UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CAMPUS ACATLÁN

INTERNET PROTOCOL VERSIÓN 4 Y 6

ASESOR: LIC. ALEJANDRO ROBERTO RUBIO PÉREZ

MARISOL GUADALUPE ÁGUILA PÉREZ

NO. CUENTA: 095087379

INTRODUCCIÓN

La expansión de los protocolos TCP/IP se debe a que se utilizan como el estándar para Internet, dado que su estructura permite el envío y recepción de paquetes de información con un gran manejo del tráfico de éstos, como se explica más adelante. Su estructura permite la libre interactividad entre diferentes sistemas de la familia de protocolos de Internet. El nombre de TCP/IP proviene de dos protocolos importantes, el TCP, Protocolo de Control de Transmisión (Transmission Control Protocol) y el IP, Protocolo de Internet (Internet Protocol).

TCP/IP no solo es utilizado en Internet, sino también por empresas que hacen su Internet privada. A este tipo de redes se les conoce con el nombre de Intranet.

La importancia de estos protocolos es que ayudaría a muchos programadores y computólogos a comprender, diseñar, implementar y depurar aplicaciones.

CAPÍTULO I

I.1 HISTORIA Y CONCEPTOS BÁSICOS

TCP/IP fue desarrollado por el informático estadounidense Vinton Cerf como parte de un proyecto dirigido por el ingeniero norteamericano Robert Kahn y patrocinado por DARPA la Agencia de Defensa Avanzada de Búsqueda de Proyectos (The Defense Advanced Research Projects Agency) entre los años 1972 y 1973, para interconectar varios departamentos de cómputo de defensa.

TCP nos permite intercomunicarnos entre computadoras de diferentes tipos (todos están incluidos), que manejan diferentes tipos de sistemas operativos en redes de área local (LAN, Local Area Network) y de área extensa (WAN, Wide Area Network). Cabe mencionar que actualmente la distancia geográfica no es un problema para comunicarse entre computadoras.

Internet es una *red de redes*. Una red de computadoras está formada por un conjunto de maquinas interconectadas. Esto permite intercambiar mensajes, archivos, datos, etc. En Internet no es solo varias computadoras sino que son miles de redes en ella. Algunas de estas redes pertenecen a instituciones públicas, privadas, hospitales, escuelas, etc., y actualmente existen un sinnúmero de compañías que existen dentro de la red de Internet y se encuentran localizadas alrededor del mundo.

TCP/IP proporcionan a los usuarios servicios de comunicación universales tales como:

- Transferencia de archivos: usado para poder enviar y recibir información entre dos o más computadoras.
- Conexión remota o terminal virtual: para conectarse desde una terminal diferente.
- Correo electrónico: uso de correo electrónico.
- Acceso a archivos distribuidos: se puede tener acceso todos los archivos dentro de Internet que sean de uso público y/o compartido.

- Manejo de ventanas: el manejo ya no es por ventanas de “MS-DOS” ahora es una interfaz grafica como en Windows, Unix y otros diferentes sistemas operativos.

Anteriormente eran menos los servicios y más pasos a seguir para poder tener conexión directa.

I.2 QUE ES UN PROTOCOLO

Hasta el momento hemos estado hablando de “protocolos” pero en realidad, ¿de qué estamos hablando?

Un protocolo se define como un conjunto de reglas, formatos de datos y convenciones que para poder tener comunicación entre dos o más entidades necesitan ser respetadas.

Establece que tipos de módems, computadoras y programas van a emplearse en una comunicación. Como IP, TCP, UDP, FTP, etc., son ejemplos de protocolos.

Podemos decir que un protocolo es una “conversación computacional”, es decir, una de las computadoras trata de enviar información hacia otra, pero primero se encarga de ver si la destinataria está “disponible” (es decir, conectada a la red); ya que se ha establecido una conexión procede a enviar la información, la destinataria la recibe y confirma que le llegó enviando un mensaje a la computadora remitente.

Suena algo complicado, pero es como una llamada telefónica: nosotros queremos establecer comunicación con alguien y nuestra forma de hacerlo es marcando el número del teléfono (dirección IP) cuando la bocina se levanta hemos establecido conexión (computadora conectada), después hablamos “hola, ¿está Juanita? Para establecer con quien queremos hablar (verificar la computadora), cuando nos comunican con Juanita empezamos nuestra conversación (entrega de paquetes) y cuando Juanita responde es porque si lo recibió (confirmación de llegada de paquete), y continuamos enviando información (paquetes) hasta que se termine el enlace (ambas bocinas cuelgan)

Ya relacionamos un poco la conexión y la información, pero seguimos sin saber que es un protocolo, tomando el mismo ejemplo, si Juanita habla chino y nosotros francés ninguno de los dos entendería lo que queremos comunicarnos, en cambio de Pedrito sabe ambos idiomas él es nuestro interprete, lo que Juanita me diga en chino, Pedrito me lo dirá en francés y viceversa. Entonces Pedro es nuestro intérprete y los protocolos serian los idiomas chino y francés.

Para que nuestro protocolo funcione mejor necesitamos de un intérprete para que pueda haber comunicación entre dos o más computadoras.

I.3 ESTRUCTURA DE TCP/IP

TCP/IP es una familia de protocolos desarrollados para permitir la comunicación entre cualquier par de computadoras de cualquier red o fabricante, respetando los protocolos de cada red individual; transmite datos ensamblados en paquetes o datagramas. Cada datagrama empieza con una cabecera que controla la información y a ésta le siguen los datos; se asegura que los datos sean entregados tal y como son enviados, es decir, lo que se recibió es lo mismo que se envió y los paquetes se ordenen en la forma en la que fueron enviados.

Mediante los protocolos se define la forma en que se van a comunicar las computadoras. Es como los idiomas: si varias personas de distinta nacionalidad hablan varios idiomas (inglés, alemán, japonés, etc.) tendrán que ponerse de acuerdo para hablar, por ejemplo, en español para darse a entender.

TCP/IP se encuentra generalmente en el sistema operativo y es compartido por todos los programas de aplicación, así que tiene que “organizar” los procesos que cada programa envía a TCP/IP.

Estos procesos son controlados por diferentes procesos o subprocesos de control, son varios e indispensables para el buen funcionamiento dentro de una computadora. Muchas veces llegan al mismo tiempo para que TCP/IP los coloque en el lugar que les corresponde, pero TCP/IP necesita saber cuáles son los más importantes para llevarse a cabo primero. Así que cada proceso tiene una prioridad; con esta prioridad los procesos se realzan de tal manera que no interrumpen los más importantes. Para que los procesos “entren” en orden se necesita un sincronizador en los procesos.

Empecemos con los *semáforos de conteo*, que son un mecanismo que permite sincronizar la ejecución de procesos independientes. Este semáforo cuando es creado tiene dos funciones “wait" que hace bloquear los procesos cuando llegan y “signal”

que hace que se empiece a ejecutar el proceso y ambos manipulan o controlan el conteo.

Cuando un proceso llega, el semáforo llama a la función “wait” para detener el proceso e identificar con “signal” si hay algún otro proceso ejecutándose, cuando “signal” le manda respuesta de “no-ejecución”, entonces ”wait” manda el proceso a “signal” para ejecutarse. Si llegan varios procesos a la vez entonces todos serán bloqueados por “wait” hasta que el espacio en “signal” esté vacio, así el semáforo controla el recibo de procesos a la vez. Cuando "signal" se desocupa entonces "wait" manda un solo proceso y los demás en "wait" estarán bloqueados, cuando termina "signal" entones "wait" manda el siguiente y así sucesivamente hasta que el contero del semáforo se encuentre en cero.

La sincronización de procesos es para el acceso a las colas que son de una capacidad limitada o finita. Es decir, los semáforos es el primer bloqueo de un proceso, después que el semáforo lo pasó, existen las colas que tienen un productor (aquel que inserta elementos en la cola) y un consumidor (aquel que extrae elementos en la cola)

Podemos pensar entonces que un puerto es una cola de eventos que serán procesados por TCP. Cuando TCP produce un datagrama, éste lo envía a IP. Para ser entregado, IP elige una interfaz de red a través de la cual debe enviarse al datagrama y lo transfiere al proceso de salida correspondiente.

Dado que IP nos permite manejar las aplicaciones libremente dentro de una red sin que el hardware utilizado sea un problema, es decir, puede utilizarse o ejecutarse en cualquier topología como por ejemplo Ethernet, Token Ring o X.25

I.4 FAMILIA DE PROTOCOLOS TCP/IP

Este protocolo es el que hoy en día constituye la base tecnológica de la mayor red de computadoras en el mundo: Internet.

En una red TCP/IP cada nodo (típicamente, en una computadora o enrutador) está unido a la red mediante al menos una interfaz de red. Cada una de esas interfaces está perfectamente identificada por una dirección IP. Puesto que el tema que nos preocupa son las aplicaciones distribuidas, que se ejecutan en computadoras (no en enrutadores) y la mayor parte de las computadoras están conectadas a la red por un única interfaz, podemos decir sin ser demasiado imprecisos que cada computadora tiene una dirección IP única.

Cada interfaz de una computadora está conectada a una subred con una cierta tecnología. Típicamente, nos encontramos accesos a redes Ethernet, Token Ring, líneas serie, Frame Relay, etc. Sin la intervención de los protocolos TCP/IP, cada computadora solo puede dialogar con los interlocutores que se encuentran en la misma subred, utilizando para ello direcciones propias de esa subred que no tienen ninguna relación con las direcciones IP.

Gracias al protocolo IP (Internet Protocol), y a la presencia en la red de dispositivos encaminadores, un conjunto de subredes puede constituir una internet, una red de redes. Cualquier par de nodos de esa red (se reserva el termino subred para cada uno de sus constituyentes) puede comunicarse mediante un intercambio de datagramas, elaborados de acuerdo al formato definido por el protocolo IP. Todos los datagramas IP van etiquetados con las direcciones IP del emisor y del receptor. Además llevan un campo denominado protocolo, cuya utilización veremos más adelante.

El modelo de comunicación ofrecido por IP, basado en datagramas, no siempre satisface las necesidades de las aplicaciones distribuidas. IP puede perder, estropear, desordenar o duplicar datagramas. Por muy bien que funcione, no ofrece las garantías de fiabilidad deseables por la mayoría de las aplicaciones. Por otra parte, IP está diseñado para llevar datagramas de una maquina a otra, pero no tiene mecanismos para determinar qué proceso dentro de una maquina es el destinatario del datagrama.

En un entorno como el de UNIX, donde en cada máquina se ejecutan concurrentemente múltiples procesos, esta es una limitación seria. Con el objeto de ofrecer algo más adecuado a las necesidades de las aplicaciones, por encima de IP se ha desarrollado el protocolo TCP que soluciona estos problemas.

CAPÍTULO II

II.1 IPv4 PROTOCOLO DE INTERNET VERSIÓN 4 (INTERNET PROTOCOL)

IPv4 es un protocolo no orientado a la conexión tanto en el destino como en el origen para comunicar datos en una red de paquetes transferidos.

Los datos basado en IPv4 se envían en bloques llamados datagramas o paquetes, algo especifico en IP, es que no necesita una configuración previa para que algún equipo trate de enviar paquetes a otro al que nunca se ha comunicado anteriormente.

IPv4 trabaja con un servicio llamado “el mejor esfuerzo” (*best effort*) que no asegura la entrega de datagramas. IP solo recibe un acuse de recibo de su cabecera y no de los datos que se transmiten, así que pueden llegar estos dañados, incompletos o no llegar a su destino, de estos errores se encarga el protocolo TCP.

Un datagrama involucra una parte de encabezado y otra de texto. El encabezado se transmite con el modo *big endian* y se lee de izquierda a derecha comenzando por el bit de orden mayor del campo versión, que en este caso es 4.

Una de las complicaciones que tiene IPv4 al enviar datagramas es que algunas veces son demasiado grandes y tiene que fraccionar ese datagrama para que llegue completo. Para esto recurre a dos procesos: Fragmentación y Reensamblado.

II.2 FRAGMENTACIÓN

Los datagramas los envía IPv4, pero algunas veces éstos suelen ser demasiado grandes de tamaño y el MTU (unidad de tamaño máximo de transferencia, Maxim transference unit) no les permite utilizar el camino elegido, así que se divide el datagrama.

La fragmentación consiste en dividir el datagrama en pequeñas partes, cada uno identificado con la cabecera original y “etiquetado” con un bit de fragmentación para que no se “pierda” en el camino.

Todos los fragmentos tiene este bit, a excepción del último, éste no lo tiene indicando así que es el último fragmento del datagrama.

Los fragmentos se hacen en tres partes: el primer fragmento empieza en el byte 0 y son 1480 bytes en el campo de datos, su bandera comienza en 1 indicando que hay mas fragmentos que le siguen. El segundo fragmento comienza en el byte 1480 en el campo de información su bandera también es 1. El tercer y último fragmento es de 1020 bytes y se inserta en el byte 2960, su bandera es 0 indicando que es el último fragmento.

II.3 REENSAMBLADO

El reensamblado consiste en reunir todos y cada uno de los fragmentos pertenecientes al mismo datagrama, identificándolos con el bit y la cabecera IPv4 que se grabó a cada fragmento. Como pueden no llegar en orden, se procede a ordenarlos. Al tener todos los fragmentos, se unen formando el datagrama original. El datagrama ha llegado completo a su destino.

Si alguno de los fragmentos no llega, entonces no se puede reensamblar el datagrama y éste se descarta y es eliminado del camino así como la ruta y red especificadas.

Los casos por los cuales no llega alguno de los fragmentos son:

- Diferentes puertas de enlace (diferentes caminos).
- El tiempo de vida se agotó eliminando al fragmento del camino.

Como se mencionó anteriormente, cuando un datagrama es demasiado grande, IPv4 tiene que dividirlo para poder enviarlo, a este método se le llama fragmentación.

IPv4 tiene que elegir el mejor camino para el envío. Si el datagrama rebasa el límite permitido por cada red y sus caminos, se tiene que fragmentar, ya que estos caminos están “restringidos” por el MTU.

Cuando el datagrama necesita ser fragmentado, IPV4 crea datagramas del mismo datagrama solo que los marca con un bit de fragmentación activo a cada uno de ellos, los fragmentos llevan información consecutiva del datagrama original.

Para identificar que sean fragmentos de un mismo datagrama, la cabecera de IPV4 activa el bit de “mas fragmentos” en cada uno de ellos (recuérdese que cada fragmento lleva una “copia” de la cabecera original), a excepción del último fragmento que lleva los últimos datos.

En casos especiales, los fragmentos pueden ser todavía muy grandes para poder enviarlos, así que se tienen que volver a fragmentar. Esto ocurre cuando el datagrama pasa por dos o más puertas de enlace. Si alguna de estas puertas fragmenta el datagrama original, entonces se tiene que fragmentar nuevamente con el bit de “más fragmentos”, es decir, activa en el fragmento este bit, como si fuera un datagrama original a todos los subfragmentos excepto el último.

Cuando un fragmento se subfragmenta y no es el fragmento final, la puerta de enlace activa el bit para todos los subfragmentos producidos dado que ninguno es el fragmento final del datagrama original.

Ya que el datagrama original (que sea muy grande) es fragmentado y cumple las especificaciones para poder ser enviado por las rutas a seguir, es decir, que no exceda el MTU de dadas rutas, es entonces cuando puede llegar a su destino y lo último que se espera es cuando llega el datagrama a su destino y se tiene que reensamblar cada uno de los fragmentos del datagrama y ordenarlos ya sea como van llegando si tienen un orden, o se tienen que ordenar cuando no han sido enviados en un orden o enviados por un mismo canal de enlace.

A este proceso se le llama reensamblado. Esencialmente reúne los fragmentos enviados ya que están en la dirección de envío y los empieza a ordenar y juntar hasta tener el datagrama completo, teniendo en cuenta la cabecera de cada uno de los fragmentos, para reconocer si pertenecen al mismo datagrama.

Dado que los fragmentos pueden llegar por caminos diferentes se tiene que verificar la cabecera para comprobar que pertenezcan al mismo datagrama, ya establecidos todos los fragmentos se procede al reensamblado.

Algunos casos en los que no se puede realizar el reensamblado es cuando no llegan todos los fragmentos, es decir, pueden no llegar el último fragmento o alguno de los intermedios, entonces el datagrama es descartado. Otra situación es cuando el TTL del datagrama llega a cero cuando todavía no llegan a todos los fragmentos, cuando esto sucede el datagrama también es descartado así como la red especificada y la ruta.

Dado que en una misma red se pueden enviar varios datagramas fragmentados, los fragmentos se pueden mezclar con otros de diferentes datagramas, es entonces cuando se tienen que implementar los datos para identificar los fragmentos que pertenecen a un mismo datagrama y poder reensamblarlo. Se realiza por medio de una lista que va almacenando los fragmentos que sean de un único datagrama.

Es análogamente como los envíos de cartas, llegan a un mismo lugar, se seleccionan por dirección de destino, ya seleccionadas se busca el número de la calle y se entrega la carta.

Solo que en este caso nos preocupa el hecho de que la confirmación de la entrega del datagrama no es fiel porque IPV4 solo confirma la llegada de su cabecera no de los datos.

Entonces es cuando se tiene que implementar los datos haciendo un arreglo en donde se enlistan o enfilan, todos los fragmentos de un mismo datagrama, aunque estén entre mezclados con otro datagrama o sus fragmentos. Para esto debe de permitir una ubicación rápida de los fragmentos que sean de un solo datagrama, la rápida inserción de un fragmento que pertenezca al datagrama, la comprobación de que un datagrama llegó completo, el tiempo de espera de los fragmentos y la eliminación de estos si el tiempo termina antes de que se realice y termine el reensamble.

Ya realizados todos estos aspectos y verificados se procede a verificar que un datagrama esté completo.

Se verifica en el arreglo todas las listas de cada uno de los datagramas enviados, en cada lista se verifica que todos los fragmentos contengan la misma cabecera IPV4, comprobar que contengan todos los fragmentos (que todos hayan llegado), si se encuentran todos se realiza el reensamblado.

El primer fragmento contiene en la cabecera el tamaño total del datagrama. Dado que cada fragmento contiene datos y pueden estar o no en orden, se realiza un bucle que esta rectificando a todos los fragmentos para organizar el orden en el que está organizado del datagrama original, hasta que el datagrama esta ordenado y completo, aquí el reensamblado concluye su función.

II.4 DIRECCIONAMIENTO DE IPv4

En el capitulo anterior se mencionó el enrutamiento y las diferentes redes que existen para el envío de datagramas y las direcciones de red, en este capítulo se describirán estos diferentes tipos de redes.

Existen cinco clases de redes, aunque solamente son utilizados tres de ellos, porque son las redes más comunes que se utilizan.

Se dividen en clases: A, B, C, D y E. cada una de ellas se identifica por las características especificas que tienen.

Tipo de Clases

Clase A: se tratan de pocas rede con un número mayor de computadoras que las demás redes.

Clase B: es un número mediano de redes con un número medio de computadoras.

Clase C: es un número mayor de redes pero con pocas computadoras en cada una de ellas.

Es necesario definir los números de bits, cada uno de diferente para las distintas clases. Todos los bits de estas direcciones son organizados en cuatro octetos, que son los que identifican la dirección IPV4, la identificación de red y la identificación de host (netid-hostid).

Expliquemos la distribución de estos octetos para cada una de las redes:

Para las direcciones de la clase A todas comienzan con un bit 0, el primer octeto define el identificador de red y los octetos restantes (tres) son para el identificador del host.

En las direcciones de clase B se comienzan con los bits 10. Dos octetos son para el netid y dos para el hostid.

En la clase C las direcciones comienzan con bits 110, tres primeros octetos para el netid y el último para el hostid.

Las direcciones de clase D comienzan con bits 1110, estas direcciones son utilizadas para multidifusiones.

En las direcciones de clase E comienzan con los bits 11110 aunque su uso todavía es experimental.

La siguiente grafica describirá las clases y su distribución de octetos.

Clase	bits		
A	0	Netid	Hostid
B	10	Netid	Hostid
C	110	Netid	Hostid
D	1110	Netid	Hostid
E	11110	Reservado	

II.4.1 Tabla de Clases de IPv4 y la distribución de los octetos.

Todas estas direcciones y sus clases surgieron de la necesidad de cubrir a todos los usuarios de la red. Se amplió las clases D y E porque las direcciones de la clase A se agotaron hace ya mucho tiempo; las direcciones disponibles de B son reservadas para pesos “muy pesados” de la industria; la clase C tiene muy pocas direcciones disponibles porque los iniciadores de IPV4 no creyeron que Internet tuviera tanto auge a un tiempo futuro. Aquí surge IP de nueva generación (IPng).

Para definir las direcciones que tienen disponibles cada una de estas clases se tiene que revisar primero una serie de restricciones que tienen éstas, algunas por ser de uso especial y otra porque no identifican ni al host ni a la red.

Las direcciones IPV4 esencialmente es en código binario (ceros y unos) y las combinaciones de éstos, pero a los humanos se nos haría muy difícil recordarlos, así que para representar a cada octeto se decidió representarlos en forma de número decimal.

Recordemos que son cuatro octetos de bits, y para ordenarlos se necesita ceros y unos, por ejemplo:

La dirección del centro de cómputo del área de Linux en Acatlan es: 192.168.4.254, este número en código binario es:

11000000 10101000 00000100 11111110

192 168 4 254

¡Ciertamente un número difícil de recordar!

Tomando el ejemplo de la dirección del área de Linux, demostraremos las restricciones de las direcciones IPV4.

Los netid y hostid con valor de 0 (00000000, en binario), no se permiten dado que identifican la red, en el ejemplo seria:

192.168.0.0

Identifica la red 192.168 y la dirección 0.0.0.2 identifica el host 2 de la red local.

El netid 127 (01111111) es de uso especial porque es una dirección de retorno que se usa para verificar la configuración de la red. Los mensajes que se dirigen al netid 127 se devuelven en vez de enviarse a la red.

Los netid con valor 255 (11111111) son restringidos para las difusiones. Los mensajes dirigidos a la dirección 255.255.255.255 se envían a todos los hosts de la red. Un mensaje dirigido a la dirección 192.168.255.255 se envía a todos los hosts de la red 192.168.

Otra restricción es que el último octeto en una dirección IPV4 no puede ser 0 ni 255.

Para expresar las anteriores reglas en la forma de los dos campos Netid y Hostid existen casos especiales mencionados a continuación:

(Netid – hostid) nos identifica un host específico dentro de la red y se puede utilizar como dirección destino o de origen, tomando el ejemplo es: 192.168.4.123

(Netid 0 – hostid 0) esta dirección identifica la red y el host de un mensaje de origen, es decir, este host de esta red es el mismo 0.0.0.0

(Netid 0 – hostid) identifica a un host especifico de esta red y solo corresponde al host de origen del mensaje 0.0.0.0

(Netid – hostid 0) cuando se origina un mensaje esta restricción identifica al host del mensaje y el número de red de éste. Solo corresponde al host origen de un mensaje 192.168.0.0

(Netid 1 – hostid 1) este tipo de direcciones pertenecen a difusión local. Los mensajes enviados llegan a todos los hosts de la red local y no a otras redes. Es utilizado para identificar el host destino de un mensaje, ejemplo: 255.255.255.255

(Netid – hostid 1) corresponde a una dirección de difusión de la red específica. Los mensajes se encaminan hacia la red adecuada de destino. Es utilizada para identificar la red destino de un mensaje de difusión, ejemplo: 192.168.255.255

Es importante destacar que estas restricciones no contemplan a las subredes, aunque el identificado de subred funciona como una extensión de la dirección de red, así que el concepto de un identificador de red netid 0 ó netid 1 se puede ampliar para combinaciones de red-subred ya sea todos 0’s o todos 1’s.

Todas las clases de redes tienen un número de direcciones disponibles, considerando las restricciones anteriores.

Todas las direcciones están representadas en la siguiente tabla:

Clase	desde	Hasta	Redes	Hosts
A	1	126	126	16,777,214
B	128	191	4095	65,534
C	192	223	2,097,152	254

II.4.2 Tabla de direcciones disponibles en las clases de IPv4.

Se reservaron tres rangos de direcciones IPV4 que no se pueden utilizar en Internet, dado que no son enrutables y los routers de Internet no las envía.

Clase A: 10.0.0.0 a la dirección 10.255.255.255

Clase B: 172.16.0.0 a la dirección 172.16.255.255

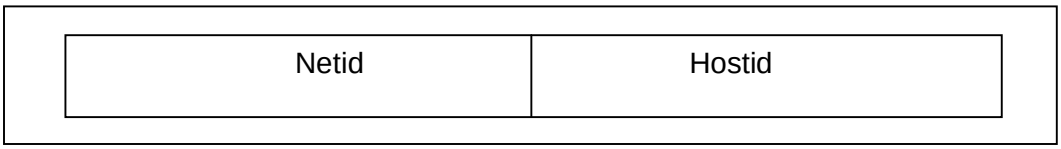
Clase C: 192.168.0.0 a la dirección 192.168.255.255

Clase D: 224.0.0.0 a la dirección 239.255.255.255

II.4.1 DIRECCIONES IPV4

Estas direcciones se encargan de que los datos entre computadoras se realicen con eficacia, como el ejemplo de la llamada telefónica, se tiene un número conocido y se establece comunicación con ella.

IPv4 utiliza las direcciones IP para poder identificar a cada host (nodo en la red, que puede ser una computadora o enrutador o encaminador), de tal manera que cada nodo tenga asignada una dirección única, inequívoca y precisa en Internet. Cada dirección IP está conformada de dos partes: Netid y Hostid, como ya se mencionó las características de éstos.



II.4.1.1 Formato de la dirección IP

ESTRUCTURA DE LA DIRECCIÓN INTERNET

Las direcciones IPv4 tienen designado un número entero de 32 bits, que están formadas por cuatro campos de 8 bits separados por comas. Cada campo está formado por valores comprendidos entre 0 y 255.

Cada dirección está conformada por un par (netid, hostid) donde el netid identifica la red y el hostid identifica al anfitrión dentro de la red. Las direcciones de red son asignadas dependiendo del tipo de la clase de la red A, B o C, que se diferencian entre sí por el número de computadoras que tiene cada red.

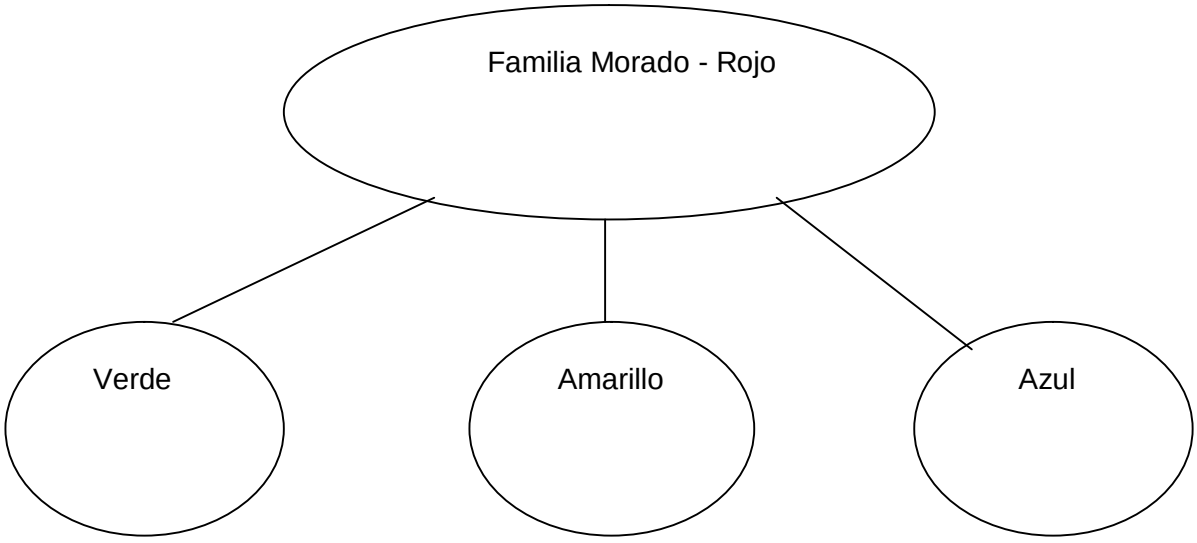
Para transmitir un datagrama a una dirección IP destino, la dirección IP tendrá que convertirse en una dirección de la red física. Esta conversión en ocasiones es bastante simple y solo necesita aplicar un algoritmo a esta dirección, pero a veces estas conversiones necesitan “ayuda” adicional para poder localizar la dirección física del nodo destino.

II.4.2 NÚMEROS DE ANFITRIONA Y NOMBRE DE DOMINIO

Se necesita un número de anfitriona (aquella computadora que va a conectarse a la red) para poder accesar. Un número de anfitriona podría ser 192.254.4.138, con este número, la computadora queda registrada como un host en Internet. El número de anfitriona sirve para la comunicación entre varias computadoras dentro de la red.

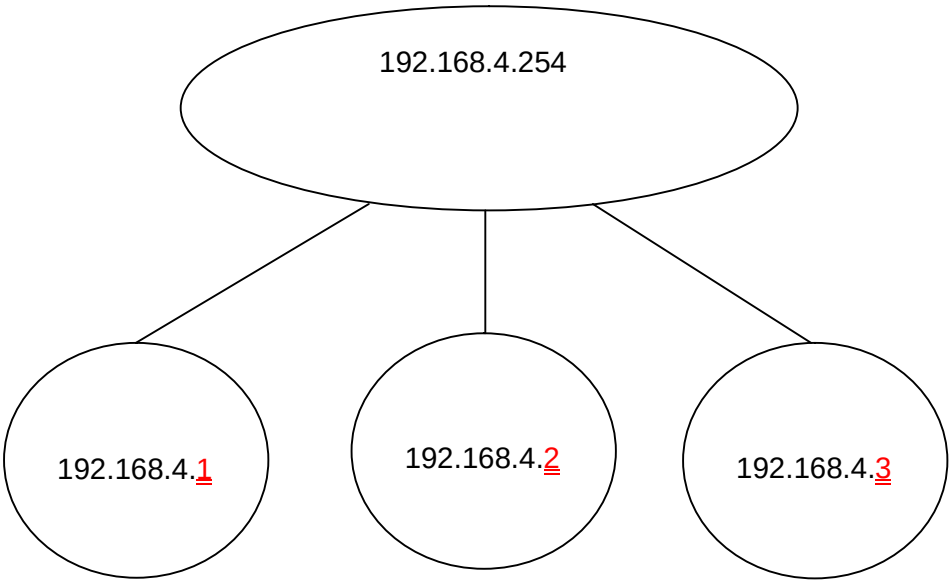
Todas las computadoras que se conectan directamente a Internet son denominadas anfitrionas y a cada una de ellas se les asigna un número específico. Este número es la dirección que Internet utiliza para poder enviar mensajes a los destinatarios. Dado que Internet es una red de redes, los números sirven para la identificación de la red en la que se encuentra la anfitriona. Ejemplo, la dirección del centro de cómputo de la FES Acatlan: 192.168.4.254

El dominio es aquel que también se le designa a esa máquina, es el identificador de máquina. Ejemplo análogo, padres e hijos, la familia Morado Rojo que está formada por papá Morado y mamá Rojo y para distinguir a sus hijos le ponen los nombres Verde, Amarillo y Azul, su grafica seria así:



II.4.2.1 Gráfica de ejemplo nombre de dominio

Ahora dese el punto de vista “computacional” cada uno de los integrantes va a tener un número propio de identificación, la “familia” a la que van a pertenecer será el dominio 192.168.4.254 y las computadoras tendrán los números 192.168.4.1, 192.168.4.2 y 192.168.4.3, estas serian las computadoras anfitrionas, la grafica sería así:



II.4.2.2 Gráfica de ejemplo nombre de dominio de una red

El dominio nos dirá a qué tipo de dominio pertenece, por ejemplo en la Web las terminaciones de “.com”, “.gob”, etc. Nos indican que son dominios de “compañías” o de “gobiernos”. Aquí unos ejemplos de los más usados:

.edu instituciones educativas, universidades y escuelas forman parte de este dominio
.com dominios comerciales como compañías
.gob dominios gubernamentales
.mx dominios de México

Existen muchos más tipos. Los números de anfitriona son los que realmente importan, dado que el dominio no especifica si una computadora participa o no en un procedimiento cualquiera.

II.5 MASCARAS DE RED

Supóngase que la maquina A con dirección IP 192.168.1.1 desea mandar un paquete a la maquina B que tiene como dirección IP 192.168.3.1, el envío no puede hacerse directamente porque no pertenece a la misma red; esto se hace con una resta de las direcciones

192.168.1.1 - 192.168.3.3 = 0.0.2.0

Con esta diferencia de “2” comprueba que no está dentro de la misma red. Entonces se hace uso de un router, que identifica por medio de una tabla las redes a las que pertenecen dichas maquinas, pero ¿Cómo se identifican a las maquinas que existen dentro de una red y a la red misma? Esto se logra con las mascaras de red

Las mascaras de red agrupan todas las direcciones de una red; todas las maquinas entonces ya agrupadas por la máscara “pertenecen a esta red”; es útil esto cuando se está buscando una maquina en particular, porque, no es necesario buscar dirección IP por dirección IP, si no que identificando la máscara ya se sabe dentro de que red está y su localización es mucho más sencilla; ahorra tiempo y esfuerzo, facilitando el envío “optimo” de los datagramas, en pocas palabras indica cuantas direcciones IP tendrá la red.

Para cada clase de dirección hay una máscara de red:

Clase	Máscara de red	Broadcast
A	255.0.0.0	127.255.255.255
B	255.255.0.0	191.255.255.255
C	255.255.255.0	223.255.255.255
D	Multidifusión	239.255.255.255
E	reservada	255.255.255.255

II.5.1 Clases de direcciones y sus máscaras de red

Cuando las direcciones ya se agruparon, dos bits se quedan reservadas, el primero y el último, que se utilizan para el identificador de host y otro para el identificador de red.

Ejemplo, si tenemos una red y hay 3 empresas que quieren una misma red, pero cada quien necesita su propia subred entonces las mascaras de red se utilizan para “encapsular” todas las direcciones IP de cada empresa, la repartición usando mascaras de red quedaría de la siguiente manera:

Subred	Maquinas	Máscara asignada
1	30	255.255.255.224
2	20	255.255.255.224
3	10	255.255.255.240

II.5.2 Ejemplo de una red con diferente número de maquinas y la máscara asignada a cada red.

II.6 LAS DIRECCIONES DE IPV4 Y SU DESPERDICIO

IPv4 utiliza direcciones de 32 bits, esto la limita a 4,294’967,296 direcciones únicas (2^{32}), muchas de las cuales están dedicadas a redes locales (LAN’s). Por el crecimiento enorme que ha tenido Internet estas direcciones ya casi se terminan, por lo tanto, se pensó en mejorar el protocolo IP para incrementar el número de direcciones. Aquí es donde existe IP de nueva generación, IPng.

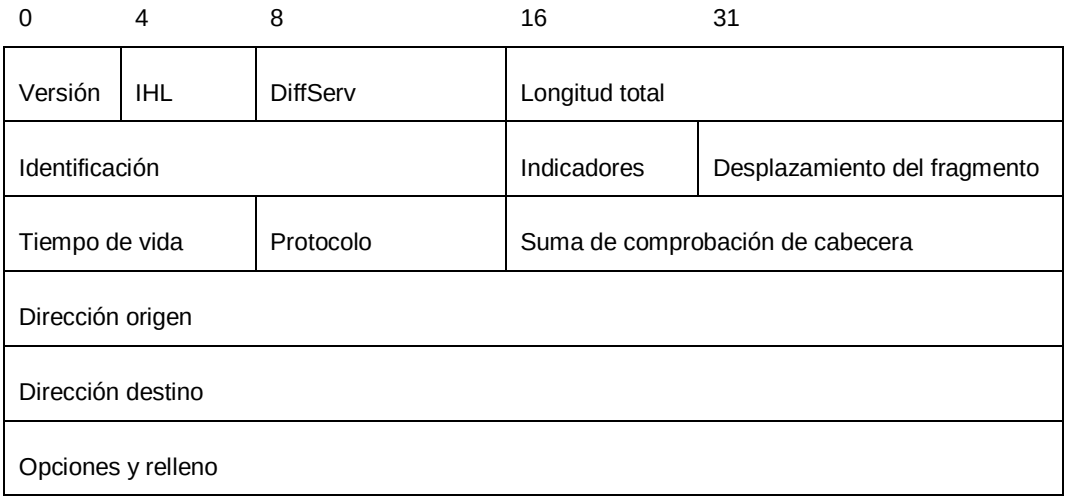
En primera instancia así se llamaría, IP de nueva generación (IPng), pero estuvieron probando diferentes versiones, la versión 5 de IP solo fue experimental, pero ya estaba asignada, entonces la que continuaba era la versión 6.

Con la estructura original de las clases de redes A, B, C, D, que permitían un número diferente de computadoras y números de IP, los problemas de asignación de direcciones comenzó a agrandarse.

Dando demasiadas IP reservadas (privadas) no se permitió que esas direcciones se utilizaran y muchas no fueron utilizadas, dando un desperdicio y una necesidad más grande por mayor cantidad de direcciones disponibles.

II.7 CABECERA IPv4

A continuación se muestra el modo en que está constituida la cabecera de IP y se describen cada uno de sus campos.



II.7.1 Cabecera IP versión 4

Descripción de cada uno de los campos:

Versión (Version): utiliza 4 bits. Define que tipo de versión IP se utiliza (4 o 6), en este caso es 4

Longitud de cabecera IHL (Internet Header Lenght): utiliza 4 bits. Establece la longitud de la cabecera en 32 bits. Su valor mínimo es de 5, que corresponde a una longitud mínima de 20 octetos.

Diferentes Servicios (Differentiated Services): Se le llamaba anteriormente ToS (Type of Service, utiliza 8 bits. Nos ayuda a la especificación de los parámetros de fiabilidad, prioridad, retardo y rendimiento. Es muy raro que se utilice. Utiliza 6 bits para el campo de servicios diferenciados, los dos bits restantes se reservan para un campo de notificación explicita de congestión que señala explícitamente la congestión de una manera similar a la discutida para retransmisión de tramas.

Longitud total (Total Length): utilize 16 bits. Se mide en octetos la longitud total del datagrama.

Identificación (IDENTIFICATION): utiliza 16 bits. Es un número de secuencia que trabaja junto con la dirección origen y destino y el protocolo de usuario (UDP) que identifica a un datagrama de forma única. Este identificador es único para la dirección origen y destino del datagrama así como para UDP durante el tiempo en el que el datagrama está en la red.

Indicadores (flags): utiliza 3 bits. Se encuentra definidos únicamente 2 bits. Un bit el de "más datos" (MF, more fragments) es utilizado para la fragmentación y reensamblado. El bit de no fragmentación (DF, don't fragment) evita ésta cuando su valor es 1, es útil este bit cuando el destino no puede reensamblar los fragmentos, sin embargo, si su valor es 1, el datagrama es descartado si el MTU (Maximun Transmission Unit, unidad máxima de transmisión) es muy pequeño en la ruta de la red. Así es que cuando el valor es 1 es preferible utilizar un enrutamiento del origen para evitar estas redes con tamaño de paquete máximos pequeños.

Desplazamiento del fragmento (Fragment Offset): utiliza 13 bits. Especifica el lugar donde se encuentra un fragmento dentro del datagrama original, está medido en unidades de 64 bits, esto es, que todos los fragmentos, excepto el último, tienen un campo de datos con una longitud con múltiplos de 64 bits.

Tiempo de vida (TTL Time To Live): utiliza 8 bits. El datagrama tiene un tiempo permitido para permanecer en la red, este campo lo hace y se mide en segundos. Algunas veces se entiende como "saltos" del datagrama en la red, cuando un datagrama se envía y se encuentra en la red el tiempo que tiene va decreciendo, hasta llegar a un valor 0, cuando esto sucede el datagrama en la red es descartado, si llegó a su destino también el contador llega a cero.

Protocolo (Protocol): utiliza 8 bits. Identifica el protocolo de la capa superior que recibirá el campo de datos en el destino, así es que este capo identifica que cabecera siguiente recibirá el paquete después de la cabecera de IP.

Suma de comprobación (Checksum): utiliza 16 bits. Este campo es un código para detectar los errores que se aplica solo a la cabecera. Algunos campos cambian a lo largo del envío como pueden ser el tiempo de vida y los de segmentación, estos valores son verificados y calculados en cada dispositivo de enrutamiento. Su valor se inicia con todo cero.

Dirección origen (Source Address): utiliza 32 bits. Indica el número de red y host origen especificando el número de bits.

Dirección destino (Destination Address): utiliza 32 bits. Especifica con un número de bits la red y el host destino.

Opciones (options): utiliza un número variado de bits. Da las opciones que envía el usuario que las solicitó.

Relleno: utiliza un número variable de bits. Usado para asegurar que la cabecera del datagrama tiene una longitud múltiplo de 32 bits.

Datos: utiliza un número variable de bits. Longitud múltiplo de 8 bits. La longitud máxima de un datagrama es de 65,535 octetos.

II.8 TCP

El protocolo de control de transmisión (Transmission Control Protocol) controla el envío y recepción de información, cuida que realmente llegue la información completa y necesita de un acuse de recibo, es decir, espera un mensaje de la fuente destino asegurando que la información recibida llegó completa y sin ningún error, se esto ocurre la computadora receptora manda un mensaje a la emisora para que se envíe ésta información a las aplicaciones que correspondan.

El protocolo TCP radica en el sistema operativo. Realiza el envío, recepción y verificación de datagramas utilizando subprocesos que facilitan el tráfico.

Cada subproceso tiene una función diferente: bloquean, temporizan, envían, colocan y reciben datagramas, ya sea de forma independiente o paralela.

Está orientado a la conexión, es un método muy seguro y eficiente de mover el tráfico de la red en el servicio cliente/servidor.

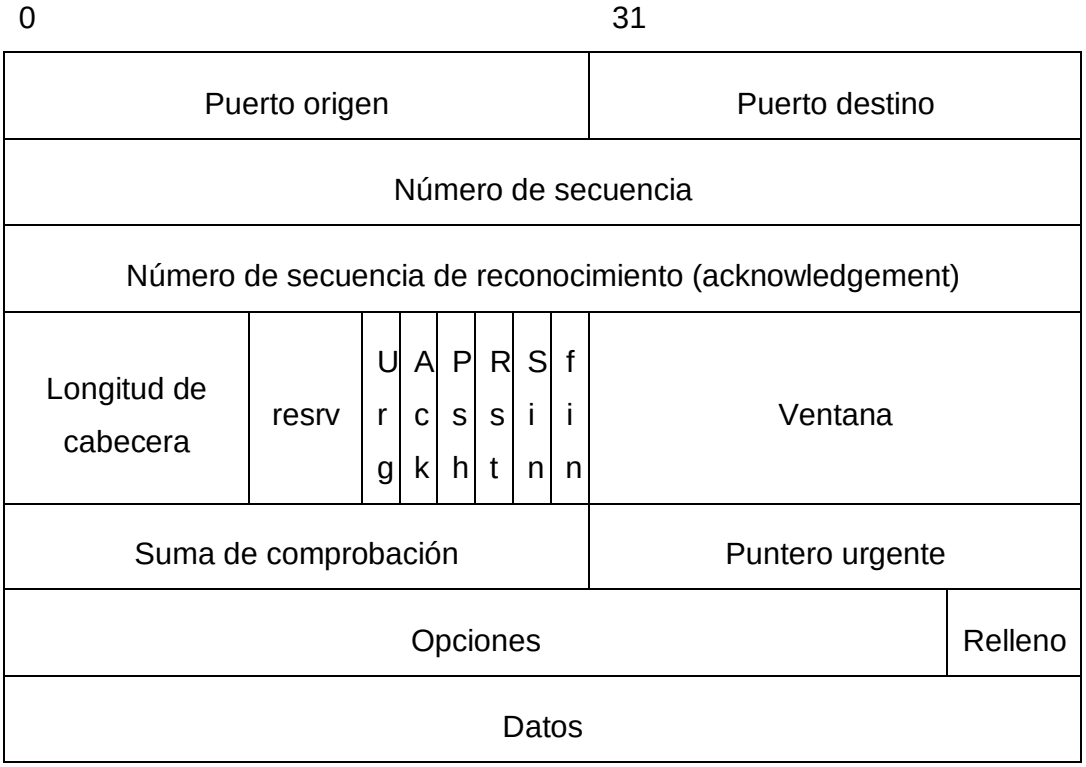
Es muy afable la información de un lado a otro (de puerto origen a puerto destino). TCP se asegura de que el tamaño de los segmentos sea óptimo para el canal de envío, que no rebase el límite permitido, y la velocidad indicada para ese canal.

Ofrece un mecanismo de comunicación basado en la idea de un canal fiable y bidireccional para el intercambio de octetos (no datagramas); no pierde datos, ni los desordena, resuelve problemas de control de flujo de los que también adolece IP; es capaz de atender a varios usuarios (procesos) a la vez, manteniendo múltiples conexiones simultaneas utilizando puertos. Una conexión viene identificada por dos direcciones IP y dos números de puerto. Antes de utilizar TCP, un proceso tiene que asociarse a uno de sus puertos. Tras ello ya puede solicitar la apertura de conexiones.

El protocolo TCP está basado en un intercambio de *segmentos*, con un formato característico que incluye, entre otra información, una serie de bytes con datos de usuario, el puerto de origen y el puerto de destino. El intercambio de segmentos entre dos entidades TCP se realiza gracias a IP; los segmentos van encapsulados dentro de datagramas IP, etiquetados con un valor en el campo protocolo antes mencionado que indica precisamente que el contenido del datagrama es un segmento TCP.

II.8.1 CABECERA TCP

El segmento de la cabecera TCP consta de 20 bytes aunque ésta puede ser más grande si se agregan las opciones adicionales. Se muestra la cabecera en la siguiente tabla:



II.8.1.1 Cabecera TCP

Puerto origen (Source Port Address): número de puerto de donde se manda la información.

Puerto destino (Source Destiny Address): número de puerto donde se recibe la información.

Número de secuencia (sequence number): los bytes se enumeran de forma aleatoria para el segmento TCP; los segmentos siguientes enviados por los hosts llevan el

número de secuencia que se les asignó al principio más el número de bytes que son enviados hasta el momento.

Número de secuencia de reconocimiento (acknowledgement number): valida los diferentes segmentos que va recibiendo, lo hace colocando el número de campo anterior (número de secuencia) y le suma 1, este byte es el que espera que reciba en el siguiente envío.

Longitud de cabecera (Header Lenght): nos indica el tamaño de la cabecera, ésta es medida en múltiplos de 32 bits. Es necesario este campo porque la cabecera puede variar de tamaño en el caso que se utilicen las opciones. Este campo es de 4 bits.

Reservado (reserved): su valor debe de ser cero. Está reservado para un uso futuro.

Banderas (flags): estos campos indican el propósito y contenidos del segmento.

Urgente: indica que se envían datos urgentes, trabaja con el campo puntero urgente (urgent pointer)

Acuse de recibo (Acknowledgement): como los datagramas no llegan en orden, solo se confirma los que llegan, este campo tiene un número que viaja en el campo acknowledgement number y tiene que ser válido, así este campo lo indica.

Push: los datos que transporta TCP tienen que ser enviados a una aplicación rápidamente y luego realizar una operación PUSH esa operación manda de inmediato los datos que se recibieron al nivel de aplicación.

Reset: la conexión debe ser reiniciada.

Sincronización: nos da el proceso de sincronización durante la conexión.

Fin: este campo es activado cuando el host quiere terminar la conexión. La conexión finaliza.

Ventana (window size): un host dispone el número de bytes que pretende recibir sin tener que validar esta información y este campo de ventana determina este número,

además el tamaño de este campo puede variar durante la conexión. Este campo es de 16 bits.

Suma de comprobación (Checksum): cuando la cabecera y/o el texto está incompleto, es decir, no cubre un octeto o le falta números para que este el octeto completo este es rellenado con bits (de valor cero) a la derecha. Se completa con los bits necesarios para cubrir los 16 bits. Este campo se asume que su valor es cero para verificar la comprobación.

Puntero urgente (Urgent pointer): es utilizado en conjunto con la bandera URG. Este campo nos indica el número de secuencia del último byte considerado como parte de los datos urgentes. Estos datos son necesarios para diferentes situaciones como el aborto de una conexión telnet o algún proceso que trabaja incorrectamente en la maquina remota. TCP especifica al emisor que estos datos a su llegada se deben notificar inmediatamente, no importando su posición dentro del segmento. El nodo (host) al reconocer los datos urgentes, TCP notifica al programa de aplicación que estos datos existen.

Opciones: se encarga de realizar y verificar varias funciones como por ejemplo checar el tamaño máximo de segmento, escalado de ventana, datos opcionales, etc.

Relleno: tiene octetos con valor cero que se pueden añadir a la cabecera y así redondear la longitud a 32 bits.

Datos: es el datagrama que se envía y espera recibirse.

II.8.2 UDP

Para aquellos casos en los que no se necesite la fiabilidad aportada por TCP (o no se quiera pagar el precio de la sobrecarga de proceso que supone), se puede utilizar el protocolo UDP (User Datagram Protocol).

Usar UDP es parecido a utilizar el protocolo IP directamente en el sentido de que se “sufren” todas sus deficiencias. Sin embargo, UDP incorpora, como TCP, el concepto de puerto para determinar, al recibir un datagrama, a que proceso va destinado. Los datagramas UDP van encapsulados dentro de datagramas IP etiquetados con el valor correspondiente en el campo protocolo.

Para UDP Protocolo de Datagramas de Usuario (User Datagram Protocol) la estructura de proceso es diferente y más sencilla que la usada para TCP.

Los mensajes de UDP no tienen control

II.8.2.1 TRANSFERENCIAS DE DATAGRAMAS ENTRANTES A UDP.

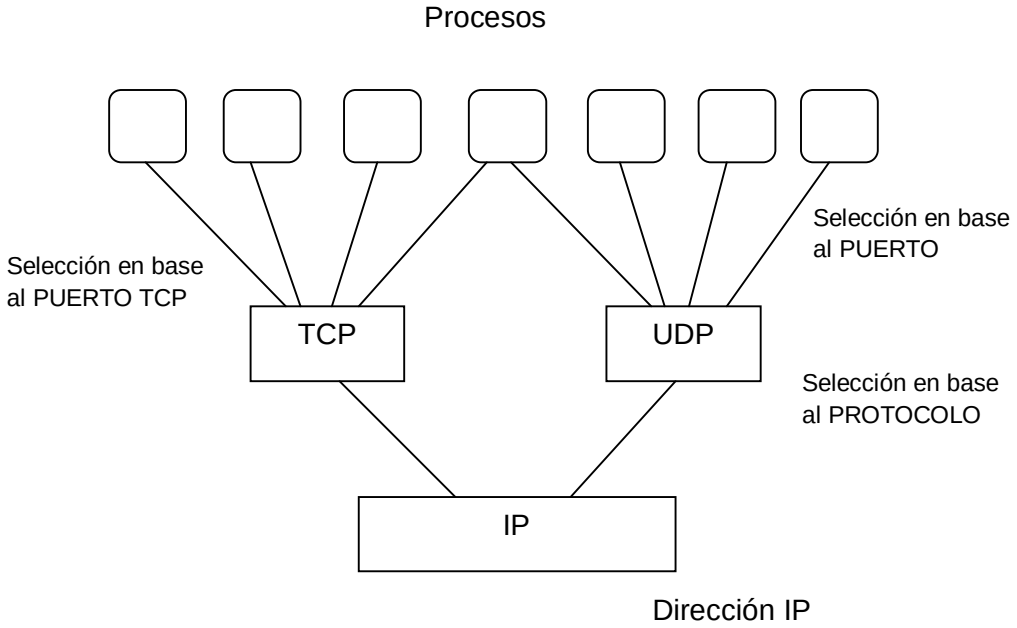
UDP no maneja procesos independientes, en vez de eso usa procedimientos que ejecutan el proceso IP. Para manejar un datagrama UDP entrante, estos procedimientos examinan el número de puerto destino de UDP y lo usa para seleccionar un puerto (cola) del sistema operativo para los datagramas entrantes. Entonces IP coloca el datagrama UDP en el puerto correspondiente de donde un consumidor lo extrae.

II.8.2.2 SALIDA UDP

La salida y el tráfico son más sencillos, dado que UDP no garantiza que un paquete sea entregado, cuando es enviado, la maquina que lo envió no conserva ninguna copia del datagrama y no necesita programar retransmisiones. Así que cuando se crea el datagrama se puede transmitir y el emisor puede descartar esta copia.

Todos los procesos que envían datagramas UDP deben ejecutar procesos UDP muy útiles para poder darles un formato, así como los procedimientos para encapsularlos, y regresar el datagrama IP resultante al proceso IP.

La relación entre IP, TCP, UDP y los procesos de los usuarios se refleja en la siguiente figura. Obsérvese que un proceso puede elegir comunicarse con otro a través de un puerto TCP, a través de un puerto UDP o incluso utilizar varios puertos (tanto TCP como UDP) simultáneamente. Si un proceso utiliza a la vez TCP y UDP, no tiene porqué hacerlo a través del mismo número de puerto.



II.8.2.2.1 Gráfica del Proceso de envío de datos desde IP

Es importante destacar que los puertos TCP y UDP son disjuntos. No hay peligro de que los datos enviados a un puerto TCP lleguen en un datagrama UDP, ni viceversa. Esto es posible gracias a que los datagramas que llevan esos datos están etiquetados con distinto protocolo.

Los procesos finales utilizan una serie de operaciones para comunicarse mediante TCP o UDP. La definición de estos protocolos incluye la descripción de una interfaz abstracta bastante sencilla. De cómo se implementa esta interfaz mediante una colección de funciones basadas en el concepto de socket, o de TSAP si se emplea la interfaz TLI se hablara más adelante.

Si un proceso quiere comunicarse con otro utilizando UDP necesita conocer la dirección IP y el número de puerto UDP de su interlocutor. Puede resultar conveniente, aunque no imprescindible, conocer su propia dirección IP y su puerto UDP. Dispondrá de una función de envío de datagramas con al menos tres parámetros: los datos a enviar, la dirección IP del receptor y el puerto del receptor. También dispondrá de una función para recibir datagramas que, además de facilitar los datos que se reciban, informara sobre la identidad (dirección IP, puerto) del emisor.

La interacción con TCP es algo más compleja, dado que el servicio ofrecido está orientado a conexión: la comunicación debe pasar por una fase de apertura, otra de intercambio de datos y una última de cierre.

Para establecer una conexión tiene que haber un proceso que asuma un papel activo y otro que asuma el papel pasivo. El extremo pasivo utiliza una función que sirve para quedarse esperando a que alguien quiera establecer una conexión con él, facilitando el puerto TCP en el que espera. El extremo activo utiliza otra función distinta para abrir la conexión, facilitando la dirección IP y el puerto del otro extremo (el pasivo). De forma explícita o implícita. El extremo activo facilita, para que el otro tenga constancia, su propia dirección IP y su puerto. La conexión, una vez establecida, queda identificada por las dos direcciones IP y los dos puertos de los extremos.

Cuando la conexión ya está abierta, se dispone de un canal bidireccional para el intercambio de octetos. Cada extremo puede leer de la conexión los datos que el otro ponga en ella. El sistema operativo ofrece medios para que un extremo se bloquee si es que quiere leer datos pero no hay ninguno disponible, o bien si es que quiere escribir y no es posible debido a que se ha puesto en marcha algún mecanismo de control de flujo.

Por último, cualquiera de los dos extremos puede optar por cerrar la conexión. El cierre de conexiones “normal” de TCP es ordenado. El extremo que cierra la conexión deja de enviar datos, pero aun puede seguir leyendo aquellos que queden pendientes de recepción del otro extremo. De esta forma, el cierre de la conexión no implica ninguna pérdida de datos. En ocasiones se puede utilizar un cierre abrupto: la conexión se cierra inmediatamente y los datos pendientes de recepción se pierden.

II.8.2.3 CABECERA UDP

0	16	31
0	Puerto Destino	Puerto Origen
32	Longitud del Mensaje	Suma de Verificación
64	DATOS	

II.8.2.3.1 Cabecera UDP

Definición de campos:

Puerto Destino: Campo de 16 bits. Identifica el proceso de recepción de los datos.

Dado que UDP no tiene servidor de estado y no recibe respuestas este campo puede ser opcional; de no ser utilizado el campo, se pondrá en 0.

Puerto Origen: Campo de 16 bits. Identifica el proceso de envío de los datos.

Longitud del Mensaje: Campo de Datos del datagrama. Campo obligatorio de 0 bits si no hay datos ó 16 bits usados para el datagrama.

Suma de verificación (Checksum): Campo opcional pero usado generalmente en envíos o recepción de datos. Campo de 16 bits.

El protocolo UDP es usado cuando se transmite audio o video; la rapidez de entrega es prioridad para el protocolo y no la entrega fiel de datos, provocando que bytes no lleguen completos.

II.9 ICMPv4

El Protocolo de Mensajes de Control de Internet (Internetworking Control Message Protocol) está dentro de la familia de protocolos de TCP/IP, su función es de notificar si existen errores cuando son enviados paquetes a través de la red. No los corrige, únicamente notifica que un error existe, por ejemplo, la red especificada no es encontrada, el datagrama no llegó a su destino o llegó incompleto.

Los mensajes de ICMP son generados y procesados por el software de TCP/IP, no el usuario, con esto se requiere de ningún número de puerto en la cabecera ICMP. Los mensajes están encapsulados en datagramas de IP, estos mensajes son considerados por IP como datos.

ICMP no asegura la fiabilidad del protocolo de IP ya que se podrían perder los datagramas sin advertirlo, los protocolos de nivel superior como TCP resuelven esto.

Este protocolo tiene su propia versión para IPv4 y otra para IPv6, conocidas como ICMPv4 e ICMPv6.

II.9.1 Cabecera ICMPv4

La siguiente figura muestra la encapsulación de los mensajes ICMP dentro de IP

Cabecera IP	Cabecera ICMP	Datos

II.9.1.1 Encapsulación de mensajes de ICMP dentro de IP

La estructura de los mensajes ICMP son de la siguiente manera:

Tipo	Código	Suma de comprobación
Datos ICMP		

II.9.1.2 Estructura de los mensajes de ICMP

Tipo: especifica el tipo de mensaje.

Código: es el código de error que afectó al datagrama referido de IP.

Suma de comprobación: utiliza el complemento a 1

Datos ICMP: habitualmente contiene alguna parte del mensaje original IP que es el que causa el mensaje ICMP.

Los mensajes ICMP tienen un formato, están con tipos de mensajes dependiendo del error que se haya generado, aquí los más comunes:

Tipo	Mensaje ICMP	Descripción
0	Respuesta al eco	Cuando el host no responde
3	Destino inalcanzable	El destino no existe
4	Fuente saturada	La fuente tiene demasiados procesos anteriores
5	Redirección de ruta	La ruta que se estableció no fue la indicada
8	Solicitud de eco	El host no envía el eco
11	Tiempo del datagrama excedido	El tiempo de vida llego a cero
12	Parámetro problema en datagrama	El datagrama contiene errores
13	Requerimiento de hora y fecha	No especifica su fecha
14	Respuesta de host y fecha	La fecha puede no coincidir
17	Requerimiento de máscara de dirección	La máscara no es la de la red
18	Respuesta de máscara de dirección	La respuesta no es valida

1.9 Tabla de errores más comunes de ICMP

CAPÍTULO III

III.1 IP VERSIÓN 6

Los creadores de Internet jamás imaginaron el auge que tendría Internet y creyeron que con las direcciones de IPv4 serian suficientes, pero no fue así, el gran incremento del uso de Internet alrededor de todo el mundo crecía rápidamente y las direcciones escaseaban de la misma manera.

Las versiones 0 a 3 de IP no fueron comercializadas porque o son de uso privado o de uso experimental como en el caso de la versión 5.

IPng ahora es conocido como IPv6. Sus creadores Steve Deering de Xerox PARC y Craig Mudge.

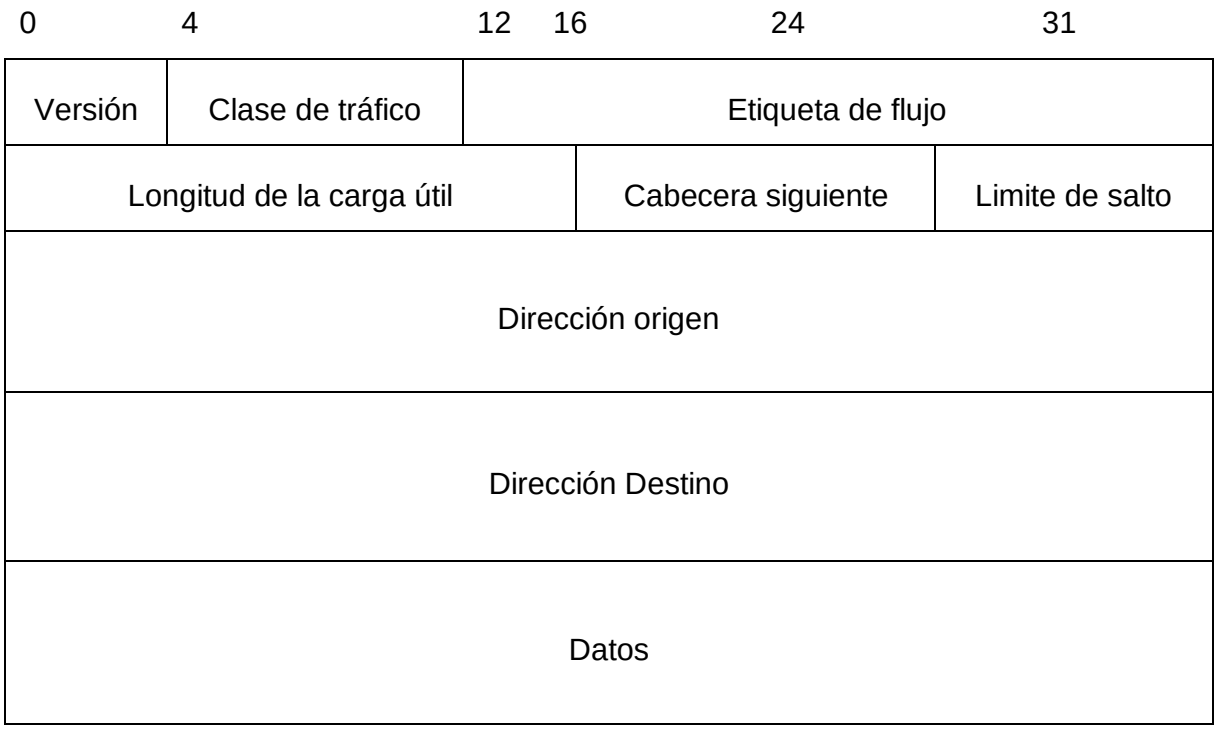
IPv6 soporta 340 millones de direcciones, aproximadamente (2^{128}), esto nos da una capacidad de direcciones que puedan alcanzar hasta el 2020 según estudiosos en el tema.

El Grupo de Trabajo en Ingeniería de Internet, IETF (Internet Engineering Task Force), en la década de los 90's empezó a desarrollar una nueva versión mejorada de IP de nueva generación, para resolver el problema del escaseo de direcciones por el rápido crecimiento de Internet

III.2 CABECERA DE IPV6

Las mejoras que se hicieron a la versión 4 de IP se ven reflejadas en la cabecera de la versión 6, se eliminaron varios campos que solo reducían mas la capacidad de los datos; la fragmentación se realiza ahora desde el puerto origen, permitiendo que el tiempo de vida sea más amplio para el envío.

III.2.1 FORMATO DE LA CABECERA IPV6



Descripción de los campos de la cabecera de IPv6

Versión: este campo especifica el número de versión de IP que se está utilizando, en este caso el valor es 6. Campo de 4 bits.

Clase de tráfico: define la prioridad del datagrama con respecto a la congestión del tráfico, es decir, cuando otros datagramas son enviados desde la misma fuente. IPv6 divide el tráfico en dos categorías: control de congestión y de descongestión:

Control de congestión:

Tráfico controlado: cuando varios paquetes salen de una misma fuente, éste tráfico controla el flujo. Tiene una prioridad de 0 a 7. Siendo 0 la más baja y 7 la más alta prioridad.

Prioridad	Definición
0	Tráfico no especificado
1	Último dato
2	Tráfico sin datos
3	Reservado
4	Tráfico de datos de mayor prioridad
5	reservado
6	Tráfico interactivo
7	Control de trafico

Tráfico no especificado: la prioridad 0 es asignada al paquete cuando los procesos no tienen definida una prioridad.

Último dato: este grupo de prioridad 1 usualmente define el último dato entregado.

Tráfico sin datos: si un usuario no espera datos que será recibido, tiene prioridad 2. Los correos electrónicos pertenecen a este grupo. El destinatario de un “e-mail” no sabe cuando el mensaje llegó.

Tráfico seguido de la mayoría de datos: un protocolo que transfiere datos mientras el usuario está esperando a recibirlos (probablemente con retraso) tiene prioridad 4. Los protocolos FTP y pertenecen a este grupo.

Tráfico interactivo: los protocolos tales como TELNET que necesitan usar la interacción son asignadas como las segunda prioridades más alta (6) en este grupo.

Control de tráfico: tiene la prioridad más alta. Los protocolos de ruteo como OSPF, RIP y los de administración como SNMP tienen esta prioridad.

Tráfico controlado de descongestión: es el tipo de tráfico que espera el mínimo retraso. El descarte de datagramas no es conveniente. La retransmisión es, en la mayoría de los casos, imposible. Es decir, la fuente no se adapta a la congestión. Los archivos de video y audio en tiempo real son ejemplos de este tipo de tráfico.

Las prioridades asignadas a este tipo de tráfico van del 8 al 15, aunque todavía no sean el estándar particular, asignadas basadas en que tanta afecta la calidad de los datos recibidos por el descarte de algunos paquetes.

El contenido de datos menos redundante (como la baja fidelidad en audio y video) se les da la prioridad más alta (15). Los datos con más redundancia (como la alta fidelidad del audio y video) se le dan la prioridad más baja (8).

Etiqueta de flujo: una secuencia de datagramas, enviados de una fuente particular a un destino particular, necesita un manejo especial por ruteadores llamado un flujo de datagramas. La combinación de la dirección fuente y el valor de la etiqueta de flujo definen un flujo de datagramas.

Para un router, un flujo es una secuencia de paquetes (datagramas) que comparten las mismas características, así como la misma ruta de viaje, usando las mismas fuentes, teniendo el mismo tipo de seguridad.

Un router que soporta el manejo de etiquetas de flujo tiene una tabla de etiqueta de flujo. La tabla tiene una entrada para cada etiqueta de flujo activa; cada entrada define los servicios requeridos por la correspondiente etiqueta de flujo. Cuando el router recibe un paquete, éste consulta la tabla para encontrar la entrada correspondiente para el valor definido en el datagrama y entonces provee al paquete con los servicios mencionados en la entrada. No obstante es importante mencionar que la etiqueta de flujo no provee la información por sí misma, esta información es provista por otros protocolos como salto por salto.

Sencillamente se puede usar a la etiqueta de flujo como un acelerador del procesamiento de un datagrama por un router. Cuando un router recibe un datagrama, en lugar de consultar la tabla de routing e ir directo a un algoritmo para definir la dirección del siguiente salto, éste puede fácilmente mirar en la tabla de etiqueta de flujo para el siguiente salto.

El video o audio en tiempo real, particularmente en forma digital, requiere fuentes como una alta banda ancha, grandes buffers y tiempo largo de procesamiento. Un proceso puede hacer una reservación para estas fuentes de antemano garantizando que el dato de tiempo real no se retrase a través de las capas de fuentes.

El uso de tiempo real y reservación de estos recursos requieren otros protocolos como lo son RTP y el protocolo de reservación de recursos RSVP junto con IPv6.

Para permitir el uso eficaz de las etiquetas de flujo, se han definido 3 reglas:

1. La etiqueta de flujo es asignada al datagrama por el host fuente. La etiqueta está en un número al azar entre 1 y 2^{24}-1. Una fuente no debe de reutilizar una etiqueta de flujo para un nuevo flujo mientras ésta todavía esté activa.

2. Si un host no soporta una etiqueta de flujo, se fija el campo a cero; éste simplemente lo ignora.
3. Todos los datagramas pertenecientes al mismo flujo tienen la misma fuente, mismo destino, misma prioridad y mismas opciones.

Longitud de carga útil: este campo es de 16 bits. La longitud se limita a 65,535 bytes. Se puede enviar cargas útiles mucho más grandes y para esto se utiliza la cabecera de extensión, que se explicaran más adelante.

Cabecera siguiente: este campo identifica el tipo de cabecera de extensión que le sigue a la cabecera básica. La longitud de la cabecera base está fijada en 40 bytes. Sin embargo par darle más funcionalidad al datagrama de IP, ésta cabecera puede seguirle alrededor de 6 cabeceras de extensión. Muchas de éstas cabeceras son las "opciones" en la cabecera de IPv4.

La cabecera fija del datagrama en IPv6 puede ser seguida de una o varias cabeceras de extensión que son opcionales, cuando se envían datagramas hay algunas opciones que no se utilizan y ocupan una parte importante de cada trama, a eso se suma que las direcciones de IPv6, por su tamaño, incrementan el problema de ineficiencia. Aquí surge el dilema de qué opciones son las que se utilizaran, los diseñadores no pueden saber cuáles de esos recursos serán necesarios.

Los encabezados de extensión de IPv6 son similares a las opciones de IPv4; un emisor puede decidir que encabezados incluye en el datagrama y cuáles no.

En general, las cabezas de extensión dan mayor flexibilidad

Limite de salto: este campo reemplaza al que sería el de TTL en la versión 4 de IP. Este campo da números de saltos que puede realizar un datagrama antes de que se descarte en el enrutador.

Direcciones origen y destino: solamente nos identifican las computadoras origen y destino. Más adelante se especificará el formato de la dirección.

Datos: en este campo vienen los datos que serán enviados; cuando llegan a su destino la carga útil se remueve del datagrama de IP y pasa al protocolo especificado en la cabecera siguiente.

Los campos que se eliminaron del formato de la versión 4 a la versión 6 de IP fueron:

Fragmentación y reensamblado: ahora estas acciones se desarrollan en la computadora origen y en la computadora destino, respectivamente, ya no es necesario utilizar “intermediarios” para realizarlos.

Suma de comprobación: al igual que la fragmentación y el reensamblado este campo también ocupaba mucho espacio.

Opciones: este campo no se eliminó, solo pasó a formar parte del campo “cabecera siguiente”.

III.3 FRAGMENTACIÓN Y REENSAMBLADO EN IPV6

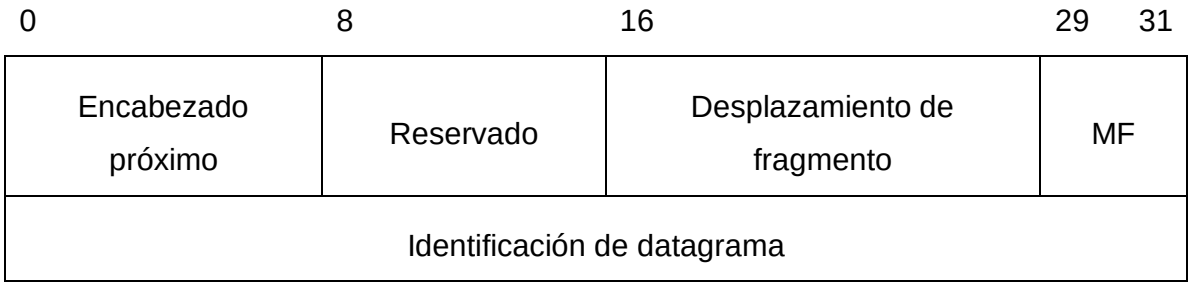
La fragmentación en IPv6 se realiza desde la fuente origen, no en el camino elegido como lo hacía IPv4, ahora la cabecera identifica si el datagrama necesito ser o no fragmentado antes de ser enviado. Cuando identifica esto determina si el datagrama es fragmentado y subfragmentado.

El reensamblado se realiza en la fuente destino, cada fragmento y subfragmento va etiquetado con la cabecera, esto facilita el reensamblado porque no importa el orden en como lleguen, la fuente destino los identifica (fragmentos iniciales, medios y finales) y procede a reunirlos.

La fragmentación y reensamblado en IPv6 es muy similar al de IPv4, en cuanto a que tiene que cubrir el espacio requerido por el MTU, la diferencia existe en que se hace desde el origen y el destino.

Antes de envía un datagrama, la fuente realiza una búsqueda del MTU (path del MTU) de la ruta que se va a utilizar, identificando este MTU la fuente fragmenta el datagrama de manera que cada fragmento sea más pequeño a la medida establecida del MTU

En el encabezado base de IPv6 se inserta entonces un encabezado de extensión para cada fragmento.



Formato del encabezado de extensión de fragmento

La fragmentación de IPv6 conserva muchas de las características del IPv4; los fragmentos son múltiplos de 8 octetos, un bit en el campo MF marca el último fragmento como el bit del IPv4, More Fragments (MF, mas fragmentos) y el campo Datagram Identification (identificación del datagrama) – IPv6 aumenta este campo a 32 bits para la adaptación a las redes de alta velocidad - transporta una id única que el receptor utiliza para el grupo de fragmentos.

Este nuevo método de fragmentación significa ventajas y desventajas; las más significativas son:

Ventajas:

- Reduce la congestión y utilización máxima de los ruteadores facilitando la unidad de tiempo para la entrega
- Al haber ruta directa es improbable la perdida de los datos
- Se establece una ruta directa
- El path MTU permite encontrar la ruta y así mismo fragmenta el datagrama a
- manera de que cada fragmento es menor al tamaño de la ruta.

Desventajas:

- La ruta puede cambiarse y pasar por diferentes ruteadores y algunas rutas pueden cambiar el tamaño del MTU y el ruteador tiene que volverá fragmentar el datagrama
- No tiene mayor flexibilidad de envío como en IPv4

III.4 TIPOS DE DIRECCIONES IPV6

Se define un nuevo tipo de dirección en IPv6: la dirección Anycast (dirección a cualquier nodo de la red), son parecidas a las direcciones multidirigidas, pero los datos enviados a las direcciones de este tipo deben entregarse a cualquier dirección en el grupo en lugar de a todas las direcciones que pertenezcan a este grupo.

Cuando surgieron los cambios en los protocolos de TCP/IP se notó que no sería suficiente agregar nuevos protocolos, como fue el caso de RSVP, IPsec y MPLS. También se necesitaría cambiar/modificar el protocolo IP.

Para modificar totalmente a IP se enfocaron en los siguientes objetivos:

- La creación de un método de direccionamiento escalable
- Reducción en las operaciones hechas por los ruteadores.
- Garantizar calidad en el servicio de transporte
- Asegurar la protección de los datos que son transmitidos a través de la red

Esto se inició en 1992 habiendo diversas versiones alternativas de IPng: ipv7 (diseñado por Ullman), ENCAPS (diseñado por Hinden), SLP (por Deering) y PIP (por Francis) se unieron los 3 últimos dando como resultado de la estructura SIPP, ésta se convirtió en la base para el desarrollo de IPng.

Han habido varias propuestas para el cambio de IP; modificación del protocolo CLNS de OSI, el SIP (simple IP) que propone conservar la mayor parte del protocolo IP pero extenderlo para su adaptación a direcciones más extensas; una versión extendida de SIP, el SIPP (Simple IP Plus).

Otro problema es el nombre que se le asignaría a las modificaciones hechas, al “nuevo” IP

El IAB informó con una declaración política que el nombre seria “IP version7” lo cual hizo polémica al dejar “fuera” a las versiones 5 y6; la versión 5 resultó ineficaz porque

estaba ya destinada al protocolo ST que era para un futuro reservado, así el IETF definió el nuevo nombre como IP “la nueva generación” así “nació” IPng.

Ya que las versiones 1 a la 3 nunca fueron formalmente asignadas, la 4ª versión estaba ya en uso y la 5ª reservada, así IPng se decidió asignarle el número 6 de la versión para distinguirlo de la anterior versión (IPv4), dando como resultado IPv6 (algunos lo mencionan como IP6)

IPv6 soporta la entrega sin conexión es decir el datagrama puede ser ruteado independientemente.

Ambas versiones de IP conservan varias similitudes (encabezado fijo, forma de datagrama, etc.) pero IPv6 cambia algunos detalles: direcciones más largas, una fragmentación desde el inicio, agrega encabezados de formato fijo, etc.

Los cambios significativos de IP fueron en el sistema de direccionamiento de estas redes. Estos cambios se relacionan principalmente con el incremento de la capacidad del bit de dirección.

El objetivo principal de la modificación de direccionamiento no fue ampliar el espacio de las direcciones sino ampliar la funcionalidad a expensas de introducir nuevos campos.

IPv6 proporciona 4 niveles de jerarquía, tres de ellos identifica la red y el otro identifica el host, esto soporta de manera eficaz la tecnología CIDR, que se menciona en el capítulo 2.

Disminuye los gastos de enrutamiento.

La forma de las direcciones IPv6 ahora son hexadecimales (por puros cambios cosméticos). Está formada por 4 dígitos hexadecimales separadas por dos puntos “:”

Ejemplo:

FECD:0A89:0000:0000:0000:0000:7654:3120

Una de las mejoras es que si existe una secuencia de ceros contínua puede ser reducida de la siguiente manera:

FECD:0A89:0:0:0:0:7654:3120

Reduciéndose aun más así:

FECD:0A89::7654:3120

La abreviación de cadena de ceros "::" puede escribirse una sola vez dentro de una dirección. Al inicio de un campo de dirección también se pueden eliminar los ceros:

FECD:A89::7654:3120

Las redes que soportan ambas versiones de IP (4 y 6), permiten la forma decimal de IPv4 y la hexadecimal para IPv6 para los 4 bytes menos significativos de la dirección conocidos como el prefijo de formato (Format Prefix, FP)

UNICAST (unidirigidas): entrega los paquetes al nodo de destino a una interfase de red específica. Aquí no existen las clases de red (A, B, C, D) como en el caso de IPv4.

Este tipo de direcciones define una sola computadora. El paquete debe de ser entregado a una dirección unidirigida de esa computadora específica.

Este tipo de direcciones se dividen en varios tipos que reflejan las situaciones más comunes para las redes contemporáneas.

MULTICAST (multidirigidas): Este tipo de direcciones son usadas para definir un grupo de host en lugar de solo uno. Tiene un prefijo de formato de la siguiente manera:

11111111 identificando grupos de interfases que generalmente se relacionan con diferentes hosts; de esta forma el datagrama es entregado a todas las interfases que la dirección tiene. Este prefijo se usa en todas las direcciones en el primer campo.

Cuando un paquete es enviado se entrega a cada uno de los miembros del grupo.

En IPv6 este tipo de direcciones son utilizadas para reemplazar las direcciones de transmisión o difusión, esto se hace introduciendo una dirección de grupo especial que conecta a todas las interfases de la subred.

ANYCAST (cualquier nodo): parecido al multicast a excepción de que el datagrama solamente es entregado a una sola de las interfases, la más próxima de acuerdo a la métrica de los protocolos de enrutamiento.

Este tipo de direcciones define un grupo de computadoras con direcciones que tienen el mismo prefijo.

Cuando un paquete se envía a una dirección cualquier nodo debe ser entregado exactamente a uno de los miembros del grupo, el más cercano o el más fácilmente accesible.

Direcciones privadas: para uso en redes autónomas. Tienen un formato especial. Existen dos variables en IPv6 para las direcciones de uso local:

1) Para direcciones no divididas en subredes y no utilizan el enrutamiento conocidas como direcciones de enlace local, con un prefijo de 10 bits: 1111 1110 10
   Dado que no existe subred, la dirección de enlace local solo tiene 64 bits de identificador de interfase, los demás bits están establecidos en cero, a excepción del primero.

2) Direcciones locales destinadas para usarlas en redes divididas en subredes conocidas como direcciones de sitio local. Con prefijo 1111 1110 11 conteniendo un campo adicional de 2 bytes del número de subred

   Las direcciones únicas agregadas son el subtipo principal de las direcciones unidirigidas.

   Las direcciones agregadas de IPv6 tienen 6 campos:

III.5 ICMP VERSIÓN 6

Al cambiar la versión de IP y sus campos de cabecera, también cambiaron los probables errores existentes, por consiguiente el protocolo encargado de manejar y de notificar estos errores también fue modificado, el Protocolo de Mensajes de Control de Internet ICMP (Internetworking Control Message Protocol)

Los mensajes están agrupados en dos clases, mensajes de error y mensajes de información.

ICMP tiene 4 mensajes de error:

1. Destino inalcanzable
2. Datagrama muy grande
3. Tiempo excedido
4. Problema de parámetros

Y los mensajes de información:

- Envío y respuesta al eco.

Los mensajes ICMPv6 siempre están precedidos de una cabecera IPv6 y ninguna, una o más cabeceras siguientes.

El formato de la cabecera de ICMPv6 no varía mucho de la versión 4, los mensajes que salen del envío sí.

CAPÍTULO IV

IV.1 COMPARATIVO ENTRE LAS VERSIONES DE IP

Las versiones de IP se fueron cambiando por motivos de expansión en la Internet. Al haber un crecimiento rápido de computadoras y así mismo un número de direcciones IP, se requirió que la entrega y recepción de datos fuera más precisa, constante y rápida.

Los cambios realizados en las versiones han servido para la reducción de memoria en la computadora, reducción en el tiempo de entrega, mayor capacidad de paquetes, soporte en la red, entrega eficaz de los paquetes, rutas más cortas o directas.

Con estos cambios se presentan muchas ventajas pero también tiene desventajas:

- La ruta puede no ser segura ya que si el Path MTU no tiene contemplado el cambio de routers, ya que estos pueden cambiar por varios motivos (tráfico muy pesado, el router está ocupando su 100% de su capacidad), los datos puede perderse fácilmente.
- En caso de haber un cambio de ruta y que sea más pequeño que los fragmentos, éstos se vuelven a fragmentar pero no se les coloca ninguna cabecera de MF sino que se crean como si fuera un nuevo datagrama y se inserta como si fuera dato.

Los cambios realizados en la cabecera de IP son muy significativos, permiten mayor fluido de datagramas, los campos que ya no aparecen en la nueva versión reducían la capacidad de almacenamiento de los datos (a veces era más cabecera que datos en sí)

Los campos que fueron eliminados o incluidos en otros campos son opcionales, así no ocupan memoria útil si no son ocupados, los que se usan no ocupan más memoria y se realiza el trabajo

Campos	IPv6	IPv4	Pros	Contras
Versión	**ü**	**ü**	Soporta ambas versiones de IP	IPv4 solo soporta su propia versión por ser "la primera"
IHL	**ü**	**ü**	Se mantiene este campo para tener control del tiempo de envío	**x**
Tipo de servicio	**x**	**ü**	Se elimina de la nueva versión ya que solo identifica datagramas de esta versión.	Solo sirve en la versión 4
Longitud total	**x**	**ü**		Utiliza tiempo valioso en el envío y recepción de datos.
Identificación	**x**	**ü**		Se eliminó porque no es necesaria ninguna identificación
TTL	**x**	**ü**	Controlaba el tiempo que un datagrama podía estar en una vía de comunicación	El tiempo a veces no era suficiente para enviar correctamente el datagrama
Protocol	**x**	**ü**	La versión 6 reconoce ambas versiones del protocolo. Este campo se cambió en la actual versión	Solo la versión 6 reconoce ambos protocolos (4 y 6)
Comprobación de cabecera	**x**	**ü**		Ocupaba demasiado espacio en la cabecera así como tiempo
Dirección fuente	**ü**	**ü**	Necesaria para el envío	
Dirección destino	**ü**	**ü**	Necesaria para la recepción	
Opciones	**ü**	**ü**	Dan la posibilidad de "partir" un datagrama al pasar por una red de bajo MTU. Se quedó el campo en ambas versiones	
Datos	**ü**	**ü**	Los datos del datagrama a enviar.	
Banderas	**ü**	**ü**		Ocupaban demasiado espacio en la cabecera. Algunas se eliminaron por completo. Otras se "unieron" en otro campo de la nueva versión

Tabla IV.1.1 Diferencias de los campos en ambas versiones, sus pros y contras.

Existen 3 puntos importantes para mejorar la versión de IP

- Problemas que surgen en las subredes de Internet, es decir, ya que Internet es una red de redes, las redes primarias generan problemas antes de que lleguen a la carretera principal.
- Los fundadores de Internet: compañías, gobierno que lo utilizan, realizan proyectos que impactan el diseño de Internet.
- Los investigadores que participan en el protocolo TCP/IP la utilizan diariamente, motivo para mejorar el servicio y ampliar la funcionalidad; ellos identifican rápida y primeramente los problemas que se presentan.

El protocolo TCP/IP es inestable, por tanto hay cambios permanentes, la evolución de la tecnología avanza.

Al aumentar las direcciones de IP, hubo un incremento en:

Envíos masivos de paquetes, saturación de rutas, ruteadores, deficiencia en el envío, datagramas incompletos, perdidos o enviados a otra dirección IP.

El equipo de trabajo de TCP/IP revisa la nueva tecnología, utilizándola inmediatamente como servidores o ruteadores para el envío de datagramas.

La implementación de nuevas tecnologías “obliga” al cambio de IP ya que los protocolos en servicio no pueden dar esos servicios como audio y video, con las demandas que existen por tener un buen video en tiempo real, y el audio con un poco de retardo, así como los protocolos que sincronicen ambos con flujos de datos.

Al ser Internet la “red de redes” el aumento de tamaño y carga de tráfico se aumentó exponencialmente, sin embargo las redes no han aumentado en la misma forma. Además de este factor, en Internet ahora no solo es usado por ingenieros,

académicos, e investigadores, también el público en general lo utiliza, sin caer en dudas, en un mayor porcentaje.

Otro factor importante es, como se mencionó, el envío de video y audio en tiempo real, estos generan más tráfico que las aplicaciones de texto. Las herramientas de búsqueda hacen un tráfico de cantidad enorme haciendo lento el sondeo para encontrar datos en las localidades de Internet.

El protocolo IPv4 se mantuvo prácticamente sin cambios desde sus inicios en los 70's. Su estructura es flexible y poderosa, se ha adaptado a los cambios producidos en la red. A pesar de éstas ventajas, éste protocolo sufre fracturas en su estructura.

IPv4 se mantenía casi intacto, pero el incremento de memoria en las computadoras, incremento de ancho de banda, mayor utilización de tecnologías (WAN y LAN) y el factor de que los cambios no sucedieron simultáneamente, hacen que el protocolo se vea afectado.

Anteriormente el espacio de 32 bits eran suficientes, con el incremento de redes LAN y WAN, este espacio es insuficiente porque no se adapta al crecimiento de la red.

Las nuevas aplicaciones como el envío y recepción de audio y video en tiempo real necesitan garantías en los retardos, la nueva versión de IP debe proporcionar el mecanismo que logra asociar el datagrama con una reservación de fuente pre asignado. Todas estas nuevas aplicaciones también requieren comunicaciones seguras. La nueva versión de IP debe incluir capacidades para autenticar al emisor.

Se propusieron varias versiones para el nuevo IP. Algunas eran muy costosas, otras se basaban en otros protocolos y algunos más proponían adaptarlo a direcciones extensas. Todos los protocolos eran muy buenos, pero se requería de un cambio radical, lo que suponía un cambio no solo en el protocolo en sí, si no en toda la red, algo que realmente sería muy costoso y tardado.

La propuesta de modificar y mejorar el protocolo ya implementado fue la mejor solución. Se analizó a detalle los campos y se identificó cuales serian necesarios y cuales los desechables, sin embargo hay campos que no pueden eliminarse pero que

en la cabecera principal ocupaban mucha memoria útil, la solución: incluirlos en otros campos de la misma línea.

IV.2 BENEFICIOS DEL CAMBIO

Muchos han especulado acerca de los beneficios del cambio en la cabecera de IP, ¿Los campos eliminados realmente no eran tan útiles? eran útiles en cuanto a funcionamiento, pero ocupaban mucha memoria útil

Con los cambios efectuados los beneficios han sido en todos aspectos. En una computadora personal, los programas, datos, comandos, etc., son más rápidos, fluyen de manera cómoda, con menos errores, más disponibilidad de espacio y menos pérdida de información.

En una red hace que el envío-recepción de datos se haga más ligero. Por ejemplo, una red interna, los datos se perderían menos, se actualizarían sin necesidad de consumir memoria útil y serían entregados en el momento.

En el ámbito de instituciones, todas son beneficiadas, salud, economía, educación, etc.

Otros de los beneficios es el aumento o ampliación en las direcciones IP, con el auge de computadoras y dar direcciones al por mayor, era obvio que se agotaran en un tiempo relativamente corto.

Al haber cambios en la cabecera los hubo también en el formato de las direcciones, ampliando mas direcciones de IP y evitando que se agotaran las pocas accesibles de la versión 4, éstas últimas ya otorgadas en el mes de febrero.

Cuatro mil millones de direcciones IP de la versión 4 fueron asignadas y agotadas en un periodo de 32 años aproximadamente. Razón suficiente para actualizar, modificar, aumentar direcciones y optimizar el protocolo de internet.

Con el aumento de direcciones, se espera que tengamos unos años mas antes de generar otras nuevas modificaciones al protocolo, al menos se pronostica hasta el año 2030 con las nuevas direcciones.

Los datos que se beneficiarán y actualizarán serán los de voz, audio, video en tiempo real, como las videoconferencias, haciendo que las imágenes concuerden con la voz, el tiempo de desfase sea mínimo o nulo en la transmisión, la calidad del audio, como en video juegos en línea, música, sea la más óptima.

Algunas compañías telefónicas de servicio fijo y móvil ya implementaron esta versión en sus nuevos equipos y programas mejorando el servicio.

En Internet varias páginas, entre ellas Google y Yahoo!, han usado este protocolo por ser barato, hasta gratuito incluso, fácil de manejar, y crea muchos ingresos a las compañías por ser económicamente bajo en cuanto a mantenimiento, sus características para voz (VoIP) aceleran las paginas, el ambiente sigue siendo eficaz y amigable, a los usuarios les facilita el tiempo de búsqueda por ejemplo.

Las empresas dedicadas a investigaciones de mercado les resulta muy eficaz esta versión porque ahora no son solo escritas las encuestas, también pueden usar sonidos e imágenes sin crear una distracción al usuario, aminorando la carga de las paginas, haciéndolas ligeras para abrir ya que no importa que versión tenga la computadora o dispositivo móvil, IPv6 se acopla a ambas versiones existentes.

La transición de las versiones generará millones de dólares de ingresos en las compañías, dado que mejorarán y crearán nuevos equipos móviles con servicios mejorados de voz y video móvil.

IV.3 CONTROVERSIAS SOBRE EL CAMBIO DE VERSIONES

El cambio de versiones significa muchos movimientos en las computadoras, en las redes y en la población, la comunidad computacional es la más escéptica con el cambio.

Los computólogos siguen siendo reacios a los cambios de protocolo, por temor a que programas y aplicaciones no respondan correctamente a los mismos. ¿Todos los programas identificarán el tipo de versión del protocolo? ¿Se modificarán datos de una versión a otra? Son algunas preguntas que están en el aire con este cambio.

Está establecido que la versión 6 identifica el protocolo de la versión 4, pero los programas que están “acostumbrados” a la antigua versión, así como los MTU y los niveles de red, ¿También lo identificarán?

Los programas tales como video y audio en tiempo real, no sufrirán con el cambio tan radicalmente, ya que éstos mejoran día con día y su tecnología avanza rápidamente.

La tecnología actual tiene muchas ventajas, información, datos, audio, video, son en tiempo real, lo que significa una actualización inmensa de datos, aquí la nueva versión de IP certifica la mejora de los cambios.

La ruta de envío de una versión a otra está en duda si no fracturará la información, el MTU arregla los problemas de tamaños tanto de red como de datos. El reensamblado entonces es mucho más fácil, los datos sufren menos pérdida o daño en la fragmentación, en este aspecto el MTU ya no está inmiscuido en el proceso.

Algunas compañías implementarán IPv6 pero no quitarán del todo IPv4, comenzarán con algunos programas, algunas sucursales, algunas computadoras.

Probablemente solo se necesite un breve ajuste a los módems, enrutadores y servidores para lograr la sincronía entre ambas versiones, a modo de una transición entre estas y así lograr un tráfico de datos eficaz.

Algunas compañías ya tienen resuelto el problema de la transición con módems que soportan y tienen ambas versiones instaladas.

CONCLUSIONES

El cambio de versiones en el protocolo IP se hizo debido a la gran demanda de direcciones IP, envío de datos de imagen, voz y video que hacían más lenta la entrega.

Los cambios a la cabecera también influyeron en otros protocolos de otro nivel como ICMP, UDP, etc.

Los cambios realizados mejorarán por un tiempo las necesidades y requerimientos de usuarios así como de programas utilizados.

Los costos para implementar este nuevo protocolo serán prácticamente bajos; el tiempo de inversión será algo largo, sin embargo, al reconocer ambas versiones, el cambio será casi irreconocible.

Empresas, agencias gubernamentales, salud, educación, serán los medios más beneficiados, así como medios privados, públicos e individuales.

La nueva versión de IP otorgará mas direcciones, una cantidad muy significativa, que durarán por lo menos una década de años, al menos si la demanda no crece exponencialmente.

GLOSARIO

Acknowledgement: Acuse de recibo. Respuesta enviada del puerto destino indicando que los datos llegaron satisfactoriamente. Este tipo de acuses pueden implantar a cualquier nivel, físico, de enlace, etc.

ARP: Address Resolution Protocol, protocolo de resolución de dirección. Protocolo utilizado para asignar una dirección IP de alto nivel a una de bajo nivel.

Best effort: mejor esfuerzo. Característica del protocolo IP que no garantiza la entrega de datagramas. Hace su “mejor esfuerzo” para entregar o recibir pero no se “responsabiliza” si no llegan datos.

Bit: Dígito binario. Unidad más pequeña de información, con dos valores únicos de 0 y 1.

Byte: Unidad de información formada por un octeto (8 bits).

Cabecera (Header): Formato de cada protocolo para el envío, recepción de datos, mensajes de error; que cuenta con diferentes campos cada protocolo.

Campo: Zona de datos en una cabecera, por ejemplo, versión, puertos, datos, etc.

Checksum: Suma de verificación.

Datagrama: Paquete de datos encaminados por diferentes redes, no necesariamente conectados por un circuito virtual o real.

Difusión (broadcast): Transmisión de datos enviada simultáneamente hacia varios destinatarios desde la misma fuente.

Encaminador (router): Nodo encargado de encaminar los datagramas por la mejor ruta, el protocolo opera a nivel de red, el router no.

Fragmentación: Cuando un datagrama pasa por una red de menor capacidad es particionado en partes más pequeñas, se fragmenta.

HTTP: HyperText Transport Protocol, protocolo de transporte de hipertexto. Protocolo de la aplicación de la pila de protocolos TCP/IP del IETF, usado en la comunicación entre clientes y servidores web.

IETF: Internet Engineering Task Force. Grupo de personas que trabajan en el diseño y la ingeniería de la suite TCP/IP y la red global de internet

Máscara de red: Cifra de 32 bits que especifica la parte de subred de una dirección IP. Los bits no cubiertos por esta máscara pertenecen al *host.*

Multidifusión: Técnica que permite pasar copias de un paquete a un subconjunto seleccionado de posibles destinos.

Protocolo: Reglas a seguir para llegar a una meta deseada.

Reensamblado: Unificación de la partes del datagrama particionado.

BIBLIOGRAFÍA

1. Comer, Douglas E., Internet working with TCP/IP, Principles, protocols and architectures, Vol. 1 Fourth Edition, Ed. Prentice Hall

2. Raya Cabrera, José Luis y Raya Pérez, Cristina, Redes locales y TCP/IP, Ed. Ra-Ma, España

3. Black, Uyless. Redes de computadores: protocolos, normas e interfaces. Ed. Addison-Wesley Iberoamericana Ra-Ma

4. López, Ángel y Novo, Alejandro, Protocolos de Internet, diseño e implementación en sistemas Unix. Ed. Alfa-Omega/Ra-Ma

5. Alonso, José Miguel. TCP/IP en Unix: programación de aplicaciones distribuidas. Ed. Alfa-Omega/Ra-Ma

6. García Tomás, Jesús, Raya Cabrera José Luis y Raya, Víctor Rodrigo. Alta velocidad y calidad de servicio en redes IP. Ed. Alfa-Omega/Ra-Ma

7. Sportack, Mark A., Fundamentos de enrutamiento IP. Ed. Cisco

8. Comer, Douglas E. y Stevens, David L., Interconectividad de redes con TCP/IP, diseño e implementación, Vol. II. Ed. Prentice Hall

9. Tanenbaum, Andrew. Computer networks. E: Prentice Hall. 4th Edition

10. Kurose, James F. y Ross, Keith W. Computer networking, a top-down approach featuring the internet. Ed. Pearson Education

Bibliografía electrónica:

Protocolo de transmisión de control TCP http://es.wikipedia.org/wiki/TCP

Protocolo de Internet IP versión 4 http://en.wikipedia.org/wiki/Ipv4

Protocolo de Internet IP versión 6 http://en.wikipedia.org/wiki/IPv6

Protocolo de datagrama de usuario UDP http://es.wikipedia.org/wiki/Udp

Protocolo de internet IP versión 6 ftp://ftp.rfc-editor.org/in-notes/rfc2460.txt

Protocolo de Internet IP versión 6 http://www.protocols.com/pbook/tcpip2.htm#IPv6

Máscara de red http://es.wikipedia.org/wiki/M%C3%A1scara_de_red