Universidad del Azuay
Facultad de:
“Ciencias de la Administración”
“TCP/IP Capa de Aplicación”
Realizado por:
Gerardo Torres López

Dedicatoria

A mis Padres y Hermanos, pilar fundamental de mi existencia, por sembrar en mi la semilla de la sensibilidad y la comprensión, por su cariño, amor y por enseñarme a que debemos tener la fortaleza de continuar hacia adelante no importa las circunstancias que la vida nos presenta.

A mi familia por apoyarme en todos los momentos difíciles de mi vida, y colaborar con mis padres en mi formación.

Agradecimiento

Al personal docente de la Universidad del Azuay que colaboraron con sus conocimientos a lo largo de mi formación académica.

A los miembros del programa de postgrado de la Universidad Politécnica de Madrid.

A mis amigos y demás persona que de una u otra manera incentivaron en mi el coraje para seguir adelante en mis e estudios.

Capitulo I

1 Introducción

Las redes se han convertido en una parte fundamental, si no la más importante, de los actuales sistemas de información. Constituyen el pilar en el uso compartido de la información en empresas así como en grupos gubernamentales y científicos. Esta información puede adoptar distintas formas, sea como documentos, datos a ser procesados por otro ordenador.

La mayoría de estas redes se instalaron a finales de los años 60 y 70, cuando el diseño de redes se consideraba como la piedra filosofal de la investigación informática y la tecnología punta. Dio lugar a numerosos modelos de redes como la tecnología de conmutación de paquetes, redes de área local con detección de colisión, redes jerárquicas en empresas, y muchas otras de elevada calidad.

Desde comienzos de los 70, otro aspecto de la tecnología de redes cobró importancia: el modelo de pila de protocolo, que permite la interoperabilidad entre aplicaciones. Toda una gama de arquitecturas fue propuesta e implementada por diversos equipos de investigación y fabricantes de ordenadores.

El resultado de todos estos conocimientos tan prácticos es que hoy en día cualquier grupo de usuarios puede hallar una red física y una arquitectura adecuada a sus necesidades específicas, desde líneas asíncronas de bajo coste, sin otro método de recuperación de errores que una función de paridad bit a bit, pasando por funciones completas de redes de área extensa(pública o privada) con protocolos fiables como redes públicas de conmutación de paquetes o redes privadas SNA, hasta las redes de área local, de alta velocidad pero distancia limitada.

El lado negativo de esta explosión de la información es la penosa situación que se produce cuando un grupo de usuarios desea extender su sistema informático a otro grupo de usuarios, que resulta que tiene una tecnología y unos protocolos de red diferentes. En consecuencia, aunque pudieran ponerse de acuerdo en el tipo de tecnología de red para conectar físicamente sus instalaciones, las aplicaciones(como por ejemplo sistemas de correo) serían aún incapaces de comunicarse entre sí debido a los diferentes protocolos.

Se tomó conciencia de esta situación bastante temprano(a comienzo de los 70), gracias a un grupo de investigadores en los Estados Unidos, que fueron artífices de un nuevo paradigma: *la interconexión de redes*. Otras organizaciones oficiales se implicaron en la interconexión de redes, tales como ITU-T e ISO. Todas trataban de definir un conjunto de protocolos, distribuidos en un conjunto bien definido de capas, de modo que las aplicaciones pudieran comunicarse entre sí, con independencia de la tecnología de red subyacente y del sistema operativo sobre el que se ejecutaba cada aplicación.

Capitulo II

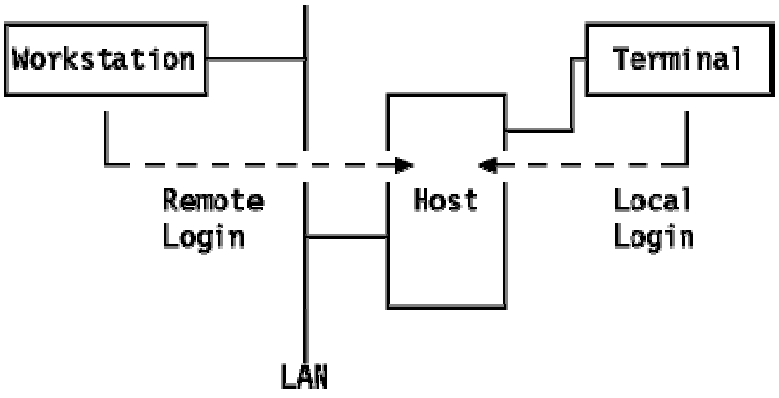
2 Arquitectura y protocolos del Nivel de Aplicación

2.1 TELNET Protocolo de conexión remota.

TELNET es un protocolo estándar siendo su número STD de 8. Su status es recomendado. Se describe en el RFC 854 - Especificaciones del protocolo TELNET y RFC 855 - "TELNET Option Specifications".

El protocolo TELNET proporciona una interfaz estandarizada, a través de la cual un programa de un host(el cliente de TELNET) puede acceder a los recursos de otro host (el servidor de TELNET) como si el cliente fuera una terminal local conectada al servidor.

Por ejemplo, un usuario de una estación de trabajo situada en una LAN se puede conectar al host. Por supuesto, TELNET se puede usar tanto en LANs como en WANs.



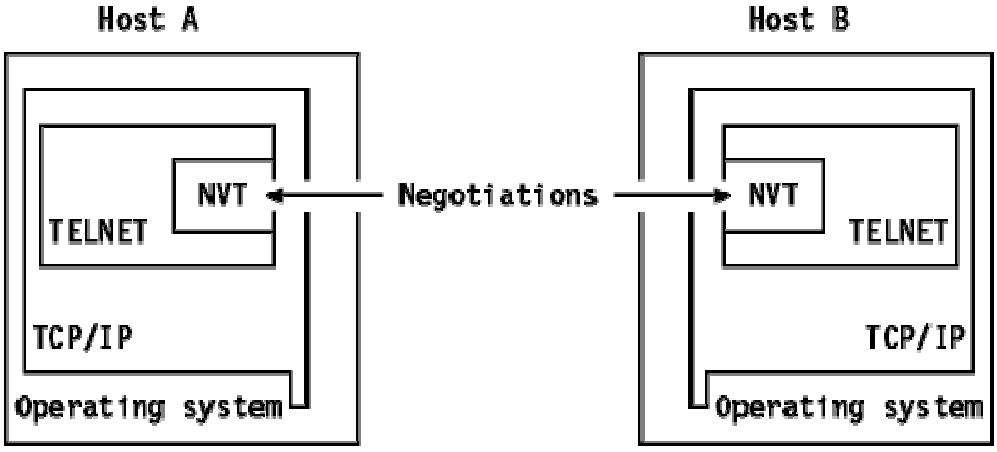
2.1.1 Funcionamiento de TELNET

TELNET es un protocolo basado en tres ideas:

- El concepto de NVT(Network Virtual Terminal) (NVT). Una NVT es un dispositivo imaginario que posee una estructura básica común a una amplia gama de terminales reales. Cada host mapea las características de su propia terminal sobre las de su correspondiente NVT, y asume todos los demás hosts harán lo mismo.
- Una perspectiva simétrica de las terminales y los procesos.
- Negociación de las opciones de la terminal. El protocolo TELNET usa el principio de opciones negociadas, ya que muchos host pueden desear suministrar servicios adicionales, más allá de los disponibles en la NVT. Se

pueden negociar diversas opciones. El cliente y el servidor utilizan una serie de convenciones para establecer las características operacionales de su conexión TELNET a través de los mecanismos "DO, DON'T, WILL, WON'T"("hazlo, no lo hagas, lo harás, no lo harás").

Los dos hosts comienzan verificando que existe una comprensión mutua entre ellos. Una vez que se ha completado esta negociación inicial, son capaces de trabajar en el nivel mínimo implementado por la NVT. Después de haber logrado este entendimiento mutuo, pueden negociar opciones adicionales para ampliar las capacidades de la NVT y así reflejar con precisión la capacidad del hardware real que se está usando. Debido al modelo simétrico usado por TELNET, tanto el cliente como el servidor pueden proponer el uso de opciones adicionales.

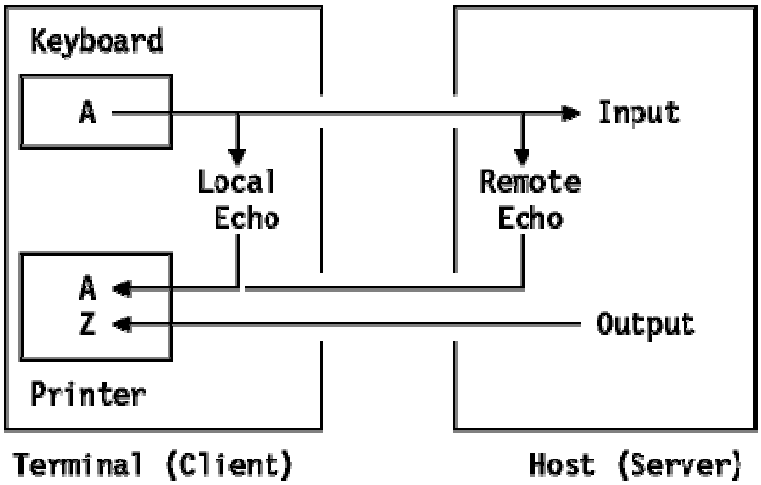


La NVT cuenta con un monitor o "display" y un teclado. El teclado produce datos de salida, que se envían por la conexión TELNET. El monitor recibe los datos de entrada que llegan. Las características básicas de una NVT, a menos que sean modificadas por opciones establecidas de común acuerdo, son:

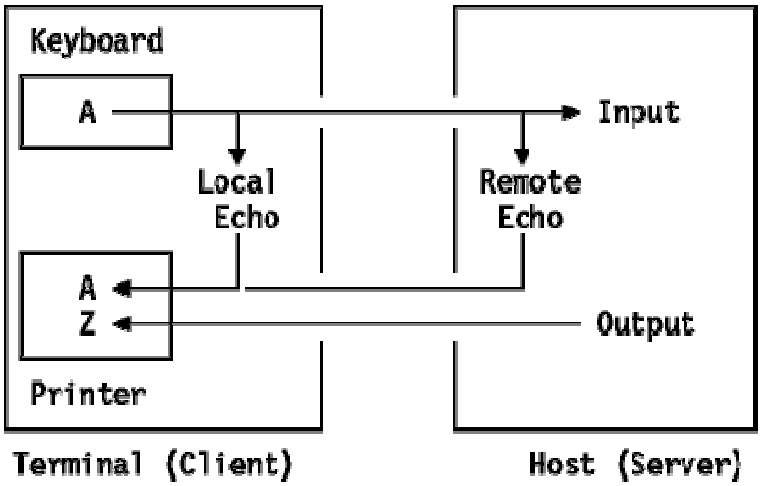
Los datos se representan en código ASCII de 7 bits, transmitido en bytes de 8 bits.

- La NVT es un dispositivo semi-duplex que opera en modo de buffer en línea.
- La NVT proporciona una función de eco local.

Todas estas opciones pueden ser negociadas por los dos hosts. Por ejemplo, se prefiere el eco local porque la carga de la red es inferior y el rendimiento superior pero existe la opción de usar el eco remoto, aunque no se le requiera a ningún host.



La anchura del retorno de carro y la longitud de la página en un monitor NVT no están especificados. Puede manejar caracteres ASCII imprimibles(códigos ASCII del 32 al 126) y puede entender algunos caracteres ASCII de control.

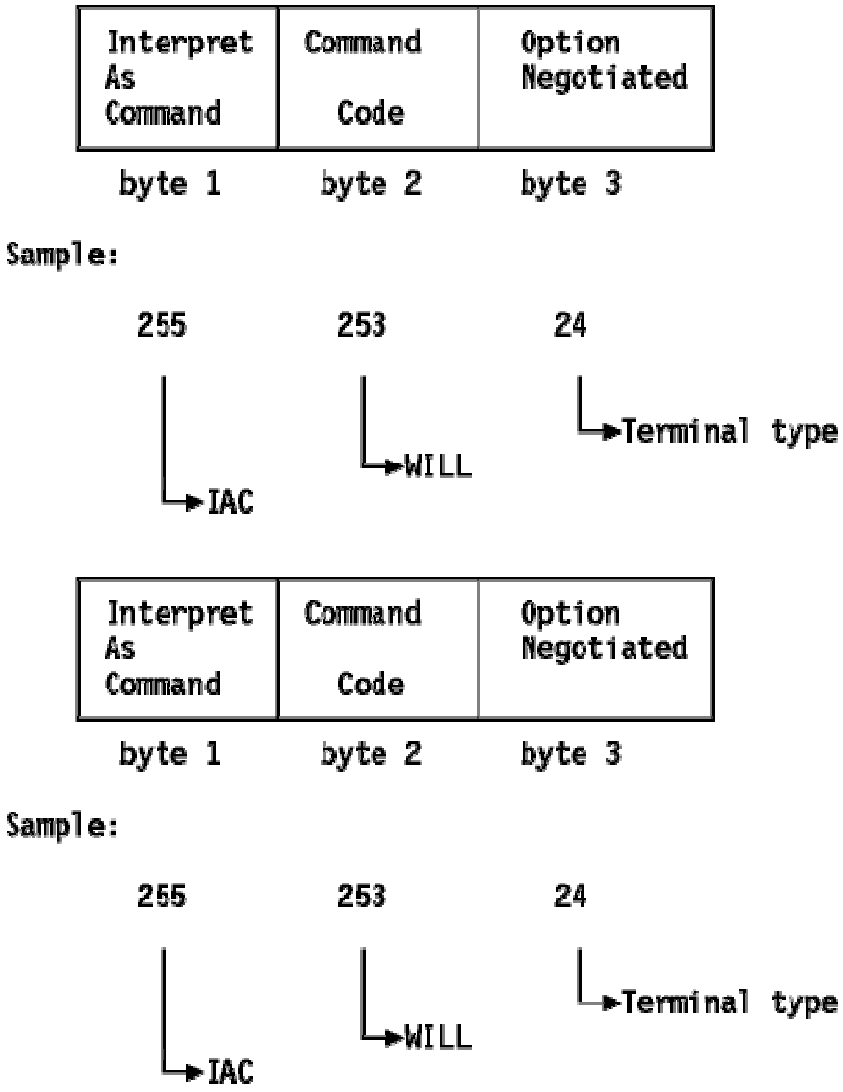


2.1.2 Estructura de comandos en TELNET

La comunicación entre cliente y servidor es manejada por comandos internos, que no son accesibles a los usuarios. Todos los comandos internos de TELNET consisten en secuencias de 2 o 3 bytes, dependiendo del tipo de comando.

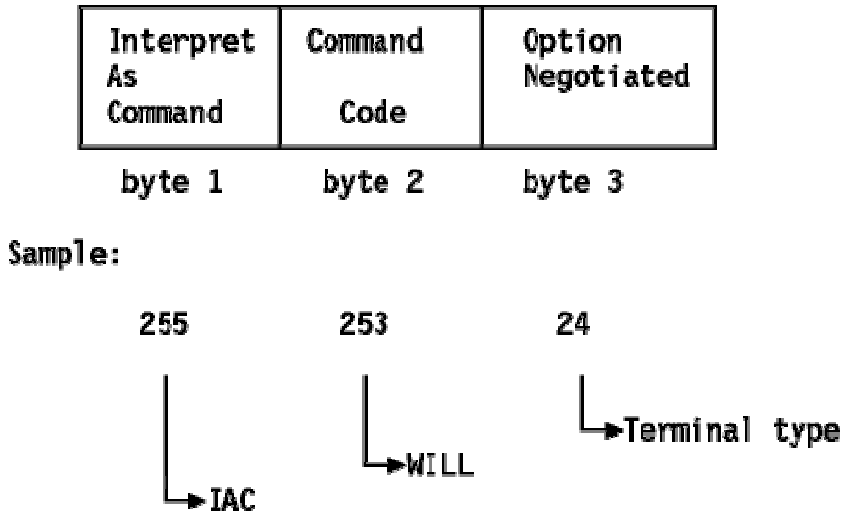
El carácter IAC("Interpret As Command"; Interpretar Como Comando) es seguido de un código de comando. Si este comando trata con opciones de

negociación, el comando tendrá un tercer byte para mostrar el código asociado a la opción indicada.



2.1.3 Negociación de opciones

Usando los comandos internos, TELNET es capaz de negociar opciones en cada host. La base inicial de la negociación es la NVT: cada host que se quiera conectar debe estar de acuerdo con este mínimo. Cada opción se puede negociar haciendo uso de los cuatro códigos de comando "WILL, WON'T, DO, DON'T". Además, algunas opciones tienen a su vez sub-opciones: si ambas partes acuerdan una opción, usarán los comandos SB y SE para llevar a cabo la sub-negociación. Aquí se muestra un ejemplo simplificado de como funciona la negociación de opciones:



2.1.4 Comandos básicos de TELNET

El objetivo principal del protocolo TELNET es proporcionar una interfaz estándar para hosts en una red. Para permitir que comience una conexión. TELNET establece una representación estándar para algunas funciones:

(IP, Interrupt Process) Interrumpir proceso

Muchos sistemas proporcionan una función que suspende, interrumpe, aborta o finaliza la operación de un proceso de usuario. Esta función se usa frecuentemente cuando un usuario cree que su proceso está en un bucle infinito o cuando se ha activado sin querer un proceso no deseado. IP es la representación estándar para invocar esta función. Los desarrolladores deberían tener en cuenta que otros protocolos que usan TELNET pueden requerir esta opción y, por tanto, debería implementarse si se quiere soportar esos otros protocolos.

(AO, Abort Output) Abortar la salida

Muchos sistemas proporcionan una función que permite a un proceso que está generando salida que se ejecute hasta terminar (o que alcance el mismo punto que alcanzaría si se ejecutara hasta el final) pero sin enviar la salida al terminal del usuario, Además, esta función elimina cualquier salida que ya se haya generado pero que no se haya impreso (o mostrado) aún en el terminal del usuario. AO es la representación estándar para invocar esta función. Por ejemplo, algún subsistema podría aceptar normalmente una orden introducida por el usuario, enviar una larga cadena de texto al terminal del usuario como respuesta y, finalmente, indicar que está preparado para la siguiente orden enviando el carácter de "prompt" (precedido por) al terminal de usuario. Si se recibe el AO durante la transmisión de la cadena de texto,

una implementación razonable debería suprimir el resto de la cadena de texto pero transmitir el carácter de "prompt" precedido de . (Esto es posiblemente para distinguir de la acción a tomar si se recibe IP; IP podría provocar la supresión de la cadena de texto y la salida del subsistema).

Se debe tener en cuenta por los servidores que ofrezcan esta función que puede haber espacios de almacenamiento intermedios externos al sistema (en la red y en el ordenador local del usuario) que deberían borrarse; la forma apropiada de hacer esto es mediante la transmisión de la señal "Synch" (descrita más adelante) al sistema del usuario.

(AYT, Are You There) ¿Estás ahí?

Muchos sistemas proporcionan una función que ofrece al usuario alguna evidencia visible de que el sistema está encendido y en funcionamiento. EL usuario puede invocar esta función cuando el sistema está inesperadamente "silencioso" durante un periodo largo de tiempo a causa de la imprevista (por el usuario) duración de una operación, una inusual elevada carga del sistema, etc. AYT es la representación estándar para invocar esta función.

(EC, Erase Character) Borrar carácter

Muchos sistemas ofrecen una función que borra el último carácter o "posición de impresión"* del flujo de datos introducido por el usuario. Esta función se usa típicamente para editar la entrada desde el teclado cuando se cometen errores. EC es la representación estándar para invocar esta función.

(EL, Erase Line) Borrar línea

Muchos sistemas proporcionan una función que borra todos los datos de la línea actual de entrada. Esta función se suele usar para editar la entrada por teclado. EL es la representación estándar para invocar esta Función.

SYNCH Sincronizar

Muchos sistemas de tiempo compartido ofrecen mecanismos para permitir a un terminal recuperar el control de un proceso en ejecución; las funciones IP y AO descritas previamente son ejemplos de estos mecanismos. En sistemas como esos, usados localmente, se tiene acceso a todas las señales generadas por el usuario, ya sean estas caracteres normales o señales "fuera de banda" especiales como las generadas por la tecla "BREAK" o la tecla "ATTN" en algunos terminales. Esto no siempre se cumple cuando el terminal se conecta al sistema a través de la red; los mecanismos de control de flujo de la red pueden provocar que una señal de este tipo se almacene en cualquier otra parte, por ejemplo en el ordenador del usuario.

Para evitar este problema, se ha creado el mecanismo "Synch" de TELNET. Una señal Synch esta formada por una notificación urgente de TCP junto con la orden TELNET de MARCA DE DATOS. La notificación urgente, que no está sujeta al control de flujo relativo a la conexión TELNET, se usa para que se trate de manera especial el flujo de datos por los procesos que lo reciban. De esta forma, el flujo de datos se explora inmediatamente en busca de señales "interesantes" tal y como se definen más abajo, descartando otros datos. La orden TELNET de MARCA DE DATOS (DM, DATA MARK) es la indicación de sincronismo en el flujo de datos que indica que cualquier señal especial ha sido generada y que el receptor puede volver a procesar el flujo de datos.

El Synch se envía mediante la operación de envío TCP con el indicador de Urgente activado y DM como el último (o único) octeto de datos.

Cuando se envían varios Synch rápidamente, se pueden mezclar varias notificaciones de urgente. No es posible contar dichas notificaciones ya que el número recibido será menor o igual que el de enviados. Normalmente, un DM no tiene ningún efecto; si se está en modo urgente, indica el final del procesamiento urgente.

Si TCP señala el final de los datos urgentes antes de que se encuentre el DM, TELNET debería continuar el tratamiento especial del flujo de datos hasta que se encuentre un DM.

Si TCP indica que hay más datos urgentes después de encontrar el DM, sólo puede ser por un Synch posterior. TELNET debería continuar el tratamiento especial del flujo de datos hasta encontrar otro DM.

Señales "interesantes" son: las representaciones TELNET estándar de IP, AO y AYT (pero no EC no EL); los análogos locales de estas operaciones (si los hay); otras señales definidas por el servidor cuya acción se puede llevar a cabo sin retrasar la búsqueda en el flujo de datos.

Como uno de los efectos del mecanismo SYNCH es el descarte de prácticamente todos los caracteres (excepto órdenes TELNET) entre el que lo envía y el que lo recibe, se especifica este mecanismo como la forma estándar de borrar los datos cuando se desee. Por ejemplo, si un usuario provoca que se transmita un AO, el servidor que lo recibe (si proporciona esa función) debería devolver un Synch al usuario.

Por último, al igual que TELNET necesita la notificación urgente de TCP como una señal indicando datos fuera-de-banda, otros protocolos que usan TELNET pueden requerir órdenes TELNET que pueden ser vistas en otro nivel como señales de datos fuera de banda.

Por convenio la secuencia [IP, Synch] se usa como señal para indicar esto. Por ejemplo, supongamos que ese otro protocolo, que usa TELNET, define el

carácter FINAL de cadena igual que la orden TELNET AO. Imaginemos que un usuario de este protocolo desea que un servidor procese el FINAL de cadena, pero la conexión está bloqueada porque el servidor está procesando otros datos. El usuario debería hacer que su sistema:

- Envíe el carácter TELNET IP;
- Envíe la secuencia TELNET SYNCH, esto es: Envíe la MARCA DE DATOS (DM) como el único carácter en una operación de envío TCP urgente.
- Envíe el carácter FINAL de cadena; y
- Envíe el análogo en el otro protocolo del TELNET DM, si lo hay.

El usuario debe transmitir la secuencia TELNET SYNCH del paso 2 anterior para asegurarse de que el TELNET IP llega al intérprete TELNET del servidor.

El envío urgente debería despertar al proceso TELNET; el IP debería despertar el siguiente proceso de más alto nivel.

2.2 FTP (File Transfer Protocol):

FTP es un protocolo estándar con STD número 9. Su estado es recomendado y se describe en el RFC 959 - Protocolo de Transferencia de Ficheros (FTP).

Una de las operaciones que más se usa es la copia de ficheros de una máquina a otra. El cliente puede enviar un fichero al servidor. Puede también pedir un fichero de este servidor.

Para acceder a un fichero remoto, el usuario debe identificarse al servidor. En este momento el servidor es responsable de autentificar al cliente antes de permitir la transferencia del fichero.

Toda conexión FTP implica la existencia de una máquina que actúa como servidor (aquella en la que se cogen o dejan fichero) y un cliente. Lo mas habitual es que los usuarios particulares utilicen programas clientes de FTP para conseguir programas albergados en servidores FTP, que se suelen encontrar en universidades, empresas, o proveedores de Internet.

Para conectarse a un servidor FTP es necesario un programa cliente. Los navegadores, como Netscape Navigator o Microsoft Explorer, suelen tener incorporados programas que actúan como clientes y que permiten tomar ficheros de un servidor. Para poder dejar ficheros en un servidor es necesario un programa de transferencia de FTP (además, el servidor ha de permitir que ese usuario tenga derecho a dejar ficheros). Windows'95 tiene la orden FTP, que puede ejecutar desde la línea de comandos.

Los servidores FTP se organizan de manera similar a como lo hace el Administrador de Archivos del Win'3.1 o el Explorador de Win'95: como una

estructura de directorios en forma de árbol. Esto significa que cada carpeta que seleccionamos está compuesta a su vez de carpetas y archivos, hasta que una carpeta está compuesta únicamente por archivos. Para coger un archivo basta hacer click sobre él (si se trata de un navegador) o utilizar la orden get del FTP en la linea de comandos.

Se pueden enviar o recibir toda clases de ficheros, ya sean de texto, gráficos, sonido, etc. Normalmente los ficheros de los servidores se encuentran comprimidos (formatos .zip o .arj para PC, .hqx o .sit para Macintosh, .tar o .gz para Unix, etc.) con el objeto de ocupar el menor espacio posible tanto en el disco como en la transferencia. Para poder descomprimirlos es necesario un programa descompresor.

Existen dos tipos de accesos a un servidor FTP:

- Como usuario registrado. El administrador del sistema concede una cuenta al sistema (similar a la de acceso a Internet), lo que da derecho a acceder a algunos directorios, dependiendo del tipo de cuenta.

- Como usuario anónimos. En este tipo de acceso el login es anonymous y el password la dirección de correo. Esta es la cuenta que usan por defecto los navegadores.

2.2.1 FTP Offline:

Es enviar un email a un servidor de FTP: se envía un email con la petición de un fichero, te desconectas, y después el fichero es enviado a tu cuenta de email.

No todos los servidores de FTP-mail funcionan de la misma forma para obtener ayuda especifica de un servidor en concreto debes de enviar un email a ese servidor y escribir el cuerpo únicamente: Help

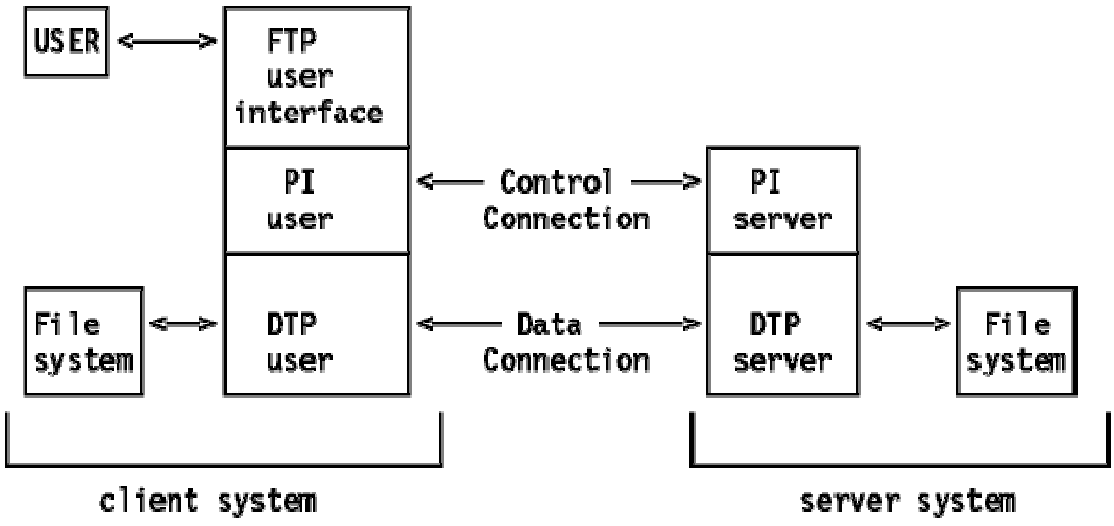
2.2.2 Descripción de FTP

FTP usa TCP como protocolo de transporte para proporcionar conexiones fiables entre los extremos. Se emplean dos conexiones: la primera es para el login y sigue el protocolo TELNET y la segunda es para gestionar la transferencia de datos. Como es necesario hacer un login en el host remoto, el usuario debe tener un nombre de usuario y un password para acceder a ficheros y a directorios. El usuario que inicia la conexión asume la función de cliente, mientras que el host remoto adopta la función de servidor

En ambos extremos del enlace, la aplicación FTP se construye con intérprete de protocolo(PI), un proceso de transferencia de datos, y una interfaz de usuario

La interfaz de usuario se comunica con el PI, que está a cargo del control de la conexión. Este intérprete de protocolo ha de comunicar la información necesaria a su propio sistema de archivos.

En el otro extremo de la conexión, el PI, además de su función de responder al protocolo TELNET, ha de iniciar la conexión de datos. Durante la transferencia de ficheros, los DTPs se ocupan de gestionar la transferencia de datos. Una vez que la operación del usuario se ha completado, el PI ha de cerrar la conexión de control.



2.2.3 Operaciones de FTP

Al usar FTP, el usuario realizará alguna de las siguientes operaciones:

- Conexión a un host remoto
- Selección de un directorio
- Listado de ficheros disponibles para una transferencia
- Especificación del modo de transferencia
- Copiar ficheros de o a el host remoto
- Desconectar del host remoto

2.2.3.1 Conexión a un host remoto

Para ejecutar una transferencia de ficheros, el usuario comienza haciendo un login en el host remoto. Este es el método primario para manejar la seguridad. El

usuario debe tener un identificador y un password para el host remoto, a menos que use un FTP anónimo.

Se usan tres comandos:

Open
: Selecciona el host remoto de inicia la sesión con el login

User
: Identifica al ID del usuario remoto

Pass
: Autentifica al usuario

Site
: Envía información al host remoto utilizado para proporcionar servicios específicos para ese host

2.2.3.2 Selección de un directorio

Cuando se establece el enlace de control, el usuario puede emplear el subcomando cd("change directory") para seleccionar un directorio remoto de trabajo. Obviamente, el usuario sólo podrá acceder a directorios a los que su ID le da acceso. El usuario puede seleccionar un directorio local con el comando lcd("local change directory"). La sintaxis de estos comando depende del sistema operativo.

2.2.3.3 Listado de ficheros disponibles para una transferencia

Se hace con los subcomandos dir o ls.

2.2.3.4 Especificación del modo de transferencia

La transferencia de datos entre sistemas diferentes suele requerir transformaciones de los datos como parte del proceso de transferencia. El usuario ha de decidir dos aspectos de la manipulación de los datos:

- La forma en qué se transferirán los bits.
- Las distintas representaciones de los datos en la arquitectura del sistema.

Esto se controla por medio de dos subcomandos:

Mode
: Especifica si el fichero se ha de tratar como si tuviera estructura de registros o como un flujo de bytes.

Block
: Se respetan las separaciones lógicas entre registros.

Stream
: El fichero se trata como un flujo de bytes. Esta es la opción por defecto, y proporciona una transferencia más eficiente, pero puede que no produzca los

resultados deseados cuando se trabaja con un ficheros estructurados por registros.

Type

Especifica el conjunto de caracteres usado para los datos.

ASCII

Indica que ambos host están basados en ASCII, o que si uno está basado en ASCII y el otro en EBCDIC, se debería realizar una traducción ASCII-EBCDIC.

EBCDIC

Indica que ambos host se basan en EBCDIC.

Image

Indica que los datos deben tratarse como bits contiguos empaquetados en bytes de 8 bits.

Debido a que estos subcomandos no cubren todas las posibles diferencias entre sistemas, el subcomando SITE está disponible para lanzar comandos dependientes del sistema.

2.2.3.5 Copia de ficheros

Get

Copia un fichero del host remoto al host local.

Put

Copia un fichero del host local al host remoto.

2.2.3.6 Finalización de la sesión de transferencia

Quit

Desconecta del host remoto y cierra el FTP. Algunas implementaciones usan el subcomando BYE.

Close

Desconecta del host remoto pero deja al cliente FTP ejecutándose. Se puede lanzar un comando open para trabajar con otro host remoto.

2.2.4 Códigos de respuesta

Con el fin de gestionar estas operaciones, el cliente y el servidor mantienen un diálogo por medio de TELNET. El cliente lanza comandos, y el servidor contesta con códigos de respuesta. Las respuestas incluyen también comentarios para el usuario, pero el cliente usa sólo los códigos.

Los códigos de respuesta tienen tres dígitos, siendo el primero el más significante.

Reply Code	Description
1xx	Positive Preliminary Reply
2xx	Positive Completion Reply

3xx	Positive Intermediate Reply
4xx	Transient negative completion Reply
5xx	Permanent negative completion Reply

Ejemplo

Para comando de usuario, mostrado **así**, el servidor FTP responde con un mensaje que comienza con un código de 3 dígitos, mostrado así:

FTP foreignhost
220 service ready
USERNAME cms01
331 user name okay
PASSWORD xyxyx
230 user logged in
TYPE Image
200 command okay

Ejemplo de una sesión FTP

[C:\SAMPLES]**ftp host01.itsc.raleigh.ibm.com**
Connected to host01.itsc.raleigh.ibm.com.
220 host01 FTP server (Version 4.1 Sat Nov 23 12:52:09 CST 1991) ready.
Name (rs60002): **cms01**
331 Password required for cms01.
Password: **xxxxxx**
230 User cms01 logged in.
ftp> **put file01.tst file01.tst**
200 PORT command successful.
150 Opening data connection for file01.tst (1252 bytes).
226 Transfer complete.
local: file01.tst remote: file01.tst
1285 bytes received in 0.062 seconds (20 Kbytes/s)
ftp> **close**
221 Goodbye.
ftp> **quit**

2.3 SMTP("Simple Mail Transfer Protocol")

El correo electrónico (E-mail) es probablemente la aplicación TCP/IP más usada. Los protocolos de correo básicos de correo proporcionan intercambio de correo y mensajes entre hosts TCP/IP hosts; se han añadido servicios para la transmisión de datos que no se pueden representar con texto ASCII de 7 bits.

Hay tres protocolos estándares que se aplican a este tipo de correo. Todos son recomendados. El término SMTP se emplea con frecuencia para referirse a la combinación de los tres protocolos, por su estrecha interrelación, pero estrictamente hablando, SMTP es sólo uno de los tres. Normalmente, el contexto hace evidente de cuál de los tres se está hablando. Cuando haya ambigüedad, se emplearán los números STD o RFC. Los tres estándares son:

- Un estándar para el intercambio de correo entre dos ordenadores(STD 10/RFC 821), que especifica el protocolo usado para enviar correo entre hosts TCP/IP. Este estándar es SMTP.
- Un estándar(STD 11) para el formato de los mensajes de correo, contenido en dos RFCs. El RFC 822 describe la sintaxis de las cabeceras y su interpretación. El RFC 1049 describe como un conjunto de documentos de tipos diferentes del texto ASCII plano se pueden usar en el cuerpo del correo(los mismos documentos están en ASCII de 7 bits con información de formato embebida: Postscript, Scribe, SGML, TEX, TROFF y DVI aparecen en el estándar).

  El nombre oficial del protocolo para este estándar es MAIL.

- Un estándar para el encaminamiento de correo usando el DNS, descrito en el RFC 974. El nombre oficial del protocolo para este estándar es DNS-MX.

El STD 10/RFC 821 establece que los datos enviados por SMTP son ASCII de 7-bis, con el bit de orden superior a cero. Esto es adecuado para mensajes en inglés, pero no para otros lenguajes o datos que no sean texto. Hay dos estrategias para superar estas limitaciones:

- MIME("Multipurpose Internet Mail Extensions"), definido en los RFCs 1521 y 1522, que especifica un mecanismo para codificar texto y datos binarios en ASCII de 7 bits en el mensaje RFC 822.
- SMTPSE("SMTP Service Extensions"), que define un mecanismo para extender las posibilidades de SMTP más allá de las limitaciones impuestas por RFC 821. Actualmente hay tres RFCs que lo describen:

Un estándar para que un receptor SMTP informe al emisor que extensiones de servicio soporta(SMTPSE) soporta (RFC 1651).

El RFC 1651 modifica el 821 para permitir que un cliente agente SMTP solicite al servidor una lista de las extensiones de servicio que soporta el inicio de una sesión SMTP. Si el servidor no soporta este RFC, responderá con un error y el cliente podrá terminar la sesión o intentar iniciar una sesión según las reglas RFC 821. Si sí lo soporta, puede responder con una lista de las extensiones que soporta. IANA mantiene un registro de servicios: la lista inicial del RFC 1651 contiene los comandos listados en el RFC 1123 - Requerimientos para hosts de Internet - Aplicación y soporte como opcionales en servidores SMTP. Se han definido otras

extensiones con RFCs del modo habitual. Los dos siguientes RFCs definen extensiones específicas:

Un protocolo para transmisión de texto de 8 bits(RFC 1652) que permite a un servidor SMTP indicar que puede aceptar datos formados por bytes de 8 bits. Un servidor que informa que dispone de esta extensión no debe modificar el bit de orden superior de los bytes recibidos en un mensaje SMTP si el cliente así se lo pide.

Las extensiones de MIME y SMTP son estrategias que se complementan más que competir entre sí. En particular, el RFC 1652 se titula SMTPSE para transporte MIME en codificación "8bit", ya que MIME permite declarar mensajes con bytes de 8 bits, en vez de 7. Tales mensajes no se pueden transmitir con agentes SMTP que sigan estrictamente el RFC 821, pero se pueden transmitir cuando tanto el cliente como el servidor siguen los RFCs 1651 y 1652. Siempre que un cliente intenta enviar datos de 8 bits a un servidor que no soporta esta extensión, el cliente SMTP debe codificar el mensaje a 7 bits según el estándar MIME o devolver un mensaje de error permanente al usuario.

Esta extensión no permite el envío de datos binarios arbitrarios porque el RFC 821 fija la longitud máxima de las líneas aceptadas por un servidor SMTP a 1000 caracteres. Los datos que no son texto pueden tener con facilidad secuencias de más de 1000 caracteres sin una secuencia <CRLF>.

Las extensiones limitan específicamente el uso de caracteres no ASCII(aquellos con valor decimal superior a 127) al cuerpo de los mensajes - no están permitidos en las cabeceras RFC 822.

Un protocolo para la declaración del tamaño del mensaje(RFC 1653) que permite a un servidor informar al cliente del tamaño máximo de mensaje que puede aceptar. Sin esta extensión, un cliente sólo puede ser informado de que un mensaje ha excedido el tamaño máximo(sea fijo o temporal, por falta de espacio en el servidor) tras transmitir todo el mensaje. Cuando esto sucede, el servidor desecha el mensaje. Con ella, el cliente puede declarar el tamaño estimado del mensaje y el servidor devolverá un error si es demasiado grande.

Todas estas extensiones son borradores y tienen status electivo.

2.4 Protocolo HTTP

El protocolo de transferencia de hipertexto (HyperText Transfer Protocol) es un protocolo del nivel de aplicación usado para la transferencia de información entre sistemas, de forma clara y rápida. Este protocolo ha sido usado por el World-Wide Web desde 1990.

Este protocolo permite usar una serie de métodos para indicar la finalidad de la petición. Se basa en otros conceptos y estándares como Uniform Resource Identifier (URI), Uniform Resource Location (URL) y Uniform Resource Name

(URN), para indicar el recurso al que hace referencia la petición. Los mensajes se pasan con un formato similar al usado por el Internet Mail y el Multipurpose Internet Mail Extensions (MIME).

El protocolo HTTP se basa en un paradigma de peticiones y respuestas. Un cliente envía una petición en forma de método, una URI, y una versión de protocolo seguida de los modificadores de la petición de forma parecida a un mensaje MIME, información sobre el cliente y al final un posible contenido. El servidor contesta con una línea de estado que incluye la versión del protocolo y un código que indica éxito o error, seguido de la información del servidor en forma de mensaje MIME y un posible contenido.

Generalmente es el cliente el que inicia la comunicación HTTP y consiste en la petición de un recurso del servidor. Puede hacerse de forma directa al servidor o a través de intermediarios.

Se han utilizado los protocolos HTTP/0.9, HTTP/1.0 y HTTP/1.1.

Capitulo III

3 DNS("Domain Name System")

El protocolo DNS es un protocolo estándar (STD 13). Su status es recomendado. Es descrito en:

- RFC 1034 - Nombres de dominio - conceptos y servicios
- RFC 1035 - Nombres de dominio - implementación y especificación

Las configuraciones iniciales de Internet requerían que los usuarios emplearan sólo direcciones IP numéricas. Esto evolucionó hacia el uso de nombres de host simbólicos muy rápidamente. Por ejemplo, en vez de escribir `TELNET 128.12.7.14`, se podría escribir `TELNET eduvm9`, y eduvm9 se traduciría de alguna forma a la dirección IP 128.12.7.14. Esto introduce el problema de mantener la correspondencia entre direcciones IP y nombres de máquina de alto nivel de forma coordinada y centralizada.

Inicialmente, el NIC("Network Information Center") mantenía el mapeado de nombres a direcciones en un sólo fichero(HOSTS.TXT) que todos los hosts obtenían vía FTP. Se denominó espacio de nombres plano.

Debido al crecimiento explosivo del número de hosts, este mecanismo se volvió demasiado tosco(considerar el trabajo necesario sólo para añadir un host a Internet) y fue sustituido por un nuevo concepto: DNS("Domain Name System"). Los hosts pueden seguir usando un espacio de nombres local plano(el fichero HOSTS.LOCAL) en vez o además del DNS, pero fuera de redes pequeñas, el DNS es prácticamente esencial. El DNS permite que un programa ejecutándose en un host le haga a otro host el mapeo de un nombre simbólico de nivel superior a una dirección IP, sin que sea necesario que cada host tenga una base de datos completa de los nombres simbólicos y las direcciones IP.

3.1 Elementos del DNS

3.1.1 El espacio de nombres distribuido

El DNS usa el concepto de espacio de nombres distribuido. Los nombres simbólicos se agrupan en zonas de autoridad, o más comúnmente, zonas. En cada una de estas zonas, uno o más hosts tienen la tarea de mantener una base de datos de nombres simbólicos y direcciones IP y de suministrar la función de servidor para los clientes que deseen traducir nombres simbólicos a direcciones IP. Estos servidores de nombres locales se interconectan lógicamente en una árbol jerárquico de dominios. Cada zona contiene una parte del árbol o sub árbol y los nombres de esa zona se administran con independencia de los de otras zonas. La autoridad sobre zonas se

delega en los servidores de nombres. Normalmente, los servidores de nombres que tienen autoridad en zona tendrán nombres de dominio de la misma, aunque no es imprescindible. En los puntos en los que un dominio contiene un sub árbol que cae en una zona diferente, se dice que el servidor o servidores de nombres con autoridad sobre el dominio superior delegan autoridad al servidor o servidores de nombres con autoridad sobre los subdominios. Los servidores de nombres también pueden delegar autoridad en sí mismos; en este caso, el espacio de nombres sigue dividido en zonas, pero la autoridad para ambas las ejerce el mismo servidor . La división por zonas se realiza utilizando registros de recursos guardados en el DNS:

Registro SOA("Start of Authority")

Define el inicio de una zona

Registro NS("Name Server")

Marca el fin de una zona iniciada por un SOA y apunta a un servidor de nombres con autoridad sobre la zona siguiente

En este contexto, el comienzo de un dominio está más cerca a la raíz del árbol que a sus terminaciones. En la raíz, no puede haber servidores de nombres superiores para delegar autoridad: la autoridad para la raíz se deposita en un conjunto de servidores de nombres de la raíz.

El resultado de este esquema es:

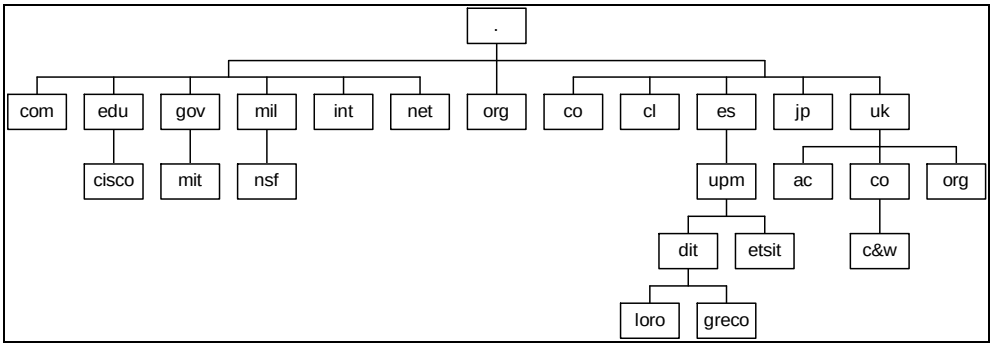
- En vez de tener un servidor central para la base de datos, el trabajo implicado en mantenerla se reparte entre los hosts a lo largo y ancho del espacio de nombres.
- La autoridad para crear y cambiar nombres simbólicos de hosts y la responsabilidad de mantener una base de datos para ellos le corresponde a la organización propietaria de la zona que los contiene.
- Desde el punto de vista del usuario, hay una sola base de datos que trata la resolución de las direcciones.

Aunque los dominios dentro del espacio de nombres se mapean con frecuencia a redes y subredes con el mecanismo de direccionamiento IP, este no es un requisito del DNS. Considerar un "router" entre dos subredes: tiene dos direcciones IP, una por cada adaptador, pero normalmente no tiene por qué poseer dos nombres simbólicos.

3.2 Espacio de nombres

El DNS organiza los nombres de máquina (hostname) en una jerarquía de dominios. Un dominio es una colección de nodos relacionados de alguna forma porque están en la misma red, tal como los nodos de una universidad.

En la siguiente figura vemos una parte del espacio de nombres. La raíz del árbol, que se identifica con un punto sencillo, es lo que se denomina dominio raíz y es el origen de todos los dominios. Para indicar que un nombre es FQDN, a veces se termina su escritura en un punto. Este punto significa que el último componente del nombre es el dominio raíz.



Dependiendo de su localización en la jerarquía, un dominio puede ser de primer nivel (top-level), segundo nivel o tercer nivel. Se pueden añadir todos los niveles que queramos, pero no son habituales. Los que siguen son los dominios de primer nivel que veremos con frecuencia:

Edu	Instituciones universitarias, casi todas norteamericanas.
Com	Organizaciones comerciales.
Org	Organizaciones no comerciales. Las redes privadas UUCP suelen estar en este dominio.
Net	Pasarelas y otras redes administrativas.
Mil	El ejército norteamericano.
Uucp	Dominio para redes UUCP.
Gov	El gobierno norteamericano.
Aero	Aerolíneas
Coop	Cooperativas
museum	Museos
Name	Personas y nombres personales
Pro	Profesionales
Biz	Comercial o de negocios
Info	Información

Inicialmente los cuatro primeros dominios de la lista anterior pertenecían solo a los Estados Unidos, sin embargo, los cambios de política posteriores han hecho que estos dominios, llamados de dominios globales primer nivel (gTLD) sean realmente globales. Además se están negociando nuevos dominios de primer nivel.

Fuera de los Estados Unidos, cada país suele tener su propio dominio de primer nivel codificado con las dos letras del país definidas en la tabla ISO-3166. Finlandia, por ejemplo, usa el dominio fi; en España se usa el dominio es; en México se usa mx; en Argentina, ar, Ecuador, ec, etc. Por debajo de cada dominio de primer nivel, cada país organiza los dominios a su manera. Algunos crean a segundo nivel una serie de dominios similares a los gTLD. Por ejemplo, en Ecuador encontramos los dominios com.ec para las empresas, y org.ec para las organizaciones sin ánimo de lucro. Otros países, como España, ponen directamente como nombres de segundo nivel las instituciones o empresas que los solicitan. Por ejemplo, tenemos hispalinux.es.

Por supuesto, el hecho de que un nombre esté en uno de estos dominios nacionales, no implica que la máquina esté realmente en ese país; significa simplemente que ha sido registrada en el NIC de ese país. Un fabricante sueco puede tener oficinas en Australia y tener sus ordenadores de allá registrados en el dominio se.

La organización del espacio de nombres en una jerarquía de nombres de dominio sirve para resolver fácilmente el problema de la unicidad de los nombres; además muchos nombres completamente cualificados son fáciles de recordar. Bajo esta premisa es conveniente dividir un dominio con gran número de máquinas en subdominios.

El sistema DNS hace más cosas. Permite delegar la autoridad de un subdominio a sus administradores. Por ejemplo, los responsables del Centro de investigaciones de la universidad politécnica pueden crear un subdominio para cada departamento, y delegar su control a éstos. Así, cada departamento puede definir libremente todos los nodos que quiera dentro de su subdominio e incluso crear nuevos subdominios y delegarlos.

3.3 Administración de nombres

3.3.1 Administración de dominios delegados

La Autoridad de Números Asignados en Internet (IANA) es responsable de la coordinación y mantenimiento del Sistema de Nombres de Dominio (DNS), y especialmente de la delegación de parte del espacio de nombres llamado dominios de nivel superior. La mayoría de los dominios de nivel superior son códigos de país de dos letras extraídos del estándar ISO 3166.

Se ha seleccionado y designado un Registro de Internet ("Internet Register, IR" o RI) central para manejar el grueso de la administración diaria del Sistema de Nombres de Dominio. Las peticiones de nuevos dominios de nivel superior (p.ej., dominios de código de país) las lleva el RI, con consultas de la IANA. El RI central es INTER NIC.NET. Los dominios de segundo nivel en COM, EDU, ORG, NET y GOV se registran a través del RI en InterNIC. Los dominios de segundo nivel en MIL los registra el registro DDN en NIC.DDN.MIL. Los nombres de segundo nivel en INT se registran a través de PVM en ISI.EDU.

Mientras que las peticiones de dominios de nivel superior deben enviarse a Internic (a hostmaster@internic.net), a veces se pide ayuda a los registros regionales para la administración del DNS, especialmente para solucionar problemas con la administración de un país. Actualmente, RIPE NCC es el registro regional de Europa y APNIC el de la región Pacífico-Asiática, mientras que INTERNIC administra la región de Norteamérica y todas las regiones sin delegación.

Los contactos para estos registros regionales son:

INTERNIC hostmaster@internic.net

APNIC hostmaster@apnic.net

RIPE NCC ncc@ripe.net

A continuación se describe la política a seguir en el establecimiento de un nuevo dominio de nivel superior. También se mencionan los aspectos que aparecen cuando es necesario cambiar la delegación de un dominio establecido de una parte a otra.

Normalmente se crea y se delega el mantenimiento de un nuevo dominio de orden superior a un "administrador designado", todo a la vez.

La mayoría de estos aspectos son relevantes cuando se delega un subdominio y en general se aplican estos principios recursivamente a todas las delegaciones del espacio de nombres del Sistema de Nombres de Dominio de Internet.

La mayor preocupación en la designación del administrador de un dominio es que sea capaz de cumplir con las responsabilidades necesarias y que tenga la capacidad de realizar un trabajo equitativo, justo, honesto y competente.

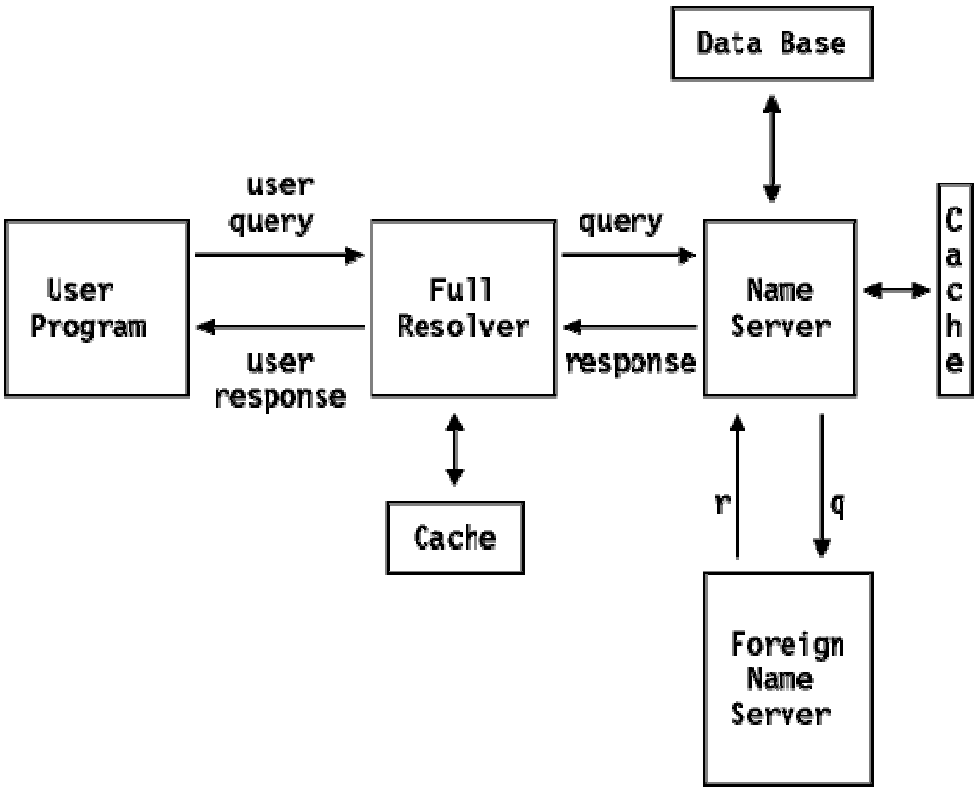
- El requisito clave es que para cada dominio haya un administrador asignado para supervisar el espacio de nombres de ese dominio. En el caso de dominios de orden superior que sean códigos de país esto significa que hay un administrador que supervisa los nombres de dominio y opera el sistema de nombres de dominio en ese país.
- Estas autoridades designadas son depositarios del dominio delegado y tienen el deber de servir a la comunidad.

- El administrador designado debe ser equitativo con todos los grupos en el dominio que soliciten nombres de dominio.
- Las partes del dominio interesadas deberían coincidir en que el administrador designado es el correcto.
- El administrador designado debe realizar un trabajo satisfactorio al operar el servicio DNS del dominio.
- Para cualquier transferencia de administración de una organización a otra, el administrador del dominio del nivel inmediatamente superior (la IANA en el caso de dominios de nivel superior) debe recibir comunicaciones de la organización antigua y de la nueva que asegure a la IANA que la transferencia ha sido aceptada por las dos partes y que la nueva organización acepta y conoce sus responsabilidades.

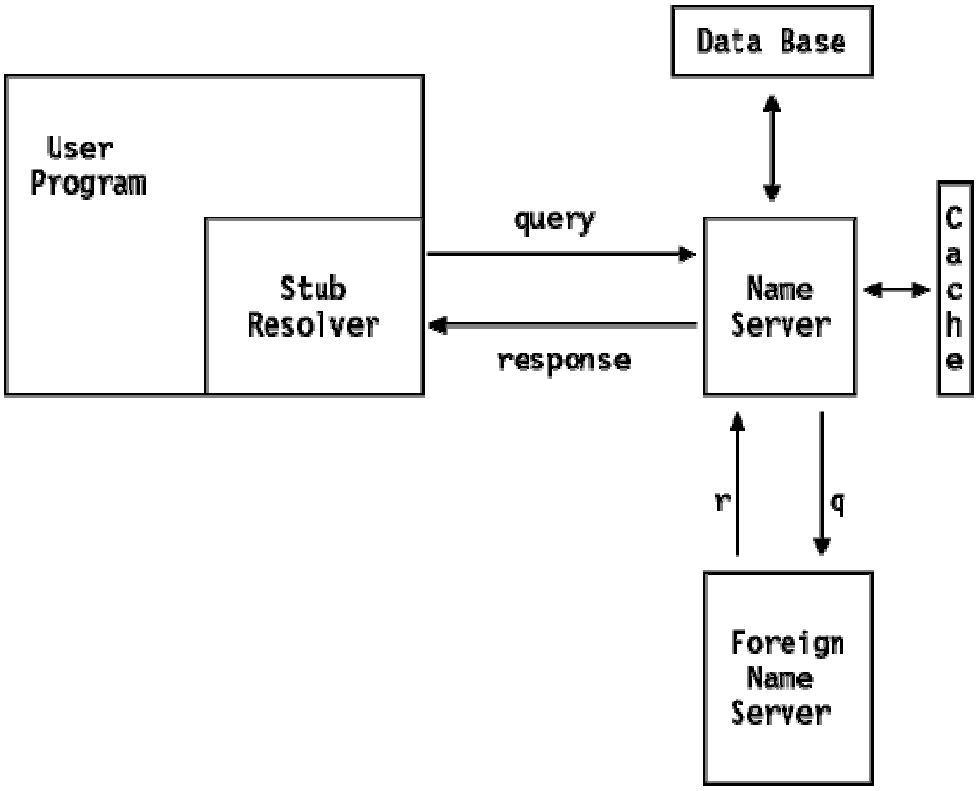
3.4 Resolución de Nombres

3.4.1 Proceso de Resolución

La resolución de nombres de dominio es un proceso cliente/ servidor . La función del cliente(el "resolver" o "name resolver") es transparente al usuario y la llama una aplicación para mapear nombres de alto nivel a direcciones IP o viceversa. El servidor de nombres(llamado también servidor de nombres de dominio) es una aplicación servidora que traduce los nombres de alto nivel de las máquinas a direcciones IP. El proceso básico se muestra a continuación usando un full resolver



y usando un stub resolver para la resolución de nombres de dominio.



En el primer gráfico se muestra un programa denominado "full resolver", distinto del programa de usuario , que envía todas las peticiones al servidor de nombres . El servidor de nombres cachea las respuestas para su uso en el futuro; posiblemente será él mismo el que las use.

El último muestra un "stub resolver", que es una rutina enlazada con el programa de usuario, que envía las peticiones a un servidor de nombres. El servidor de nombres suele cachear las respuestas, aunque esto depende de la implementación.

En UNIX, el "stub resolver" se implementa con dos funciones de librería: gethostbyname() y gethostbyaddr() para convertir nombres de hosts a direcciones IP y viceversa. Otras plataformas tienen rutinas iguales o parecidas. Los "stub resolvers" son mucho más comunes que los "full resolvers".

3.4.2 Registros de recursos del DNS

La base de datos distribuida del DNS se compone de RRs("resource records"). Estos proporcionan una mapeado entre nombres de dominio y objetos de red. Los objetos de red más comunes son las dirección de los hosts, pero el DNS está diseñado para acomodarse a una variada gama de distintos objetos. El formato general del registro de recurso es:

name ttl class type rdata

donde:

name

Es el nombre de dominio a definir. El DNS es muy general en las reglas de composición de nombres de dominio. Sin embargo, se recomienda una sintaxis para crearlos que minimiza la probabilidad de que las aplicaciones que usen un "resolver"(es decir, la mayoría de las aplicaciones TCP/IP) los malinterpreten. Un nombre de dominio que siga esta sintaxis consistirá en una serie de etiquetas formadas por caracteres alfanuméricos o guiones, cada etiqueta con una longitud de 1 a 63 caracteres, comenzando con un carácter alfabético. Cada par de etiquetas está separado por un punto en forma legible para el ojo humano, pero no en la misma forma que se usa dentro de los mensajes DNS. Los nombres de dominio no son sensibles a mayúsculas y minúsculas.

ttl

Es el "time-to-live" o tiempo en segundos que el registro será válido en la caché de un servidor de nombres. Se almacena en el DNS como un valor de 32 bits sin signo. 86400(un día) es un valor típico para registros que apuntan a una dirección IP.

class

Identifica la familia del protocolo. Valores habituales son:

IN

Sistema de Internet

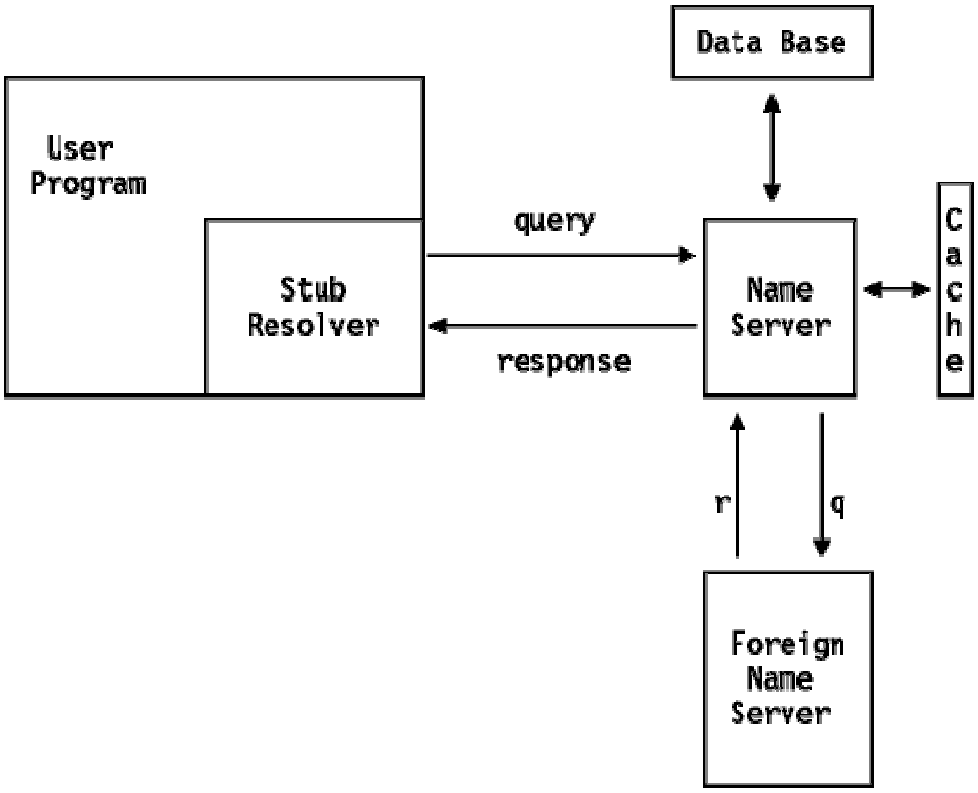
CH

Sistema Chaos

type

Identifica el tipo de recurso del registro.

Los diferentes tipos se describen en detalle en los RFCs 1034, 1035 y 1706. Cada tipo tienen un nombre y un valor. Valores corrientes incluyen:



Rdata

El valor depende del tipo, por ejemplo:

A

Un dirección IP de 32 bits(si la clase es IN)

CNAME

Un nombre de dominio

MX

Un valor por defecto de 16 bits(se prefieren valores bajos) seguido de un nombre de dominio.

NS

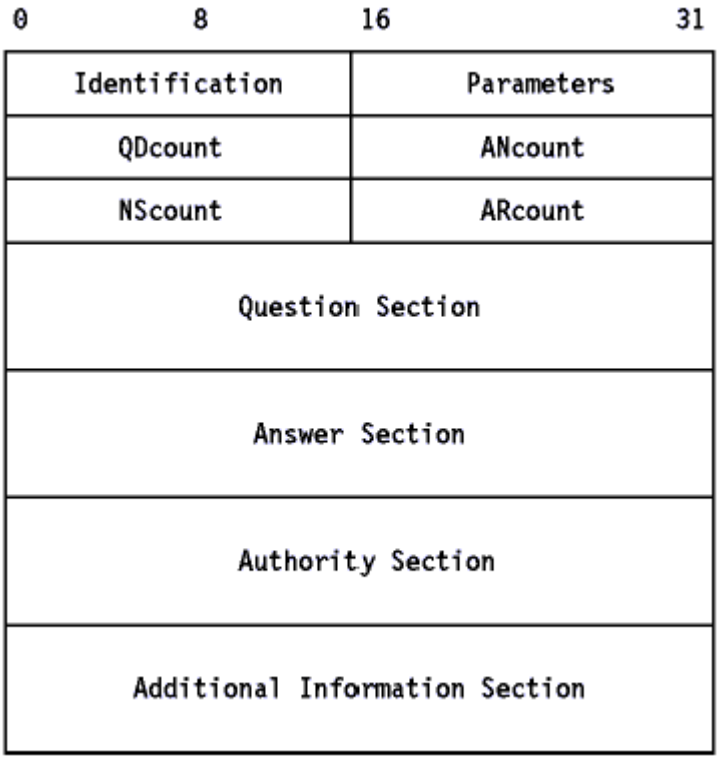
Un nombre de host.

PTR

Un nombre de dominio.

3.4.3 Mensajes del DNS

Todos los mensajes del DNS utilizan un único formato. El "resolver" envía la trama al servidor de nombres. Sólo la cabecera y la sección "question" se utilizan para la consulta. Las respuestas o retransmisiones de las consultas usan la misma trama, pero llenan más secciones de la misma(las secciones "answer/authority/additional").



3.4.3.1 Formato de la cabecera

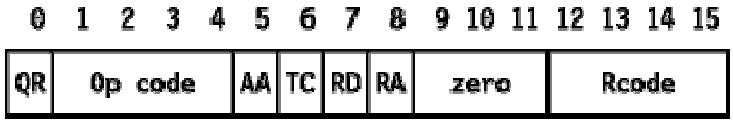
La sección de cabecera siempre ha de aparecer y tiene una longitud fija de 12 bytes. Las otras secciones son de longitud variable.

ID

Un identificador de 16 bits asignado por el programa. Este identificador se copia en la respuesta correspondiente del servidor de nombres y se puede usar para diferenciar respuestas cuando concurren múltiples consultas.

Parameters

Un valor de 16 bits con el siguiente formato:



QR

Flag que indica consulta(0) o respuesta(1)

Op code

Campo de 4-bit especificando el tipo de consulta:

0

consulta estándar(QUERY)

1

consulta inversa(IQUERY)

2

solicitud del estado del servidor(STATUS) Se reservan los otros valores para su uso en el futuro

AA

Flag de respuesta autoritativa. Si está activo en una respuesta, especifica que el servidor de nombres que responde tiene autoridad para el nombre de dominio enviado en la consulta.

TC

Flag de truncado. Activo si el mensaje es más largo de lo que permite el canal.

RD

Flag de recursividad. Este bit indica al servidor de nombres que se pide resolución recursiva. El bit se copia en la respuesta.

RA

Flag de recursividad disponible. Indica si el servidor de nombres soporta resolución recursiva.

zero

3 bits reservados para uso futuro. Deben ser cero.

Rcode

Código de respuesta de 4 bits. Posibles valores son:

0

Ningún error.

1

Error de formato. El servidor fue incapaz de interpretar el mensaje.

2

Fallo en el servidor. El mensaje no fue procesado debido a un problema con el servidor.

3

Error e nombre. El nombre de dominio de la consulta no existe. Sólo válido si el bit AA está activo en la respuesta.

4

No implementado. El tipo solicitado de consulta no está implementado en el servidor de nombres.

5

Rechazado. El servidor rechaza responder por razones políticas. Los demás valores se reservan para su usuario en el futuro.

QDcount

Un entero sin signo de 16 bits que especifica el número de entradas en la sección "question".

ANcount

Un entero sin signo de 16 bits que especifica el número de RRs en la sección "answer".

NScount

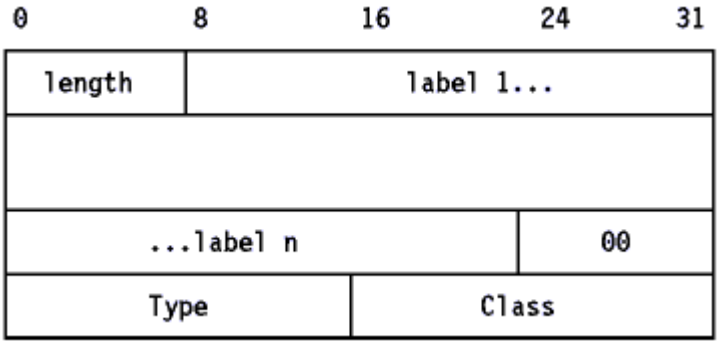
Un entero sin signo de 16 bits que especifica el número de RRs en la sección "authority".

ARcount

Un entero sin signo de 16 bits que especifica el número de RRs en la sección "additional records".

3.4.3.2 Sección "Question"

La siguiente sección contiene las consultas al servidor de nombres. Contiene QDcount(generalmente 1) entradas, cada una con el formato mostrado a continuación.



length

Un byte que indica la longitud de la siguiente etiqueta.

label

Un elemento del nombre de dominio. El nombre de dominio se almacena como una serie de etiquetas de longitud variable, cada una precedida por un campo "length".

00

X'00' indica el fin del dominio y representa la etiqueta nula del dominio raíz.

Type

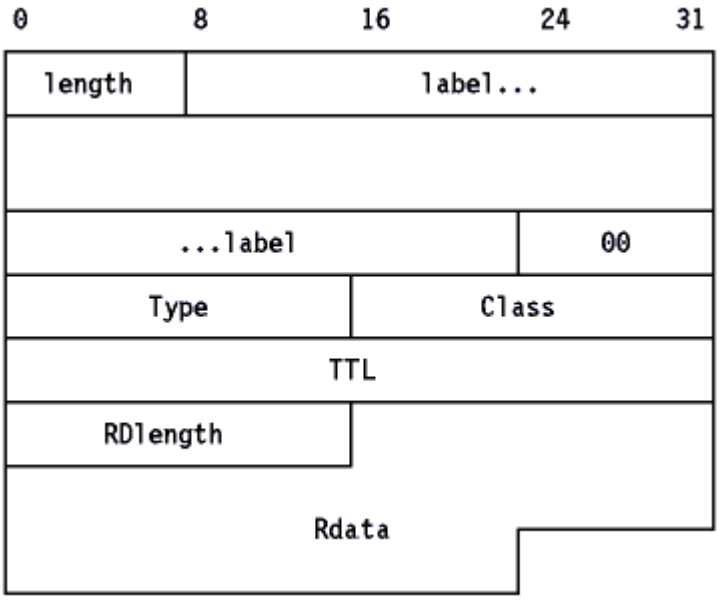
2 bytes especificando el tipo de consulta. Puede tener cualquier valor del campo "Type" del registro.

Class

2 bytes especificando la clase de consulta. Para consultas en Internet, será "IN".

3.4.3.3 Secciones "Answer", "Authority" y "Additional Resource"

Estas tres secciones contienen un número variable de registros de recursos. El número se especifica en el campo correspondiente de la cabecera. Los registros tienen el formato mostrado en la siguiente figura.



Los campos que preceden al TTL tienen el mismo significado que en la sección "question" y:

TTL

TTL de 32-bit medido en segundos. Define cuánto tiempo se puede considerar válido el recurso.

RDlength

Longitud de 16 bits para el campo Rdata.

Rdata

Ristra de longitud variable cuya interpretación depende del campo "Type".

3.4.3.4 Compresión de mensajes

Con el fin de reducir el tamaño del mensaje, se utiliza un esquema de compresión para eliminar la repetición de nombres de dominio en los diversos RRs. Cualquier dominio o lista de etiquetas duplicada se sustituye por un puntero a la ocurrencia anterior. El puntero tiene la forma de un campo de 2 bytes:

- Los primeros 2 bits distinguen al puntero de una etiqueta normal, que está restringida a una longitud de 63 bytes además de el byte de longitud(con el valor de <64).
- El campo de "offset especifica un desplazamiento desde el comienzo el mensaje. Un "offset" de cero especifica el primer byte del campo ID de la cabecera.
- Si se usa compresión en un campo en el campo "Rdata" de una sección "answer", "authority" o "additional", el campo "RDlength" precedente contiene, después de haber efectuado la compresión, la longitud real .

Transporte

Los mensajes DNS se transmiten como datagramas(UDP) o sobre un canal(TCP).

UDP

Puerto del servidor: 53 (decimal).

Los mensajes transportados por UDP se restringen a 512 bytes. Los mensajes más largos se truncan y el bit TC de la cabecera se activa. Ya que las tramas UDP se pueden perder, hace falta una estrategia de retransmisión.

TCP

Puerto del servidor: 53 (decimal).

En este caso, el mensaje va precedido de un campo de 2 bytes que indica la longitud total de la trama.

El STD 3 - Requerimientos del host requiere que:

- Un "resolver" del DNS o un servidor que envía una consulta que no supone una transferencia de zona debe enviar una consulta UDP primero. Si la sección "answer" de la respuesta está truncada y el solicitante soporta TCP,

debería intentarlo de nuevo usando TCP. Se prefiere UDP a TCP para las consultas porque UDP tiene un factor de carga mucho menor, y su uso es esencial para un servidor fuertemente cargado. El truncamiento de mensajes no suele ser un problema dados los contenidos actuales de la base de datos del DNS, ya que típicamente se pueden enviar en un datagrama 15 registros, pero esto podría cambiar a medida que se añaden nuevos tipos de registro al DNS.

- TCP debe usarse para actividades de transferencia de zonas debido a que el límite de 512 bytes de UDP siempre será inadecuado para una transferencia de zona.
- Los servidores de nombres deben soportar ambos tipos de transporte.

Capitulo IV

4 Correo Electrónico

4.1 Protocolos de correo Electrónico.

El correo electrónico de Internet se implementó originalmente como una función del protocolo FTP. En 1980 Suzanne Sluizer y Jon Postel realizaron trabajos con un protocolo que posteriormente se denominaría SMTP ("Simple Mail Transfer Protocol") [9]. Hoy en día se sigue utilizando este protocolo, con los avances lógicos que requiere el tipo de transferencia actual.

El protocolo SMTP fue desarrollado pensando en que los sistemas que intercambiarían mensajes, eran grandes computadoras, de tiempo compartido y multiusuario conectados permanentemente a la red Internet. Sin embargo, con la aparición de los computadoras personales, que tienen una conectividad ocasional, se hizo necesaria una solución para que el correo llegase a estos equipos. Para solventar esta limitación, en 1984 surge POP ("Post Office Protocol").

Este protocolo, en su especificación inicial, solo permite funciones básicas como recuperar todos los mensajes, mantenerlos en el servidor y borrarlos.

En sucesivas versiones del protocolo (POP2 y POP3) se han ampliado las funciones, permitiendo una mejor gestión del correo.

Estos dos protocolos son los encargados de transportar el correo por toda la red Internet, pero sólo son capaces de transportar mensajes en formato texto ASCII. Para superar esta limitación, se utilizaba hasta hace poco tiempo, programas como UUEncode y UUDecode.

En 1992, surge MIME ("Multipurpose Internet Mail Extensions"), que permite el correo electrónico en otras lenguas que no sea el inglés, además de la transmisión de sonido, gráficos, vídeo, etc. En la actualidad el estándar MIME, es el que se utiliza para la transferencia de datos no tradicionales a través de correo electrónico.

4.1.1 SMTP

SMTP está basado en *la entrega punto-a-punto*; un cliente SMTP contactará con el servidor SMTP del host de destino directamente para entregar el correo. Guardará el correo hasta que se haya copiado con éxito en el receptor. Esto difiere del principio de retransmisión común a muchos sistemas de correo en las que el correo atraviesa un número de host intermedios de la misma red y donde una transmisión con éxito implica sólo que el correo ha alcanzado el host correspondiente al siguiente salto.

En varias implementaciones, existe la posibilidad de intercambiar correo entre los sistemas de correo locales y SMTP. Estas aplicaciones se denominan *pasarelas o puentes de correo*. Enviar correo a través de una pasarela puede alterar la entrega punto-a-punto, ya que SMTP sólo garantiza la entrega fiable a la pasarela, no al host de destino, más allá de la red local. La transmisión punto SMTP en estos casos es host-pasarela, pasarela-host o pasarela-pasarela; SMTP no define lo que ocurre más allá de la pasarela. CSNET proporciona un interesante ejemplo de servicio de pasarela de correo. Diseñada en principio como un servicio barato para interconectar centros científicos y de investigación, CSNET opera una pasarela que permite a sus suscriptores enviar y recibir correo en Internet con sólo un módem con dial. La pasarela sondea a los suscriptores a intervalos regulares, les entrega su correo y recoge el correo de salida. A pesar de no ser una entrega punto-a-punto, ha demostrado ser un sistema muy útil.

Cada mensaje tiene:

- Una cabecera, o sobre, con estructura RFC 822. La cabecera termina con una línea nula(una línea con sólo la secuencia <CRLF>).
- Contents

Todo lo que hay tras la línea nula es el cuerpo del mensaje, una secuencia de líneas con caracteres ASCII(aquellos con valor menor del 128 decimal).

El RFC 821 define un protocolo cliente/servidor. Como siempre, el cliente SMTP es el que inicia la sesión (el emisor) y el servidor el que responde a la solicitud de sesión(el receptor). Sin embargo, como el cliente suele actuar como servidor para un programa de correo del usuario, es más sencillo referirse a él como emisor SMTP, y al servidor como receptor SMTP.

4.1.2 POP("Post Office Protocol")

Debido a que un receptor de correo SMTP es un servidor, y SMTP es una aplicación punto-a-punto más que de retransmisión, es necesario que el servidor esté disponible cuando un cliente desea enviarle correo. Si el servidor SMTP reside en una estación de trabajo o en un PC, ese host debe estar ejecutando el cuando el cliente quiera transmitir. Esto no suele ser un problema en sistemas multiusuario porque están disponibles la mayor parte del tiempo. En sistemas monousuario, sin embargo, este no es el caso, y se requiere un método para asegurar que el usuario tiene un buzón disponible en otro servidor. Hay varias razones por las que es deseable descargar a la estación de trabajo de las funciones del servidor de correo, entre ellas la falta de recursos en estaciones de trabajo pequeñas, la falta o encarecimiento de la conectividad TCP, etc.

La estrategia más simple es, por supuesto, usar un sistema multiusuario para las funciones de correo, pero esto no suele ser deseable -- quizá el usuario no lo va a usar para nada más, o quiere tener acceso a Alternativamente, el usuario final puede ejecutar un cliente que comunique con un programa servidor en un host. Este

servidor actúa tanto como emisor como receptor. Recibe y envía el correo del usuario.

Un método intermedio es descargar la función de servidor SMTP de la estación de trabajo del usuario final, pero no la función de cliente. Es decir, el usuario envía correo directamente desde la estación, pero tiene un buzón en un servidor. El usuario debe conectar con el servidor para recoger su correo.

El POP describe cómo un programa que se ejecuta en una estación de trabajo final puede recibir correo almacenado en sistema servidor de correo. POP usa el término "maildrop" para referirse a un buzón gestionado por un servidor POP. POP 3 es un borrador y su status es s*elective*. Las versión es un *protocolo histórico* con status *no recomendado*.

4.1.3 MIME ("Multipurpose Internet Mail Extensions")

El E-mail es probablemente la aplicación TCP/IP más usada. Sin embargo, SMTP se limita a texto ASCII de 7 bits con una longitud de línea máxima de 100 caracteres lo que da lugar a diversas limitaciones.

- SMTP no puede transmitir ficheros ejecutables u otros objetos binarios. Hay un número de métodos ad hoc para encapsular objetos binarios en correos SMTP, por ejemplo:

  - Codificar el fichero en formato hexadecimal puro.
  - Las utilidades UNIX UUencode y UUdecode usadas para codificar datos binarios en el sistema de correo UUCP para superar las limitaciones del formato de 7 bits.
  - La representación Andrew Toolkit.

  Ninguna de ellas se puede considerar un estándar de facto. UUencode es quizás la más extendida debido a que los sistemas UNIX son los pioneros de Internet.

- SMTP no puede transmitir texto que incluya caracteres nacionales, ya que estos se representan con valores iguales o mayores de 128 en los juegos de caracteres basados en ASCII.
- Los servidores SMTP pueden rechazar los correos mayores de un tamaño concreto. Cualquier servidor puede tener límites permanentes y/o temporales a la máxima cantidad de correo que puede aceptar de un cliente en un momento dado.
- Las pasarelas SMTP que traducen de ASCII a EBCDIC y viceversa no emplean un conjunto consistente de mapeados de páginas de código, dando lugar a problemas de traducción.
- Las pasarelas SMTP a redes X.400 no pueden manejar datos que no sean texto en los correos de X.400. Los estándares para mapear X.400 al STD 11/RFC 822 anterior al MIME requieren que las partes que no sean texto del

cuerpo de un mensaje X.400 se conviertan o se desechen y el receptor RFC 822 informe que se ha desechado esa información.

Obviamente, esto no es deseable ya que se trata de información presumiblemente importante aunque el sistema de correo no la pueda entender. Desecharla significa que será inaccesible al usuario. Convertirla a ASCII implica poner a cero el byte de orden superior, lo que supone la corrupción de los datos más haya de cualquier forma de recuperación. El problema es particularmente agudo en el caso de que emisor y receptor se hallen en redes X.400 pero el correo pase por una red RFC 822 intermedia("tunnelling") ya que los usuarios de X.400 esperan recibir correos X.400 sin perder información.

- Algunas implementaciones de SMTP u otros MTAs("Mail Transport Agents") de Internet no se adhieren por completo al estándar SMTP definido en el RFC 821. Problemas habituales son:

    - Eliminación de espacios en blanco terminales(TABs y SPACEs)
    - Relleno de todas las líneas de un mensaje para que tengan la misma longitud.
    - Empaquetado de las líneas de más de 76 caracteres.
    - Cambio de secuencias de nueva línea para diferentes convenios(por ejemplo, los caracteres <CR> se pueden convertir a secuencias<CRLF>).
    - Conversión de caracteres TAB a múltiples caracteres SPACE.

MIME es un estándar que incluye mecanismos para resolver estos problemas en una forma con un alto grado de compatibilidad con los estándares RFC 822. Debido a que los mensajes de correo suelen enviarse a través de pasarelas de correo, a un cliente SMTP no le es posible distinguir entre un servidor que gestiona el buzón del receptor y uno que actúa como pasarela a otra red. Como el correo que atraviesa una pasarela se puede dirigir a otras pasarela, que pueden usar distintos protocolos de mensajería, en el caso general al emisor no le es posible determinar el mínimo común denominador de las capacidades de cada una de las etapas que atraviesa el mensaje. Por este motivo, MIME asume el peor caso: transporte ASCII de 7 bits que puede no seguir estrictamente el RFC 821. No define ninguna extensión al RFC 821, sino que se limita sólo a las extensiones incluidas en el RFC 822. De esta forma, un mensaje MIME puede ser enrutado a través de cualquier número de redes capaces de transmitir mensajes RFC 821. Se describe en dos partes:

- Protocolos para incluir objetos distintos de los mensajes de correo US ASCII. dentro de mensajes RFC 822. El RFC 1521 los describe.
- Un protocolo para codificar texto no US ASCII en los campos de cabecera según el RFC 822. El RFC 1522 lo describe.

Aunque el RFC 1521 proporciona un mecanismo adecuado para describir dato de texto de mensajes X.400 en una forma no compatible con el RFC 822, no

dice como se han de mapear las partes de los mensajes X.400 a las de los mensajes MIME. Esta conversión está definida en los RFCs 1494, 1495 y 1496 que actualizan los protocolos de conversión de RFC 822 a X.400.

4.2 Formato de Mensajes.

4.2.1 Formato de la cabecera

Normalmente, el usuario no tiene por qué preocuparse de la cabecera, que es responsabilidad de SMTP.

El RFC 822 contiene un análisis completo de la cabecera. La sintaxis es BNF("Backus-Naur Form") extendida. El RFC 822 contiene una descripción de BNF, y muchos RFCs relacionados usan el mismo formato. Además describe como convertir una cabecera a su *forma canónica*, uniendo las líneas de continuación, los espacios no significativos, los comentarios, etc. Es una sintaxis poderosa, pero relativamente difícil de analizar. Aquí se incluye una breve descripción.

Brevemente, la cabecera es una lista de líneas de la forma:

field-name: field-value

Los campos comienzan en la columna 1: las líneas que comienzan con caracteres en blanco(SPACE o TAB) son líneas de continuación que se unen para crear una sola línea para cada campo en la forma canónica. Las cadenas entre comillas ASCII señalan que los caracteres especiales que limitan no son significativos sintácticamente. Muchos valores importantes(como los de los campos "To" y "From") son buzones. Las formas más corrientes para estos son:

- Isim0234@master.etsit.upm.es
- Alumno <isim0234@master.etsit.upm.es>
- "Alumno" <isim0234@master.etsit.upm.es>

La cadena "Alumno" ha de ser leída por receptores humanos y es el nombre del propietario del buzón. "isim0234@master.etsit.upm.es" es la dirección para la máquina del buzón(el > y el < delimitan la dirección pero no forman parte de ella). Se ve que esta forma de direccionamiento está relacionada con DNS. De hecho, el cliente SMTP utiliza el DNS para determinar la dirección de destino del buzón.

Algunos campos habituales son:

keyword valor

to Receptores primarios del mensaje.

cc Receptores Secundario("carbon-copy") del mensaje.

from Identidad del emisor.

reply-to El buzón al que se han de enviar las repuestas. Este campo lo añade el emisor.

return-path Dirección y ruta hasta el emisor. Lo añade el sistema de transporte final que entrega el correo.

Subject Resumen del mensaje. Suele proporcionarlo el usuario.

4.3 Entrega de Mensajes

4.3.1 SMTP y el DNS

Si la red usa el concepto de dominio, un SMTP no puede entregar simplemente correo a master.etsit.upm.es abriendo una conexión TCP con master.etsit.upm.es. Primero debe consultar al servidor de nombres para hallar a que host(en un nombre de dominio) debería entregar el mensaje.

Para la entrega de mensajes, el servidor de nombres almacena los RRs("resource records") denominados MX RRs. Mapean un nombre de dominio a dos valores:

- Un valor de preferencia. Como pueden existir múltiples RRs MX para el mismo nombre de dominio, se les asigna una prioridad. El valor de prioridad más bajo corresponde al registro de mayor preferencia. Esto es útil siempre que el host de mayor preferencia sea inalcanzable; el emisor SMTP intenta conectar con el siguiente host en orden de prioridad.
- Un nombre de host.

También es posible que el servidor de nombres responda con una lista vacía de RRs MX. Esto significa que el nombre de dominio se halla bajo la autoridad del servidor, pero no tiene ningún MX asignado. En este caso, el emisor SMTP puede intentar establecer la conexión con el mismo nombre del host.

El RFC 974 da una recomendación importante. Recomienda que tras obtener los registros MX, el emisor SMTP debería consultar los registros WKS(*Well-Known Services*) del host, y chequear que el host referenciado tiene como entrada WKS a SMTP.

Esto es sólo una opción del protocolo, aunque aparece en numerosas implementaciones.

ejemplo de RRs MX:

```
fsc5.stn.mlv.fr.    IN   MX 0  fsc5.stn.mlv.fr.
                    IN   MX 2  psfred.stn.mlv.fr.
                    IN   MX 4  mvs.stn.mlv.fr.
                    IN   WKS   152.9.250.150 TCP (SMTP)
```

En el ejemplo anterior, el correo para fsc5.stn.mlv.fr debería, por prioridad, ser entregado al propio host, pero en caso de que el host sea inalcanzable, el correo también podría ser entregado a psfred.stn.mlv.fr o a mvs.stn.mlv.fr (si psfred.stn.mlv.fr no se pudiera alcanzar tampoco).

4.3.2 Direccionando buzones en servidores

Cuando un usuario emplea un servidor para las funciones de correo, la dirección del buzón que ven otros usuarios SMTP se refiere exclusivamente al servidor. Por ejemplo, si dos sistemas se llaman:

- hayes.itso.ral.ibm.comando y
- itso180.itso.ral.ibm.comando

usándose el primero como cliente y el segundo como servidor, la dirección de correo podría ser:

- hayes@itso180.itso.ral.ibm.comando

Esta dirección de buzón aparecería en el campo "From:" de la cabecera de todo el correo saliente y en los comandos SMTP a servidores remotos lanzados por el servidor.

Sin embargo, cuando el usuarios emplea un servidor POP, la dirección de correo contiene el nombre de host de la estación de trabajo(por ejemplo xxx@hayes.itso.ral.ibm.comando). En este caso, el emisor debería incluir un campo "Reply-To:" en la cabecera para indicar que las réplicas *no* se deberían enviar al emisor.

Por ejemplo, la cabecera podría tener este aspecto:

```
Date: Fri, 10 Feb 95 15:38:23
From: xxxx@hayes.itso.ral.ibm.comando
To: "Xxxx " <tsgsh@gford1.warwick.uk.ibm.comando>
Reply-To: hayes@itso180.itso.ral.ibm.comando
Subject: Test Reply-To: header field
```

Se espera que el agente de correo envíe las respuestas a la dirección "Reply-To:" y no a "From:".

4.3.3 DNS para dirigir correo

Una alternativa al uso del campo "Reply-To:" es usar el DNS para dirigir el correo al buzón correcto. El administrador del DNS con autoridad para el dominio que contiene la estación del usuario y el servidor de nombres pueden añadir registros MX al DNS para dirigir el correo. Por ejemplo, los siguientes registros MX indican a los clientes SMTP que, si el servidor SMTP en hayes.itso.ral.ibm.comando no está disponible, hay un servidor de correo en itso.180.ral.ibm.comando (9.24.104.180) que se debería usar en su lugar.

```
itso180.itso.ral.ibm.comando.  IN     WKS  9.24.104.180 TCP (SMTP)

hayes.itso.ral.ibm.comando.    IN     MX 0 hayes.itso.ral.ibm.comando.
                               IN     MX 1 itso180.itso.ral.ibm.comando.
```

Capitulo V

5 El servicio WWW

La World Wide Web (WWW) es un sistema de publicación y distribución electrónica de información basado en Hipertexto. El WWW es la herramienta de mas potencia para el acceso a la red Internet. Desde su aparición, se ha generado una extraordinaria actividad económica alrededor del WWW. Las aplicaciones se multiplican todos los días alcanzando en la actualidad desde el acceso a mapas de predicción meteorológica, o a información de tráfico, o guías turísticas, servicios de soporte técnico de las principales empresas del sector informático, publicidad e información comercial. Se bautizado como Web a la colección de todos los documentos Hipertexto de la Internet a los que se puede acceder mediante los visores de WWW.

Es EE.UU. donde se ha desarrollado con mayor velocidad el nuevo mercado de aplicaciones telemáticas en torno al WWW. Han aparecido productos para la instalación de servidores de WWW, editores de HTML (el lenguaje en el que se escriben los documentos del WWW), visores y clientes para el acceso a la información. El crecimiento del sector ha provocado la aparición de empresas especializadas en la Ingeniería del WWW, desarrollo de instalaciones, gabinetes de diseño especializados en la distribución electrónica de información y empresas que ofrecen alquiler de espacio telemático para soportar servidores WWW de terceros.

Como se ha dicho, el WWW es un sistema de distribución de información que permite la presentación de información multimedia: texto mas gráficos mas sonido a usuarios que acceden a la misma a través de una conexión telemática. La información se organiza en forma de hipertexto, es decir, unos documentos enlazan con otros a través de palabras o iconos señalados en el documento. Esto permite diseñar interfaces de usuario avanzados, superiores en facilidad de manejo a los basado en menús o comandos. Es la tendencia que siguen todas las herramientas de autor multimedia o las documentación on-line que incorporan las nuevas aplicaciones.

El servicio WWW es bidireccional, no solo podemos recuperar información del servidor, sino que también el usuario puede enviar información. De esta forma se permiten las consultas a bases de datos, la introducción de información personal, la emisión de ordenes de compra, o la autorización de operaciones.

Las especificaciones del WWW son de dominio público y se mantienen por el organismo de normalización internacional IETF (Internet Engineering Task Force) dependiente del ISO (International Standard Organization). Esto unido a la penetración alcanzada por el sistema dentro de la red Internet y en países como EE.UU.; Canadá, Reino Unido o Alemania, establecen al WWW como estándar de hecho para la distribución electrónica de información.

5.1 Elementos

5.1.1 URL. (“Universal Resource Locator”)

URL (“Universal Resource Locator”) es el mecanismo con el cual el World Wide Web asigna una dirección única a cada uno de los recursos de información de cualquier lugar de la Internet. Existe un URL único para cada página de todos los documentos HTML de la Internet, para cada archivo en un directorio de acceso público de la Internet, para todos los elementos del Gopher y todos los grupos de debate USENET.

El URL combina el "path" y el nombre del archivo con el nombre de la máquina que sirve el archivo a la red, así como el protocolo a usar para recuperar los datos.

El formato general de un URL es

servicio://*dirección de la máquina:puerto*/*directorio/archivo*

5.1.2 URLs para archivos

El nombre del servicio en el URL de archivos es file. Un URL de un archivo identifica un archivo en un servidor de ftp anonymous. No se puede acceder a través del URL a archivos en directorios privados de una máquina ya que el acceso se realiza a través de los servidores de ftp.

file://ftp.yoyodyne.com/pub.files.foobar.txt

si no indicamos el directorio en el URL

file://ftp.yoyodyne.com/

este hace referencia al archivo. También se puede indicar el directorio sin nombre de archivo y el URL hará referencia al directorio indicado.

5.1.3 URL para Gopher

Gopher indica el servicio Gopher en un URL.

gopher://gopher.yoyodyne.com/

el URL hace referencia el menú de entrada de un Gopher. Algunos servidores Gopher utilizan un puerto TCP distinto al estándar. El URL permite indicar el puerto del servicio a continuación del nombre de la máquina y separando ambas informaciones por dos puntos.

gopher://gopher.yoyodyne.com:1234/

5.1.4 URL para noticias

Los URL también pueden indicar grupos de noticias. El URL utiliza la palabra de servicio news seguida por el nombre del grupo separado por dos puntos.

news:alt.com.windows

El cliente de WWW accederá al servidor de noticias indicado en la configuración y accederá al grupo de noticias. Los grupos de noticias son iguales en toda la Internet independientemente del servidor a través del que se acceda. Por eso no es necesario indicar la máquina en el URL.

5.1.5 URL para HTTP

http es el nombre de servicio en los URLs que hacen referencia a páginas HTML. HTTP es el nombre del protocolo que permite el intercambio de información hipertexto en la Internet.

El URL se compone de una dirección de maquina, directorio y nombre del archivo. Los archivos de HTML tienen extensión html. Al igual que en el caso de los URLs de archivos, el camino del archivo puede no ser el camino absoluto, ya que el servidor de WWW procesa las consultas.

http://www.yoyodyne.com/pub/files/foobar.html

5.1.6 URL parciales

Una vez el cliente ha accedido a una documento HTML pueden utilizarse URL relativos a esta página para hacer referencia a otras páginas o recursos de información. Si el documento esta en la misma máquina que el primer documento no es necesario indicar la máquina en el URL. Igualmente si el documento esta en el mismo directorio que el anterior no es necesario indicar el directorio en el URL. En estos casos no se indica el nombre del servicio y únicamente el nombre del archivo sirve para direccionarlo. Este es un mecanismo conveniente para la edición de documentos HTML con muchos enlaces.

5.1.7 El lenguaje HTML

El servicio WWW utiliza una organización de los documentos en forma de hipertexto. El hipertexto es un método de organización de la información en el cual los diferentes elementos de información se enlazan a través de elementos del propio texto. El usuario a partir de una página de portada hojea el documento seleccionando elementos del propio documento que le transportan a otras partes del documento donde se desarrollan los conceptos con mayor extensión, y así sucesivamente. De esta forma los documentos no tienen una única lectura sino que el lector accede a

aquella información que le interesa siguiendo una línea de lectura personal. El formato hipertexto se esta utilizando con mucho éxito en aplicaciones tales como documentación técnica de productos o manuales de ayuda on-line.

El lenguaje HTML (Hiper Text Mark-up Languaje) permite construir los documentos hipertexto del WWW. El HTML es un lenguaje de marcas, es decir, se añaden marcas a los documentos que se presentan en el WWW que definen la presentación gráfica de los documentos y los enlaces entre las páginas del mismo. Los recursos de la presentación incluyen resaltados, separación de párrafos, negrita, subrayado, etc.

5.2 El protocolo http

El Protocolo de Transferencia de HiperTexto (Hypertext Transfer Protocol) es un sencillo protocolo cliente-servidor que articula los intercambios de información entre los clientes Web y los servidores HTTP. La especificación completa del protocolo HTTP 1/0 está recogida en el RFC 1945. Fue propuesto por Tim Berners-Lee, atendiendo a las necesidades de un sistema global de distribución de información como el World Wide Web.

Desde el punto de vista de las comunicaciones, está soportado sobre los servicios de conexión TCP/IP, y funciona de la misma forma que el resto de los servicios comunes de los entornos UNIX: un proceso servidor escucha en un puerto de comunicaciones TCP (por defecto, el 80), y espera las solicitudes de conexión de los clientes Web. Una vez que se establece la conexión, el protocolo TCP se encarga de mantener la comunicación y garantizar un intercambio de datos libre de errores.

HTTP se basa en sencillas operaciones de solicitud/respuesta. Un cliente establece una conexión con un servidor y envía un mensaje con los datos de la solicitud. El servidor responde con un mensaje similar, que contiene el estado de la operación y su posible resultado. Todas las operaciones pueden adjuntar un objeto o recurso sobre el que actúan; cada objeto Web (documento HTML, fichero multimedia o aplicación CGI) es conocido por su URL.

Las principales características del protocolo HTTP son:

- Toda la comunicación entre los clientes y servidores se realiza a partir de caracteres de 8 bits. De esta forma, se puede transmitir cualquier tipo de documento: texto, binario, etc., respetando su formato original.
- Permite la transferencia de objetos multimedia. El contenido de cada objeto intercambiado está identificado por su clasificación MIME.
- Existen tres verbos básicos (hay más, pero por lo general no se utilizan) que un cliente puede utilizar para dialogar con el servidor: GET, para recoger un objeto, POST, para enviar información al servidor y HEAD, para solicitar las características de un objeto (por ejemplo, la fecha de modificación de un documento HTML).

- Cada operación HTTP implica una conexión con el servidor, que es liberada al término de la misma. Es decir, en una operación se puede recoger un único objeto.
- No mantiene estado. Cada petición de un cliente a un servidor no es influida por las transacciones anteriores. El servidor trata cada petición como una operación totalmente independiente del resto.
- Cada objeto al que se aplican los verbos del protocolo está identificado a través de la información de situación del final de la URL.

5.2.1 Etapas de una transacción HTTP

Cada vez que un cliente realiza una petición a un servidor, se ejecutan los siguientes pasos:

- Un usuario accede a una URL, seleccionando un enlace de un documento HTML o introduciéndola directamente en el campo Location del cliente Web.
- El cliente Web descodifica la URL, separando sus diferentes partes. Así identifica el protocolo de acceso, la dirección DNS o IP del servidor, el posible puerto opcional (el valor por defecto es 80) y el objeto requerido del servidor.
- Se abre una conexión TCP/IP con el servidor, llamando al puerto TCP correspondiente.
- Se realiza la petición. Para ello, se envía el comando necesario (GET, POST, HEAD,…), la dirección del objeto requerido (el contenido de la URL que sigue a la dirección del servidor), la versión del protocolo HTTP empleada (casi siempre HTTP/1.0) y un conjunto variable de información, que incluye datos sobre las capacidades del browser, datos opcionales para el servidor,…
- El servidor devuelve la respuesta al cliente. Consiste en un código de estado y el tipo de dato MIME de la información de retorno, seguido de la propia información.
- Se cierra la conexión TCP.

Este proceso se repite en cada acceso al servidor HTTP. Por ejemplo, si se recoge un documento HTML en cuyo interior están insertadas cuatro imágenes, el proceso anterior se repite cinco veces, una para el documento HTML y cuatro para las imágenes.

En la actualidad se ha mejorado este procedimiento, permitiendo que una misma conexión se mantenga activa durante un cierto periodo de tiempo, de forma que sea utilizada en sucesivas transacciones. Este mecanismo, denominado HTTP Keep Alive, es empleado por la mayoría de los clientes y servidores modernos. Esta mejora es imprescindible en una Internet saturada, en la que el establecimiento de cada nueva conexión es un proceso lento y costoso.

ejemplo:

1. Desde un cliente se solicita la URL http://www.upm.es/default.html
2. Se abre una conexión TCP/IP con el puerto 80 del sistema www.upm.es.
3. El cliente realiza la solicitud, enviando algo similar a esto:

   GET /default.html HTTP/1.0 Operación solicitada+objeto+versión de HTTP
   Accept: text/plain Lista de tipos MIME que acepta o entiende
   Accept: text/html el cliente
   Accept: audio/*
   Accept: video/mpeg
   .. .. ..
   Accept: */* Indica que acepta otros posibles tipos MIME
   User-Agent: Mozilla/3.0 (WinNT; I) Información sobre el tipo de cliente
   Línea en blanco, indica el final de la petición
4. El servidor responde con la siguiente información:

   HTTP/1.0 200 OK Status de la operación; en este caso, correcto
   Date: Monday, 7-Oct-96 18:00:00 Fecha de la operación
   Server: NCSA 1.4 Tipo y versión del servidor
   MIME-version: 1.0 Versión de MIME que maneja
   Content-type: text/html Definición MIME del tipo de datos a devolver
   Content-length: 254 Longitud de los datos que siguen
   Last-modified: 6-Oct-96 12:30:00 Fecha de modificación de los datos
   Línea en blanco
   <HTML> Comienzo de los datos
   <HEAD><TITLE>Recursos de investigación en UNICAN</TITLE></HEAD>
   <BODY>
   .. .. ..
   .. .. ..
   </HTML>
5. Se cierra la conexión.

5.2.2 Estructura de los mensajes HTTP

El diálogo con los servidores HTTP se establece a través de mensajes formados por líneas de texto, cada una de las cuales contiene los diferentes comandos y opciones del protocolo. Sólo existen dos tipos de mensajes, uno para realizar peticiones y otro para devolver la correspondiente respuesta. La estructura general de los dos tipos de mensajes se puede ver en el siguiente esquema:

Mensaje de solicitud	**Mensaje de respuesta**
Comando HTTP + parámetros	Resultado de la solicitud
Cabeceras del requerimiento	Cabeceras de la respuesta
(línea en blanco)	(línea en blanco)
Información opcional	Información opcional

La primera línea del mensaje de solicitud contiene el comando que se solicita al servidor HTTP, mientras que en la respuesta contiene el resultado de la operación, un código numérico que permite conocer el éxito o fracaso de la operación. Después aparece, para ambos tipos de mensajes, un conjunto de cabeceras (unas obligatorias y otras opcionales), que condicionan el funcionamiento del protocolo.

La separación entre cada línea del mensaje se realiza con un par CR-LF (retorno de carro más nueva línea). El final de las cabeceras se indica con una línea en blanco, tras la cual se pueden incluir los datos transportados por el protocolo, por ejemplo, el documento HTML que devuelve un servidor o el contenido de un formulario que envía un cliente .

5.2.3 Comandos del protocolo

Los comandos o verbos de HTTP representan las diferentes operaciones que se pueden solicitar a un servidor HTTP. El formato general de un comando es:

Nombre del Comando	Objeto sobre el que se aplica	Versión http utilizada

Cada comando actúa sobre un objeto del servidor, normalmente un fichero o aplicación, que se toma de la URL de activación. La última parte de esta URL, que representa la dirección de un objeto dentro de un servidor HTTP, es el parámetro sobre el que se aplica el comando. Se compone de una serie de nombres de directorios y ficheros, además de parámetros opcionales para las aplicaciones CGI (ver Ejecución de programas en un servidor HTTP en la página <http://www.upm.es/libro/HTTP.htm>).

El estándar HTTP/1.0 recoge únicamente tres comandos, que representan las operaciones de recepción y envío de información y chequeo de estado:

- GET Se utiliza para recoger cualquier tipo de información del servidor. Se utiliza siempre que se pulsa sobre un enlace o se teclea directamente a una

URL. Como resultado, el servidor HTTP envía el documento correspondiente a la URL seleccionada, o bien activa un módulo CGI, que generará a su vez la información de retorno.

- HEAD Solicita información sobre un objeto (fichero): tamaño, tipo, fecha de modificación… Es utilizado por los gestores de caches de páginas o los servidores proxy, para conocer cuándo es necesario actualizar la copia que se mantiene de un fichero.

- POST Sirve para enviar información al servidor, por ejemplo los datos contenidos en un formulario. El servidor pasará esta información a un proceso encargado de su tratamiento (generalmente una aplicación CGI). La operación que se realiza con la información proporcionada depende de la URL utilizada. Se utiliza, sobre todo, en los formularios.

Un cliente Web selecciona automáticamente los comandos HTTP necesarios para recoger la información requerida por el usuario. Así, ante la activación de un enlace, siempre se ejecuta una operación GET para recoger el documento correspondiente. El envío del contenido de un formulario utiliza GET o POST, en función del atributo de <FORM METHOD="...">. Además, si el cliente Web tiene un caché de páginas recientemente visitadas, puede utilizar HEAD para comprobar la última fecha de modificación de un fichero, antes de traer una nueva copia del mismo.

Posteriormente se han definido algunos comandos adicionales, que sólo están disponibles en determinadas versiones de servidores HTTP, con motivos eminentemente experimentales. La última versión de HTTP, denominada 1.1, recoge estas y otras novedades, que se pueden utilizar, por ejemplo, para editar las páginas de un servidor Web trabajando en remoto.

- PUT Actualiza información sobre un objeto del servidor. Es similar a POST, pero en este caso, la información enviada al servidor debe ser almacenada en la URL que acompaña al comando. Así se puede actualizar el contenido de un documento.

- DELETE Elimina el documento especificado del servidor.

- LINK Crea una relación entre documentos.

- UNLINK Elimina una relación existente entre documentos del servidor.

5.2.4 Las cabeceras

Son un conjunto de variables que se incluyen en los mensajes HTTP, para modificar su comportamiento o incluir información de interés. En función de su

nombre, pueden aparecer en los requerimientos de un cliente, en las respuestas del servidor o en ambos tipos de mensajes. El formato general de una cabecera es:

Nombre de la variable	Cadena ASCII con su valor

Los nombres de variables se pueden escribir con cualquier combinación de mayúsculas y minúsculas. Además, se debe incluir un espacio en blanco entre el signo : y su valor. En caso de que el valor de una variable ocupe varias líneas, éstas deberán comenzar, al menos, con un espacio en blanco o un tabulador.

Cabeceras comunes para peticiones y respuestas

- Content-Type: descripción MIME de la información contenida en este mensaje. Es la referencia que utilizan las aplicaciones Web para dar el correcto tratamiento a los datos que reciben.

- Content-Length: longitud en bytes de los datos enviados, expresado en base decimal.

- Content-Encoding: formato de codificación de los datos enviados en este mensaje. Sirve, por ejemplo, para enviar datos comprimidos (x-gzip o x-compress) o encriptados.

- Date: fecha local de la operación. Las fechas deben incluir la zona horaria en que reside el sistema que genera la operación. Por ejemplo: Sunday, 12-Dec-96 12:21:22 GMT+01. No existe un formato único en las fechas; incluso es posible encontrar casos en los que no se dispone de la zona horaria correspondiente, con los problemas de sincronización que esto produce. Los formatos de fecha a emplear están recogidos en los RFC 1036 y 1123.

- Pragma: permite incluir información variada relacionada con el protocolo HTTP en el requerimiento o respuesta que se está realizando. Por ejemplo, un cliente envía un Pragma: no-cache para informar de que desea una copia nueva del recurso especificado.

5.2.4.1 Cabeceras sólo para peticiones del cliente

- Accept: campo opcional que contiene una lista de tipos MIME aceptados por el cliente. Se pueden utilizar * para indicar rangos de tipos de datos; tipo/* indica todos los subtipos de un determinado medio, mientras que */* representa a cualquier tipo de dato disponible.

- Authorization: clave de acceso que envía un cliente para acceder a un recurso de uso protegido o limitado. La información incluye el formato de

autorización empleado, seguido de la clave de acceso propiamente dicha. La explicación se incluye más adelante.

- From: campo opcional que contiene la dirección de correo electrónico del usuario del cliente Web que realiza el acceso.

- If-Modified-Since: permite realizar operaciones GET condicionales, en función de si la fecha de modificación del objeto requerido es anterior o posterior a la fecha proporcionada. Puede ser utilizada por los sistemas de almacenamiento temporal de páginas. Es equivalente a realizar un HEAD seguido de un GET normal.

- Referer: contiene la URL del documento desde donde se ha activado este enlace. De esta forma, un servidor puede informar al creador de ese documento de cambios o actualizaciones en los enlaces que contiene. No todos los clientes lo envían.

- User-agent: cadena que identifica el tipo y versión del cliente que realiza la petición. Por ejemplo, los browsers de Netscape envían cadenas del tipo User-Agent: Mozilla/3.0 (WinNT; I)

5.2.4.2 Cabeceras sólo para respuestas del servidor http

- Allow: informa de los comandos HTTP opcionales que se pueden aplicar sobre el objeto al que se refiere esta respuesta. Por ejemplo, Allow: GET, POST.

- Expires: fecha de expiración del objeto enviado. Los sistemas de cache deben descartar las posibles copias del objeto pasada esta fecha. Por ejemplo, Expires: Thu, 12 Jan 97 00:00:00 GMT+1. No todos los sistemas lo envían. Puede cambiarse utilizando un <META EXPIRES> en el encabezado de cada documento.

- Last-modified: fecha local de modificación del objeto devuelto. Se puede corresponder con la fecha de modificación de un fichero en disco, o, para información generada dinámicamente desde una base de datos, con la fecha de modificación del registro de datos correspondiente.

- Location: informa sobre la dirección exacta del recurso al que se ha accedido. Cuando el servidor proporciona un código de respuesta de la serie 3xx, este parámetro contiene la URL necesaria para accesos posteriores a este recurso.

- Server: cadena que identifica el tipo y versión del servidor HTTP. Por ejemplo, Server: NCSA 1.4.

- WWW-Autenticate: cuando se accede a un recurso protegido o de acceso restringido, el servidor devuelve un código de estado 401, y utiliza este campo para informar de los modelos de autentificación válidos para acceder a este recurso.

5.2.5 Códigos de estado del servidor

Ante cada transacción con un servidor HTTP, éste devuelve un código numérico que informa sobre el resultado de la operación, como primera línea del mensaje de respuesta. Estos códigos aparecen en algunos casos en la pantalla del cliente, cuando se produce un error. El formato de la línea de estado es:

Versión de protocolo HTTP utilizada	Código numérico de estado (tres dígitos)	Descripción del código numérico

Existen cinco categorías de mensajes de estado, organizadas por el primer dígito del código numérico de la respuesta:

1xx	mensajes informativos. Por ahora (en HTTP/1.0) no se utilizan, y están reservados para un futuro uso.
2xx	mensajes asociados con operaciones realizadas correctamente.
3xx	mensajes de redirección, que informan de operaciones complementarias que se deben realizar para finalizar la operación.
4xx	errores del cliente; el requerimiento contiene algún error, o no puede ser realizado.
5xx	errores del servidor, que no ha podido llevar a cabo una solicitud.

5.3 Caches de páginas y servidores proxy

Muchos clientes Web utilizan un sistema para reducir el número de accesos y transferencias de información a través de Internet, y así agilizar la presentación de documentos previamente visitados. Para ello, almacenan en el disco del cliente una copia de las últimas páginas a las que se ha accedido. Este mecanismo, denominado "caché de páginas", mantiene la fecha de acceso a un documento y comprueba, a través de un comando HEAD, la fecha actual de modificación del mismo. En caso de que se detecte un cambio o actualización, el cliente accederá, ahora a través de un GET, a recoger la nueva versión del fichero. En caso contrario, se procederá a utilizar la copia local.

Un sistema parecido, pero con más funciones es el denominado "servidor proxy". Este tipo de servidor, una mezcla entre servidor HTTP y cliente Web, realiza las funciones de cache de páginas para un gran número de clientes. Todos los clientes conectados a un proxy dejan que éste sea el encargado de recoger las URLs solicitadas. De esta forma, en caso de que varios clientes accedan a la misma página,

el servidor proxy la podrá proporcionar con un único acceso a la información original.

La principal ventaja de ambos sistemas es la drástica reducción de conexiones a Internet necesarias, en caso de que los clientes accedan a un conjunto similar de páginas, como suele ocurrir con mucha frecuencia. Además, determinadas organizaciones limitan, por motivos de seguridad, los accesos desde su organización al exterior y viceversa. Para ello, se dispone de sistemas denominados "cortafuegos" (firewalls), que son los únicos habilitados para conectarse con el exterior. En este caso, el uso de un servidor proxy se vuelve indispensable.

En determinadas situaciones, el almacenamiento de páginas en un caché o en un proxy puede hacer que se mantengan copias no actualizadas de la información, como por ejemplo en el caso de trabajar con documentos generados dinámicamente. Para estas situaciones, los servidores HTTP pueden informar a los clientes de la expiración del documento, o de la imposibilidad de ser almacenado en un caché, utilizando la variable Expires en la respuesta del servidor.

Capitulo VI

6.1 Conclusión

El conjunto de protocolos TCP/IP ha sido de vital importancia para el desarrollo de las redes de comunicación, sobre todo para Internet. El ritmo de expansión de Internet también es una consecuencia de estos protocolos, sin los cuales, conectar redes de distintas naturalezas (diferente *Hardware*, sistema operativo, etc..), hubiera sido mucho mas difícil, por no decir imposible.

Además, los protocolos interred contienen una gran cantidad de protocolos de nivel de aplicación, como TELNET, el protocolo de transferencia de archivos (FTP) y el protocolo simple de transferencia de correo (SMTP), son muchas las ventajas que Internet nos ofrece, podríamos llenar páginas enteras de bondades.

Todas estas ventajas nos han permitido a lo largo de la evolución de la tecnología hasta nuestros idas contar con un "A*cceso Global"* que nos permite ingresar a la red a través de una llamada telefónica o una línea alquilada directa a Internet y el acceso a la información no posee un costo de comunicación extra para la información este donde este.

Este proceso de creación de diferentes protocolos no hubiese sido posible sin la dirección y colaboración del IETF (Internet Engineering Task Force), que es una gran comunidad de carácter abierto formada por diseñadores de redes, operadores, usuarios, etc. Todos los protocolos agrupados normalmente bajo el nombre TCP/IP son estándares de Internet cuyo desarrollo depende del IETF. Las actividades que realiza el IETF se dividen en distintos grupos, llamados Working Groups (WG) con finalidades específicas, los cuales se clasifican en distintas áreas comunes (Aplicaciones, seguridad, estandarización, servicios de transporte, etc.). El IESG (Internet Engineering Steering Group) se encarga de coordinar y dirigir al IETF por medio de los directores de área, que controlan las actividades número de los Working Groups que se encuentren dentro de cada área.

Todos los protocolos y estándares que se consolidan como propios de Internet han de ser organizados y dirigidos de alguna manera. Esta es la misión principal del IETF (Internet Engineering Task Force), que es una gran comunidad de carácter abierto formada por diseñadores de redes, operadores, usuarios, etc. Todos los protocolos agrupados normalmente bajo el nombre TCP/IP son estándares de Internet cuyo desarrollo depende del IETF. Las actividades que realiza el IETF se dividen en distintos grupos, llamados Working Groups (WG) con finalidades específicas, los cuales se clasifican en distintas áreas comunes (Aplicaciones, seguridad, estandarización, servicios de transporte, etc.). El IESG (Internet Engineering Steering Group) se encarga de coordinar y dirigir al IETF por medio de los directores de área, que controlan las actividades número de los Working Groups que se encuentren dentro de cada área.

Todas las personas que deseen pueden colaborar directamente con estas instituciones mediante los RFC (Request for Comments), los mismos que contienen información de gran interés acerca de Internet. Existen miles de estos documentos con información sobre cualquier aspecto relacionado con la red.

De esta forma y a lo largo de la historia se ha ido recopilando información que ha permitido la creación de los protocolos que hoy en día millones de personas usan diariamente y tal vez sin darse cuenta de la complejidad que hay dentro de las redes de comunicaciones.

Así pues, podemos decir que los protocolos TCP/IP fueron y son el motor necesario para que las redes en general, e Internet en particular, se mejoren y se pueda lograr una buena "autopista de la información".

6.2 Glosario

Resolución de direcciones(address resolution)
: Un medio para mapear direcciones del nivel de red a direcciones específicas del medio.

ADMD
: Administration Management Domain. Un servicio de transporte público MHS("Message Handling System" o sistema de manejo de mensajes) X.400.

Agente(agent)
: En el modelo cliente servidor, la parte del sistema que prepara e intercambia la información para una aplicación cliente o servidor.

Modelo cliente/servidor(client/server model)
: Una forma habitual de describir servicios de red y el modelo de usuario(programas) de esos servicios. Ejemplos pueden ser el paradigma "name-server/name-resolver" del DNS o relaciones servidor de ficheros/cliente de ficheros como NFS y hosts sin disco.

CSNET
: Computer Science Network. Una gran red, localizada principalmente en los U.S, pero con conexiones internacionales. Los sitios CSNET incluyen universidades, laboratorios de investigación, y algunas compañías comerciales.

DDN
: Defense Data Network. Contiene MILNET y otras varias redes DoD.

DNS
: Domain Name System. El mecanismo distribuido de nombres/direcciones usado en Internet.

Dominio(domain)
: En Internet, una parte de una jerarquía de nombres. Sintácticamente, un nombre de dominio consiste en una secuencia de nombres(etiquetas) separados por puntos, por ejemplo, "tundra.mpk.ca.us." En OSI, "dominio" se usa generalmente como partición administrativa de un complejo sistema distribuido.

FTP
: File Transfer Protocol. El protocolo(y programa) de Internet para transferir ficheros entre hosts.

Gopher

Herramienta de navegación por Internet que proporciona un menú de acceso a información. Muchas organizaciones la usan en vez del FTP anónimo.

IANA

Internet Assigned Numbers Authority. El cuerpo técnico dentro del IAB que gestiona los estándares de protocolos en Internet. Coordina la asignación de valores a los parámetros de los protocolos.

IESG

Internet Engineering Steering Group. El comité ejecutivo del IETF.

IETF

Internet Engineering Task Force. Una de las fuerzas de trabajo del IAB. El IETF es responsable de resolver las necesidades de ingeniería de Internet a corto plazo. Tiene más de 40 grupos de trabajo.

Interred(internet)

Colección de redes interconectadas por un conjunto de "routers" que la permiten funcionar como una sola gran red virtual).

Dirección de Internet(internet address)

Una dirección de 32 bits asignada a los host que usan TCP/IP. Ver notación decimal con puntos("dotted decimal notation").

IP

Internet Protocol. El protocolo de red de la pila de protocolos de Internet.

IP datagram

La unidad fundamental de información transmitida a través de Internet. Contiene las direcciones fuente y destino junto con datos y una serie de campos que definen la longitud del datagrama, el checksum de la cabecera y flags para indicar cuando el datagrama ha sido(o puede ser) fragmentado.

RDSI(ISDN)

Integrated Services Digital Network. Una tecnología emergente que está comenzando a ser ofertado por las empresas telefónicas de todo el mundo. RDSI combina servicios de voz y digitales en un solo medio .

ISO

International Organization for Standardization. Mejor conocido como el modelo de referencia OSI de siete capas. Ver OSI.

ME(mail exploder)

Parte de un sistema de entrega de correo electrónico que permite que un mensaje sea entregado en una lista de direcciones. Los MEs se emplean para implementar listas de correo. Los usuarios envían mensajes a una sola dirección(por ejemplo hacks@somehost.edu) y el SE ME encarga de distribuirlos a cada uno de los buzones de la lista de correo.

Pasarela de correo(mail gateway)

Una máquina que conecta dos o más sistemas de correo electrónico(especialmente sistemas de correo distintos en redes diferentes) y transfiere mensajes entre ellos. A veces el mapeo y la traducción pueden ser bastante complejas, y generalmente requieren un esquema de almacenamiento-retransmisión por el que un sistema ha de recibir completamente un mensaje antes de poder realizar las traducciones pertinentes y enviarlo al siguiente sistema.

MIME

Multipurpose Internet Mail Extensions. Protocolo de correo que proporciona soporte para multimedia(gráficos, audio, video) además de una compatibilidad básica con SMTP. Se describe en los RFCs 1521 y 1522.

MTA

Message Transfer Agent. Un proceso de aplicación OSI empleado para almacenar y retransmitir mensajes en el sistema de manejo de mensajes X.400. Equivalente a un agente de correo de Internet.

Resolución de nombres(name resolution)

El proceso de mapear un nombre a su correspondiente dirección. Ver DNS.

NetBIOS

Network Basic Input Output System. La interfaz estándar para redes en el IBM PC y sistemas compatibles.

NIC

Network Information Center. Originalmente sólo había uno, localizado en el SRI International, que tenía la tarea de servir a la comunidad ARPANET (y más tarde a DDN). Hoy en día, hay muchos NICs, operados por redes locales, regionales y nacionales por todo el mundo. Tales centros proporcionan asistencia al usuario, servicios de documentación, formación, y mucho más.

OSI

Open Systems Interconnection. Un programa de estandarización internacional para facilitar las comunicaciones entre ordenadores de distintos fabricantes.

port

La abstracción que utilizan los protocolo de transporte de Internet para distinguir entre distintas conexiones simultáneas con el mismo host.

Protocolo

Una descripción formal de los mensajes a intercambiar y de las reglas que dos o más sistemas han de seguir para intercambiar información.

RFC

Request For Comments. La serie de documentos, comenzada en 1969, que describe la pila de protocolos de Internet y los experimentos relacionados. No todos los RFCs describen estándares de Internet(de hecho, sólo lo hacen unos cuántos), pero todos los estándar de Internet están descritos en forma de RFCs.

Router

Un sistema responsable de tomar decisiones acerca de la ruta que seguirá el tráfico de una red. Para hacerlo, utiliza un protocolo de encaminamiento con el fin de obtener información sobre la red, y algoritmos para elegir el mejor camino, basados en diversos criterios conocidos como "métricas de encaminamiento". En la terminología OSI, un "router" es la capa de red de un sistema intermedio.

SMTP

Simple Mail Transfer Protocol. El protocolo de correo electrónico de Internet. Definido en el RFC 821, con descripciones del formato de mensajes asociados en el RFC 822.

TCP

Transmission Control Protocol. El principal protocolo de transporte de la pila de protocolos, que proporciona flujos fiables, orientados a conexión y en full-duplex. Empleado para la entrega de paquetes IP.

Telnet

El protocolo de terminal virtual en la pila de protocolos de Internet. Permite que los usuarios de un host entren en sesión en un host remoto e interactúen como usuarios normales del mismo.

UDP

User Datagram Protocol. Un protocolo de transporte en la pila de protocolos de Internet. A al igual que TCP, se usa para el transporte de paquetes IP, pero el intercambio de datagramas que proporciona no ofrece reconocimiento o garantía de entrega.

UUCP

UNIX to UNIX Copy Program. Un protocolo usado para la comunicación entre sistemas UNIX.

6.3 Bibliografía

Tanenbaum, Andrew S.: Computer Networks, Fourth Edition, Pearson 2003.

Request For Comments: http://www.cis.ohio-state.edu/hypertext/information/rfc.htm

Request For Comments: http:// www.ietf.com

Indice