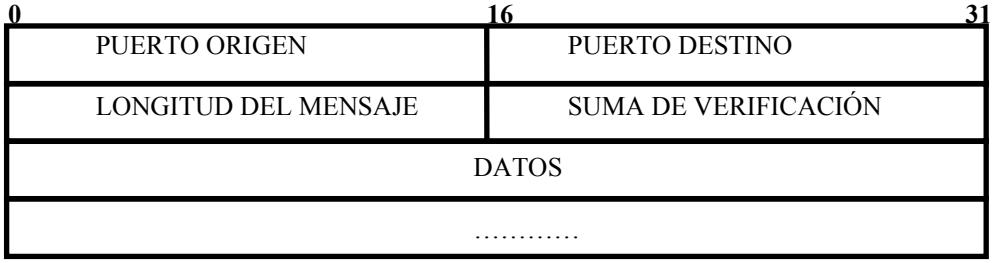

Figura 1.8 Formato de un *datagrama* UDP.

1.2.4.2 Protocolo de control de transmisión (TCP)

Para las aplicaciones que deben enviar o recibir grandes volúmenes de datos, la entrega de *datagramas* no fiables puede convertirse en una carga. Del mismo modo que los *datagramas* UDP, los segmentos TCP se encapsulan en un *datagrama* IP. TCP guarda el flujo en el buffer y espera a que un *datagrama* de tamaño grande se llene de datos antes de enviarlo, este flujo se caracteriza por carecer de estructura, de ahí que tanto la aplicación emisora como la receptora

tengan que llegar a un acuerdo sobre el contenido del mismo antes de iniciar la transmisión. El protocolo TCP usa una transmisión dúplex integral (*full-duplex*), es decir, que pueden enviarse dos flujos de datos simultáneamente en direcciones opuestas. En consecuencia, la aplicación de destino puede enviar información de control o datos de vuelta a la aplicación emisora mientras ésta continúa enviando datos.

El protocolo TCP asigna un número secuencial a cada segmento. La aplicación que se encuentra en el extremo receptor de la conexión, verifica los números de secuencia para asegurar que todos los segmentos se reciban y procesen en orden. El receptor envía un reconocimiento al emisor indicando los segmentos recibidos. TCP permite que el emisor tenga varios segmentos pendientes antes de que el receptor envíe un reconocimiento. Cuando el nodo emisor recibe el reconocimiento, indica a la aplicación que los últimos datos se enviaron satisfactoriamente, si el nodo emisor no recibe el reconocimiento de un segmento, en un período de tiempo determinado, volverá a retransmitir este segmento. Este esquema, llamado retransmisión con acuse de recibo, asegura que la entrega de flujo sea fiable.

En la figura 1.9 se muestra el formato del segmento TCP, el cual tiene un encabezado mínimo de 20 bytes. Este encabezado esta formado por los siguientes elementos: los campos PUERTO FUENTE y PUERTO DESTINO contienen los números de puertos TCP que identifican a los programas de aplicación en los extremos de la conexión. El campo NÚMERO DE SECUENCIA identifica la posición de los datos del segmento en el flujo de datos de transmisión. El campo NÚMERO DE ACUSE DE RECIBO identifica el número de bytes que la fuente espera recibir después. El campo HLEN contiene un número que especifica la longitud del encabezado del segmento, medida en múltiplos de 32 bits. El campo RESERVADO es de 6 bits que se reservan para ser usados en el futuro. El campo CODIGO de 6 bits es utilizado para determinar el propósito y contenido del segmento. Los seis bits indican cómo interpretar otros campos en el encabezado, de acuerdo con la tabla 1.3.

Tabla 1.3 Bits del campo CODIGO en el encabezado TCP.

Bit (de izquierda a derecha)	Significado si el bit está puesto a 1
URG	El campo de apuntador de urgente es válido
ACK	El campo de acuse de recibo es válido
PSH	Este segmento solicita una operación push
RST	Inicio de la conexión
SYN	Sincroniza el número de secuencia
FIN	Final de la conexión.

El campo VENTANA es utilizado para informar sobre cuántos datos está dispuesto a aceptar cada vez que se envía un segmento. El campo SUMA DE VERIFICACIÓN contiene una suma de verificación para comprobar la integridad de los datos así como el encabezado. El campo APUNTADOR DE URGENCIA solo es válido si la bandera URG esta encendida, éste campo especifica la posición en la que terminan los datos urgentes dentro del segmento. El campo OPCIONES se utiliza comúnmente para especificar el tamaño máximo de segmento (MSS) que se está dispuesto a recibir.

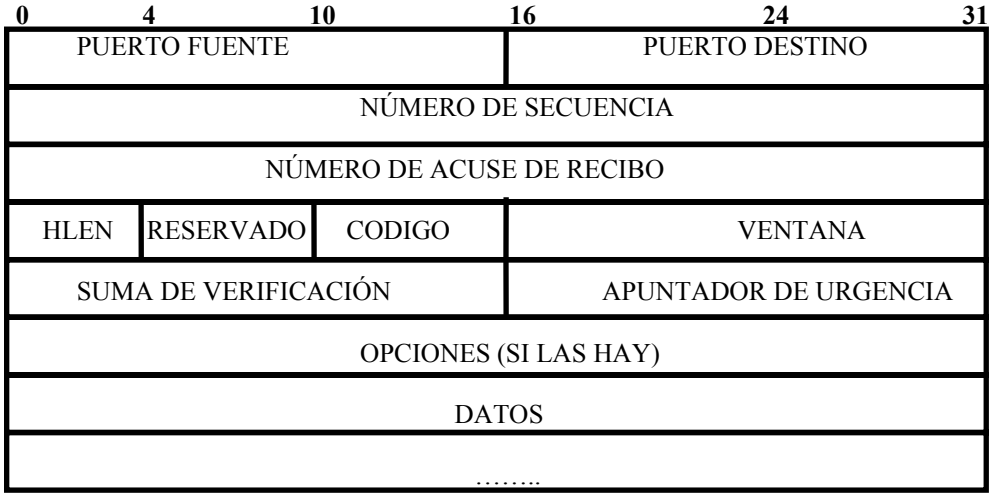

Figura 1.9 Formato de un segmento TCP.

1.2.5 Protocolo de resolución de direcciones (ARP)

Los diferentes tipos de red utilizan formatos particulares para los paquetes que envían a través de sus nodos. La estructura de estos paquetes incluye, entre otros elementos, la dirección física del nodo destino. Todos los medios físicos tienen una dirección física asignada a los nodos del medio, estas direcciones se denominan direcciones de control de acceso a medios (MAC). Las redes Ethernet o *Token Ring* representan sus direcciones MAC con 6 bytes y ARCNET las representa con 1 byte, aquí surge un problema dado que el protocolo TCP/IP utiliza direcciones de longitud de 32 bits para especificar el destino del paquete que se envía. El *protocolo de resolución de direcciones* (ARP) soluciona este problema al implementar un procedimiento de descubrimiento dinámico para el mapeo de las direcciones IP en las direcciones del hardware (Ethernet, *Token Ring*, etc.). La manera en que funciona este protocolo es la siguiente: supongamos que la red conectada a nuestra máquina es una red Ethernet, antes de que el protocolo IP mande un paquete a través de la red, el protocolo ARP consulta una tabla local para ver si existe un mapeo entre la dirección Internet de 32 bits destino y la dirección Ethernet de 48 bits del destino, si no existe, ARP manda un paquete de *broadcast* a todas las máquinas de la red, requiriendo la dirección Ethernet correspondiente a la dirección Internet de 32 bits que se tiene, el *host* con la dirección IP requerida contesta especificando su dirección Ethernet y la máquina origen recibe el mensaje y añade la entrada en su tabla de mapeo que asocia la dirección IP con la dirección Ethernet y envía el paquete a su destino.
En la figura 1.10 se muestra el formato de un paquete de solicitud o respuesta ARP usado para resolver una dirección IP en una red Ethernet. Los primeros dos campos en el encabezado Ethernet son las *direcciones destino y fuente*. Existe una dirección especial de Ethernet de destino llamada *broadcast*, la cual se reconoce porque en este campo aparecen todos los bits en uno, esta dirección se utiliza para que todas las interfaces Ethernet reciban el paquete de envío. El campo *tipo de frame* especifica el tipo de dato que se envía, para una solicitud o una respuesta ARP este campo contiene el valor hexadecimal 0806. El campo *hardware* especifica el tipo de dirección de hardware, este valor es 1 para una red Ethernet. El campo *protocolo* especifica el tipo de dirección de protocolo de mapeo, el valor 0800_{16} aparece cuando es una dirección IP. Los siguientes dos campos, *tamaño de hardware y tamaño de protocolo*, especifican el tamaño en bytes de las direcciones de hardware y las direcciones del protocolo. En una solicitud o una respuesta ARP para una dirección IP o una Ethernet aparece un 6 y un 4 respectivamente.

El campo *operación* especifica si la operación es una solicitud ARP (valor de 1), una respuesta ARP (2), una solicitud RARP (3) o una respuesta RARP (4).

Los siguientes cuatro campos son las direcciones del hardware de envío (una dirección Ethernet en este ejemplo), la dirección del protocolo de envío (una dirección IP), la dirección de hardware de la tarjeta y la dirección del protocolo de la tarjeta.

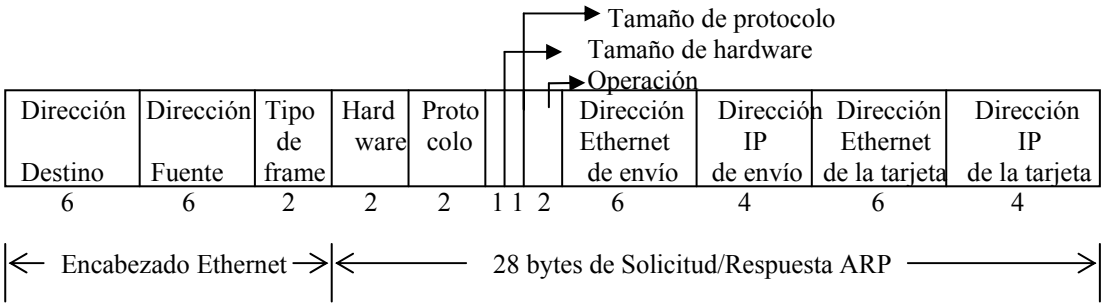

Figura 1.10 Paquete de solicitud o respuesta ARP en Ethernet .

1.2.6 Protocolo de asociación de direcciones por réplica (RARP)

Una máquina sin disco utiliza el protocolo de asociación de direcciones por réplica (RARP) a fin de obtener su dirección IP de un servidor. Al igual que un mensaje ARP, un mensaje RARP se envía de una máquina a otra, encapsulado en la porción de datos de una trama de red. Una trama Ethernet que transporta una solicitud RARP tiene el preámbulo usual, las direcciones Ethernet tanto destino como fuente y el campo de tipo de paquete al comienzo de la trama. El tipo de trama contiene el valor hexadecimal 8035 para identificar que en el contenido de la trama se transporta un mensaje RARP.

Para concluir este capítulo podemos decir lo siguiente: para que un dato sea transportado por la red, este debe pasar por varios protocolos, ellos colocan un encabezado en el dato con la finalidad de que se reciba correctamente. Cuando el dato se recibe se deben quitar y analizar estos encabezados, esto con el fin de verificar si no ha ocurrido algún imprevisto en el envío.

Los encabezados que se colocan para enviar un dato son: encabezado de aplicación, encabezado de transporte, encabezado IP y encabezado de enlace físico. Cuando se recibe el dato se lleva a cabo la operación inversa, es decir, se quitan todos los encabezados comenzando con el encabezado físico y finalizando con la aplicación que lo envío.