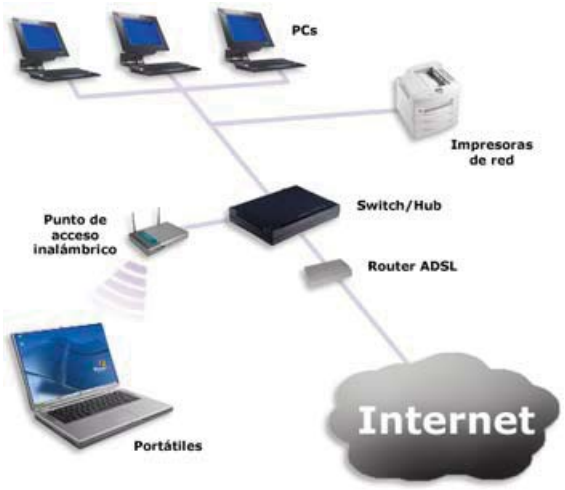
élogos

COMUNICACIÓN DE APLICACIONES EN RED

élogos

Teoría de REDES

Una red es un sistema de ordenadores y otros dispositivos conectados por cables entre sí. La red más simple posible la forman dos ordenadores conectados mediante un cable. A partir de aquí su complejidad puede aumentar hasta conectar miles de ordenadores en todo el mundo. El ejemplo más conocido de este último caso es Internet. Las redes, en general, consisten en «compartir recursos», y uno de sus objetivos es hacer que todos los programas, datos y equipos estén disponibles para cualquier miembro de la red que así lo solicite.



Una red conectada en un área limitada se conoce como Red de Área Local (LAN). Una LAN está contenida a menudo en una sola ubicación. En una LAN, los recursos u ordenadores intercambian información entre sí, permitiendo compartirla. Lo compartido puede ser la información contenida en el disco, una impresora o un módem.

Una Red de Área Extensa (WAN) es un grupo de dispositivos, o varias LAN, conectados en una área geográficamente mayor, a menudo por medio de líneas telefónicas u otro formato de cableado como puede ser una línea de alta velocidad, fibra o enlace vía satélite. Una de los mayores ejemplos de WAN es la propia Internet.

En una red se puede compartir la información y los recursos. Gracias a esta facilidad contamos con una serie de ventajas para nuestro trabajo en los centros:

- Podemos compartir los periféricos caros, como pueden ser las impresoras. En una red, todos los ordenadores pueden acceder a la misma impresora.

- Podemos transferir datos entre los usuarios sin utilizar disquetes. La transferencia de archivos a través de la red elimina el tiempo que se pierde copiando archivos en dispositivos de almacenamiento y luego en otros PCs (por ejemplo, la actualización de nuestro antivirus). Además, hay menos restricciones en el tamaño del archivo que se transfiere a través de la red.

- Se puede crear una copia de seguridad del archivo automáticamente. Se puede utilizar un programa para hacer copias de seguridad de archivos automáticamente, con lo que se ahorra tiempo y se garantiza que todo el trabajo ha quedado guardado.

- Se puede enviar y recibir correo electrónico a y desde cualquier punto del globo, comunicar mensajes y avisos a mucha gente, en un sinfín de diferentes áreas, rápida y económicamente.

- Se puede acceder a los vastos recursos de Internet y de la Web mundial.

Los modelos más comunes son el *Cliente-Servidor* y el modelo *Par a Par*. En los centros de trabajo nos encontramos con el modelo *Cliente-Servidor*, en donde todas las estaciones de trabajo (equipos conectados a la red) pueden actuar como clientes conectados a Ordenadores Centrales que actúan como Servidores. A continuación se muestran brevemente las características de estos modelos.

Cliente-Servidor

Éste es un modelo de proceso en el que las tareas se reparten entre programas que se ejecutan en el servidor y otros en la estación de trabajo del usuario. En una red cualquier equipo puede ser el servidor o el cliente. El cliente es la entidad que solicita la realización de una tarea, el servidor es quien la realiza en nombre del cliente. Éste es el caso de aplicaciones de acceso a bases de datos, en las cuales las estaciones ejecutan las tareas del interfaz de usuario (pantallas de entrada de datos o consultas, listados, etc.) y el servidor realiza las actualizaciones y recuperaciones de datos en la base.

Redes de pares

Este modelo permite la comunicación entre usuarios (estaciones) directamente sin tener que pasar por un equipo central para la transferencia. Todos los equipos conectados pueden desempeñar el papel de servidor y de estación de trabajo al mismo tiempo. En este caso, si alguien quisiera compartir un recurso, podría ofrecerlo a los demás (incluso, por ejemplo, su disco duro) o utilizar los recursos ofrecidos por otro ordenador. Éste es un tipo de red para trabajos simples, donde el volumen de información intercambiado es pequeño y la seguridad no es un factor crítico.

COMUNICACIÓN DE APLICACIONES EN RED

El protocolo es un conjunto de reglas que permite que uno o más recursos intercambien información utilizando la red física (cables y placas de comunicación). Dentro de estas reglas existen formas por las cuales se puede identificar y distinguir cada uno de los recursos y lo que ellos pueden ofrecer a todos los demás recursos de la red.

Un sistema operativo puede soportar varios protocolos, pero solamente los dispositivos que utilizan el mismo protocolo pueden comunicarse entre sí.

Cuando se conecta un ordenador a la red (utilizando una tarjeta NIC, módem), el ordenador asocia automáticamente un protocolo con dicho dispositivo. El protocolo asociado por defecto con el dispositivo dependerá del sistema operativo instalado en el ordenador.

Por ejemplo, Windows 95 instala por defecto el protocolo NetBEUI y Windows 98, NT, 2000 y XP instalan por defecto el protocolo TCP/IP.

Si unos ordenadores utilizan NetBEUI y otros utilizan TCP/IP, se tendrán dos redes diferentes. Los ordenadores que utilizan NetBEUI podrán reconocer y comunicar solamente con otros ordenadores que utilicen NetBEUI. Los ordenadores que utilizan TCP/IP podrán comunicar solamente con otros ordenadores que utilicen TCP/IP.

Para solucionar este problema, hay que asegurarse de que todos los ordenadores de la red utilizan el mismo protocolo.

Más adelante se dará una explicación de cómo configurar TCP/IP.

El modelo de referencia OSI

El modelo de referencia OSI ofrece varias funciones a la comunidad que participa del **internetworking:**

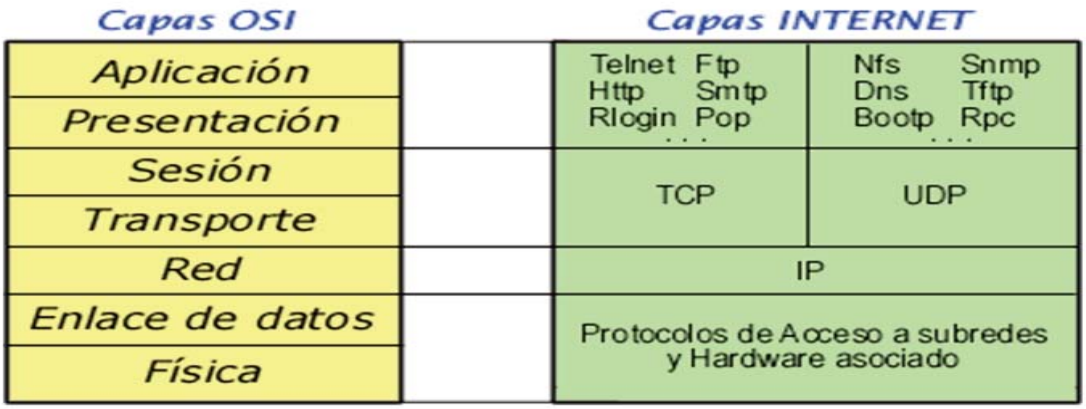
- Proporciona una forma de entender cómo opera un **internetworking** de redes.
- Sirve de guía o marco de trabajo para crear e implementar estándares de red, dispositivos y esquemas de **internetworking.**

Estas son algunas de las ventajas de utilizar un modelo estructurado en capas.

- Separa la compleja operación de **internetworking** en elementos más simples.
- Permite a los ingenieros centrarse en el diseño y desarrollo de funciones modulares.
- Proporciona la posibilidad de definir interfaces estándar para compatibilidad “plug-and-play” e integración multifabricante.

El modelo de referencia OSI consta de siete capas.

Las cuatro capas de nivel inferior definen rutas para que los puestos finales puedan conectarse unos con otros y poder intercambiar datos. Las tres capas superiores definen cómo han de comunicarse las aplicaciones de los puestos de trabajo finales entre ellas y con los usuarios.



Descripción de las capas OSI:

Capa de aplicación. Es la capa de nivel superior del modelo.

Aquí, el usuario o la aplicación dialogan con los protocolos para acceder a la red. Por ejemplo, se accede a un procesador de textos por el servicio de transferencia de archivos de esta capa.

Capa de presentación. La capa de presentación proporciona diversas funciones de conversión y codificación que se aplican a los datos de la capa de aplicación. Estas funciones aseguran que los datos enviados desde la capa de aplicación de un sistema podrán ser leídos por la capa de aplicación de otro sistema. Un ejemplo de funciones de codificación sería el cifrado de datos una vez que éstos salen de una aplicación. Otro ejemplo podrían ser los formatos de imágenes jpeg y gif que se muestran en páginas Web. Este formato asegura que todos los navegadores web podrán mostrar las imágenes, con independencia del sistema operativo utilizado.

Capa de sesión. La capa de sesión es la responsable de establecer, administrar y concluir las sesiones de comunicaciones entre entidades de la capa de presentación. La comunicación en esta capa consiste en peticiones de servicios y respuestas entre aplicaciones ubicadas en diferentes dispositivos. Un ejemplo de este tipo de coordinación podría ser el que tiene lugar entre un servidor y un cliente de base de datos.

Capa de trasporte

En este nivel se realiza y se garantiza la calidad de la comunicación, ya que asegura la integridad de los datos. Es aquí donde se realizan las retransmisiones cuando la información fue corrompida o porque alguna trama (del nivel 2) detectó errores en el formato y se requiere volver a enviar el paquete o datagrama.

El nivel de transporte notifica a las capas superiores si se está logrando la calidad requerida. Este nivel utiliza reconocimientos, números de secuencia y control de flujo.

Capa de red

Es la encargada de preparar la información codificada en forma binaria en formatos previamente definidos por el protocolo a utilizar.

Tiene su aplicación en el contexto de redes WAN y LAN ya que como se estableció previamente la transmisión de datos no es mas que el envió en forma ordenada de bits de información. Podríamos de hecho concebir a ésta como una cadena de bits que marchan en una fila inmensa (para el caso de transmisiones seriales), cadena que carece de significado hasta el momento en que las señales binarias se agrupan bajo reglas, a fin de permitir su interpretación en el lado receptor de una manera constante.

Capa de enlace de datos

En el nivel de enlace de datos se lleva a cabo el direccionamiento físico de la información; es decir, se leerán los encabezados que definen las direcciones de los nodos (para el caso WAN) o de los segmentos (para el caso LAN) por donde viajarán las tramas. Decimos que son direcciones físicas ya que las direcciones lógicas o de la aplicación que pretendemos transmitir serán direccionadas o enrutadas en el nivel de red

En este nivel de enlace sólo se da tratamiento a las direcciones MAC (Media Access Control) para el caso de LAN y a las direcciones de las tramas síncronas como HDLC (High-Level Data Link Control), SDLC (Synchronous Data Link Control, de IBM), LAP B (Link Access Procedure Balance) por citar algunos para el caso WAN.

Capa física

Es el primer nivel del modelo OSI y en él se definen y reglamentan todas las características físicas-mecánicas y eléctricas que debe cumplir el sistema para poder operar. Como es el nivel más bajo, es el que se va a encargar de las comunicaciones físicas entre dispositivos y de cuidar su correcta operación. Es bien sabido que la información computarizada es procesada y transmitida en forma digital siendo esta de bits: 1 y 0.

El Protocolo IP

Introducción

El protocolo IP es no fiable y sin conexión. No es fiable ya que se envía un paquete pero puede ser que llegue o no al destinatario. No tiene conexión ya que cuando se envía el mensaje no se establece.

El protocolo IP se diseño para unir diferentes tipos de redes. A estas redes se les denomina redes LAN (Local Area Network). Las LAN son redes privadas de corto alcance y bastante rápidas.

Para interconectar estas redes (LAN) se utilizan las redes WAN (Wide Area Network). Internet se puede considerar una red WAN.

En estas redes (LAN) habrá un router que conectará la red LAN con Internet (WAN). El router va a tener dos o más conexiones de red. Una conexión irá a la red local y otra conexión para el exterior.

A cada ordenador se le va a dar una dirección diferente de IP. El tamaño de la dirección IP es de 32 bits, se pueden llegar a hacer 4.000.000.000 combinaciones (número máximo de ordenadores conectados a la red). No se pueden aprovechar todas las direcciones IP ya que el reparto inicial fue erróneo.

Para solucionar el problema de la asignación de las direcciones se pasará a la versión siguiente de IP (IPv6), pero esta migración generará que tengan que evolucionar los Sistemas Operativos ya que no pueden funcionar con esta nueva dirección que es más grande (aproximadamente 128 bits).

Clases de Direcciones IP

A	0	IDRed (07)	IDHost(24)
B	1 0	IDRed(14)	IDHost(16)
C	1 1 0	IDRed(21)	IDHost(08)
D	1 1 1 0	Multicast a un GroupID	(28)
E	1 1 1 1 0	Reservado para el futuro	(27)

El IDRed representa el código de la red local y quien lo asigna es INTERNIC.

El IDHost es un número que representa a cada uno de los ordenadores de la LAN y quien lo asigna es el administrador de la LAN.

La clase A habrá 2->7 = 128 LAN y2->24 = 16.000.000 Hosts / LAN. Se asignaron al principio y están organismos como la NASA o el FBI y empresas como America On Line. Rango de direcciones: 0.. 127.

La clase B habrá 2->14 = 16.000 LAN y 2 -> 16 = 65.000 Hosts / LAN. Se destinan a proveedores de acceso a Internet y a universidades. Rango de direcciones: 128 .. 191

La clase C habrá 2—>21 = 2.000.000 LAN y 2 ->8 = 256 Hosts / LAN. Se destinan a proveedores de servicios de Internet. Rango de direcciones: 192 .. 223

La clase D se utiliza para Multicast que consiste en enviar un mismo paquete a muchos destinatarios. Se utiliza para radio y televisión por Internet. Rango de direcciones: 224 .. 239

La clase E su rango de direcciones será: 240 .. 247.

Realmente en las empresas sólo existe un router con una dirección IP real mientras que el resto de los ordenadores que están dentro de la LAN tendrán direcciones falsas.

En función de la dirección IP se podrá saber qué tipo de red se está conectando.

Los Sistemas Operativos no están preparados para admitir direcciones IP por debajo de 128 y superiores a 247.

Existe también un grupo de direcciones especiales como las siguientes:

255.255.255.255	Es la dirección denominada de broadcast. Es la dirección que se utiliza para enviar un mensaje a todos los ordenadores de la red local, pero el router nunca va dejar que salga fuera.
IDRed 255.255	Sirve para realizar un broadcast a otra red local indicada por el IDRed.
0.0 IDHost	El Host de la red local (LAN) con el que se quiere hablar. Nunca sale hacia afuera.

0.0.0.0	Sirve para indicar que la dirección IP no es útil o no procede.
127.0.0.x (0..255)	Interfaz de loopback. A esta dirección se la denomina LocalHost. Tiene la característica especial de que todo mensaje enviado a esta dirección lo devuelve al mismo ordenador. Esto sirve sobre todo para realizar un cliente y un ordenador en una misma máquina. También se utiliza para comprobar si el TCP/IP está bien instalado. Para realizar estas pruebas se puede utiliza el comando ping.

Subredes

Permiten solucionar básicamente dos problemas:

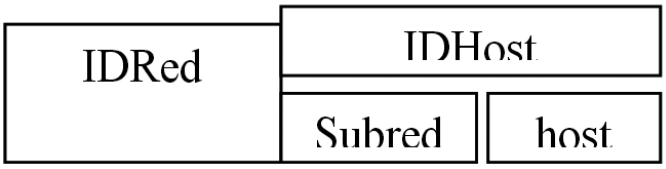
- Agrupar varias LAN arquitectónicamente distintas bajo un mismo Id de red.

- Dividir una LAN grande (donde hay muchas colisiones) en varias redes (subredes) más pequeñas.

Las subredes surgen después de la construcción del protocolo IP. Se puede encontrar dentro del RFC-950. Al ser más modernas no todos los Sistemas Operativos las soportan.

En las subredes surge el problema de que originariamente las direcciones estaban constituidas por un IDRed + IDHost, para diferenciar el IDRed y el IDHost debe tener algún sistema para diferenciar los tipos de Id. Esto se hacia a partir de las clases de direcciones IP. Por lo tanto el Sistema Operativo lo tenía fácil para diferenciar los Id utilizando las clases de direcciones IP.

El problema que se encontraron fue a la hora de dividir una LAN grande en varias subredes diferentes ya que el Sistema Operativo se confundía. Lo solucionaron utilizando una máscara de subred que determina si la dirección IP está en la misma subred o en una subred diferente. La dirección IP se modifica para estos casos:



De tal forma que ahora no sólo tiene que coincidir el número de IDRed sino el número de Subred para saber si un PC está en la misma Subred.

La máscara de Subred es un número que en la parte de red se van a poner valores de unos y valores 0 para la parte del host: 255.255.255.0.

El Protocolo UDP

Introducción

UDP (User Datagram Protocol) pertenece a los protocolos de transporte junto con TCP (Transport Control Protocol).

Diferencias entre TCP y UDP

TCP es un protocolo de flujo de bytes fiable, orientado a conexión y con control de flujo.

Es fiable ya que se va a encargar de comprobar que los datos llegan al destino. Si no llegan los devuelve al origen.

Es orientado a conexión, se va a establecer la conexión antes de enviar datos y está conexión queda establecida hasta que se cierra la conexión.

Es un protocolo de flujo de bytes ya que lo se envía y recibe son bytes en un flujo denominado Stream.

TCP tiene las siguientes responsabilidades:

- Secuencialización: según se reciben segmentos se ordenan antes de entregarlos al destino.

- Confirmación y control de flujo: TCP es el que se encarga de enviar las confirmaciones al emisor para que el emisor sepa que han llegado los datos. El control de flujo consiste en evitar que nos llegan demasiados datos que lleguen a colapsar al receptor.

- Entrega a la aplicación correcta: TCP se encarga a partir de la información de control de entregar los datos a la aplicación destino.

UDP transporta la información en datagramas que son paquetes sueltos, es un protocolo no fiable ya que no confirma la llegada de la información al destino.

UDP es un protocolo sin conexión.

UDP es un protocolo sin control de flujo, se puede llegar a saturar al receptor.

UDP se utiliza básicamente en dos casos:

- Para realizar pequeñas consultas al servidor (Servidores de DNS, ...)

- También para multimedia en tiempo real.

Los números de puerto

Tanto TCP y UDP utilizan el concepto de número de puerto para saber cuál es la aplicación destino que debe recibir el mensaje.

Para identificar la aplicación destino se va a utilizar el siguiente formato: Dirección IP + Puerto.

Los servidores utilizan los puertos a partir del rango 0 a 1024. Desde el puerto 1025 en adelante se utilizará para aplicaciones clientes.

Respecto a los puertos de los servidores están estandarizados por IANA. También son conocidos estos puertos con el nombre 'Well Knows Ports'. Todos estos puertos se encuentran en la RFC-1700.

Exiten algunos puertos conocidos. Son los siguientes:

FTP	Puerto 21
Telnet	Puerto 23
SMTP	Puerto 25
POP3	Puerto 110
HTTP	Puerto 80
DNS	Puerto 53

En los sistemas UNIX únicamente el superusuario puede instalar los puertos de servidores para evitar los Caballos de Troya.

Esta tabla indicada anteriormente es igual tanto para UDP como para TCP.

Estos puertos aparecen en un fichero:

- WINDOW c:\windows\services
- UNIX/etc/services

Los puertos altos también son denominados como 'Anonymous Ports' también se les denomina 'Ephimerals Ports'.

Los puede utilizar cualquier usuario.

Cliente (170.181.12.14)	**Servidor (53.81.15.7)**
Dir IP Origen: 170.181.12.14	Dir IP Origen 53.81.15.7
Dir IP Destino:53.81.15.7	Dir IP Destino 170.181.12.14
Puerto Origen:1300 (Puerto Anónimo)	Puerto Origen: 53 (DNS)
Puerto Destino: 53 (DNS)	Puerto Destino: 1300 (Puerto Cliente)
Consulta: www.microsoft.com	Solución: 178.142.12.4

Tipos de Servidores UDP

Un servidor iterativo se implementa como un bucle que se dedica a recoger una consulta, resolverla y devolver la solución al cliente y se vuelve a repetir el bucle. Son fáciles de implementar, pero tienen el inconveniente de que si la consulta del cliente dura mucho tiempo el hilo del servidor se queda bloqueado mucho tiempo y no puede atender a otros clientes. (UDP)

Los servidores concurrentes cada vez que llega un cliente se crea un hilo que se encarga de atender a ese cliente. Es útil para peticiones largas. (TCP)

En servidores UDP se encuentra el *wildcard*, es decir, el servidor de UDP recibe peticiones por todas las interfaces. También existe el *multicast* que consiste en enviar un mismo mensaje a varios destinatarios. También está el *broadcast* que consiste en enviar un mensaje en toda una red local (255.255.255.255). Por último esta el *unicast*, que consiste en enviar un mensaje a un único destinatario (UDP y TCP).

El Protocolo TCP

Introducción

TCP es un protocolo full-duplex (se puede enviar y recibir a la vez), dos aplicaciones pueden estar leyendo y escribiendo a la vez en el socket (conector, canal de comunicación).

TCP no predefine un tamaño (ni mínimo ni máximo) de bytes que se puedan enviar, sino que con TCP se puede enviar el número de bytes que se quiera.

Los segmentos de TCP deben ser confirmados mediante ACK (Acknowlenge). ACK que consiste en la confirmación de la llegada de los segmentos al receptor.

En TCP no existe el concepto del NAT como existe en otros protocolos. NAT consiste en que si un segmento llega defectuoso se le pide al emisor que vuelva a enviarlo. En TCP si el segmento llega defectuoso se espera a que el emisor vuelva a enviarlo de nuevo.

El último concepto asociado a TCP es el de ventana deslizante (Sliding Windows). Consiste en que se envían varios segmentos sin esperar confirmación, pero se debe fijar una cantidad máxima de información que pude recibir el receptor para no saturarle. La cantidad máxima de información que se envía sin confirmación se denomina tamaño de la ventana.

El modelo de ventana deslizante permite enviar información al receptor de manera fluida, para ello el receptor tiene que tener un tamaño de ventana aceptable. Lo óptimo es que no se produzcan paradas (Stall) en el receptor para que el envío de información sea fluido.

Para TCP el modelo de ventana deslizante se interpreta de forma distinta, el tamaño de ventana se mide en número de bytes y no en número de segmentos pendientes de confirmar.

La cabecera del segmento T

Toda la información de control del protocolo TCP se encuentra en la cabecera del segmento.

Formato:

Cabecera IP Cabecera TCPDatos

Puerto origen	Puerto destino	Nro Secuencia	ACK Number	HLEN	Reservado	Bts de control	Tamaño ventana	Checksum	URG Pointer	Opciones
16b	16b	32b	32b	4b	6b	6b	16b	16b	16b	n b

- Los puertos origen y destino identifican las aplicaciones emisoras y receptoras.

- Número de secuencia indica el offset del primer byte de datos del segmento respecto al inicio del flujo TCP.

- ACK Number es un campo que debe ser comprobado cuando el flag de ACK esté activo y contiene el número de secuencia del flujo de datos TCP respecto al inicio del flujo que espera recibir del interlocutor. Sirve para confirmar datos que se han recibido en el destino.

- HLEN contiene el tamaño de la cabecera.

- Reservado para futuros casos.

- Bits de Control indica si están activas o no unas determinadas opciones o flags , que son las siguientes:

    - URG, para el envío de datos urgentes.
    - ACK, para indicar que ACK Number tiene valor de confirmación.
    - PSH (Push), indica que los datos no se almacenen en el buffer si no que se entreguen en la aplicación en cuanto lleguen.
    - RST (Reset), sirve para reiniciar una conexión.
    - SYN, para iniciar la conexión.
    - FIN, para finalizar la conexión.

- Tamaño de ventana, se utiliza para control de flujo. Indicar, al receptor, cuántos bytes le quedan libre en el buffer. Entonces el emisor como máximo enviará la cantidad de información indicada por el receptor.

- Checksum, control de problemas.

- URG Pointer, se utiliza para indicar datos urgentes. Son datos que se quieren enviar antes que el resto de los datos que lleguen al receptor. También se les denomina: *out of band data*.

Para el envío de datos urgentes se activa el flag de URG y en URG Pointer se le indica la primera posición de información normal después de los datos urgentes respecto al inicio del segmento.

Establecimiento y cierre de una conexión en TCP

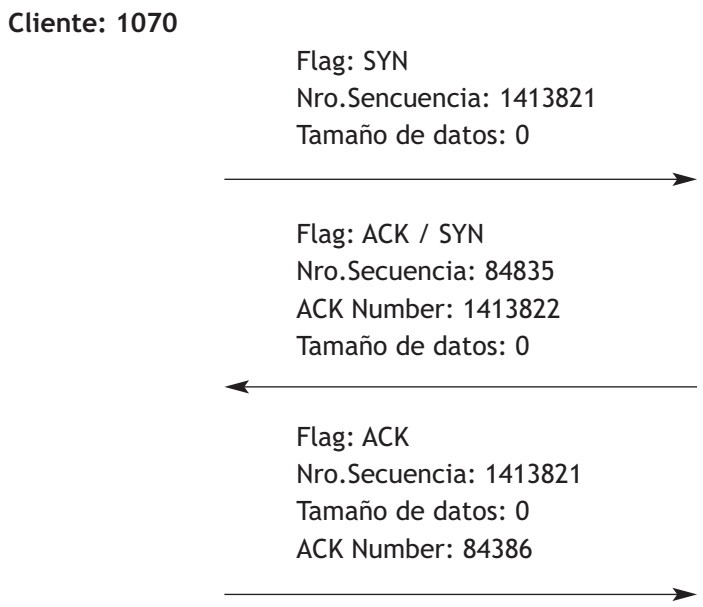
- Establecimiento

Se utiliza un algoritmo denominado *three steps handshake*. Para ello la aplicación del servidor se pone a esperar en un puerto indicando en qué puerto se está esperando (*pasive open*). Para esto se llama a la operación listen() del TCP. El hilo del servidor queda dormido hasta que alguien se conecta.

Más tarde en algún momento el cliente se conectará al servidor enviando un segmento TCP formado únicamente por la cabecera de TCP y con el flag de SYN activo indicando que quiere abrir una conexión (*active open*). También se rellena el campo número de secuencia, lo rellena con el ISN (*Initial Sequence Number*), este número es un número consecutivo, se hace así para evitar un efecto lateral de que se abra una conexión y se cierre inmediatamente y se abra de nuevo. Si se hace esto alguno de los elementos de la primera conexión se podría retrasar y llegar durante la segunda conexión con lo cual se mezclaría la información de la primera y la segunda conexión (ambas van dirigidas a la misma dirección IP y al mismo puerto) y no habría forma de separar la información.

El servidor posteriormente envía un nuevo segmento activando los flags SYN y ACK. En el campo ACK Number se introduce el ISN del cliente + 1. Se pone +1 para saber hasta cuál se confirma. Únicamente se envía la cabecera de TCP, no hay datos en el segmento. Además el servidor da un ISN para su conexión que no tiene que coincidir con el ISN del cliente, esto es porque son dos comunicaciones distintas.

El cliente manda un tercer mensaje con el flag de ACK activo y en el ACK Number confirma el ISN del servidor + 1.



- Cierre:

El cierre de conexión no es obligatorio pero sí recomendable para liberar los recursos.

La forma de cerrar la conexión se denomina *two ways handshake*. Se denomina de esta forma ya que hay dos aplicaciones conectadas, no existe cliente-servidor.

Cuando se cierra la conexión que va desde A hasta B, ya no se pueden mandar datos desde A hasta B pero sí en el otro sentido. Esto se denomina *half close*. Y cuando se cierra la conexión desde el sentido B a A, ya no queda conexión abierta, a este estado se le denomina *full close*.

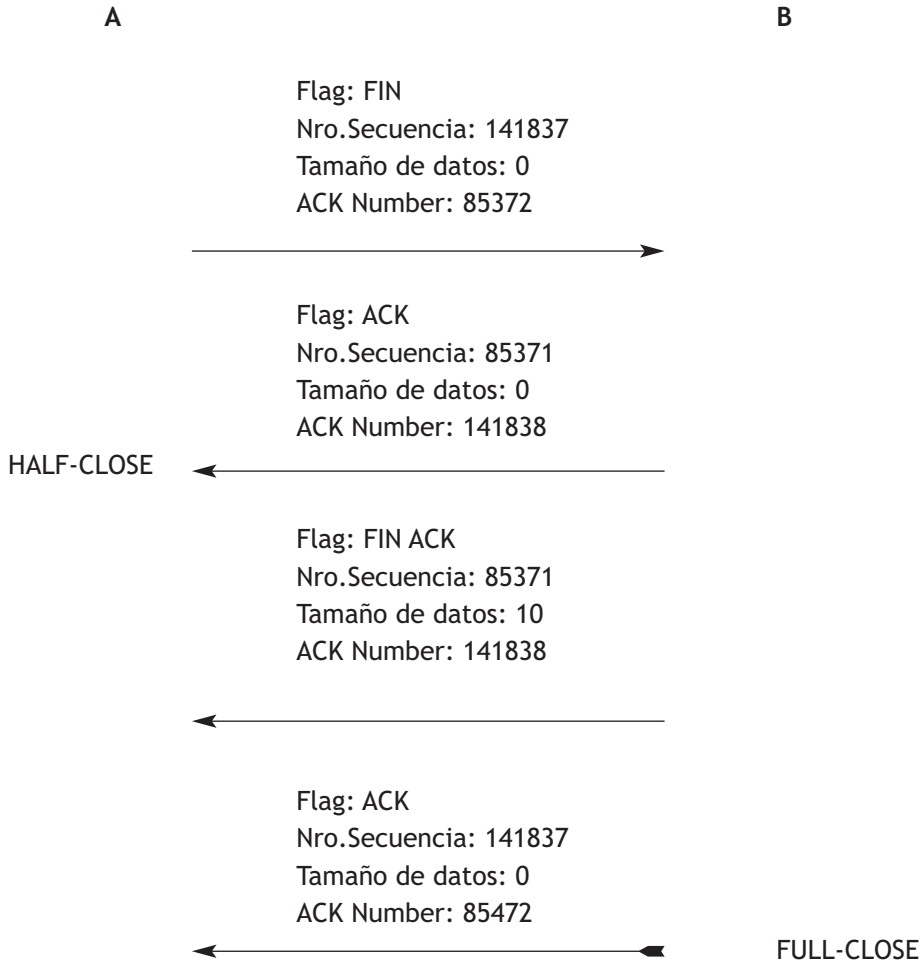
A manda cerrar la conexión con B enviando todos los datos que queden en el buffer y el flag FIN activado. Si no hubiese datos en el buffer se mandaría un segmento aislado con el flag FIN activado.

Cuando el software TCP recibe este segmento tiene que informar a la aplicación de que B ha finalizado la conexión. Surge el problema de que la aplicación puede tardar un tiempo en responder, el software de TCP manda un segmento de confirmación antes de informar a la aplicación. Este segmento va a llevar el flag de ACK activo y el campo del ACK Number. Se confirmarán todos los bytes de A.

Más tarde la aplicación de B pedirá más datos a TCP y TCP entregará los datos que queden en el buffer y la información de cierre de la comunicación.

Cuando la aplicación de B vea que no quedan más datos enviará otro segmento con el flag del FIN y ACK activados. Le envía datos, ya que la aplicación B puede ser que no haya enviado toda la información que debería dar a A

Para confirmar el cierre A mandará un último segmento con el flag ACK activado.


A
B
Flag: FIN
Nro.Secuencia: 141837
Tamaño de datos: 0
ACK Number: 85372
Flag: ACK
Nro.Secuencia: 85371
Tamaño de datos: 0
ACK Number: 141838
HALF-CLOSE
Flag: FIN ACK
Nro.Secuencia: 85371
Tamaño de datos: 10
ACK Number: 141838
Flag: ACK
Nro.Secuencia: 141837
Tamaño de datos: 0
ACK Number: 85472
FULL-CLOSE