REDES

- **SISTEMAS CERRADOS Y ABIERTOS**

En los 60', empresas como IBM o DEC, cubrían su cuota de mercado de manera integral: pantallas, teclados, hardware, conectores, etc. esto daba fiabilidad, pero encarecía mucho el producto. Éran las arquitecturas cerradas. A primeros de los 70', Con la bajada de costes de los productos informáticos, la creación de un estandar para LAN, la aparición de SO adaptables a múltiples plataformas (Unix), etc, nacieron las primeras redes compatibles, y con ellas, los sistemas abiertos.

- **MODELO DE REFERENCIA OSI vs MODELO INTERNET**

Aplicación
TCP(Transporte)
IP
Subred

Aplicación
Presentación
Sesión
Transporte
Red
Enlace
Físico

OSI: Es una torre que tiene 7 niveles Los 4 primeros son los niveles altos o de aplicación, y los 3 últimos los bajos o de red. Éstos, los de red, están implementados en todos los nodos, mientras los altos, solo en el origen y el destino.

INTERNET: Los niveles de presentación y sesión no aparecen, el IP sería una parte del del red en OSI, y el de subred englobaría el físico y de enlace, y parte del de red OSI. Sólo los niveles IP y de subrede deben estár implementados en los nodos de enlace, lo que abarata mucho los costes y agiliza el tráfico. En la red X.25, p. ej. el de transporte tb. debía implementarse en los enlaces, lo que daba más fiabilidad, más coste.

- **TRANSMISIÓN DE DATOS**

El mundo real, se nos presenta con parámetros analógicos (colores, posición exacta, etc), que sólo podríamos aspirar a representarlos como valores reales (IR), nunca exactos. Por eso, deberemos usar la representación digital, discreta, en la que siempre existirán márgenes de error.

Un modelo sencillo de comunicaciones, seria:

ORIGEN MEDIO DE DESTINO

TRANSMISIÓN

CIRCUITO DE DATOS

El medio de transmisión:

Está gestionado por los dos niveles inferiores del modelo OSI. La señal, sufre modificaciones a lo largo de su paso por el medio, debido a factores que la perturban. Estos factores, son la atenuación (debilitamiento de la señal, que mantiene sin embargo su forma. Los amplificadores nos ayudan a resolver el problema), la distorsión (la señal pierde su forma original), y el ruido.

Tipos de medios. Hay dos clasificaciones:

- Conductores (Se les aplica tensión o corriente / Dieléctricos (sensibles a ondas electromagnéticas).
- Medios de transmisión no guiados (aire, vacio, para LAN sin hilos, telefonía movil, mandos a distancia, etc.), o guiados (par, par trenzado, par apantallado, coaxial, fibra óptica).

Circuito de datos

Puede ser uni o bidireccional (simplex o duplex). A los cod−descod de fuente, que tratan y representan los datos, se les llama DTE (*Data terminal equipement*), y a los cod−decod de canal, que convierten los datos en señales y viceversa, DCE (*Circuit−terminating*) El Dte envía señales digitales al DCE, que las transforma en analógicas para que puedan viajar por el medio. Esto nos hace ver dos conceptos nuevos:

Velocidad de transmisión (*transmisión rate*): se mide en bps, y está asociada a la línea del circuito de datos DTE (digital).

Velocidad de modulación: Se asocia al la línea de transmisión DCE analógica (número de señales de n bits (símbolos) que se generan, se reciben o se transmiten por segundo), se mide en baudios (símbolos /s).

Tipos de codificaciones: codificaciones digitales, que generan señales digitales(NRZ, Manchester y bipolar) y modulaciones digitales, que las generan analógicas.

Multiplexación: mecanismo que permite compartir un medio de transmisión único a circuitos de datos diferentes. Puede ser por división de frecuencia o de tiempo.

Interfaz DTE−DCE: Pueden ser de transmisión en serie o en paralelo; síncronas (si tienen un reloj dedicado) o asíncronas (si no lo tienen); con velocidades de transmisión bajas (10 kbps, usaríamos la RS−232), o altas (SCSI, USB)

- **ENLACE DE DATOS (DLC)**

Es el nivel que hay por encima del nivel físico OSI, el cual se encarga del transporte no fiable de bits, por lo que el nivel de enlace tendrá como primera tarea el control de errores que se produce en la transmisión de bits a nivel físico. Para ello, deberá destinar una parte de los bits que se intercambian los protocolos de transmisión (DLC), a la detección y posterior gestión de esos errores. Para lograrlo, organiza los datos en TRAMAS, estructuras de bits organizadas en campos, que indican el inicio de trama, el final, etc.

Detección de errores

CRC. Códigos detectores de errores muy utilizados, que calculan un número binario, llamado CRC, a partir de los bits a proteger. En el destino, se repite el cálculo para ver si coincide o no con el CRC.

Hay también códigos que además corrigen los errores (técnica FEC), en vez de reenviar la trama.

- **REDES DE AREA LOCAL (LAN)**

Para evitar el cableado entre todas las estaciones de una red, se recurre a la multiplexación, donde un solo

cable transporta diversas comunicaciones, a través de centralitas o nodos, que también multiplexan, y en los extremos se demultiplexa (un ejemplo es la red telefónica).

Esta solución era muy cara, por lo que las primeras LAN se conectaban a un único medio compartido por todos los terminales (llamadas redes por difusión o en *broadcast*, porque todas las estaciones reciben una copia de la información transmitida por otra estación), donde se definen unas características de transmisión (tipo de cable, codificación, distancias máximas, etc.), una política de acceso y una topología física (bus, anillo, estrella). Para poder interconectar diversas LAN, necesitaríamos acudir la la filosofía de la red telefónica, y usar un conmutador que multiplexe las diferentes conexiones, busque el camino óptimo y asigne direcciones para identificar destinatarios.

Estaríamos aquí en las WAN.

Control de acceso al medio (MAC)

Es un mecanismo que decide qué estación tiene acceso al medio de transmisión para emitir una trama de información. En la trama MAC (unidad de información en este nivel), deben ir indicadas las estaciones de origen y destino, y adjuntarlas a los datos que nos da el nivel superior, par aque las estaciones puedan deducir si la trama es para ellas o no. Se han propuesto gran cantidad de protocolos de acceso al medio para todo tipo de redes, siendo importante el tema hoy en dia en las comunicaciones móviles, pero el más habitual en las LAN, es el acceso múltiple por escucha de portadora (CSMA, Carrier Sense Multiple Access), que escucha el medio y aplica un algoritmo determinado (si está libre, o transmite inmediatamente, o lo hace con una probabilidad *p*, y si está ocupado, espera a que esté libre para transmitir, o bien espera un tiempo aleatorio), con el fin de evitar en lo máximo las colisiones (transmisión de tramas simultánea en el tiempo). Una variante mas actual es el CSMA/CD (Colision Detect), que al detectar una colisión, deja de transmitir, y avisa al resto de estaciones con una señal interferente.

Las LAN más extendidas son las Ethernet con acceso al medio CSMA/CD, seguidas de las Token Ring a distancia. La tendencia es a usar una topología física en estrella, que facilita el cableado, junto con un concentrador o Hub que puede ser compartido (*shared hub*) y trabajar lógicamente (internamente) en bus (concentrador por difusión o *broadcasting hub*) o en anillo, o puede ser por conmutación (*switched hub*), que filtra la dirección destino y sólo a ella le transmite la trama.

- **REDES DE AREA AMPLIA (WAN)**

Siguen otra filosofía: son redes de conmutación, con dos tipos de nodos, los de conmutación (por el camino) y los de acceso (junto a las estaciones). La distinción no siempre es importante. Las redes de conmutación pueden ser de dos tipos: de conmutación de circuitos y de conmutación de paquetes.

Redes de conmutación de circuitos

Establecen a traves de la red, un camino físico continuo entre las dos estaciones que se quieren comunicar, como si fuera una línea punto a punto. Hasta que no se ha establecido ese camino no se inicia la transmisión (ej. red telefónica). Con alta carga, estas redes presentan bloqueos.

Redes de conmutación de paquetes

Nacieron como una alternativa a la anterior, con los objetivos de conseguir mayores niveles de eficiencia en el uso de recursos (aquí no son dedicados), poder interconectar dispositivos que trabajen a diferentes velocidades, y poder establecer diferentes conexiones simultaneas de manera flexible. La transmisión se realiza subdividiendo la información en paquetes, y se distinguen dos variantes:

- Circuitos virtuales: el camino a seguir se establece previamente a la transmisión, y es conocido. Tiene la desventaja, de que si un nodo deja de funcionar, el origen ha de encargarse de establecer otra vez el circuito para continuar la transmisión
- Datagramas: el camino no se conoce, y se decidirá en cada nodo en función del tráfico. Cada paquete se transmite independientemente al resto, por lo que en función de los retardos, pueden llegar desordenados al destino, que deberá ordenarlos (con el TCP, por ejemplo). Este método tiene la ventaja de que si un nodo deja de funcionar, la red eligirá otro camino. Los criterios de encaminamiento pueden ser en función de la distancia, o del coste.

El protocolo IP sigue la técnica de datagramas (se suele llamar servicio no orientado a la conexión), encargándose el nivel superior (TCP) en las estaciones, del control de errores y reordenación de paquetes.

Ejemplos de WAN

RTC: Red Telefónica Conmutada

RDSI: puede dar acceso a una red de conmutación de circuitos para voz, a una red de paquetes para datos, y enlaces punto a punto para conexiones permanentes.

El estándar X.25: ofrece servicios virtuales conmutados, y permanentes (con origen y destinos prefijados y asignados por el operador)

Protocolo IP: No es una WAN, sino una manera de interconectar diversas redes. Como podemos tener el protocolo IP en cualquier nivel de enlace, esto permite interconectar redes muy diferentes en topología y protocolos.

Interconexión de redes

Para conectar redes locales, la pieza clave es el puente (*bridge*) o si la LAN es conmutada, el conutador (*switch*). Si conectamos una LAN a una WAN, necesitaremos un encaminador o *router*, que forma parte de la WAN y realiza las tareas de encaminamiento.

Para conectar un ordenador a traves de una WAN, nos conectamos mediante un modem y un número de teléfono a uno de los modems del encaminador (ISP, proveedor de internet). A partir del establecimiento de la conexión, la estación entra a formar parte de la red de conmutación de paquetes y puede conectarse con otras máquinas conectadas a esta red. El modem cambia el soporte físico (de bits a tonos sobre la línea telefónica) y nada más.

INTERNET

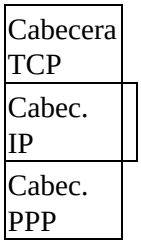
- **ESTRUCTURA DE PROTOCOLOS EN INTERNET**

NIVELES CARACTERÍSTICAS BÁSICAS DEL MODELO INTERNET

APLICACIÓN	Clientes y servidores WWW, e−mail, FTP, etc. Sólo están en las estaciones Protocolos: HTTP, Telnet, SMTP, (SNMP)
TCP	Nivel de transporte. Controla errores y ordena paquetes. Los routers no lo necesitan Protocolos: TCP, (UDP)
IP	Nivel Internet. Da unidad a la red. Direcciona y encamina. Está en toda los equipos

	Protocolos: IP, (ICMP, ARP)
RED	LAN o WAN punto a punto. Red física. Driver de red, Tarjeta de red Protocolos: Ethernet, PPP (point to point protocol)

Los protocolos, funcionan todos con unas estructuras básicas conocidas como PDU (Protocol Data Units), que en Ethernet o PPP son llamadas tramas, en IP y ARP paquetes, en TCP segmentos y en UDP datagramas. El encapsulamiento consiste en que cada protocolo superior, engloba al inferior (pirámide PDU: para enviar un byte de datos, se necesita un segmento TCP de 21 B, un paquete IP de 20 más y una trama PPP de 8 más). Veamos la llamada pirámide PDU:



Datos (1 byte)

Segmento TCP

Paquete IP

Trama PPP

- **REDES DE ACCESO A INTERNET**

Red telefónica conmutada (RTC)

Utiliza el protocolo PPP. Las tramas PPP pueden transportar diferentes protocolos en ellas: LCP para el test de enlace, paquetes IP, NCP para gestionar los protocolos que transporta el enlace, etc.

Con la compresión de cabeceras, se soluciona el problema de la pirámide PDU, que para transmitir un bit usa un montón de ellos. Se basa en el hecho de que puede mantener una tabla de hasta 16 conexiones simultaneas TCP/IP, a las que asignará 16 indicadores diferentes. Como las cabeceras TCP/IP no varían mucho en una conexión, puede combinar PPP i TCP/IP, ganando en eficiencia.

Acceso LAN: el protocolo Ethernet

Es el más utilizado, más por su simplicidad que por sus prestaciones. Usa CSMA/CD como control de acceso al medio (escucha el medio, acceso aleatorio y detección de colisiones. No olvidemos que en las LAN, lo que una estación envía, lo reciben todas, en broadcast).

La trama Ethernet está formada unos 1500 bytes, de los que 48 son la dirección física MAC del origen, 48 la MAC del destino, 32 son CRC (control de errores), y entre 46 y 1400 son datos. Una variante de este protocolo es el IEEE802.3. Los medios físicos en Ethernet son actualmente el 10/100BaseT, en estrella y hub, el 10Base2 en coaxial y el 10BaseF sobre fibra óptica.

La dirección MAC universal (dirección física), en una Lan Ethernet, tiene un formato de 6 bytes, y se escribe en hexadecimal (ej. 08:00:00:10:97:00). La dirección broadcast, sería poner todos los bits a 1.

- **ELPROTOCOLO IP**

Es un protocolo de interconexión de red orientado a datagrama (ver Redes, punto 6), por lo que no dispone de circuito virtual, y no es capaz de recuperar tramas perdidas, ni garantizar el orden de recepción correcto, pero es fácil de implementar, y barato.

Las direcciones IP (direcciones lógicas), son únicas para cada máquina e interfaz, y son de 4bytes.

Las de clase A usan el primer byte para identificar la red, y los otros tres para la estación (0.0.0.0. a 127.255.255.255). Están agotadas.

Las de clase B usan dos bytes para la red, y dos para la estación (128.0.0.0 a 191.255.255.255). Quedan pocas.

Las de clase C tienen 3 bytes para la red, y el último para la estación (192.0.0.0 a 223.255.255.255).

Las de clase D, desde la 224.0.0.0 a la 239.255.255.255, son reservadas para el tráfico multicast IP.

Las de clase E, se reservan para ser asignadas a nuevos servicios.

Mascaras de red

Se usan para gestionar diversas direcciones IP, al configurar internamente diversas LAN, al no servir los identificadores de clase. Con la máscara, de 32 bits, identificaremos la red con los bits a 1, y la estación con los bits a 0. Así, el identificador de red será la porción de la dirección IP que encaja con los bits a 1 de la máscara.

Este concepto es fundamental, , pues permite decidir a una estación si el destino al que se envia un paquete se encuentra en la misma LAN o en una LAN remota, y por lo tanto, delegar la transmisión a otro equipo de su misma LAN (el Router), a cuya IP dirigirá el paquete.

Todas las estaciones de una misma LAN deben usar el mismo identificador de red, y todas sus estaciones, la misma máscara.

Ejemplo: 2 estaciones, con IP 147.83.153.100 y 147.83.153.200, estarán conectadas a la misma LAN, si su máscara es 255.255.255.0, pero no lo estarán si es 255.255.2545.128, por ejemplo.

Una notación alternativa de la máscara es: x.x.x.x./n, con n= nº de bits a 1 (ej. 147.83.153.100/24).

Direcciones de propósito especial

Dirección de red: la de la estación 147.83.153.100/24, sería con el último bit a cero, o sea, 147.83.153.0/24.

Dirección 127.0.0.1 (o 127.x.x.x) es la *loopback*, en la que el SO de vuelve los paquetes al destino (máquina local). No es válida para paquetes IP.

Dirección 255.255.255.255 (*broadcast* local), sólo válida como destino de un paquete. El cual transmite a todas las estaciones su misma LAN. Para enviar uno *broadcast* remoto (desde una estación remota, fuera de la LAN), a la red 147.83.153.0/24, lo haremos mediante la dirección 147.83.153.255.

Dirección 0.0.0.0: señala al ordenador que la envía. Se usa en paquetes iniciales cuando no hay una IP definida, hasta que un protocolo como el DHCP u otro asigna una. También para indicar una ruta por defecto a los programadores de gestión de encaminamiento.

Direcciones 10.0.0.0/8, 169.0.0.0/8, 172.16.0.0/16 a 172.31.0.0/16, y 192.168.0.0/24 a 192.168.255.0/16. Corresponden respectivamente a las redes de clase A, B, C, y nunca serán asignadas en Internet. Se utilizan en redes que no tiene previsto conectarse a Internet, pero que en caso de conectarse, no se quiere que sus direcciones se confundan con otras estaciones que sí están en Internet. Un ejemplo son las estaciones parcialmente conectadas mediante *firewalls* o *proxies*.

Encaminadores (Routers)

Interconectan dos o más subredes IP y se encargan de encaminar el tráfico destinado a estaciones remotas. Cada encaminador decide, basándose en la dirección destino que incorpora el paquete IP, el trozo de ruta en que participa, hasta el próximo encaminador (emisor y receptor del paquete, respectivamente).. Entre un encaminador y otro, puede haber una LAN, o una conexión punto a punto.

La tabla de encaminamiento

En el ejemplo siguiente, de una estación conectada a una LAN (147.83.153.103/24), con una tarjeta Ethernet.

Tabla de encaminamiento de la estación				
	Dirección	**Máscara de red**	**Encaminador (Puerta de enlace)**	**Interfaz**
1	147.183.153.103	255.255.255.255	127.0.0.1	Loopback
2	127.0.0.0	255.0.0.0	127.0.0.1	Loopback
3	147.83.153.0	255.255.255.0	147.83.153.103	Ethernet
4	255.255.255.255	255.255.255.255	147.83.153.103	Ethernet
5	0.0.0.0	0.0.0.0	147.83.153.5	Ethernet

En los casos 1 y 2, transmite paquetes IP a la la dirección local y a todas las que empiecen por 127, utilizando el encaminador o *gateway* (puerta de enlace) 127.0.01, que es el de *loopback*, o sea, devuelve los paquetes a la máquina local.

La tercera y cuarta entradas, las tomarán los paquetes destinado a a la máquina local y al broadcast, usando el *gateway* local (la IP local).

La quinta, es la ruta por defecto, que nos permite comunicarnos con estaciones remotas. La máscara tiene todos los bits a 1, y nos dice que todas las IP remotas irán por aquí, a través del gateway del encaminador, que suponemos en este ejemplo 147.83.153.5.

Para visualizar esta tabla, se usa el comando **route print** (route en UNIX). Ver anexo de comandos.

Ignoraremos las direcciones que empiecen con 255 o 224, y las que acaben con 255, pues son reservadas.

En un router, la tabla tiene lógicamente más entradas, pues conecta dos redes, pero funciona igual.

OTROS CONCEPTOS

Cojamos dicho Ejemplo y vamos a poner un problema: Supongamos que en nuestra Intranet, está en la red 169.254.x.x (direcciones directamente alcanzables desde nuestra tarjeta de red), y además tenemos otras direcciones, por ejemplo 169.100.x.x, conectadas físicamente a nuestra red.

Estas direcciones, en principio son inalcanzables, ya que si queremos llegar a ellas de alguna manera, El TCP se recorre nuestra tabla de rutas de abajo a arriba, no lo encuentra y entonces como en el ultimo nivel tiene 0.0.0.0 con el gateway, la dirección de nuestro modem, pues entonces intenta localizar las maquinas a las que

queremos llegar en Internet en vez de en nuestra intranet. Evidentemente no se puede.

Pues bien, podemos proceder a modificar la tabla de rutas. Si dais route /?. el comando os explicará la sintaxis, en nuestro caso particular, lo que queremos es añadir la ruta 169.100.x.x a través de nuestra tarjeta de red. Por tanto debemos dar el comando:

route add 169.100.0.0 mask 255.255.0.0 169.254.1.15

Fijaros que acabo de poner como "gateway" a la dirección IP de mi tarjeta "física" de red. Con esto cualquier petición a dicho tramo de nuestra red, ya será alcanzable.

Y ahora por fin, solucionemos en este caso un ultimo problema. Las maquinas en ese tramo, tienen un "nombre". Pero nosotros no podemos acceder por nombre debido a que nuestro DNS es nuestro proveedor de Internet vía modem, y este, no conoce las maquinas de nuestra intranet.

Entonces, si tenemos una maquina llamada ZEUS, en la dirección 169.100.1.10, no podremos hacer por ejemplo:

telnet ZEUS

pero en cambio, si que nos funcionará:

telnet 169.100.1.10

¿como podemos resolverlo?. Fácil también. Recordad que en el articulo anterior, habiamos dicho que nuestro PC, para resolver el tema de nombres, siempre busca primero en el fichero "hosts" del directorio de windows a ver si lo encuentra, y sino, se lo pide al DNS.

Por tanto si nos creamos un fichero hosts, o si ya existiese, si le añadimos a él la línea (con el notepad, por ejemplo):

169.100.1.10 ZEUS

ya podremos referirnos a ZEUS por su nombre. Nos lo encontrará.

- **PROTOCOLOS ARP Y ICMP**

El primero realiza la resolución automática del mapeado entre las direcciones físicas MAC, y las lógicas IP, mediante una tabla que las relaciona. El segundo gestiona las diferentes incidencias que pueden ocurrir en una red IP, y nos envía mensajes, que viajan dentro de los paquetes IP.

El programa ping

Permite descubrir s una estación está activa, simplemente haciendo: *ping <dirección_destino>* (vale también poner su URL, con letras, vaya). En realidad está enviando un mensaje ICMP de tipo 8 (petición de eco), al que el receptor ha de responder con una respuesta de eco (ICMP tipo 0).

El programa traceroute (tracert en Windows)

Permite conocer las rutas entre un origen y un destino. Se basa en mensajes genéricos ICMP (no específicos como en ping), y básicamente explota dos errores:

- El tiempo de vida agotado: El router, si el valor del campo TTL (tiempo de vida restante) es cero, elimina el paquete, pero lo notifica al originador del mismo mediante un mensaje ICMP.
- Puerto inalcanzable: cuando una estación recibe un datagrama UDP o segmento TCP destinado a un puerto que la máquina no escucha (espera sin recibir respuesta), ésta lo notifica con un mensaje de error.

Así traceroute lo que hace es enviar paquetes al destino con TTL secuencialmente ascendentes, de modo que cada router enviará un mensaje de tiempo de vida agotado. Al llegar al destino nos dará el mensaje de puerto inalcanzable, porque traceroute dirige expresamente el paquete a un puerto no usado. Hacer por ejemplo tracert www.nasa.gov .

Mensajes de redireccionamiento

Cuando en una LAN hay más de un router, y se envía un paquete a un router equivocado, éste dispone de un mensaje ICMP de *redirect*, que lo envía al router correcto, y notifica a la estación que modifique su tabla de encaminamiento, que se actualizará, para en el futuro optimizar el rendimiento del sistema.

- **PROTOCOLOS DEL NIVEL DE TRANSPORTE**

Actúa de interfaz entre los niveles orientados a aplicación y los orientados a red. Oculta a los niveles altos el tipo de tecnología (red) a que está conectado el terminal. En la jerarquía de protocolos TCP/IP, se definen dos protocolos: el **UDP** (User Datagram Protocol), y el **TCP** (Transmissión Control Protocol).

También se definen aquí, dos direcciones: la **IP**, que identifica un subsistema dentro de una red, y el **puerto**.

El puerto

Identifica la aplicación que requiere la comunicación. Para identificar las diferentes aplicaciones, los protocolos TCP/IP marcan cada paquete con un identificador de 16 bits llamado puerto. Su verdadera utilidad es que permite multiplexar aplicaciones, permitiendo que un mismo protocolo lleve información de diferentes aplicaciones , que son identificadas por su puerto. Los puertos más conocidos, reservados para aplicaciones conocidas, son:

Puerto para aplicaciones que usan el UDP	**Puertos y servicios que usan el TCP**
7, para el servidor de eco	20 y 21, FTP y Control.
53, para el servidor DNS	23, Telnet
69, para la transferencia de ficheros virtual	25, SMTP
	79, finger
	80, para el http (WWW). Navegadores
	nntp (news)

OTROS CONCEPTOS QUE DEBEMOS CONOCER

Host: debemos ser conscientes que en TCP/IP todas las maquinas son hosts. Desde nuestro humilde PC hasta el mainframe mas gigantesco que exista.

Un "puerto" no es nada más que un sitio al donde podemos ir via TCP en nuestro PC. Es un numero de 2 bytes, por tanto puede ser desde el 0 al 65535. Pero para que vayamos a un puerto, y seamos capaces de hacer algo, "alguien" deberá estar en espera en ese puerto.

El TCP instalado en nuestro PC, se reserva para sí, los puertos 0 a 1023, y en muchos de ellos, coloca a ese "alguien" en espera. (no es otra cosa que un programa que "escucha" en dicho puerto.

Para que dos PCs hablen entre sí, hay que establecer un canal de comunicación. Esto es lo que se llama un "SOCKET". Para establecer dicho canal (o dicho SOCKET), es necesario una dirección y puerto origen y una dirección y puerto destino. Es decir, esta combinación de 2 direcciones y dos puertos es lo que se llama un socket. Evidentemente habrá, un programa en nuestro PC que "escuche" y escriba en el puerto de nuestro PC y otro programa en el destino que escuche y escriba en el otro puerto. Muchas veces para indicar una direccion/puerto, se separan por dos puntos.

Supongo que como el mundo curioso, habréis estado por ciertas paginas de hackers o de crakers. Allí habréis visto ciertas direcciones del tipo:

123.23.43.66:2021

Esto, escribiéndolo en un navegador de la forma:

ftp://123.23.43.66:2021

Está solicitando que se abra un socket desde nuestro PC y puerto 21 al ordenador 123.23.43.66 puerto 2021 (el puerto 21 es el puerto "estándar" del ftp, y por tanto no es necesario especificarlo, pero en ese caso, la maquina destino, utiliza un puerto no estándar, el 2021 por ejemplo).

Bien, una vez abierto el socket, en la otra maquina habrá un servidor de FTP que atenderá a las peticiones del navegador. El navegador, aunque nosotros no lo veamos, una vez establecido el socket, lo que hará será dar un comando DIR en la maquina destino y nos mostrará el "directorio" que está ofreciendo el servidor.

Realmente la sintaxis completa del FTP es:

ftp://usuario:password@direccionIP:puerto

El usuario/password es obligado siempre en una comunicación FTP (por definición), lo que pasa es que el navegador es "listo" y normalmente sino está especificada, no envía como usuario "guest" (invitado). Y el servidor FTP, puede o no, aceptar a dicho usuario.

El protocolo UDP

Es un protocolo no orientado a la conexión, o sea, no tiene un control de errores − y si los detecta, descarta el datagrama−, ni ordena los paquetes. Ha de ser la aplicación la que controle los errores ¿qué utilidad tiene?.

Recordemos que el protocolo IP también es no orientado a la conexión, con lo que el UDP explota esas características, además de ser simple y sencillo. Esta simplicidad, lo hace bueno para aplicaciones que requieren pocos retardos, como aplicaciones en tiempo real, o para sistemas limitados que no pueden implementar el TCP.

El protocolo TCP

Protocolo sí orientado a la conexión, lo que da fiabilidad a la aplicación, al realizar un control de flujo y errores y encargarse del ordenamiento de paquetes, garantizando la transmisión libre de errores, el libramiento de la información, y la no duplicación de ésta, Esta fiabilidad conlleva mayores retardos que con el UDP, y mayor complejidad.

Realiza tres fases: establecimiento de la conexión, transmisión de la información, y terminación de la conexión. Usa métodos como el sliding window (los routers se avisan continuamente de su capacidad de recibir nuevos datos para evitar el desbordamiento de sus memorias intermedias), en la transmisión de datos

de gran volumen. También usa complejos algoritmos como el slow star para ralentizar el envío de datos, realizándolos progresivamente, y así evitar congestiones , o el congestion avoidance, que complementa al anterior, al reducir la recepción y reiniciar el slow star.

Anexos

Ver anexo con las aplicaciones estándar de Internet y su utilización: ping, tracert, arp, route print, ipconfig, netstat, telnet, y las no estándar netcat y tcpdump.

UTILIDADES TCP/IP INCORPORADAS EN WIN 95/98
−−

Bueno, vamos empezar la casa por el tejado. Aunque en principio parecería mas logico comentar un poco como se estable una comunicacion IP. que son los puertos, que es un socket, etc..... vamos a ver primero una serie de "comandos" o utilidades que incorpora win 98. La mayoria son utilidades "estandard" que están en cualquier otro sistema operativo, sea o nó de Microsoft. Recordar que la mayoria de estos comandos, mientras no diga lo contrario son comandos para dar en una ventana MsDOS. Es decir con comandos de "consola". Estos comandos podemos ejecutarlos con /? para que nos muestren una pequeña ayuda.

PING : Este comando nos permite enviar un "paquete" de prueba a otro ordenador. Sí el otro ordenador está vivo, nos responderá. Aunque tenemos practicamente la garantia de que el PING nos funcione contra cualquier ordenador, esto puede limitarse. Me explico: el comando PING no utiliza IP, utiliza ICMP. El trafico ICMP puede cortarse. Algunos servidores que quieren permanecer ocultos en la red, pueden filtrar este trafico y nunca responderá ese servidor a un PING.

WINIPCFG (este es un comando windows). Nos dará la direccion IP sobre cualquier adaptador de red. Igualmente nos informará del gateway y del DNS que tenemos en ese momento. Esto puede tener una utilidad que se me ocurre ahora mismo. Imaginemos que tenemos camara de video y queremos establecer videoconferencia "privada" con NetMeeting (sin pasar por ningun servidor de internet), directamente con otra persona. Para ello, podemos querdar a una hora determinada, y mediante ICQ o mail, por ejemplo, informarnos del IP de la otra persona e igualmente enviarle nuestro IP. Con esto, ahora podemos configurar MetMeeting en directo a ese PC para establecer entre nostros una videocinferencia. Esto es en directo, no pasa por ningun servidor ni lista de directorio en servidores y por tanto la comunicacion es mucho mas rapida.

IPCONFIG : nos muetra una informacion detallada de datos sobre todos nuestros adaptadores de red. Lo mejor es ejecutar un IPCONFIG /ALL y como la salida de datos es muy grande deberemos o bien redirigirla a un fichero o simplemente teclear: IPCONFIG /ALL | MORE

NETSTAT : sirve para ver el estado de los puertos (mas adelante veremos este concepto), en nuestro PC. En principio utilizarlo con el parametro −a. Es decir NETSTAT −a Esto nos mostrará los puertos que está utilizando. Como introduccion, debemos comentar que existen 65535 puertos disponivbles. Del 0 al 1024 son reservados para el TCP, y del 1024 en adelante para aplicaciones de usuario. En particular una posible "aplicacion" de usuario es el famoso BO (Back Orifice). Este "pilla" un puerto 3x.xxx (treinta y tantos mil). Tiene un valor por defecto, pero cuidado: se puede "configurar".

FTP : nos permite de una manera un poco "rupestre" (entrar en él y darle HELP, el transferir datos entre ordenadores en TCP/IP. Es el cliente FTP que trae windows. Tambien en la actualidad, el propio navegador de internet (IE), es capaz de realizar una transferencia FPT, siendo en este caso mas comodo. Pero tambien mas lento. Igualmente recuerd que podemos utilizar programas FTP de libre distribucin de terceros. Y alguno realmente buenos que merece la pena tener en cualquier instalacion.

(el FTP utiliza por defecto el puerto 21 del TCP/IP)

TELNET : nos permite entrar en un ordenadro remoto con emulacion de terminal. Es decir si el ordenador remoto tiene activo el servicio de telnet y tenemos usuario / password en él, nos permitirá manejarlo como si estuviesemos fisicamente en su pantalla y teclado. (el TELNET utiliza por defecto el puerto 23 del TCP/IP)

NET : Admite muchos subparametros. Nos permite por ejemplo mapearnos una unidad de red, desonectarla, y en general utilizar directamente todos los comandos para actuar sobre una red. Es el "antiguo" comando LM (Lan Manager).

COMUNICACION TCP/IP
−−−−−−−−−−−−−−−−−−

En TCP/IP, todo se basa en la filosofía cliente/servidor. Un "host" (no es nada mas que un ordenador cualquiera, un PC, o uno gigantesco), puede ofrecer servicios a la red. Este es un servidor. Y otro puede solicitar esos servicios (cliente). Lo "normal" (en un PC) es ser siempre cliente, es decir solicitamos servicios de FTP, HTML, news,....

¿Como se solicitan los servicios o como hablan dos maquinas bajo TCP?. Muy sencillo, se establece un "socket" −un canal de comunicacion− ¿y como? pues un socket no es nada mas que una conexion entre una maquina y un puerto con otra maquina y otro puerto. Es decir intervienen 2 maquinas y 2 puertos.

En el IP, se definen 65535 puertos y desde el 0 al 1023 estan reservados para los servicios estandard del TCP (html=80, ftp=21,...etc) y desde el 1024 en adelante estan reservados para programas de usuario.

Para abrir un socket (canal de comunicacion), una maquina debe solicitar una direccion ajena y un puerto. Si en ese puerto, en la otra maquina, hay un programa escuchando (LISTENING), este programa atenderá a nuestras peticiones.

Cuando abrimos el "navegador" y tecleamos una pagina, el funcionamiento sería el siguiente:

1) Nuestro PC para abrir el socket necesita conocer la direccion del destino (el puerto ya lo sabe, ya que por defecto en html es el 80). Esta direccion debe estar en formato xxx.xxx.xxx.xxx (4 numeros separados por puntos −direccion IP−)

2) Nuestro PC, investiga si conoce el ordenador destino (es decir investiga un nombre, por ejemplo www.microsoft.com). Si no lo conoce debe investigar.

3) Aunque para esta investigacion intervienen varios pasos (fichero host, mascaras de subred, etc), vamos a simplificarlo. Nuestro PC se lo pregunta al DNS de nuestro servidor (recordad que ponemos DNS al definir una conexion telefonica, o bien nuestro proveedor nos lo da automaticamente al establecerlo.

4) DNS = Domain Name Solver. Es decir nos va a resolver por nombre de dominio su direccion. Por tanto nos devuelve su direccion IP.

5) Con esto el navegador ya puede abrir un socket desde nuestra direccion, y un puerto "cualquiera" !!! libre de nuestro PC y la direccion del destino y el puerto 80.

6) Es decir lo que hace en ese momento es pedirle a nuestro propio PC un puerto libre, imaginemos que le dá el 2122, y entonces establece un canal de comunicacion entre ambas maquinas desde el puerto 2122 en el cual "escuhará" nuestro navegador para esa conexion y el 80 en el destino, que será el que nos pase la informacion (la pagina web).

Bien, esto es extensible para todos los servicios IP (FTP en el 21,TELENT en el 23,..etc). Y para cualquier cosa que queramos hacer.

Fijaros que en el ejemplo anterior, en la otra maquina, no hay un navegador, sino que hay un programa, que lo que hace es escuchar "peticiones" en el puerto 80, cuando las recibe, analiza lo que le piden, busca la pagina y nos la envia. Este es un servidor de paginas web.

- **APLICACIONES INTERNET**

Aplicación distribuida

Por aplicación distribuida entendemos aquellas formadas por diferentes partes que se ejecutan en ordenadores diferentes y que utilizan las redes de comunicaciones para interaccionar entre sí, mediante la petición de una ejecución de operaciones a un sistema remoto, y el retorno del resultado de esa ejecución como respuesta.

Las operaciones típicas de las aplicaciones distribuidas en la transferencia o acceso remoto a información, incluyen crear, borrar, listar, leer, escribir, buscar, mover, copiar, modificar, salir.

El modelo cliente/ servidor

Es en el que se basan las aplicaciones distribuidas, que son la mayoría en Internet. Permite estructurarlas, especificar sus protocolos de comunicación y sus mecanismos de implementación. También describe como implementar una aplicación distribuida a partir de una local.

Cualquier aplicación bien diseñada, tiene dos partes, la servidor y la cliente, que forman la interfaz de definición del servicio. La interfaz de aplicación la forman el conjunto de operaciones para permitir la interacción entre el usuario y la aplicación.

En las aplicaciones distribuidas, los clientes acceden al servidor a través del protocolo de acceso, en el cual un extremo actúa siempre como cliente y el otro como servidor, y de unos módulos de comunicaciones para transportar las operaciones. Los siguientes esquemas lo aclaran mejor.

Sistema servidor y comunicación entre servidores

Un sistema servidor consiste en la distribución del servicio en diversos servidores no necesariamente del mismo tipo que interaccionan entre sí para dar un servicio. Los protocolos de sistema los unen, y en función de las necesidades del momento, los módulos actuarán como clientes o como servidores

Servidor

Usuario Protoc. de acc. opcional

PSO

Interfaz de

aplicación Protocolo de sistema opcional

Protoc de acceso (PSO)

Cliente Servidor

Protoc. de acc. opcional PSO

Comunicacion

Servidor

TCP/ UDP

(lo mismo entre los servidores)

Comunicaciones

DNS. El servidor de nombres Internet

Todos los nombres de la red asociados a recursos, están organizados de manera jerárquica, formando un sistema de dominios, en forma de BD distribuida descentralizada, conocido como DNS (Domain Name Server). Pueden ser ordenadores, pero también buzones de correos, donde a la dirección usuario@campus.uoc, le corresponde el dominio usuario.campus.uoc.

El DNS está organizado en forma de árbol, donde cada nodo tiene una etiqueta que lo distingue de sus nodos hermanos. El nombre de dominio de un nodo, se define como la secuencia formada por las etiquetas que hay en el camino entre este nodo y la raíz. Sólo valen los caracteres alfanuméricos y el −, pero empezando letra y acabando por letra o número.

En este modelo, intervienen dos agentes:

- Los servidores DNS, que reciben las consultas y envían las respuestas. Cada servidor tiene acceso directo a su BD local, e indirecto a las BD de otros servidores. Así el conjunto de servidores DNS, forman un sistema servidor, ya explicado.
- Los resolvedores, que actúan como clientes del servicio, convirtiendo las peticiones en consultas, y extrayendo de las respuestas la información solicitada. Han de tener acceso al menos a un servidor DNS.

Los nodos se agrupan en zonas, donde cada zona está formada por el nodo superior y los que haya en el nivel jerárquicamente inferior. El objetivo de esto es asignar una autoridad que gestione cada zona, con capacidad de agregar o borrar nodos dentro de su zona, crear subzonas con sus delegados, etc.

La mayoría de las consultas de los clientes se realizan en modo denominado recursivo (el servidor debe hallar la respuesta o dar error, pero no referencias a otros servidores.

Telnet. Terminal virtual

Es un protocolo de terminal virtual (dispositivo imaginario que establece una correspondencia entre sus funciones y las del terminal real), con el que controlar un terminal de usuario desde un ordenador remoto. Está basado en el protocolo de transporte TCP, y sigue el modelo cliente/ servidor, donde el sistema usuario establece una conexión con el sistema proveedor, que está esperando peticiones de conexión en un puerto determinado cualquiera, aunque generalmente se usa como estándar el 23.

Anexos: comandos de Telnet, los protocolos rlogin (login), rsh (shell) y sus implementaciones.

FTP. Transferencia de ficheros

Este protocolo se basa también en TCP y en el modelo C/S, y permite la transferencia en los dos sentidos, así

como que el cliente efectúe transferencias directas de un servidor a otro, sin el paso previo por el cliente. También proporciona operaciones para que el cliente manipule el sistema de ficheros del servidor, permitiendo así la interoperatividad entre sistemas diversos. Su puerto oficial es el 21.

El interprete de protocolo del cliente, establece una conexión de control al puerto del interprete servidor, utilizando las reglas especificadas en el protocolo Telnet (mensajes de 8 bits, ASCII,...).

Anexos: comandos FTP del interprete cliente, y las respuestas FTP del interprete servidor, y ejemplos.

Correo electrónico. Protocolos

Es una aplicación distribuida para enviar y recibir mensajes a través de sistemas informáticos. Para la transferencia, se usa el protocolo SMTP (también vale para la recepción), y para el acceso a los buzones de correo, el POP3 (más sencillo que el SMTP), y el IMAP4. Como formato de mensaje, se usa el MIME. Veamos cada cosa.

- Formato de mensaje: MIME: El estándar especifica que los mensajes constarán de una **cabecera** (campos de remitente−From, destinatario−To, destinatario de copia−Cc, destinatario de copia ciega−bcc , asunto, fecha, etc.), y un **cuerpo del mensaje**. El formato MIME redefine este formato llamado RFC 822, permitiendo contenido de texto no solamente ASCII, sino también texto enriquecido, y nuevos tipos de valores como image (gif y jpeg), audio (basic), video (mpeg), etc. Anexo cabeceras y MIME
- El protocolo SMTP: siguiendo el modelo del correo postal, se basa en el almacenamiento y reenvío posterior, a otra *oficina* o al destinatario final, que deberá disponer de un buzón para recibir mensajes, así como de un receptor SMTP o POP3. El SMTP es independiente de los subsistemas de transmisión, y sólo requiere un canal de datos. No obstante, se suele implementar la comunicación a través del protocolo Telnet y sobre TCP. Su puerto asignado es el 25. Anexo comandos, respuestas y ejemplos SMTP
- El protocolo POP3: es para sistemas pequeños, donde se requiere un acceso simple a los buzones de correo, y donde SMTP no es práctico, pues implica tener el sistema conectado y dispuesto a recibir mensajes en cualquier momento. Sin embargo, el POP3 implica tener sistemas donde se encuentren los buzones, y que éstos estén conectados en todo momento, tanto para recibir mensajes como para recibir peticiones de acceso a los buzones. Anexo
- El protocolo IMAP4: Permite al cliente acceder a los mensajes de correo electrónico de un servidor, y manipularlos. También permite al usuario tener diferentes buzones estructurados jerárquicamente y manipularlos de forma remota, como lo haría en local. Este protocolo no especifica ningún método de envío, el cual deja a otros protocolos. Anexos: modelo funcional del protocolo IMAP4 integrado en un sistema SMTP y POP3: comandos y ejemplos.

Servidores de noticias. El protocolo NNTP

Las news, permiten el envío de mensajes, como en el correo electrónico, pero con la diferencia que no se indica destinatario, sino que cualquier usuario con acceso al servicio puede leer el mensaje. Sería como un tablero de anuncios. El servidor de noticias en Internet se conoce como Usenet, nombre de la red donde se originó este servicio. Aunque la distribución de noticias está muy extendida en la Red, en la práctica podemos restringir el ámbito donde distribuir los mensajes, por ejemplo por zona geográfica.

El modelo NNTP: en las news, la distribución de mensajes, anuncios, artículos, se efectúa de manera descentralizada, donde cada servidor de noticias reenvía los artículos a una serie de servidores cercanos, y así sucesivamente. Así, idealmente, un artículo estará disponible en todos los servidores de la red. Naturalmente, los servidores sólo almacenan el artículo durante un cierto tiempo, bien porque tenga fecha de caducidad, bien a discreción del administrador. Lo servidores se comunican entre sí mediante el protocolo TNP (Network News Transfer Protocol).

Lo normal, es que el usuario no acceda directamente a un servidor de noticias, sino a otro sistema que actuará de cliente ante el servidor, y se entenderá con el usuario mediante un programa, el denominado lector de noticias, que actuará de interfaz.

Los principales grupos en que se organizan los artículos, según la temática, son:

Comp. (computadores, news (noticias sobre las news), rec (hobbies), soc (temas sociales, culturales, etc.), sci (ciencia), talk (debates), misc (miscelánea), alt (grupos no oficiales), fr, es, (destinados aestos paises), etc.
Anexos sobre el formato de los mensajes, comandos.

WWW. El protocolo HTTP

El protocolo http (Hipertext Transfer Protocol), es la base del servicio WWW. Creado a finales de los 80 por un grupo de investigadores del CERN, se extendió por todo el mundo y dio lugar al nacimiento del WWW.

- WWW: En el WWW, cada recurso (gif, texto, banner, etc) existente en un documento, tiene una dirección, denominada *identificador uniforme de recurso* o URI, cuya forma general es:

esquema: identificador URI absoluto. Ejemplo: http://servidor

En los URI relativos omitiríamos el esquema y el separador :.

Un URI, puede ser un localizador (**URL**), si especifica como acceder al recurso, o un nombre (URN), si identifica al recurso mediante una serie de atributos. Veamos las sintaxis de **URL** más conocidas:

ftp:// [usuario [:contraseña] @] servidor/ruta

news: identificador

malito: dirección

telnet:// [usuario [:contraseña] @] servidor [: puerto] [/]

http:// servidor [: puerto] [camino absoluto] (el puerto por defecto es el 80)

Se pueden indicar URL relativos, tal como lo haríamos en UNIX: ../../, /ruta, #final.

Anexo sintaxis protocolo HTTP

Ver libro y otra bibliografía.para HTML

Llamadas a procedimientos remotos. RPC

Los RPC son métodos que permiten, de forma sencilla, la especificación e implementación de aplicaciones distribuidas como las vistas anteriormente. El RPC es uno de los mecanismos que permite la implementación de los dos módulos de comunicaciones descritos en el modelo C/S: el de la parte cliente y el de la parte servidor.

Servicio de directorio X.500

Es un servicio que permite el almacenamiento de todo tipo de información de forma distribuida, destinado a aplicaciones que sólo necesitan acceso simple de lectura/ escritura al directorio, y normalmente sobre

conexiones TCP/ IP. A diferencia del DNS, puede incluir información no sólo de PC's, sino también de cualquier tipo de objeto y protocolo para acceder a él. No lo veo claro.

Protocolos Internet en tiempo real

Para la transmisión de datos en tiempo real, hay diversos protocolos:

- El estándar RTP consta de dos protocolos: RTP y RTCP.

RTP (Real−Time Transport Protocol): proporciona servicio de envío punto a punto para datos como audio y video.

RTCP: para supervisar la calidad del servicio y conseguir información de los participantes en la sesión. Se usa en conferencias en tiempo real de grupos de cualquier tamaño.

El protocolo RTP se usa normalmente sobre redes de tipo UDP/ IP, aunque se hacen esfuerzos para que sea independiente del medio de transporte. En las sesiones multimedia, es necesaria una sesión RTP separada para cada tipo de dato. No está orientado a la reserva de recursos, para lo cual se usa el RSVP.

- El protocolo RTSP: es a nivel de aplicación, y proporciona el control del flujo de datos multimedia , ya sea unicast como multicast.
- El protocolo RSVP (Resouce reSerVation Protocol): reserva recursos para proporcionar calidad de servicio (QoS) en aspectos como retardos, pérdida de datos, ancho de banda, variando los parámetros de calidad en función del tipo de datos. Opera sobre el IP, ocupando el sitio del protocolo de transporte, pero sin transportar datos, sólo para determinar a donde irá a parar las peticiones de reserva de recursos. Para ello, todos los sistemas que hay entre el cliente y el servidor, deberán soportar el RSVP.

Seguridad en las comunicaciones

Las acciones que alguien puede realizar malintencionadamente en una comunicación entre dos usuarios, pueden ser: interrumpir el envío, interceptarlo, modificarlo, o ir de impostor, simulando ser otro. En criptografía, el principal mecanismo de seguridad, hay dos tipos de cifrados:

- Simétrico: se usa la misma clave para cifrar y descifrar.
- Asimétrico hay una clave para cifrar y otra para descifrar, de modo que a partir de la primera −clave privada−, se puede obtener la segunda −clave pública−, que se puede dar a conocer a todo el mundo, pero al revés es teóricamente imposible (sistema de los números primos, por ejemplo).

El segundo método no requiere que las partes intercambien las claves, pero necesita más tiempo de proceso, por su complejidad. Para cifrar el mensaje, se puede utilizar la clave pública, y la clave privada para descifrarlo (sólo el destinatario podrá leerlo), o al revés, cifrándolo con la clave privada, sin importar quien lo lea, pero con la seguridad de que sólo ha podido ser generado por su creador (por ejemplo las firmas digitales).

Origen

Codificación

Fuente DTE

Codific

Canal DCE

Descod

Canal DCE

Descod

Font DTE

Destino