Escuela Politécnica Superior
Universidad de Huelva

TERCER CURSO. TECNOLOGÍA DE REDES

Tema 4:
World Wide Web

Manuel Sánchez Raya
Versión 0.1
5 de Febrero de 2004

ÍNDICE

BIBLIOGRAFÍA

Apuntes año 2002-2003. Estefanía Cortés Ancos.
Apuntes Universidad de Oviedo. J.A.Sirgo, Rafael C. González
Academia de Networking de Cisco Systems. Guía del Primer Año.
Internetworking with TCP/IP. Vol. I. D.E. Comer.

1. Introducción al World Wide Web.

La World Wide Web (conocida como WWW, Web o W3) es el universo de información accesible a través de una red. Se trata de un servicio de distribución de la información proporcionado al usuario final por la capa de aplicación. Cuando nos referimos a la WWW hablamos de la infraestructura que permite la publicación, consulta de documentos y servicios a escala mundial, con hiperenlaces entre ellos. La World Wide Web está revolucionando muchos elementos de la sociedad, como el comercio, la política, la literatura, los servicios sociales.

1.1.-Historia de la WWW.

En 1945 Vannevar Bush escribe un artículo acerca de un dispositivo que podrá crear y seguir enlaces en documentos en microfichas. Posteriormente, en 1960s Doug Engelbart hace un prototipo de "oNLine System" (NLS) que permite editar hipertextos, correo y otras cosas. Inventa el ratón para este sistema. Ted Nelson acuña el término hipertexto.

En 1980 Tim Berners-Lee escribe un programa con enlaces entre notas mientras es consultor para el CERN (Centro Europeo de Investigación Nuclear). Posteriormente, en 1989 el mismo Berners-Lee escribe una propuesta de sistema de información basado en hipertexto, y la hace circular en el CERN. En 1990 construye un prototipo de browser y editor, lo llama World Wide Web.

Posterioremente en los años 1991-2 se producen desarrollos y popularización del sistema. Los sistemas disponibles en aquella epoca se reducian a 50 servidores.

En 1993 el CERN permite el libre uso de la tecnología WWW. Habra entorno a 200 servidores. Posteriormente, en 1994 se crea el WWW Consortium (http://www.w3.org). Organización encargada del desarrollo de la WWW, estandarización de protocolos y el fomento de interoperabilidad entre las instalaciones.

1.2.- Arquitectura de WWW.

El WWW está diseñado siguiendo una arquitectura distribuida cliente servidor. Un *cliente de WWW* es un programa que envía peticiones de documentos a cualquier servidor de WWW a través de una conexión TCP. Un *servidor de WWW* es un programa que, al recibir una petición, envía el documento requerido al cliente. El cliente se puede estar ejecutando en una maquina diferente a la del servidor, en otra habitación, país o continente. El cliente se encarga de la presentación de los documentos al usuario final. El servidor se encarga del almacenamiento de los documentos.

1.2.1.- El cliente de WWW.

Desde el punto de vista del cliente la web consiste en un enorme conjunto a nivel mundial de documentos (páginas). Cada página puede contener vínculos con otras páginas: *hipertexto*. Las páginas pueden verse a través de un programa llamado visor (*browser*). Los browser más importantes son Netscape o Mozilla y Microsoft Internet Explorer.

En el web los clientes demandan hipertextos a los servidores. El visor obtiene la página solicitada, interpreta el texto y los comandos de formateo que contiene, y exhibe la página adecuadamente formateada.

Para desarrollar un sistema de este tipo ha sido necesario:

a) Un nuevo protocolo que permite saltos hipertextuales, es decir, de un nodo origen a otro destino, que puede ser texto, imágenes, sonido, animaciones, vídeo, etc. Este protocolo se denomina HTTP (HiperText Transfer Protocol) y es el lenguaje que hablan los servidores.

b) Un nuevo lenguaje para representar hipertextos que incluyera información sobre la estructura y formato de representación y, especialmente, indicara el origen y destinos de los saltos de hipertexto. Este lenguaje es el HTML (HyperText Markup Language).

c) Una forma de codificar las instrucciones para los saltos hipertextuales de un objeto a otro de la Internet.

d) Aplicaciones cliente para todo tipo de plataformas y resolver el problema de y cómo se accede a la información que está almacenada, y que ésta sea disponible a través de los diversos protocolos (FTP, HTTP, WAIS...) que representen a su vez información multiformato (texto, imágenes, animaciones, etc.). Con este fin aparecen varios clientes, entre los que destacan MOSAIC del NCSA (Universidad de Chicago) y NETSCAPE Navigator de Netscape Communications Corporation.

El servicio ofrecido por la www se basa en un sistema de *hipertexto*. En un sistema Hipertexto el texto contiene enlaces a otros documentos. Seleccionando estos enlaces con el ratón o teclado se puede acceder a otro documento que, a su vez, puede ser un hipertexto.

Las páginas además de contener texto normal e hipertexto suelen contener iconos, mapas, fotografías, … cada uno de ellos puede vincularse a otra página. Algunas páginas contienen pistas de audio, fragmentos de vídeo o ambas cosas. El resultado de mezclar páginas de hipertexto con otros medios se conoce como *hipermedia*. En un sistema hipermedia los documentos hipermedia contienen enlaces a otros medios: sonido, imágenes, video, programas (Java) o otros documentos hipermedia.

Las propias imágenes pueden tener asociados enlaces a otros elementos. Algunos visores utilizan el disco local para poner en caché las páginas que han traído. Antes de traer una página, se comprueba si ya está en la caché local. Si es así, sólo es necesario verificar si la página está actualizada, así la página no será descargada nuevamente.

1.2.2.- El servidor de WWW.

Los servidores de WWW utilizan el protocolo HTTP para comunicarse con los clientes. HTTP es un protocolo basado en TCP/IP que se ejecuta sobre el puerto 80. Permite servir ficheros reales o virtuales (generados por scripts o programas). También soporta formularios y zonas activas sobre imágenes.

Los servidores pueden utilizarse también como un *servidor proxy* que permite a clientes un acceso común y no completo a la Web a través de él (para proporcionar seguridad o filtrado).

Si el Proxy tiene caché permite a todo cliente WWW que lo utilice como proxy, guardar temporalmente en un almacén local las últimas peticiones realizadas. También actúa como pasarelas entre el visualizador y servidores viejos que utilizan otro sistema de transferencia como Gopher o FTP.

1.3.- Esquema de funcionamiento.

El usuario hace clic en alguna parte del texto que apunta a la página cuyo nombre (URL localizador uniforme de recursos) es:

http:// www.w3.org/hypertext/WWW/TheProject.html.

- El visualizador determina el URL.
- El visualizador solicita al DNS la dirección IP de www.w3.org
- EL DNS contesta con 18.23.0.23
- El visualizador establece una conexión con el puerto 80 en 18.23.0.23
- El visualizador emite un comando GET */hypertext/WWW/TheProject.html*
- El servidor www.w3.org envía el archivo *TheProject.html*
- Se libera la conexión TCP
- El visualizador presenta todo el texto de *TheProject.html*
- El visualizador trae y presenta todas las imágenes de *TheProject.html*

2.- Protocolo HTTP.

El servicio de WWW proporciona un interfaz común para acceder a diferentes tipos de servicios/documentos a través de un sistema de nombres: *Universal Resource Locator* (URL). Describe una forma de incluir enlaces a URL's en documentos textuales: *HyperText Markup Language* (HTML). Para acceder a WWW se utiliza un programa visor que se encarga de obtener documentos hipermedia desde su lugar de origen, utilizando el protocolo *HyperText Transfer Protocol* (HTTP).

El HTTP es el protocolo de alto nivel del World Wide Web que rige el intercambio de mensajes entre clientes y servidores del Web. Se trata de un protocolo genérico orientado a objetos que no mantiene la conexión entre transacciones. Se diseñó especialmente para entender las exigencias de un sistema hipermedia distribuido como es el World Wide Web.

Sus principales características son:

- **Ligereza**: reduce la comunicación entre clientes y servidores a intercambios discretos, de modo que no sobrecarga la red y permite saltos hipertextuales rápidos.
- **Generalidad**: puede utilizarse para transferir cualquier tipo de datos (según el estándar MIME sobre el tráfico multimedia que incluye también los que se desarrollen en el futuro.)
- **Extensibilidad**: contempla distintos tipos de transacciones entre clientes y servidores y el futuro desarrollo de otros nuevos.

El esquema básico de cualquier transacción HTTP entre un cliente y un servidor es el siguiente:

- Conexión: El cliente establece una conexión con el servidor a través del puerto 80 (puerto estándar), u otro especificado.
- Petición: El cliente envía una petición al servidor.
- Respuesta: El servidor envía al cliente la respuesta (es decir, el objeto demandado o un código de error).
- Cierre: Ambas partes cierran la conexión.

La eficiencia del HTTP posibilita la transmisión de objetos multimedia y la realización de saltos hipertextuales con gran rapidez. Se basa en solicitudes ASCII seguidas de respuestas tipo MIME, bajo TCP. Normalmente, puerto 80. Los tipos MIME permiten el envío de objetos arbitrarios de una manera estándar. *Multipurpose Internet Mail Extensions* es un estándar abierto para enviar datos multimedia a través de correo-e, utilizado también para intercambiar documentos entre cliente y servidor en WWW.

El estándar MIME clasifica los contenidos según tipo/subtipo:

text/html
text/plain
image/gif
image/jpeg
video/mpeg
audio/basic
application/java
application/x-tex

El protocolo HTTP no mantiene el estado (no hay información sobre las conexiones entre una petición y otra). El protocolo reconoce dos tipos de solicitudes: sencillas y completas.

- Una solicitud sencilla es sólo una línea GET que nombra la página deseada, sin la versión del protocolo. La respuesta es la página en bruto, sin cabeceras, sin MIME y sin codificación.

- Una solicitud completa contiene GET, el nombre de la página deseada, la versión del protocolo HTTP y varias líneas con cabeceras RFC 822 seguidas de una línea en blanco.

No olvidar que HTTP puede servir tanto contenido estático (ficheros) como dinámico (el resultado de ejecutar programas en el servidor).

2.1.- Formato de los mensajes

Los mensajes están compuestos por líneas de texto:

- Línea inicial (diferente para petición y respuesta), terminada en CRLF.
- Cero o más líneas de cabecera, cada una terminada en CRLF:
- Línea en blanco (CRLF).
- Cuerpo del mensaje (opcional).

Además de CRLF, deberán tratarse adecuadamente líneas terminadas en LF.

2.2- Línea inicial (peticiones).

Especifica el recurso que se solicita, y que se quiere de él:

- Nombre de método:
  - GET : solicita leer una página web.
  - HEAD : solicita leer la cabecera de una página web.
  - PUT: solicita almacenar una página web, reemplaza la existente.
  - POST : anexa a la página existente.
  - DELETE: Elimina la página web.
  - LINK: Conecta dos recursos existentes.
  - UNLINK: Rompe una conexión existente entre dos recursos.
- Camino de acceso (path).
- Versión de HTTP (siempre HTTP/x.x).

Ejemplo: GET /directorio/otro/fichero.html HTTP/1.0

Si a la solicitud GET le sigue una cabecera *If-Modified-Since* el servidor sólo enviará los datos si han sido modificados después de la fecha proporcionada.

2.3.- Línea inicial (respuestas)

Cada solicitud recibe una respuesta que proporciona información de estado:

- Versión de HTTP (siempre HTTP/x.x).
- Código numérico de estado.
- Código de estado “en inglés".

Los códigos de estado pueden ser algunos de los siguientes:

1xx: Mensaje informativo.
2xx: Resultado exitoso (200 OK).
3xx: Redirección del cliente a otra URL (301 Moved permanently, 303 See Other).
4xx: Error en el lado del cliente (404 Not Found).
5xx: Error en el lado del servidor (500 Server Error).

Las líneas de cabecera tienen el mismo formato que las cabeceras de correo y News (RFC 822, sección 3). El protocolo HTTP/1.0 define 16 cabeceras, ninguna obligatoria. El protocolo HTTP/1.1 define 46 cabeceras.

El cuerpo del mensaje en las peticiones contiene datos de usuario o ficheros para subir. En las respuestas contiene el recurso pedido o texto explicando un error. Si hay cuerpo, normalmente hay algunas cabeceras relativas a él:

"Content-Type": tipo MIME de los datos (ej: text/html, image/png).
"Content-Length": número de bytes en el cuerpo.

2.4.- Ejemplos.

A continuación se muestra un ejemplo GET de petición:

```
GET /~jgb/test.html HTTP/1.0
Connection: Keep-Alive
User-Agent: Mozilla/4.07 [en] (X11; I; Linux 2.2.15 i586; Nav) ...
Host: gsyc.escet.urjc.es
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, i ...
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
```

A continuación se muestra un ejemplo GET de respuesta:

```
HTTP/1.1 200 OK
Date: Tue, 23 Jan 2001 12:44:27 GMT
Server: Apache/1.3.9 (Unix) Debian/GNU
Last-Modified: Tue, 23 Jan 2001 12:39:45 GMT
ETag: "19e89f-22-3a6d7b91"
Accept-Ranges: bytes
Content-Length: 34
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

<html>\nEsto es una prueba\n</html>
```

A continuación se muestra un ejemplo POST de petición:

```
POST /comments.pl HTTP/1.0
From: jgb@gsyc.escet.urjc.es
User-Agent: MegaNavigator/0.0
Content-Type: application/x-www-form-urlencoded
Content-Length: 18

section=all&rank=10
```

2.5.- Protocolo HTTPS.

El protocolo HTTPS es básicamente HTTP sobre SSL (*secure socket layer*). La conexión TCP está cifrada, de forma que una tercera parte no puede conocer su contenido. Permite enviar datos "sensibles" a un servidor web, y recibirlos de él.
Necesita de otros mecanismos (certificados, etc.) para ofrecer un nivel de seguridad razonable. Las URLs son "https://".

2.6.- Galletitas (cookies).

Un cookie es un fichero que se envía a un navegador por medio de un servidor web para registrar las actividades de un usuario en un sitio web. Por ejemplo, cuando se compran artículos se colocan en los que se llama un carrito de compra virtual, esa información es almacenada en el cookie.

Cuando el navegador pide ficheros adicionales, la información del cookie es devuelta al servidor. Los cookies pueden recordar otros tipos de información personal, como la contraseña de un usuario, de manera que no haya que volver a escribirla cada vez que se visita el mismo sitio, y sus preferencias, así que la próxima vez que se visite un sitio, se es presentado con informaciones del usuario.

Se trata de una manera de hacer que los sitios web estén más personalizados. La mayoría de los cookies tienen una fecha de caducidad y, o bien residen en la memoria del ordenador del usuario hasta que se cierra el navegador, o guardados en el disco duro. Los cookies no pueden leer informaciones almacenadas en el ordenador. Se puede hacer uso de un editor de texto para ver los ficheros de los cookies. Para los usuarios de Netscape en Windows el fichero se llama cookies.txt y se encuentra en la misma carpeta que Netscape. Internet Explorer crea ficheros separados para cada cookie y los almacena en carpetas llamadas Cookies o ficheros Temporary Internet.

En resumen:

- Una cookie es un pequeño trozo de datos que entrega el programa servidor de HTTP al navegador WWW (cliente WWW) para que éste lo guarde.
- Normalmente, se trata de información sobre la conexión o los datos requeridos, de esta manera puede saber qué hizo el usuario en la última visita.

- Sirven para asociar un estado a un conjunto de transacciones (peticiones/ respuesta).
- Generalmente, son datos asociados a un usuario (carro de la compra, cuenta de usuario, etc.)
- Especificación original de Netscape, luego propuesta como RFC 2109.

Cabecera "Set-Cookie"

Cabecera puesta por un servidor cuando quiere enviar una galletita. Formato:

- "Set-Cookie:"
- Nombre de la galletita y valor ("nombre=valor").
- Fecha de caducidad ("expires=fecha").
- Dominio, camino ("domain=dominio, path=camino"). Para decidir más tarde si se envía una galletita o no.
- "secure": si esta marcada así, solo se transmitirá sobre canales seguros (HTTPS).

Ejemplo:

```
Set-Cookie: unnombre=unvalor; expires=Mon, 30-Jan-2001 12:35:23 GMT;
path=/dir; domain=mi.dominio.com; secure
```

Cabecera "Cookie"

Cuando un cliente pide una URL, buscará en su lista de galletitas si hay alguna que tenga que enviar (mirando su "domain" por la cola, y su "path" por su cabeza). Enviará todas las galletitas en una única cabecera ("Cookie").

Dentro de esta cabecera, las galletitas se ordenarán de más a menos específicas (según su "path"). No se consideran las galletitas con caducidad en el pasado (de hecho, se eliminan). Ejemplo:

```
Cookie: unnombre=unvalor; otronombre=otrovalor
```

3.- URL – Uniform Resource Locator

A cada página se le asigna un URL (Localizador Uniforme de Recursos) que sirve como nombre mundial de la página. Los URL (*Uniform Resource Locator*) son "localizadores" de direcciones dentro de la red, que relacionan un servicio con un servidor. Constituyen la herramienta esencial del Web, ya que permiten la localización y conexión con cualquier servidor y recurso de la Internet

Los URLs tienen tres partes:

- La primera parte del URL especifica el método de acceso (protocolo).

http://	Acceso mediante World Wide Web
ftp://	Acceso mediante FTP
news://	Acceso mediante News.
mailto://	Acceso mediante E-Mail
gopher://	Acceso mediante Gopher
telnet://	Acceso mediante Telnet

- A continuación viene la dirección de la computadora en la que reside, el servicio (opcionalmente puede llevar un puerto):
  http://www.urjc.es:80/ficheros/fichero1.html
  Las computadoras que ofrecen un servicio de WWW suelen nombrarse con www al principio de su dirección: www.urjc.es

- El resto de la URL especifica el camino y el nombre del fichero

Opcionalmente la URL puede llevar un nombre de sección, separado por **#**:
http://www.urjc.es/ficheros/fichero1.html#secc2
URI: Uniform Resource Identifier es un medio para localizar un recurso en Internet de manera no ambigua (RFC2396). Puede verse como un URL generalizado.

3.1.- Ejemplos de URL.

A continuación se ofrecen algunos ejemplos de URL comunes:

file://www.urjc.es/pub/sonido.au	Trae y emite el sonido
file://www.urjc.es/imagen.gif	Trae y muestra la imagen
file://www.urjc.es/pub/	Contenido del directorio
http://www.urjc.es/~pheras/index.html	Se conecta a un servidor HTTP y trae un fichero HTML
ftp://www.xerox.com/pub/file.txt.	Abre una sesión FTP con www.xerox.com y trae un fichero de texto
gopher://iluso.ci.uv.es.	Gopher de iluso.ci.uv.es
telnet://porky.urjc.es.	Abre una sesión telnet
news:gsyc.test.	Lee las news
mailto:pheras@ordago.	Envía correo electrónico mundo comercial

4.- HTML - Hypertext Markup Language

El HTML es el Lenguaje de Marcaje de Hipertexto. Se trata de un lenguaje para describir la manera en la que debe formatearse un documento. Para ello, contiene una serie de comandos de modo que el visualizador sólo tiene que entender dichos comandos de marcación para presentar el documento. Utilizado para crear y reconocer documentos hipermedia.

Permite separar la presentación del contenido al integrar los comandos de marcación dentro de cada archivo HTML y estandarizarlos, se hace posible que cualquier visor pueda leer y reformatear cualquier página web. El lenguaje HTML es el método que se utiliza para diseñar el aspecto visual de las páginas de Web. Su principal baza es su enorme sencillez: una página se puede diseñar de manera rápida e interactiva, viendo en tiempo real como va quedando el resultado.

Las páginas escritas en HTML pueden contener texto, imágenes, animaciones, música y enlaces de hipertexto. Los recursos multimedia (imágenes, animaciones y música) se describen en ficheros separados, mientras que el resto está escrito en un fichero cuyo nombre acabará en .HTML, que será el núcleo de la página.

El lenguaje se basa en las denominadas marcas, que sirven para indicar que en ese punto del documento sucederá un evento dado. Por ejemplo, para insertar una imagen entre dos párrafos basta con escribir el texto de ambos párrafos en el fichero .HTML y, en medio, poner una marca y el nombre del fichero que contiene la imagen. De la misma manera se pueden colocar enlaces, música...

Cuando nuestro software de gestión de Internet (Netscape, por ejemplo) recibe la orden de cargar una cierta página, lo que hará es:

- Cargar del servidor la página, y buscar en ella qué ficheros tiene vinculados.
- Cargar los ficheros vinculados del servidor. Si falta alguno, mostrará un error o indicará que ese fichero está corrupto
- Verificar que los enlaces hipertexto están definidos (es decir, que existen los documentos a los que se apunta)
- Mostrar la página en pantalla.

Un documento HTML es un conjunto de caracteres ASCII de 7 bits, con códigos para:

- Estilos del texto
- Títulos de documentos, secciones
- Párrafos
- Listas
- HiperEnlaces
- Formularios

Una página web escrita en lenguaje HTML consiste en una cabecera y un cuerpo encerrado entre etiquetas (comandos de formateo) <HTML> y </HTML>

4.1.- Futuro de WWW.

Con URL, cuando un documento cambia de lugar, cambia de nombre, no valen los enlaces. A través de los URC (*Universal Resource Citation*) se pueden hacer búsquedas por “campos", como título, autor, etc.

También se prevé una mejora de rendimiento con replicación y caches jerárquicas. Encriptación y autenticación de clientes. Esto permitirá nuevos servicios, como servicios de micropagos, de gran interés para el comercio electrónico.