IP

Un datagrama IP consiste en una parte de cabecera y una parte de exto. La cabecera tiene una parte fija de 20bytes y una parte opcional de longitud variableEl formato de la cabecera se muestra en la imagen inferior. Se transmite en orden *big hendian*:* de izquierda a derecha, comenzando por el bit de orden mayor del campo de *versión*.

El formato de un datagrama IP es:

<table>
<tr><td colspan="2">Versión</td><td>IHL</td><td colspan="2">Tipo De Servicio</td><td colspan="4">Longitud total</td></tr>
<tr><td colspan="5">Identificación</td><td></td><td>DF</td><td>MF</td><td colspan="2">Desplazamiento De Fragmento</td></tr>
<tr><td colspan="2">Tiempo de Vida</td><td colspan="3">Protocolo</td><td colspan="4">Suma de comprobación de la cabecera</td></tr>
<tr><td colspan="9">Dirección IP de la fuente</td></tr>
<tr><td colspan="9">Dirección IP del destino</td></tr>
<tr><td colspan="9">Opciones IP (Opcional)</td></tr>
<tr><td colspan="9">DATOS</td></tr>
</table>

Los Datagrama IP están formadas por *Palabras* de 32 bits.

Versión *:* Versión de IP que se emplea para construir el Datagrama. Se requiere para que quien la recibe lo interprete correctamente.

IHL *:* Indica la longitud en palabras de 32 bits. El valor mínimo es de 5, cifra que aplica cuando no hay opciones. El valor máximo de este campo de 4bits es de 15, lo que limita la cabecera a 60bytes y, por tanto, el campo de opciones es de 40bytes. Para algunas opciones 40bytes es muy poco, lo que hace inútil esta opción.

Tipo de Servicio*:* permite al *host* indicar a la subred el tipo de servicio que requiere. El campo mismo contiene un campo de *precedencia*; tres indicadores, D, T y R; y dos bits no usados. El campo de precedencia es una prioridad (de 0 a 7). Los tres bits indicadores permiten al *host* especificar lo que interesa más del grupo {retardo (*delay*), rendimiento (*throughput*), confiabilidad (*reliability*)}.En teoría, estos campos permiten a los enrutadores tomar decisiones. En práctica, los enrutadores actuales ignoran por completo el campo *tipo de servicio*.

Longitud total*:* incluye todo el datagrama (tanto la cabecera como los datos). La longitud máxima es de 65.535bytes.

Identificación *:* es necesario para que el *host* de destino determine a qué datagrama pertenece un fragmento recién llegado. Todos los fragmentos de un datagrama contienen el mismo valor de *identificación*.

DF *:* (1 bit) Don´t fragment, es una orden para los enrutadores de que no fragmenten el datagrama.

MF *:* (1 bit) Significa más fragmentos. Todos los fragmentos excepto el último tienen establecido este bit.

Desplazamiento del fragmento *:* indica en qué parte del datagrama actual va este fragmento. Todos los fragmentos excepto el último del datagrama deben tener un múltiplo de 8bytes, que es la unidad de fragmento elemental. La longitud máxima del datagrama puede ser 65.536bytes, uno más que el campo *longitud total*.

Tiempo de Vida *:* es un contador que sirve para limitar la vida de un paquete. Se supone que el contador cuenta la vida en segundos, permitiendo una vida máxima de 255seg. En la práctica, simplemente cuenta los saltos. Cuando el contador llega a cero, el paquete se descarta y se envía de regreso un paquete de aviso al *host* de origen.

Protocolo *:* indica la capa de transporte a la que debe entregarse (TCP, UDP...).

Suma de comprobación de la cabecera*:* verifica solamente la cabecera.

Dirección de Origen y Dirección de Destino *:* indican el nº de red y nº de *host.*

Opciones*:* para proporcionar un recurso que permitiera que las versiones subsiguientes del protocolo incluyeran información no presente en el diseño original. Las opciones son de longitud variable. Cada una empieza con un código de 1byte que identifica la opción. Actualmente hay cinco opciones definidas, que se listan a continuación:

Opción	Descripción
Seguridad	Especifica qué tan secreto es el datagrama
Enrutamiento estricto desde origen	Indica la trayectoria completa a seguir
Enrutamiento libre desde el origen	Da una lista de los enrutadores que no deben evitarse
Registrar ruta	Hace que cada enrutador agregue su dirección IP
Marca de tiempo	Hace que cada enrutador agregue su dirección y su marca de tiempo

TCP

La entidad TCP transmisora y la receptora intercambian datos en forma de segmentos. Un segmento consiste en una cabecera TCP fija de 20bytes (más una parte opcional) seguida de 0 o más bytes de datos. El software de TCP decide el tamaño de los segmentos; puede acumular datos de varias escrituras para formar un segmento, o dividir los datos de una escritura en varios segmentos. Hay dos límites que restringen el tamaño de segmento:

- Cada segmento, incluida la cabecera TCP, debe caber en la carga útil de 65.535bytes del IP.
- Cada red tiene una **unidad máxima de transferencia**, o MTU, y cada segmento debe caber en la MTU.

En la práctica, las MTU generalmente son unos cuantos miles de bytes y por tanto definen el límite superior del tamaño del segmento. Si un segmento pasa a través de una serie de redes sin fragmentarse y luego se topa con una cuya MTU es menor que el segmento, el enrutador de la frontera fragmenta el segmento en dos o más segmentos más pequeños.

Un segmento demasiado grande para transitar por una red pueden dividirse en varios segmentos mediante un enrutador. Cada segmento nuevo recibe sus propias cabeceras TCP e IP, por lo que la fragmentación en los enrutadores aumanta la carga extra total.

El protocolo básico usado por las entidades TCP es el protocolo de ventana deslizante. El protocolo TCP debe estar preparado para enfrentarse a cualquier problema que pueda surgir eficientemente, por ello se ha invertido una cantidad considerable de esfuerzo en la optimización del desempeño de corrientes TCP, incluso problemas de red.

La cabecera de segmento TCP

El formato del segmento TCP es el siguiente:

Puerto de Origen			Puerto de Destino	
Número de secuencia				
Número de Reconocimiento				
Longitud	Reservado	Code Bits	Tamaño de laVentana	
Suma de comprobación			Puntero Datos Urgentes	
Opciones				Relleno
DATOS				

Los Segmentos TCP están formadas por *Palabras* de 32 bits.

Puerto de Origen y Puerto de Destino: identifican los puertos terminales locales de la conexión. Cada *host* puede decidir por sí mismo la manera de asignar sus propios puertos comenzando por el 256.

Número de secuencia

Número de acuse de recibo

Longitud (longitud de cabecera TCP): indica la cantidad de palabras de 32 bits contenidas en la cabecera TCP. Esta información es necesaria porque el campo de opciones es de longitud variable, por lo que la cabecera también. Este campo en realidad indica el comienzo de los datos en el segmento, medido en palabras de 32bits.

Reservado: campo de 6 bits que no se usa

Code Bits: Seis banderas de 1 bit

- URG: se establece en 1 si está en uso el *apuntador urgente*.
- ACK: se establece en 1 para indicar que el *número de acuse de recibo* es válido. Si el ACK es 0, el segmento no contiene un acuse de recibo, por lo que se ignora el campo *número de acuse de recibo*.
- PSH: indica datos empujados (con PUSH). Por este medio se solicita al receptor entregar los datos a la aplicación a su llegada y no ponerlos en *buffer* hasta la recepción de un *buffer* completo.
- RST: se usa para restablecer una conexión que se ha confundido debido a una caída de *host* u otra razón; también sirve para rechazar un segmento no válido o un intento de abrir una conexión.
- SYN: se usa para establecer conexiones. La solicitud de conexión tiene SYN = 1 y ACK = 0 para indicar que el campo de acuse de recibo incorporado no está en uso. La respuesta de conexión sí lleva un reconocimiento, por lo que tiene SYN = 1 y ACK = 1. En esencia, el bit SYN se usapara denotar CONNECTION REQUEST y CONNECTION ACCEPTED, usándose el bit ACK para distinguir entre ambas posibikidades.
- FIN: se usa para liberar una conexión; especifica que el transmisor no tiene más datos que transmitir.

Tamaño de Ventana : indica la cantidad de bytes que pueden enviarse comenzando por el byte que ya se ha enviado acuse de recibo.

Suma de Comprobación: Es una suma de comprobación de la cabecera, los datos y la pseudocabecera conceptual mostrada a continuación. Al realizar este cálculo, se establece el campo de suma de comprobación del TCP en 0, y se rellena el campo de datos con un byte cero adicional si la longitud es un número non. El algoritmo de suma simplemente suma todas las palabras de 16 bits en complemento a 1 y luego obtiene el complemento a 1 de la suma. Como consecuencia, al realizar el cálculo el receptor con el segmento completo,

incluido el campo de suma de comprobación, el resultado debe ser 0.

Puntero a Datos Urgente: sirve para indicar un bytes a partir del número actual de secuencia en el que se encuentran datos urgentes. Este recurso sustituye los mensajes de interrupción.

Opciones: se diseñó para contar con una manera de agregar características extra no cubiertas por la cabecera normal.

32bits

Dirección de origen		
Dirección de destino		
0 0 0 0 0 0 0 0	Protocolo = 6	Longitud de segmento TCP

Figura: Pseudocabecera incluida en la suma de comprobación del TCP.

La pseudocabecera contiene las direcciones IP de 32bits de las máquinas de origen y de destino, el número de protocolo de TCP (6), y la cuenta de bytes del segmento TCP (incluida la cabecera). La inclusión de la pseudocabecera en el cálculo de la suma de comprobación TCP ayuda a detectar paquetes mal entregados, pero hacerlo viola la jerarquía de protocolos puesto que las direcciones IP que contiene pertenecen a la capa IP, no a la capa TCP.

FTP

Los objetivos principales de este protocolo son:

- Posibilitar la ***compartición archivos*** entre computadoras (programas y/o datos)
- Posibilitar el ***uso remoto*** de las computadoras
- ***Transferir datos*** de una forma segura y optima entre computadoras.

FTP mas que para ser usado por un usuario directamente es usado por los programas para comunicarse, lo que facilita al usuario despreocuparse de las características del sistema con que conecta.

Terminología relacionada:

- ***ASCII*** Solo se usan todos los caracteres dentro de los 8 bits en su valor bajo
- ***Access controls*** Este sirve para hablar a cerca de los privilegios (derechos en la red) de cada usuario tanto en archivos como en dispositivos.
- ***Data connection*** Habla de cuando hay una comunicación Full Duplex entre dos computadoras.
- ***DTP*** Proceso de la transferencia.
- ***Error Recovery*** Este es un procedimiento que le permite al usuario en algunos casos recuperar información perdida en el proceso de transferencia.

Tipos d e datos en la transferencia por FTP:

- El tipo ***ASCII***, es el mas común en el protocolo FTP. Se usa cuando se transfieren archivos de texto, la computadora que envía (sender), cualquiera que sea su estructura de archivos interna, debe convertir sus datos al formato genérico de 8 bits, y el que recibe (receiver) lo debe convertir de nuevo a su formato propio.
- El tipo ***EBCDIC*** es el mas eficiente cuando ambos el que recibe y el que envía lo usan como formato propio, este tipo se representa también en 8 bits pero de forma EBCDIC. Lo único en lo que cambian

es en la forma de reconocer los códigos de los caracteres.
- El formato de ***IMAGEN*** es cuando se compacta todo lo que se quiere enviar en cadenas seguidas de paquetes de 8 bits, es decir, no importa el formato en que internamente se maneje la información, cuando se va a enviar se tiene que hacer una conversión de 8 bits en 8 bits y cuando el que recibe tiene todo el paquete, el mismo debe codificarlos de nuevo para que la transmisión sea completada.

En la estructura de datos en FTP se consideran tres tipos diferentes de archivos:

- ***File*** − estructura donde no hay estructuras internas y el archivo es considerado una secuencia continua de bytes
- ***Record*** − estructura donde los archivos contienen puros registros igualitos en estructura
- ***Page*** − estructura donde los archivos contienen paginas enteras indexadas separadas.

Al establecer una conexión por FTP se debe tomar en cuenta que el mecanismo de transferencia consiste en colocar bien la transferencia de datos en los puertos adecuados y al concluir la conexión estos puertos deben ser cerrados adecuadamente. El tamaño de transferencia es de 8 bits, en ambos. El que va a transferir, debe escuchar desde el puerto hasta que el comando enviado sea recibido y este será el que de la *Dirección de la transferencia*. Una vez recibido el comando y establecido una transferencia del servidor a que solicita se inicializa la *Comunicación de la transferencia* para verificar la conexión, esta es una cabecera con un formato específico, después de esto se comienza a enviar las tramas de 8 bits sin importar el tipo de datos que sea (antes mencionado), y al finalizar se envía otra trama cabecera ya establecida confirmando la transferencia completada.

IPv6

Los días del IP en su forma actual (IPv4) están contados por lo que se comenzó a trabajar en una nueva versión IP, una que nunca se quedaría sin direcciones, resolvería varios otros problemas y sería más flexible y eficiente. Sus metas principales eran:

- Manejar miles de millones de *hosts*.
- Reducir el tamaño de las tablas de enrutamiento.
- Simplificar el protocolo, para permitir a los enrutadores el procesamiento más rápido de los paquetes.
- Proporcionar mayor seguridad (verificación de autenticidad y confidencialidad).
- Prestar mayor atención al tipo de servicio, especialmente con datos en tiempo real.
- Ayudar a la multitransmisión permitiendo la especificación de alcances.
- Posibilitar que un *host* sea móvil sin cambiar su dirección.
- Permitir que el protocolo viejo y nuevo coexistan por años.

El IPv6 cumple los objetivos bastante bien: mantiene las buenas características del IP, descarta y reduce las malas, y agrega nuevas donde se necesitan. En general, IPv6 no es compatible con IPv4, pero es compatible con todos los demás protocolos de Internet (TCP, UDP,ICMP, IGMP, OSPF, BGP Y DNS), a veces requiriendopequeñas modificaciones.

La cabecera principal de IPv6

<table>
<tr><td>Versión</td><td>Prioridad</td><td colspan="3">Etiqueta de flujo</td></tr>
<tr><td colspan="3">Longitud de carga útil</td><td>Siguiente cabecera</td><td>Límite de saltos</td></tr>
<tr><td colspan="5">Dirección de origen

(16 bytes)</td></tr>
<tr><td colspan="5">Dirección de destino</td></tr>
</table>

(16 bytes)

Cabecera fija del IPv6 (obligatoria)

32 bits

Prioridad ! se usa para distinguir entre paquetes a cuyos de orígenes se les puede controlar el flujo y aquellos a los que no. Los valores de 0−7 son para transmisiones capaces de reducir su velocidad en caso de congestionamiento. Los valores 8−15 son para tráfico de tiempo real cuya tasa de envío es constante, aún si todos los paquetes se están perdiendo. El audio y el vídeo caen en esta última categoría. Esta distinción permite a los enrutadores manejar mejor los paquetes en caso de congestionamientos. Dentro de cada grupo, los paquetes de número más bajo son menos importantes que los paquetes de número alto.

Etiqueta de flujo ! aún es experimental, pero se usará para permitir a un origen establecer una seudoconexión con propiedades y requisitos particulares. Los flujos son un intento de tener la flexibilidad de una subred de datagramas y las garantías de una subred de circuitos virtuales.

Longitud de carga útil ! indica cuántos bytes siguen a la cabecera de 40 bytes de la figura anterior.

Siguiente cabecera ! La razón por la que pudo simplificarse la cabecera es que puede haber cabeceras adicionales (opcionales) de extensión. Este campo indica cuál de las seis cabeceras de extensión sigue, de haberlas, sigue a ésta. Si esta cabecera es la última cabecera IP, el campo siguiente cabecera indica el manejador de protocolo de transporte (por ejemplo, TCP, UDP) al que se entregará el paquete.

Límite de saltos ! se usa para evitar que los paquetes vivan eternamente.

Dirección de origen y **dirección destino**! Son direcciones de 16 bytes de longitud fija.

Además de manejar direcciones unitransmisión estándar (punto a punto) y multitransmisión, el IPv6 también maneja un nuevo tipo de direccionamiento: transmisión a cualquiera. La transmisión a cualquiera (anycasting) es como la multitransmisión, en el sentido de que es destino es un grupo de direcciones, pero en lugar de tratar de entregar el paquete a todos, intenta entregarlo a uno solo, sin tener que saber quien es. La transmisión a cualquiera usa direcciones al más cercano, sin tener que saber quien es. Elegir al *host* receptor es responsabilidad del sistema de enrutamiento.

Se ha desarrollado una nueva notación para escribir direcciones de 16 bytes: se escriben como ocho grupos de cuatro dígitos hexadecimales, separados los grupos por dos puntos, como sigue:

8000:0000:0000:0000:0123:4567:89AB:CDEF

Ya que muchas direcciones tendrán muchos ceros en ellas, se han autorizado tres optimizaciones:

- Los ceros a la izquierda de un grupo pueden omitirse, por lo que 0123 puede escribirse como 123.
- Pueden reemplazarse uno o más grupos de 16 ceros por un par de signos de dos puntos. Por tanto la anterior dirección quedaría del siguiente modo:

8000::0123:4567:89AB:CDEF

- Por último, las direcciones IPv4 pueden escribirse como un par de signos de dos puntos y un número decimal anterior separado por puntos, como por ejemplo:

::192.31.20.46

Cabeceras de extensión

Estas cabeceras pueden usarse para proporcionar información extra, pero codificada de una manera eficiente. Todas son opcionales, pero si hay más de una , deben aparecer justo después de la cabecera fija, y de preferencia en el orden listado.

Cabecera de extensión	**Descripción**
Opciones salto por salto	Información diversa para los enrutadores
Enrutamiento	Ruta total o parcial a seguir
Fragmentación	Manejo de fragmentos de datagramas
Verificación de autenticidad	Comprobación de la identidad del transmisor
Carga útil cifrada de seguridad	Información sobre el contenido cifrado
Opciones de destino	Información adicional sobre el destino

Algunas cabeceras tienen un formato fijo; otras contienen un número variable de campos de longitud variable En éstos cada elemento está codificado como una tupla (tipo, longitud valor).

Tipo ! es un campo de 1 byte que indica la opción de la que se trata. Los valores de Tipo se han escogido de modo que los dos primeros bits indican a los enrutadores que no saben cómo procesar la opción lo que tienen que hacer. Las posibilidades son: saltar la opción, descartar el paquete, descartar el paquete y enviar de regreso un paquete ICMP, y lo mismo que lo anterior, pero no enviar paquetes ICMP a direcciones de multitransmisión (para evitar que un paquete de multitransmisión malo genere millones de informes ICMP).

Longitud ! es un campo de 1 byte e indica la longitud del valor (0 a 255 bytes).

Valor ! es cualquier información requerida, de hasta 255 bytes.

La cabecera de salto por salto se usa una informació que deben examinar todos los enrutadores a lo largo de la trayectoria. Hasta ahora se ha definido una opción: manejo de datagramas de más de 64K (jumbogramas). El formato es el siguiente:

Siguiente cabecera	0	194	0
Longitud de la carga útil gande			

Como todas las cabeceras de extensión, ésta comienza con 1byte que indica el tipo de cabecera que sigue. A este byte le sigue uno que indica la longitud de la cabecera salto por salto en bytes , excluyendo los primeros 8bytes, que son obligatorios. Los 2bytes siguientes indican que esta opción define el tamaño del datagrama (código 194) como número de 4bytes. Los últimos 4bytes indican el tamaño del datagrama (no se permiten tamaños menores que 65.536).

La cabecera de enrutamiento lista uno más enrutadores que deben visitarse en el camino al destino. Hay posibilidad tanto de enrutamiento estricto (se suministra la trayectoria completa) como libre (s´lo se proporcionan enrutadores seleccionados), pero están combinados. El formato es el siguiente:

Siguiente cabecera	0	Cantidad de direcciones	Siguiente dirección
	Mapa de bits		
Direcciones 1−24			

Lo primeros 4bytes de la cabecera de extensión de enrutamiento contienen 4 enteros de 1byte: el tipo de la siguiente cabecera, el tipo de enrutamiento (actualmente 0), la cantidad de direcciones presentes en esta cabecera (1 a 24) y el índice de la siguiente dirección a visitar. Este último campo comienza en 0 y aumenta con cada dirección que se visita.

Después viene el byte reservado seguido de un mapa de bits con bits para cada una de las 24 direcciones IPv6 potenciales que siguen. Estos bits si debe visitarse cada dirección directamente después de la anterior (enrutamiento estricto desde el origen) o si pueden intervenir otros enrutadores intermedios (enrutamiento libre desde el origen).

La cabecera de fragmentación maneja la fragmentación de una manera parecida a la del IPv4. La cabecera contiene el identificador del datagrama, el número de fragmento y un bit que indica si seguirán más fragmentos. Sólo el bit de origen puede fragmentar el paquete, los enrutadores a lo largo del camino no pueden hacerlo.

La cabecera de verificación de autenticidad proporciona un mecanismo mediante el cual el receptor de un paquete puede estar seguro de quién lo envió. La carga cifrada de seguridad posibilita el cifrado del contenido de un paquete de modo que sólo el receptor pretendido pueda leerlo. Estas cabeceras usan técnicas de cifrado para lograr su cometido.

8