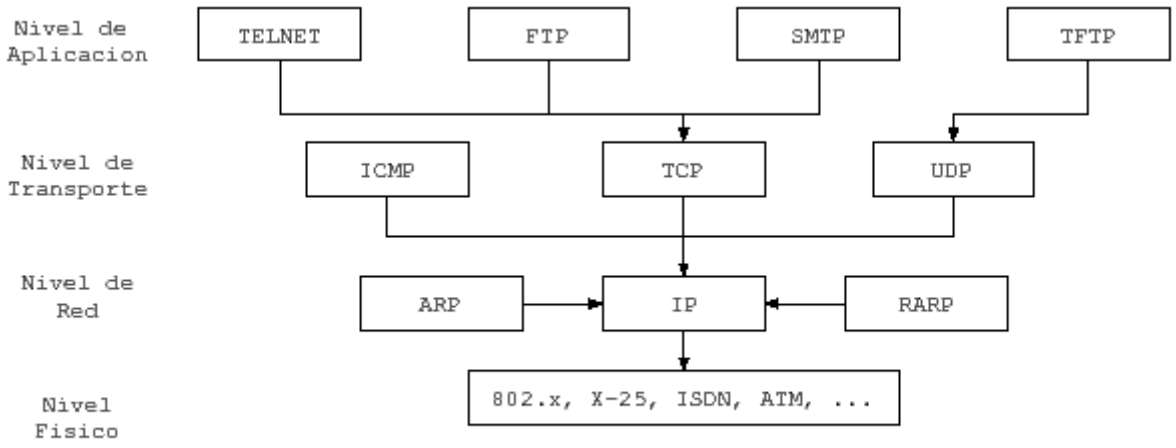
La pila de protocolos y su interrelación.

La arquitectura TCP/IP consta de cuatro niveles. Un primer nivel de acceso a la red, que englobaría el nivel físico, el de enlace de datos y parte del nivel de red del modelo de referencia OSI de la ISO. Un segundo nivel de inter-red ofrece un servicio de transmisión de la información entre dos puntos remotos de la red, no orientado a conexión y sin fiabilidad. Este mecanismo es controlado por el protocolo IP, ofreciendo unos servicios similares al nivel de red de la OSI. La unidad básica de información que maneja este nivel se denomina datagrama, según el argot de Internet. El protocolo IP no es fiable porque no asegura la entrega de datagramas. No está orientado a conexión porque el protocolo IP no mantiene la situación de los datagramas sucesivos enviados, de forma que los datagramas pueden llegar al destino duplicados, en orden incorrecto, etc.

Un tercer nivel de transporte suministra un transporte de información entre procesos funcionando en estaciones remotas, con o sin fiabilidad. En este nivel la unidad de información del nivel de transporte es el paquete. En el nivel de transporte, y ofreciendo sus servicios directamente a las aplicaciones, tenemos el protocolo TCP (*Transmission Control Protocol*) que da un servicio orientado a conexión para el transporte fiable de datos extremo a extremo, es decir, capaz de asegurar la entrega de información sin errores. Esto se consigue haciendo un control de errores y pidiendo al receptor las retransmisiones que sean necesarias al emisor. Se dice que es orientado a conexión, puesto que los dos procesos involucrados en la comunicación establecen una conexión antes de iniciar la comunicación, y se hace una reordenación de los paquetes recibidos. Por otro lado, en este nivel también está el protocolo UDP (*User Datagram Protocol*) que da un servicio no orientado a conexión, muy similar al que ofrece IP. En este sentido, y debido a que no hace retransmisiones, no verifica la entrega ni la corrección de datos. UDP permite el envío de información de forma más eficiente y rápida. Así se puede decir que TCP es más conveniente para la transferencia de ficheros, el acceso vía terminal remoto o la descarga de páginas web, mientras que UDP es más adecuado para servicios en tiempo real y que toleran algunos errores o pérdidas, como el envío de audio o vídeo (telefonía, videoconferencia, etc.).

Finalmente el cuarto nivel o nivel de aplicación, nos da la posibilidad de abrir y controlar una sesión con un nodo remoto para transferir información formateada (servicio de transferencia de ficheros), para establecer un diálogo interactivo remoto (servicio de terminal virtual), o para enviar mensajes textuales electrónicos en diferido (servicio de correo electrónico). Este nivel de la arquitectura TCP/IP engloba aproximadamente los servicios de sesión, presentación y aplicación de la OSI.

En el entorno *Internet* se habla de que la red está formada por redes o subredes, que conectan localmente los *hosts* (sistemas informáticos anfitriones de aplicaciones y usuarios), interconectadas a la vez por *gateways* (pasarelas). En la arquitectura TCP/IP habrá entonces otros protocolos para el diálogo *host-gateway* y *gateway-gateway* con objeto de gestionar el encaminamiento, controlar los flujos de datos, notificar errores, etc. Algunos de estos protocolos, digamos "auxiliares", son el ICMP (*Internet Control Message Protocol*), el RIP (*Routing Information Protocol*), el ARP (*Address Resolution Protocol*), el RARP (*Reverse Address Routing Protocol*), OSPF (*Open Shortest Path First*), etc.



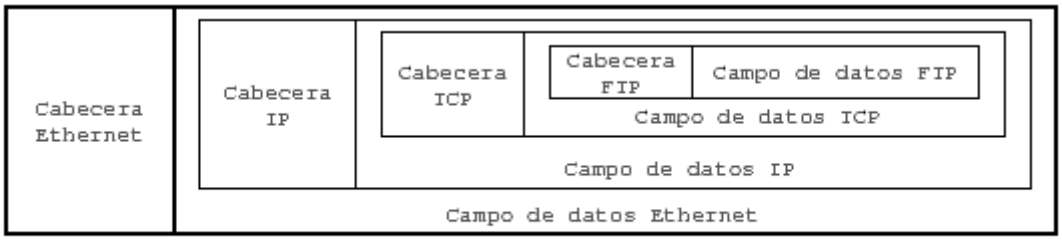
Clasificación de los protocolos más significativos de la arquitectura TCP/IP.

El hecho de que TCP/IP sea una arquitectura universal hace que los datagramas puedan viajar por cualquier medio físico, topología, o protocolo de enlace, (*Ethernet*, *Token Ring*, FDDI, X-25, ATM, etc.). En nuestro caso las estaciones de trabajo están interconectadas mediante una topología en estrella empleando el protocolo *ethernet*.

Se observa que dentro del campo de datos de la trama *ethernet*, está el datagrama, mientras que dentro del campo de datos del datagrama IP está el paquete TCP.

La comunicación entre dos nodos de nuestra red se hace mediante datagramas que son transportados dentro del campo de datos de la trama *ethernet*.

Una trama enviada por un *host* es detectada por todas las demás estaciones conectadas, pero sólo el nodo destinatario la recoge y la procesa. Si dos *hosts* intentan emitir a la vez, se produce lo que se denomina una colisión. En esta situación los dos *hosts* abortan la transmisión y realizan un reintento al cabo de un intervalo aleatorio de tiempo. Dentro de una red *ethernet*, las colisiones son un fenómeno natural. En un sistema con actividad elevada, niveles de colisión que ocupen el 30% del ancho de banda de la red pueden ser habituales.



Encapsulamiento del servicio FTP empleando una trama ethernet.

La capa de inter-red: el protocolo IP

El objetivo de el protocolo IP es convertir redes físicamente heterogéneas (como pueden ser *Ethernet*, *Token Ring*, X.25, *Frame Relay*, ATM...) en una red aparentemente homogénea, lo que se conoce como interconexión de redes. A la red resultante se la puede denominar *internet* (observar la diferencia con *la Internet*), dónde podemos destacar que:

- Hay un esquema de identificación (o direccionamiento) de todos los sistemas, uniforme y universal. Este esquema de direccionamiento tiene que ser independiente del *hardware*. Esto

se consigue asignando a cada nodo un número único de 32 bits (normalmente, puesto que suelen ser IPv4, *Internet Protocol version 4*) denominado dirección IP.
- Las comunicaciones entre usuarios siguen un método uniforme denominado encaminamiento de datagramas, independiente de la red en particular dónde residen.

El protocolo IP es un protocolo de red no orientado a conexión que proporciona un servicio de entrega de datagramas no fiable.

Encapsulación de datos

En el proceso de transmisión de información, los datos deben ser convertidos para que puedan viajar por los medios y ser interpretados por los dispositivos de la red. Este proceso se conoce como encapsulaciónProceso que codifica y transforma los datos para que puedan viajar por los medios y ser interpretados por los dispositivos de la red..

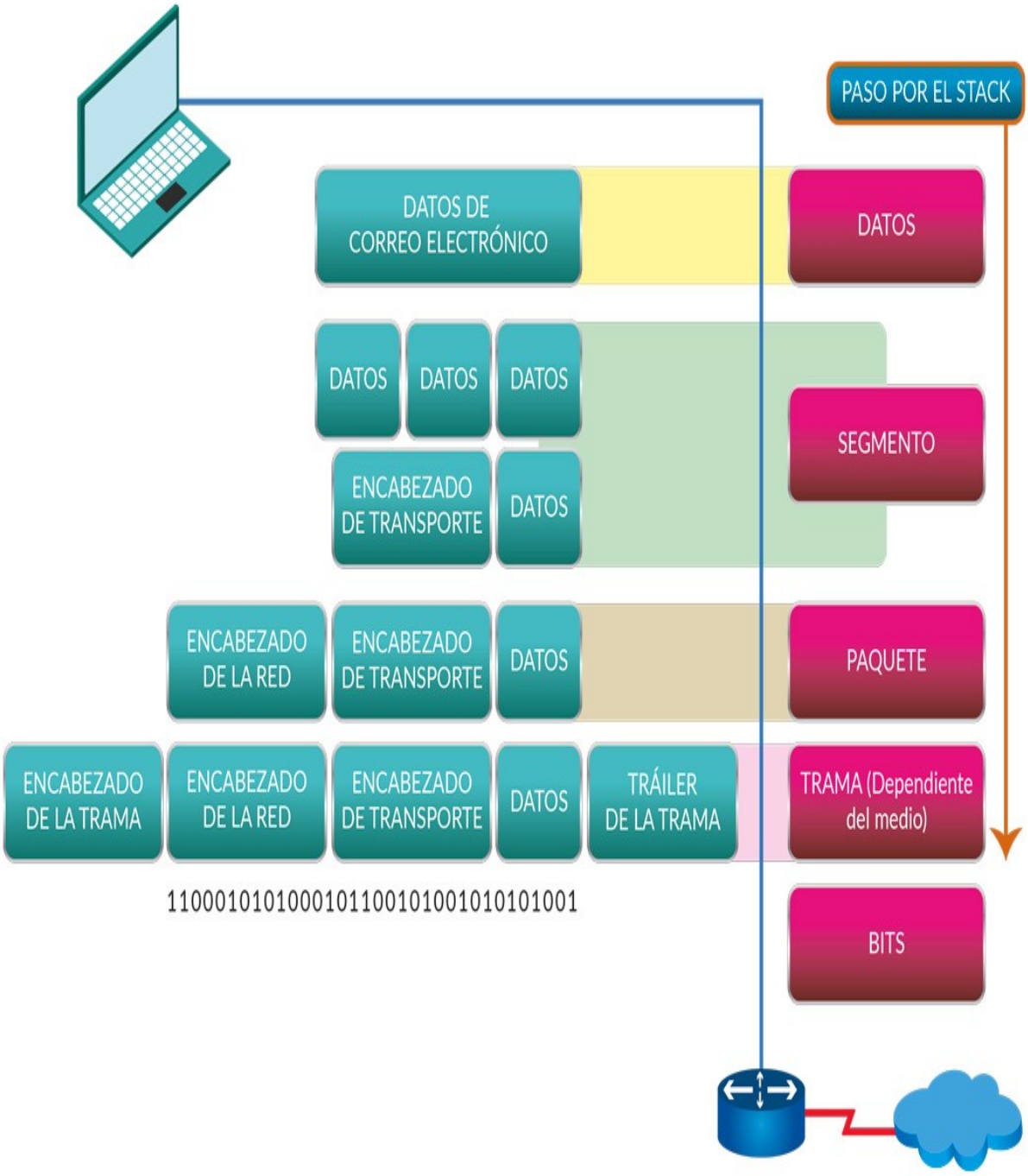
Este proceso está relacionado con las diferentes capas del modelo OSI, y consiste en dar formato a los datos y agregar la información necesaria a medida que pasan por cada capa dependiendo del protocolo que intervenga en ese momento en la comunicación.

La información que se agrega en cada capa se conoce como encabezado; estos contienen la información de control para cada dispositivo de la red y aseguran el envío correcto de los datos al receptor. Los encabezados reciben un nombre específico en cada capa conocidos como unidades de datos de protocolo (PDUUnidad de datos de protocolo. Nombre que recibe los datos al agregar el encabezado correspondiente en cada capa de los modelos OSI o TCP/IP durante el proceso de):

a. **Datos:** En la capa de aplicación, presentación y sesión.
b. **Segmentos:** En la capa de Transporte.
c. **Paquetes o datagramas:** En la capa de Red.
d. **Tramas:** en la capa de Enlace de datos.
e. **Bits:** En la capa Física.

El proceso de encapsulación (proceso que codifica y transforma los datos para que puedan viajar por los medios y ser interpretados por los dispositivos de la red) consta de los siguientes pasos:

1. Los datos son generados por el usuario en la capa de aplicación, estos pasan a las capas de presentación y sesión donde se les da el formato o representación especifica y se agrega el encabezado de protocolo correspondiente a estas capas para ser enviados a la capa de Transporte.
2. Los datos son recibidos en la capa de transporte, esta capa divide los datos en porciones más pequeñas para facilitar su transmisión. A cada porción le agrega el encabezado correspondiente con la información de los protocolos de capa de transporte. Cada porción de datos se convierte en un segmento y es enviado a la capa de red.
3. La capa de red recibe cada segmento y le agrega el encabezado correspondiente. Cada segmento se convierte en un paquete para ser enviado a la capa de enlace de datos.
4. La capa de enlace de datos recibe el paquete, le agrega su encabezado y convertir el paquete en una trama que es enviada a la capa física.
5. Por último la capa física recibe las tramas y las convierte en bits, los cuales son enviados a través de los medios de red hacía el destino.



Proceso de encapsulación

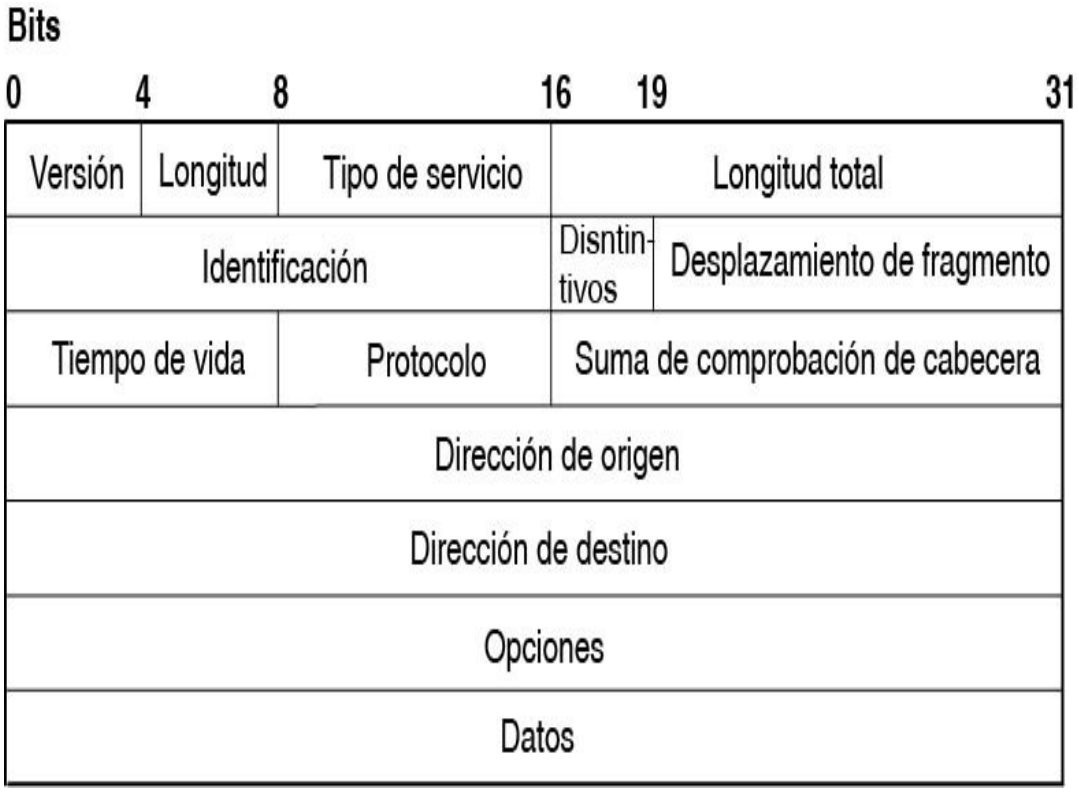
Una vez los datos son recibidos por el host receptor este comienza el proceso de convertir nuevamente los bits en datos, para ello comienza a quitar los encabezados correspondientes a cada capa del modelo OSI. Este proceso se conoce como desencapsulación.

PROTOCOLO IP

El tercer protocolo de nivel de red es **IP** (**Internet Protocol - Protocolo Internet**), que proporciona la entrega de paquetes sin conexión no fiable para Internet.

IP no tiene conexiones porque trata cada paquete de información de forma independiente. No es fiable porque no garantiza la entrada, lo que significa que no necesita reconocimientos del sistema principal de envío, del sistema principal de recepción ni de los sistemas principales intermedios.

En la red masiva conocida como Internet, los dispositivos de cómputo envían todo tipo de mensajes a otros dispositivos de cómputo. Un mensaje puede ser un pequeño ping para comprobar si otro dispositivo está en línea, o puede ser una página web completa.
Pero hay un límite al tamaño de un mensaje, ya que hay un límite a la cantidad de datos que pueden ser razonablemente transmitidos a la vez por las conexiones físicas de red entre dispositivos.
Es por eso que el protocolo ip divide cada mensaje en varios **paquetes** pequeños. El Protocolo Internet (IP) describe la estructura de los paquetes que viajan raudos y veloces por Internet.
Cada paquete IP contiene tanto un encabezado (de 20 o 24 bytes de longitud) como datos (de longitud variable). El encabezado incluye las direcciones IP de la fuente y del destino, además de otros campos que ayudan a enrutar el paquete. Los datos son el contenido real, tales como una cadena de letras o parte de una página web.



Definiciones de campo de cabecera IP

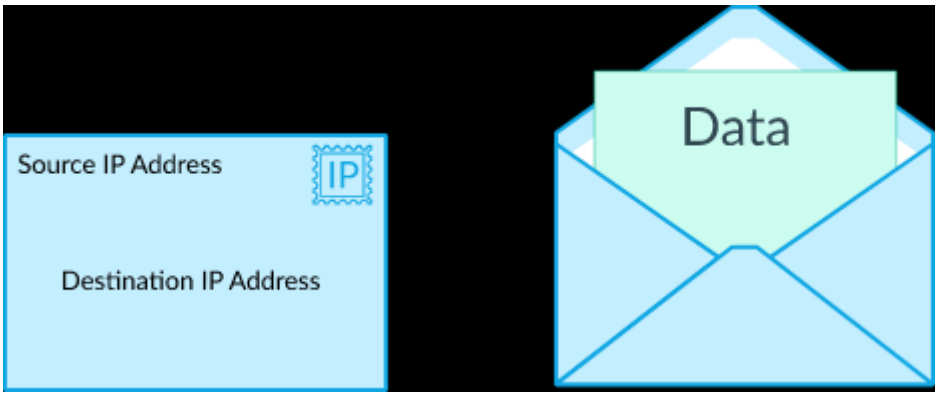
Item	Descripción
Version	(Versión) Especifica la versión de **IP** utilizada. La versión actual del protocolo **IP** es 4.
Length	(Longitud) Especifica la longitud de cabecera de datagrama, medida en palabras de 32 bits.
Type of	(Tipo de servicio) contiene cinco subcampos que especifican el tipo de

Item	Descripción
	precedencia, retardo, rendimiento y fiabilidad deseados para dicho paquete. (Internet no garantiza esta petición.) Los valores predeterminados para estos cinco subcampos son precedencia de rutina, retardo normal, rendimiento normal y fiabilidad normal. Normalmente Internet no utiliza este campo en este momento. Esta implementación de **IP** satisface los requisitos de la especificación **IP**, RFC 791, *Protocolo Internet*.
Service	
Total Length	(Longitud total) Especifica la longitud del datagrama incluyendo la cabecera y los datos medidos en octetos. Se proporciona la fragmentación de paquetes en las pasarelas, con reensamblaje en los destinos. La longitud total del paquete **IP** se puede configurar de interfaz en interfaz con el mandato ifconfig o la vía de acceso rápida de System Management Interface Tool (SMIT), `smit chinet`. Utilice SMIT para establecer los valores permanentemente en la base de datos de configuración; utilice el mandato ifconfig para establecer o cambiar los valores en el sistema en ejecución.
Identification	(Identificación) Contiene un entero exclusivo que identifica el datagrama.
Flags	(Distintivos) Controla la fragmentación de datagrama, junto con el campo de identificación. Los distintivos de fragmento especifican si el datagrama se puede fragmentar y si el fragmento actual es el último.
Fragment Offset	(Desplazamiento de fragmento) Especifica el desplazamiento de este fragmento en el datagrama original medido en unidades de 8 octetos.
Time to Live	(Tiempo de vida) Especifica cuánto tiempo puede permanecer el datagrama en Internet. Esto evita que los datagramas direccionados incorrectamente permanezcan en Internet de forma indefinida. El tiempo de vida predeterminado es 255 “saltos”.
Protocol	(Protocolo) Especifica el tipo de protocolo de alto nivel.
Header Checksum	(Suma de comprobación de cabecera) Indica un número calculado para asegurar la integridad de los valores de cabecera.
Source Address	(Dirección de origen) Especifica la dirección de Internet del sistema principal de envío.
Destination Address	(Dirección de destino) Especifica la dirección de Internet del sistema principal de recepción.
Options	(Opciones) Proporciona pruebas y depuración de red. Este campo no es necesario para cada datagrama. **End of Option List** (Fin de lista de opciones) Indica el final de la lista de opciones. Se utiliza al final en la última opción, no al final de cada opción individualmente. Sólo se debe utilizar esta opción si, de otra forma, el final de las opciones no va a coincidir con el final de la cabecera **IP**. El fin de lista de opciones se utiliza si las opciones exceden la longitud del datagrama. **No Operation** (Ninguna operación) Proporciona alineación entre otras opciones; por ejemplo, para alinear el principio de una opción subsiguiente en un límite de 32 bits. **Loose Source and Record Route** Proporciona un medio para que el origen de un datagrama de Internet proporcione la información de direccionamiento utilizada por las pasarelas al reenviar el datagrama a un destino y al registrar la información de ruta. Esta ruta de origen es *flexible*: Se permite a la pasarela o al sistema principal **IP**

Item	Descripción
	utilizar cualquier ruta de cualquier número de otras pasarelas intermedias con el fin de alcanzar la siguiente dirección de la ruta.
	Strict Source and Record Route Proporciona un medio para que el origen de un datagrama de Internet proporcione la información de direccionamiento utilizada por las pasarelas al reenviar el datagrama a un destino y al registrar la información de ruta. Esta ruta de origen es *estricta*: Para alcanzar la siguiente pasarela o el siguiente sistema principal especificado en la ruta, la pasarela o el sistema principal **IP** debe enviar el datagrama directamente a la siguiente dirección de la ruta de origen y sólo a la red directamente conectada que se indica en la siguiente dirección.
	Record Route (Ruta de registro) Proporciona un medio para registrar la ruta de un datagrama de Internet.
	Stream Identifier (Identificador de corriente) Proporciona un medio para que un identificador de corriente pase por redes que no soportan el concepto de corriente.
	Indicación de la hora de Internet (Indicación de la hora de Internet) Proporciona un registro de las indicaciones de la hora por la ruta.

En los paquetes de salida se pone automáticamente como prefijo una cabecera **IP**. En los paquetes de entrada, la cabecera **IP** se elimina antes de enviar dichos paquetes a los protocolos de nivel más alto. El protocolo **IP** proporciona el direccionamiento universal de sistemas principales en la red Internet.

Podemos pensar que los paquetes IP son como cartas de correo: el encabezado es el sobre con toda la información de enrutamiento que necesita la oficina de correos, y la carga útil es la carta que solo lee el destinatario.



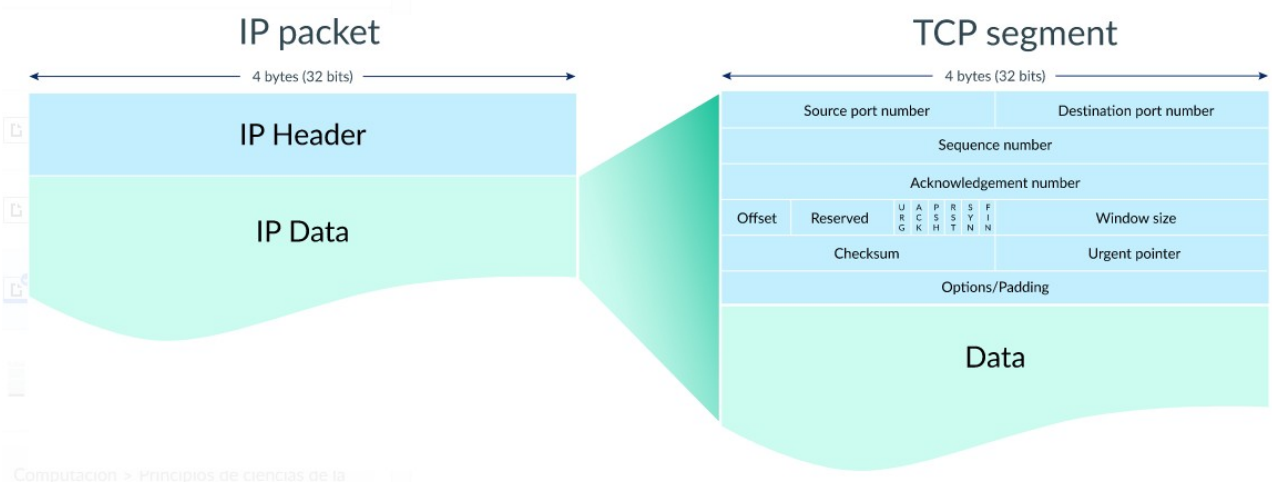
Protocolo TCP

El propósito primordial de TCP es *proporcionar circuitos lógicos confiables o servicios de conexión entre parejas de procesos.* Esto *no* implica confiabilidad desde protocolos de más bajo nivel (como IP) así que TCP debe garantizar esto por sí mismo.

TCP se puede caracterizar por las siguientes facilidades que proporciona a las aplicaciones que hacen uso de él:

- **Tranferencia de flujos de datos**. Desde el punto de vista de la aplicación, TCP transfiere *un flujo de bytes contiguo* a través de internet. La aplicación no tiene que preocuparse troceando los datos en bloques básicos o datagramas. TCP hace esto agrupando los bytes en *segmentos TCP*, que se transfieren a IP para transmitirlos al destino. TCP decide también por sí mismo cómo segmentar los datos y debe dirigir los datos a su propia conveniencia.
- **Confiabilidad.** TCP asigna un número de secuencia a cada byte transmitido y espera por un reconocimiento positivo (ACK) del receptor TCP. Si el ACK no se recibe en un intervalo fijado, los datos se retransmiten. Como los datos se transmiten en bloques (segmentos TCP) sólo el número de secuencia del primer byte de datos en los segmentos se envían al host destino.

  El receptor TCP utiliza los números de secuencia para reorganizar los segmentos cuyo lleguen fuera de orden y para eliminar segmentos duplicados.
- **Control de flujo.** El receptor TCP, cuyo envía un ACK de vuelta al emisor, también indica al emisor el número de bytes que puede recibir más allá del último segmento TCP recibido sin causar ni overrun ni desbordamiento en sus búferes internos. Este se envía en el ACK de forma of the highest sequence number it puede recibir sin problemas. Este mecanismo también se denomina mecanismo *ventana.*
- **Multiplexación.** Se logra usando puertos distintos para cada aplicación.
- **Conexiones lógicas.** La confiabilidad y los mecanismos de control de flujo descritos anteriormente requieren que TCP inicializa y mantenga cierta información de estado para cada "flujo de datos". La combinación de este estado, incluyendo sockets, números de secuencia y tamaños de ventana, se llama conexión lógica. Cada conexión se identifica unívocamente por la pareja de sockets usados por los procesos emisor y receptor.
- **Full Duplex.** TCP proporciona flujos de datos concurrentes en ambas direcciones.



Cabecera TCP

Puerto de Origen			Puerto de Destino
Número de secuencia			
Número de reconocimiento			
Offset	Reservado	Bits de Bandera (Flag)	Ventana
Suma de Control (Checksum)			Urgente

32 Bits (4 Bytes)

Puerto Origen (16 bits): Identifica el puerto emisor.

Puerto de Destino (16 bits): Identifica el puerto receptor.

Número de secuencia (32 bits): Identifica el byte del flujo de datos enviado por el emisor TCP al receptor TCP.

Número de reconocimiento o acuse de recibo (32 bits): Contiene el valor del siguiente número de secuencia que el receptor del segmento espera recibir.

Offset o Longitud de cabecera (4 bits): Especifica el tamaño de la cabecera en palabras de 32 bits.

Reservado (3 bits): Para uso futuro.

Bits de Bandera o Flag (9 bits): Nueve banderas de 1 bit para distintos propósitos.

Ventana (16 bits): Tamaño de ventana que especifica el número máximo de bytes que pueden ser metidos en el buffer de recepción.

Suma de control o verificación(16 bits): Checksum utilizado para la comprobación de errores tanto en la cabecera como en los datos.

Urgente (16 bits): Cantidad de bytes desde el número de secuencia que indica el lugar donde acaban los datos urgentes.

Aplicaciones que utilizan TCP

Número de puerto	Aplicación
20	FTP datos
21	FTP control
22	SSH
23	Telnet
25	SMTP
53	DNS
80	HTTP (WWW)
110	POP3
443	SSL

Confiabilidad en TCP

TCP provee **confiabilidad** en la transferencia de datos, llamado también **recuperación de errores**.

Para poder brindar esa confiabilidad, TCP **enumera** los bytes utilizando una **Secuencia** y **Reconocimiento (Acknowledgment)** de los campos en la cabecera TCP.

TCP logra la confiabilidad en ambas direcciones, utilizando el campo **Número de Secuencia para un dirección** combinado con el campo **Número de Reconocimiento en la dirección contraria**.