**7 Tecnologías y protocolos de red**

|  |  |
| --- | --- |
| [Nivel de aplicación](http://es.wikipedia.org/wiki/Nivel_de_aplicaci%C3%B3n) | [DNS](http://es.wikipedia.org/wiki/Domain_Name_System), [FTP](http://es.wikipedia.org/wiki/File_Transfer_Protocol), [HTTP](http://es.wikipedia.org/wiki/Hypertext_Transfer_Protocol), [IMAP](http://es.wikipedia.org/wiki/Internet_Message_Access_Protocol), [IRC](http://es.wikipedia.org/wiki/IRC), [NFS](http://es.wikipedia.org/wiki/Network_File_System), [NNTP](http://es.wikipedia.org/wiki/Network_News_Transport_Protocol), [NTP](http://es.wikipedia.org/wiki/Network_Time_Protocol), [POP3](http://es.wikipedia.org/wiki/Post_Office_Protocol), [SMB/CIFS](http://es.wikipedia.org/wiki/Server_Message_Block), [SMTP](http://es.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol), [SNMP](http://es.wikipedia.org/wiki/Simple_Network_Management_Protocol), [SSH](http://es.wikipedia.org/wiki/Secure_Shell), [Telnet](http://es.wikipedia.org/wiki/Telnet), [SIP](http://es.wikipedia.org/wiki/Session_Initiation_Protocol), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_de_nivel_de_aplicaci%C3%B3n) |
| [Nivel de presentación](http://es.wikipedia.org/wiki/Nivel_de_presentaci%C3%B3n) | [ASN.1](http://es.wikipedia.org/wiki/ASN.1), [MIME](http://es.wikipedia.org/wiki/MIME), [SSL/TLS](http://es.wikipedia.org/wiki/Transport_Layer_Security), [XML](http://es.wikipedia.org/wiki/XML), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_y_formatos_de_nivel_de_presentaci%C3%B3n) |
| [Nivel de sesión](http://es.wikipedia.org/wiki/Nivel_de_sesi%C3%B3n) | [NetBIOS](http://es.wikipedia.org/wiki/NetBIOS), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_de_nivel_de_sesi%C3%B3n) |
| [Nivel de transporte](http://es.wikipedia.org/wiki/Nivel_de_transporte) | [SCTP](http://es.wikipedia.org/wiki/Stream_Control_Transmission_Protocol), [SPX](http://es.wikipedia.org/wiki/Sequenced_packet_exchange), [TCP](http://es.wikipedia.org/wiki/Transmission_Control_Protocol), [UDP](http://es.wikipedia.org/wiki/User_Datagram_Protocol), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_de_nivel_de_transporte) |
| [Nivel de red](http://es.wikipedia.org/wiki/Nivel_de_red) | [AppleTalk](http://es.wikipedia.org/wiki/AppleTalk), [IP](http://es.wikipedia.org/wiki/Protocolo_de_Internet), [IPX](http://es.wikipedia.org/wiki/Internetwork_Packet_Exchange), [NetBEUI](http://es.wikipedia.org/wiki/NetBEUI), [X.25](http://es.wikipedia.org/wiki/Norma_X.25), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_de_nivel_de_red) |
| [Nivel de enlace](http://es.wikipedia.org/wiki/Nivel_de_enlace_de_datos) | [ATM](http://es.wikipedia.org/wiki/Asynchronous_Transfer_Mode), [Ethernet](http://es.wikipedia.org/wiki/Ethernet), [Frame Relay](http://es.wikipedia.org/wiki/Frame_Relay), [HDLC](http://es.wikipedia.org/wiki/HDLC), [PPP](http://es.wikipedia.org/wiki/Point-to-Point_Protocol), [Token Ring](http://es.wikipedia.org/wiki/Token_Ring), [Wi-Fi](http://es.wikipedia.org/wiki/Wi-Fi), [STP](http://es.wikipedia.org/wiki/Spanning_tree), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Protocolos_de_nivel_de_enlace_de_datos) |
| [Nivel físico](http://es.wikipedia.org/wiki/Nivel_f%C3%ADsico) | [Cable coaxial](http://es.wikipedia.org/wiki/Cable_coaxial), [Cable de fibra óptica](http://es.wikipedia.org/wiki/Cable_de_fibra_%C3%B3ptica), [Cable de par trenzado](http://es.wikipedia.org/wiki/Cable_de_par_trenzado), [Microondas](http://es.wikipedia.org/wiki/Red_por_microondas), [Radio](http://es.wikipedia.org/wiki/Red_por_radio), [RS-232](http://es.wikipedia.org/wiki/RS-232), [*ver más*](http://es.wikipedia.org/wiki/Categor%C3%ADa:Tecnolog%C3%ADas_y_protocolos_de_nivel_f%C3%ADsico) |
|  | |

7.1 Protocolo TCP/IP

**7.1.1. Historia**

El Protocolo de Internet IP) y el Protocolo de Transmisión (TCP), fueron desarrollados

inicialmente en 1973 por el informático estadounidense Vinton Cerf

como parte de un proyecto dirigido por el ingeniero norteamericano Robert

Kahn y patrocinado por la Agencia de Programas Avanzados de Investigación

(ARPA, siglas en ingles) del Departamento Estadounidense de Defensa. Internet

comenzó siendo una red informática de ARPA (llamada ARPAnet) que conectaba

redes de ordenadores de varias universidades y laboratorios en investigación

en Estados Unidos. World Wibe Web se desarrolló en 1989 por el informático

británico Timothy Berners-Lee para el Consejo Europeo de Investigación Nuclear

(CERN, siglas en francés).

**7.1.2. Que es el protocolo TCP/IP**

TCP/IP es un conjunto básico de protocolos de comunicaci´on de redes, popularizado

por Internet, que permiten la transmisión de información en redes de

computadores. El nombre TCP/IP proviene de dos protocolos importantes de

la familia, el Transmission Control Protocol (TCP) y el Internet Protocol (IP).

Existen tantos protocolos (dentro de estos mismos) que juntos llegan a ser más

de 100 protocolos diferentes definidos en este conjunto, algunos de ellos son el

muy usado HTTP (HyperText Transfer Protocol), para la transmisión de páginas

WEB, además de otros, el FTP para transferencia de archivos, el SMTP y

POP para correo electrónico, Telenet utilizado en el acceso remoto a servidores,

entre otros.

Hay que tener en cuenta que en Internet, y en otras redes, se encuentran

conectadas maquinas de clases muy diferentes y con hardware y software incompatibles

en muchos casos, además de todos los medios y formas posibles de

conexión. AquI se encuentra una de las grandes ventajas del TCP/IP, pues este

protocolo se encargará de que la comunicaci´on entre todos sea posible. TCP/IP

es compatible con cualquier sistema operativo y con cualquier tipo de hardware.

TCP/IP no es un ´único protocolo, sino que es en realidad lo que se conoce

con este nombre es un conjunto de protocolos que cubren los distintos niveles

del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission

Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al

conjunto. La arquitectura del TCP/IP consta de cinco niveles o capas en las

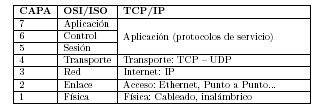
que se agrupan los protocolos.

**7.1.3. Capas del protocolo TCP/IP**

El protocolo TCP/IP trabaja sobre varios niveles de jerarquia o capas; cada

capa corresponde a una o mas capas del modelo del modelo ISO/OSI, que es el

estándar internacional de jerarquización de las redes de datos.

****

Capa fisica: El nivel más bajo es la capa física. AquI nos referimos al

medio fisico por el cual se transmite la información. Generalmente será un

cable aunque no se descarta cualquier otro medio de transmisión como

ondas o enlaces via satélite.

Capa de acceso o enlace: determina la manera en que las estaciones

(maquinas) envian y reciben la información a través del soporte fisico

proporcionado por la capa anterior. Es decir, una vez que tenemos un

cable, ¿cómo se transmite la información por ese cable? ¿Cuándo puede

una estación transmitir? ¿Tiene que esperar algún turno o transmite sin

más? ¿Como sabe una estación que un mensaje es para ella? Pues bien,

son todas estas cuestiones las que resuelve esta capa. TCP/IP necesita

funcionar sobre algún tipo de red o de medio fisico que proporcione sus

propios protocolos para el nivel de enlace. Por este motivo hay que tener

en cuenta que los protocolos utilizados en este nivel pueden ser muy diversos

y no forman parte del conjunto TCP/IP. Sin embargo, esto no debe

ser problemático puesto que una de las funciones y ventajas principales

del TCP/IP es proporcionar una abstracción del medio de forma que sea

posible el intercambio de información entre medios diferentes y tecnologias

que inicialmente son incompatibles. como ejemplos de protocolos de enlace

tenemos a Ethernet, protocolo casi común en redes de ´área local (LAN),

PPP (point to point protocol ) protocolo diseñado para acceso por linea de

telefónica, protocolos de red de ´área amplia (WAN) como ADSL, DOCSIS,

T1, T3, enlaces vía satélite; entre otros.

Capa de Red: Esta define la forma en que un mensaje se transmite a

través de distintos tipos de redes hasta llegar a su destino. El principal

protocolo de esta capa es el IP aunque también se encuentran a este nivel

los protocolos. Esta capa proporciona el direccionamiento IP y determina

la ruta ´optima a través de los ruteadores (routers) que debe seguir un

paquete desde el origen al destino.

Capa de Transporte: Aquí entran los protocolos TCP y UDP; ya no

se preocupa de la ruta que siguen los mensajes hasta llegar a su destino.

Sencillamente, considera que la comunicaci´on extremo a extremo esta establecida

y la utiliza. Además añade la noción de puertos, como veremos

más adelante.

Capa de Aplicación: Una vez que tenemos establecida la comunicaci´on

desde el origen al destino nos queda lo más importante, ¿que podemos

transmitir? La capa de aplicación nos proporciona los distintos servicios de

red o Internet: correo electrónico, páginas Web, FTP, TELNET, compartir

archivos y recursos...

**7.1.4. Caracteristicas del protocolo TCP/IP**

La familia de protocolos TCP/IP tiene caracteristicas importantes que le permite

llevas datos de un lugar a otro, sin importar su distancia, tipo de maquina

o red, de forma segura y rápida;

La tarea de IP es llevar los datos a granel (los paquetes) de un sitio a otro.

Las computadoras que encuentran las vias para llevar los datos de una

red a otra (denominadas enrutadores) utilizan IP para trasladar los datos.

En resumen IP mueve los paquetes de datos a granel, mientras TCP se

encarga del flujo y asegura que los datos est´en correctos.

Las lineas de comunicaci´on se pueden compartir entre varios usuarios.

Cualquier tipo de paquete puede transmitirse al mismo tiempo, y se ordenar

´a y combinará cuando llegue a su destino. Compare esto con la manera

en que se transmite una conversación telefónica. Una vez que establece

una conexión, se reservan algunos circuitos para usted, que no puede emplear

en otra llamada, aun si deja esperando a su interlocutor por veinte

minutos.

Los datos no tienen que enviarse directamente entre dos computadoras.

Cada paquete pasa de computadora en computadora hasta llegar a su

destino. ´Este, claro esta, es el secreto de cómo se pueden enviar datos y

mensajes entre dos computadoras aunque no est´en conectadas directamente

entre si. Lo que realmente sorprende es que solo se necesitan algunos

segundos para enviar un archivo de buen tamaño de una maquina a otra,

aunque est´en separadas por miles de kilómetros y pese a que los datos

tienen que pasar por múltiples computadoras. Una de las razones de la rapidez

es que, cuando algo anda mal, solo es necesario volver a transmitir

un paquete, no todo el mensaje.

Los paquetes no necesitan seguir la misma trayectoria. La red puede llevar

cada paquete de un lugar a otro y usar la conexión más idónea que este disponible

en ese instante. No todos los paquetes de los mensajes tienen que

viajar, necesariamente, por la misma ruta, ni necesariamente tienen que

llegar todos al mismo tiempo.

La flexibilidad del sistema lo hace muy confiable. Si un enlace se pierde,

el sistema usa otro. Cuando usted envia un mensaje, el TCP divide los

datos en paquetes, ordena ´estos en secuencia, agrega cierta información

para control de errores y después los lanza hacia fuera, y los distribuye.

En el otro extremo, el TCP recibe los paquetes, verifica si hay errores y los

vuelve a combinar para convertirlos en los datos originales. De haber error

en algún punto, el programa TCP destino envia un mensaje solicitando

que se vuelvan a enviar determinados paquetes.

7.2. Capa de Red: Protocolo IP

El protocolo IP a diferencia del protocolo TCP, que esta orientado a conexion

, es sin conexión. Está basado en la idea de los datagramas interred, los

cuales son transportados transparentemente, pero no siempre con seguridad,

desde el hostal fuente hasta el hostal destinatario, quizás recorriendo varias redes

mientras viaja. El protocolo IP trabaja de la siguiente manera; la capa de

transporte toma los mensajes y los divide en datagramas, de hasta 64K octetos

cada uno. Cada datagrama se transmite a través de la red interred, posiblemente

fragmentándose en unidades más pequeñas, durante su recorrido normal. Al

final, cuando todas las piezas llegan a la maquina destinataria, la capa de transporte

los reensambla para asi reconstruir el mensaje original. Un datagrama IP

consta de una parte de cabecera y una parte de texto. La cabecera tiene una

parte fija de 20 octetos y una parte opcional de longitud variable.

**7.2.1. Direcciones IP**

La dirección IP es el identificador de cada host (maquina) dentro de su red

de redes. Cada host conectado a una red tiene una dirección IP asignada, la

cual debe ser distinta a todas las demás direcciones que est´en vigentes en ese

momento en el conjunto de redes visibles por el host. En el caso de Internet,

no puede haber dos maquinas con 2 direcciones IP (publicas) iguales. Pero

si podriamos tener dos ordenadores con la misma dirección IP siempre y cuando

pertenezcan a redes independientes entre si (sin ningún camino posible que las

comunique).

Clasificación

Las direcciones IP se pueden clasificar según su dominio en:

Direcciones IP Publicas: Son visibles en todo Internet. Un ordenador

con una IP pública es accesible (visible) desde cualquier otro ordenador

conectado a Internet. Para conectarse a Internet es necesario tener una

direccion IP publica.

Direcciones IP Privadas:Son visibles ´únicamente por otros hosts de

su propia red o de otras redes privadas interconectadas por routers. Se

utilizan en las empresas para los puestos de trabajo. Las maquinas con

direcciones IP privadas pueden salir a Internet por medio de un router (o

proxy) que tenga una IP pública. Sin embargo, desde Internet no se puede

acceder a ordenadores con direcciones IP privadas.

A su vez, las direcciones IP se pueden ser:

Direcciones IP estáticas (fijas): Un host que se conecte a la red con

dirección IP estática siempre lo hará con una misma IP. Las direcciones

IP publicas estáticas son las que utilizan los servidores de Internet con

objeto de que est´en siempre localizables por los usuarios de Internet. Estas

direcciones hay que contratarlas.

Direcciones IP dinámicas Un host que se conecte a la red mediante

dirección IP dinámica, cada vez lo hará con una dirección IP distinta. Las

direcciones IP públicas dinámicas son las que se utilizan en las conexiones

a Internet mediante un módem. Los proveedores de Internet utilizan

direcciones IP dinámicas debido a que tienen más clientes que direcciones

IP (es muy improbable que todos se conecten a la vez).

Esta dirección es un numero de 32 bit que debe ser ´único para cada host,

y normalmente suele representarse como cuatro cifras de 8 bit separadas por

puntos, de la siguiente forma:

desde 0.0.0.0 a 255.255.255.255

La dirección de Internet (IP Address) se utiliza para identificar tanto a

la maquina en concreto como la red a la que pertenece, de manera que sea

posible distinguir a los ordenadores que se encuentran conectados a una misma

red. Con este propósito, y teniendo en cuenta que en Internet se encuentran

conectadas redes de tamaños muy diversos, se establecieron tres clases diferentes

de direcciones, las cuales se representan mediante tres rangos de valores:

Clase A: Son las que en su primer byte tienen un valor comprendido entre

1 y 126, incluyendo ambos valores. Estas direcciones utilizan ´únicamente

este primer byte para identificar la red, quedando los otros tres bytes

disponibles para cada uno de los hosts que pertenezcan a esta misma red.

Esto significa que podrán existir más de dieciséis millones de maquinas en

cada una de las redes de esta clase. Este tipo de direcciones es usado por

redes muy extensas, pero hay que tener en cuenta que solo puede haber

126 redes de este tamaño, como algunas grandes redes comerciales, aunque

son pocas las organizaciones que obtienen una dirección de ¸clase A”. Lo

normal para las grandes organizaciones es que utilicen una o varias redes

de ¸clase B”.

Clase B: Estas direcciones utilizan en su primer byte un valor comprendido

entre 128 y 191, incluyendo ambos. En este caso el identificador de

la red se obtiene de los dos primeros bytes de la dirección, teniendo que

ser un valor entre 128.1 y 191.254 (no es posible utilizar los valores 0 y

255 por tener un significado especial). Los dos ´últimos bytes de la direccion

constituyen el identificador del host permitiendo, por consiguiente,

un número máximo de 64516 maquinas en la misma red. Este tipo de

direcciones tendria que ser suficiente para la gran mayoria de las organizaciones

grandes. En caso de que el número de computadores que se

necesita conectar fuese mayor, seria posible obtener más de una dirección

de ¸clase B”, evitando de esta forma el uso de una de ¸clase A”.

Clase C: En este caso el valor del primer byte tendrá que estar comprendido

entre 192 y 223, incluyendo ambos valores. Este tercer tipo de

direcciones utiliza los tres primeros bytes para el número de la red, con

un rango desde 192.1.1 hasta 223.254.254. De esta manera queda libre

un byte para el host, lo que permite que se conecten un máximo de 254

computadores en cada red. Estas direcciones permiten un menor número

de host que las anteriores, aunque son las más numerosas pudiendo existir

un gran número redes de este tipo (más de dos millones).

Direcciones IP reservadas y especiales

En la clasificación de direcciones anterior se puede notar que ciertos números

no se usan. Algunos de ellos se encuentran reservados para un posible uso futuro,

como es el caso de las direcciones cuyo primer byte sea superior a 223, mientras

que el valor 127 en el primer byte se utiliza en algunos sistemas para propósitos

especiales. También es importante notar que los valores 0 y 255 en cualquier

byte de la dirección no pueden usarse normalmente por tener otros propósitos

especificos. El número 0 esta reservado para las maquinas que no conocen su

dirección, pudiendo utilizarse tanto en la identificación de red para maquinas

que aún no conocen el número de red a la que se encuentran conectadas, en la

identificación de host para maquinas que aún no conocen su número de host

dentro de la red, o en ambos casos. El número 255 tiene también un significado

especial, puesto que se reserva para el broadcast. El broadcast es necesario

cuando se pretende hacer que un mensaje sea visible para todos los sistemas

conectados a la misma red. Esto puede ser ´útil si se necesita enviar el mismo

datagrama a un número determinado de sistemas, resultando más eficiente que

enviar la misma información solicitada de manera individual a cada uno.

En el caso de algunas organizaciones extensas puede surgir la necesidad de

dividir la red en otras redes más pequeñas (subnets). Como ejemplo podemos

suponer una red de clase B que, naturalmente, tiene asignado como identificador

de red un número de dos bytes. En este caso seria posible utilizar el tercer byte

para indicar en que red Ethernet se encuentra un host en concreto. Esta división

no tendrá ningún significado para cualquier otro ordenador que este conectado

a una red perteneciente a otra organización, puesto que el tercer byte no

será comprobado ni tratado de forma especial. Sin embargo, en el interior de

esta red existirá una división y será necesario disponer de un software de red

especialmente diseñado para ello. De esta forma queda oculta la organización

interior de la red, siendo mucho más cómodo el acceso que si se tratara de varias

direcciones de clase C independientes.

**7.2.2. mascara de subred**

Una mascara de subred es aquella dirección que enmascarando nuestra dirección

IP, nos indica si otra dirección IP pertenece a nuestra subred o no. Las

mascaras 255.0.0.0 (clase A), 255.255.0.0 (clase B) y 255.255.255.0 (clase C)

suelen ser suficientes para la mayoria de las redes privadas. En una red de redes

TCP/IP no puede haber hosts aislados: todos pertenecen a alguna red y todos

tienen una dirección IP y una mascara de subred (si no se especifica se toma la

mascara que corresponda a su clase). Mediante esta mascara un ordenador sabe

si otro ordenador se encuentra en su misma subred o en otra distinta. Si pertenece

a su misma subred, el mensaje se entregara directamente. En cambio, si

los hosts están configurados en redes distintas, el mensaje se enviara a la puerta

de salida o router de la red del host origen. Este router pasará el mensaje al

siguiente de la cadena y asi sucesivamente hasta que se alcance la red del host

destino y se complete la entrega del mensaje.

**7.2.3. Enrutamiento (routering**)

Una red de redes esta formada por redes interconectadas mediante routers

o enrutadores. Cuando enviamos un datagrama desde un ordenador hasta otro,

´este tiene que ser capaz de encontrar la ruta más adecuada para llegar a su

destino. Esto es lo que se conoce como encaminamiento.

Los routers son los encargados de elegir las mejores rutas. Estas maquinas

pueden ser computadores con varias direcciones IP o bien, aparatos específicos.

Los routers deben conocer, al menos parcialmente, la estructura de la red que

les permita encaminar de forma correcta cada mensaje hacia su destino. Esta

información se almacena en las llamadas tablas de enrutamiento. Observemos

que debido al sistema de direccionamiento IP esta misión no es tan complicada.

Lo ´único que necesitamos almacenar en las tablas son los prefijos de las

direcciones (que nos indican la red). Por ejemplo, si el destino es la maquina

149.33.19.4 con mascara 255.255.0.0, nos basta con conocer el encaminamiento

de la red 149.33.0.0 ya que todas las que empiecen por 149.33 se enviaran hacia

el mismo sitio.

7.3. Capa de Transporte: Protocolo TCP y UDP

La capa de red transfiere datagramas entre dos computadores a través de

la red utilizando como identificadores las direcciones IP. La capa de transporte

añade la noción de puerto para distinguir entre los muchos destinos dentro de

un mismo host. No es suficiente con indicar la dirección IP del destino, además

hay que especificar la aplicación que recoger’ a el mensaje. Cada aplicación que

este esperando un mensaje utiliza un número de puerto distinto; más concretamente,

la aplicación esta a la espera de un mensaje en un puerto determinado

(escuchando un puerto).

Pero no solo se utilizan los puertos para la recepción de mensajes, también

para el envio: todos los mensajes que envie una maquina debe hacerlo a través

de uno de sus puertos.

**7.3.1. Puertos**

Un ordenador puede estar conectado con distintos servidores a la vez; por

ejemplo, con un servidor de noticias y un servidor de correo. Para distinguir las

distintas conexiones dentro de un mismo computador se utilizan los puertos.

Un puerto es un número de 16 bits, por lo que existen 65536 puertos en cada

computador. Las aplicaciones utilizan estos puertos para recibir y transmitir

mensajes.

Los números de puerto de las aplicaciones cliente son asignados dinámicamente

y generalmente son superiores al 1024. Cuando una aplicación cliente

quiere comunicarse con busca un número de puerto libre y lo utiliza.

En cambio, las aplicaciones servidoras utilizan unos números de puerto pre-

fijados: son los llamados puertos well-known (bien conocidos). Estos puertos

están predefinidos en normas internacionales, tienen un valor menor a 1024, por

ejemplo el puerto 21 es utilizado por el protocolo FTP, el 23 por Telnet, el 80

por HTTP y asi con todos los protocolos de aplicaciones.

Los puertos tienen una memoria intermedia (bu\_er) situada entre los programas

de aplicación y la red. De tal forma que las aplicaciones transmiten la

información a los puertos. Aquí se va almacenando hasta que pueda enviarse

por la red. Una vez que pueda transmitirse, la información irá llegando al puerto

destino donde se irá guardando hasta que la aplicación este preparada para

recibirla.

Los dos protocolos principales de la capa de transporte son UDP y TCP.

El primero ofrece una transferencias de mensajes no fiable y no orientada a

conexión y el segundo, una transferencia fiable y orientada a conexión.

**7.3.2. Protocolo UDP**

El protocolo UDP (User Datagram Protocol, protocolo de datagrama de

usuario) proporciona una comunicaci´on muy sencilla entre las aplicaciones de

dos ordenadores. como características del protocolo tenemos

No orientado a conexion:No se establece una conexión previa con el

otro extremo para transmitir un mensaje UDP. Los mensajes se envian

sin más y ´estos pueden duplicarse o llegar desordenados al destino.

No fiable: Los mensajes UDP se pueden perder o llegar dañados.

UDP utiliza el protocolo IP para transportar sus mensajes. Como vemos,

no añade ninguna mejora en la calidad de la transferencia; aunque si incorpora

los puertos origen y destino en su formato de mensaje. Las aplicaciones (y no

el protocolo UDP) deberán programarse teniendo en cuenta que la información

puede no llegar de forma correcta.

El encabezado del protocolo es muy simple; solo especifica los puertos de

origen y destino del mensaje, la longitud de mensaje y una verificación simple

del encabezado.

**7.3.3. Protocolo TCP**

El protocolo TCP esta basado en el protocolo IP, que no es fiable ni orientado

a la conexión; para orientarlo, es necesario establecer una conexión previa

entre las dos maquinas antes de poder transmitir ningún dato. A través de esta

conexión los datos llegaran siempre a la aplicación destino de forma ordenada

y sin duplicados. Finalmente, es necesario cerrar la conexión.

Caracteristicas

El protocolo TCP permite una comunicaci´on fiable entre dos aplicaciones.

De esta forma, las aplicaciones que lo utilicen no tienen que preocuparse de la

integridad de la información: dan por hecho que todo lo que reciben es correcto.

El flujo de datos entre una aplicación y otra viajan por un circuito virtual.

Sabemos que los datagramas IP pueden seguir rutas distintas, dependiendo del

estado de los encaminadores intermedios, para llegar a un mismo sitio. Esto

significa que los datagramas IP que transportan los mensajes siguen rutas diferentes

aunque el protocolo TCP logre la ilusión de que existe un ´único circuito

por el que viajan todos los bytes uno detrás de otro (algo así como una tuberia

entre el origen y el destino). Para que esta comunicaci´on pueda ser posible es

necesario abrir previamente una conexión. Esta conexión garantiza que los todos

los datos lleguen correctamente de forma ordenada y sin duplicados. La unidad

de datos del protocolo es el byte, de tal forma que la aplicación origen envia

bytes y la aplicación destino recibe estos bytes.

Sin embargo, cada byte no se envía inmediatamente después de ser generado

por la aplicación, sino que se espera a que haya una cierta cantidad de bytes, se

agrupan en un segmento y se envía el segmento completo. Para ello son necesarias

unas memorias intermedias o bu\_ers. Cada uno de estos segmentos viaja en

el campo de datos de un datagrama IP. Si el segmento es muy grande será necesario

fragmentar el datagrama, con la consiguiente perdida de rendimiento; y si

es muy pequeño, se estarán enviando más cabeceras que datos. Por consiguiente,

es importante elegir el mayor tamaño de segmento posible que no provoque

fragmentación.

El protocolo TCP envía un flujo de información no estructurado. Esto significa

que los datos no tienen ningún formato, son ´únicamente los bytes que una

aplicación envía a otra. Ambas aplicaciones deberán ponerse de acuerdo para

comprender la información que se están enviando.

Cada vez que se abre una conexión, se crea un canal de comunicaci´on bidireccional

en el que ambas aplicaciones pueden enviar y recibir información, es

decir, una conexión es full-d´uplex.

El transporte TCP acepta mensajes de longitud arbitrariamente grande procedentes

de los procesos de usuario, los separa en pedazos que no excedan de

64K octetos y, transmite cada pedazo como si fuera un datagrama separado.

La capa de red, no garantiza que los datagramas se entreguen apropiadamente,

por lo que TCP deberá utilizar temporizadores y retransmitir los datagramas si

es necesario. Los datagramas que consiguen llegar, pueden hacerlo en desorden;

y dependerá de TCP el hecho de reensamblarlos en mensajes, con la secuencia

correcta.

fiabilidad

¿Cómo es posible enviar información fiable basándose en un protocolo no

fiable? Es decir, si los datagramas que transportan los segmentos TCP se pueden

perder, ¿cómo pueden llegar los datos de las aplicaciones de forma correcta al

destino?

La respuesta a esta pregunta es sencilla: cada vez que llega un mensaje se

devuelve una confirmación (acknowledgement) para que el emisor sepa que ha

llegado correctamente. Si no le llega esta confirmación pasado un cierto tiempo,

el emisor reenvia el mensaje. El emisor envía un dato, arranca su temporizador

y espera su confirmación (ACK). Si recibe su ACK antes de agotar el temporizador,

envía el siguiente dato. Si se agota el temporizador antes de recibir el

ACK, reenvia el mensaje.

****