HTML Y HTTP

HYPERTEXT MARKUP LANGUAGE (HTML): En simples palabras HTML es en lenguaje usado para describir la estructura de las páginas web. El lenguaje fue originalmente desarrollado por Tim Berners-Lee en los inicios de 1990, con el fin de mejorar el acceso a la información y desde entonces se ha ido desarrollando y extendiendo mucho más allá de su forma inicial. Como ya fue mencionado antes, HTML define el diseño y formato de una página web, y da la facilidad a los creadores de estos sitios el poder utilizar referencias de hipervínculos hacía otros recursos de la Web. Aprender la sintaxis de este lenguaje es relativamente sencillo y está expresada en un formato ASCII simple.

HYPERTEXT TRANSFER PROTOCOL (HTTP): HTTP son las siglas de "Hypertext Transfer Protocol". Es un protocolo de transferencia de hipertexto que sirve para sistemas de información distribuidos, colaborativos e hipermedia. HTTP comenzó su uso en 1990 con la información global World-Wide Web (WWW). La versión de lanzamiento fue la 0.9. Esta era un protocolo simple de transferencia de datos sin procesar a través de Internet. Luego en la versión 1.0 mejoró el protocolo al dejar que los mensajes estén en formatos MIME (mensajes similares, que contienen metainformación sobre los datos transferidos y modificadores en la semántica de solicitud / respuesta). Por otro lado, no considera la jerarquía de los proxy, el almacenamiento en caché o los hosts virtuales. Es por estos problemas que necesitó crear la versión 1.1 siendo más estricta que su versión anterior para una mayor confiabilidad en sus características. HTTP permite una variedad de métodos y encabezados para los distintos propósitos de solicitudes, se rige por la referencia proporcionada por el identificador de recursos uniforme (URI), como la ubicación (URL) o nombre (URN), para saber a cuál recurso se aplicará el método. HTTP también se utiliza como un protocolo común para la comunicación entre agentes de usuario y proxies / gateways a otros sistemas de Internet. De esta forma, HTTP autoriza el acceso básico desde hipermedia a los recursos disponibles desde diversas aplicaciones.

Relación: La relación entre HTML y HTTP es el trabajo en conjunto que realizan estas dos herramientas**.** HTML es el lenguaje que nos permite definir la estructura de una página web y posibilita los enlaces hacia otras páginas en la red. Las páginas tienen funcionalidades como el acceso a base de datos y otras muchas clases de aplicaciones e información estructurada. Es aquí en donde HTTP entra en juego, debido a que se encarga de poder hacer que los usuarios accedan a esta información en múltiples plataformas a través de protocolos. Los protocolos viajan a través de los cables que conforman la red hacia los servidores que procesan los pedidos de información y devuelven los resultados.

Análisis de páginas

Página web: www.comercialeuropa.cl

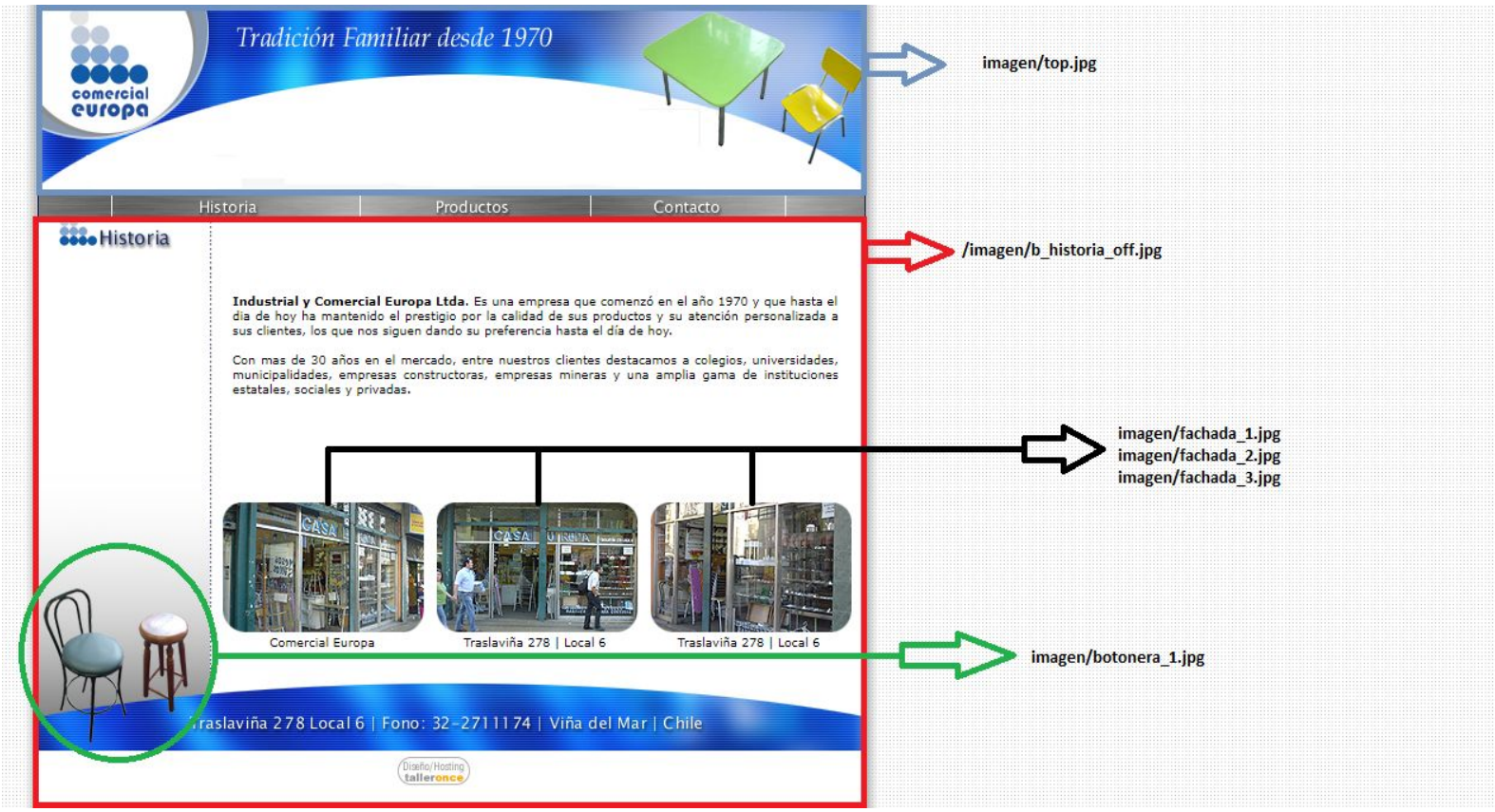


Figura 1

Página web: www.diazydiaz.cl

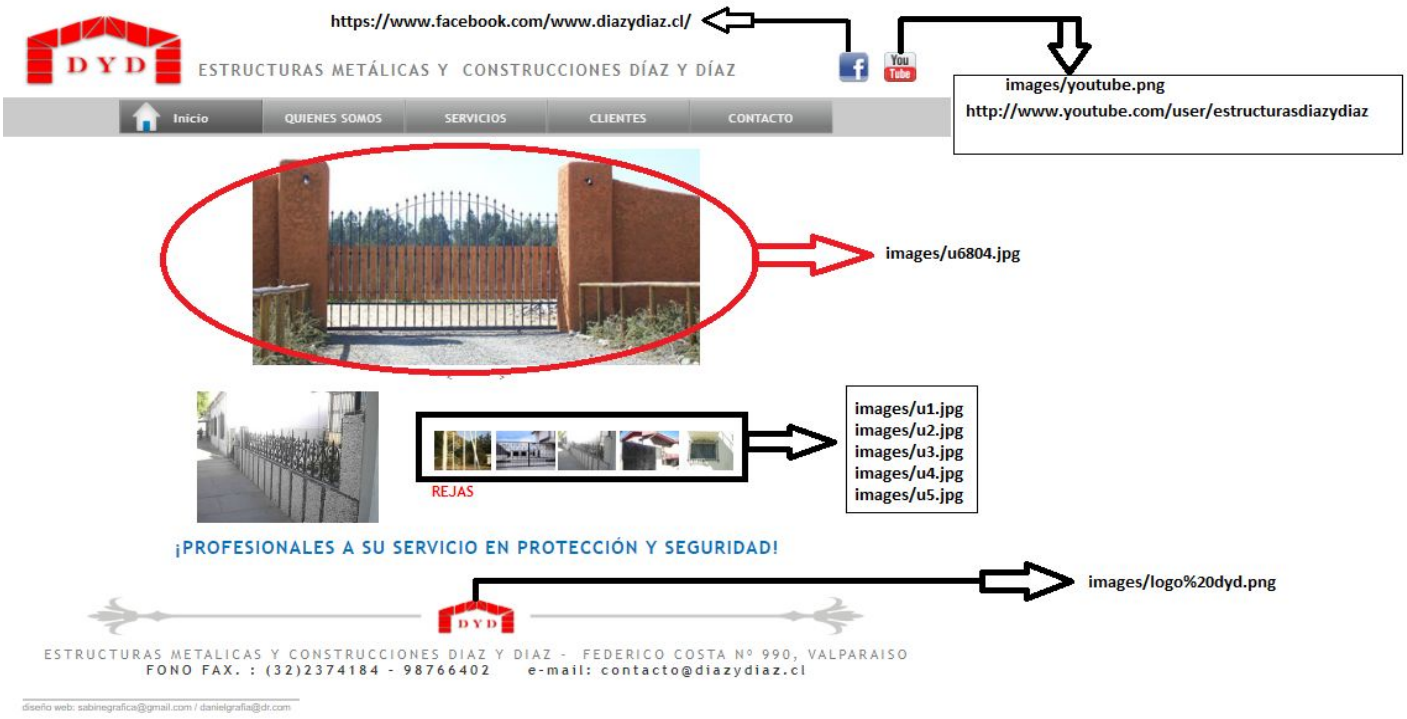

https://www.facebook.com/www.diazydiaz.cl/
ESTRUCTURAS METÁLICAS Y CONSTRUCCIONES DÍAZ Y DÍAZ
Inicio
QUIENES SOMOS
SERVICIOS
CLIENTES
CONTACTO
images/youtube.png
http://www.youtube.com/user/estructurasdiazydiaz
images/u6804.jpg
REJAS
images/u1.jpg
images/u2.jpg
images/u3.jpg
images/u4.jpg
images/u5.jpg
¡PROFESIONALES A SU SERVICIO EN PROTECCIÓN Y SEGURIDAD!
images/logo%20dyd.png
ESTRUCTURAS METALICAS Y CONSTRUCCIONES DIAZ Y DIAZ - FEDERICO COSTA Nº 990, VALPARAISO
FONO FAX. : (32)2374184 - 98766402 e-mail: contacto@diazydiaz.cl

Figura 2

HandShake

El “handshake” es utilizado para establecer una comunicación entre cliente/servidor. Por lo general, es un proceso que tiene lugar cuando un equipo está a punto de comunicarse con un dispositivo exterior para establecer las normas para la comunicación. El handshake tiene 3 pasos iniciales a través del protocolo TCP, los cuales son: **SYN**, **SYN/ACK** y **ACK.** Luego de hacer el intercambio de estos paquetes se ha establecido una comunicación óptima entre cliente y servidor.

HandShake Páginas Analizadas

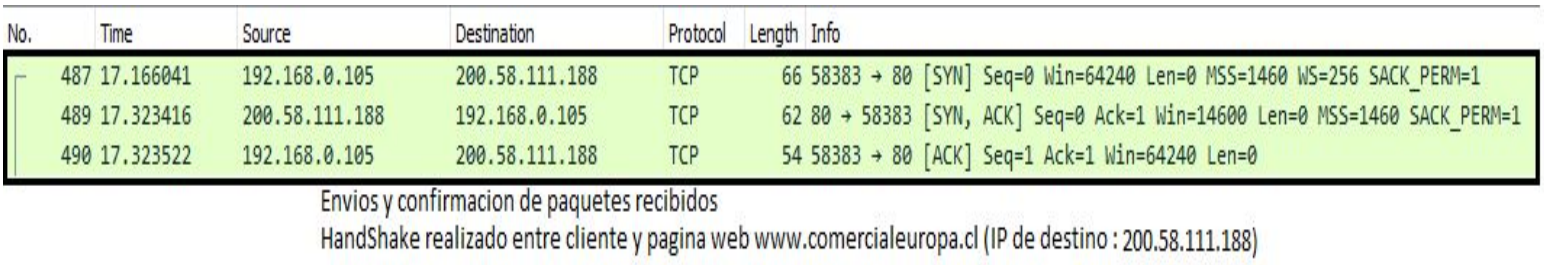


Figura 3

Luego de tener una comunicación exitosa entre ambas partes, se sigue con el mismo modelo de petición/respuesta. Es así como desde el cliente se solicita al servidor la obtención de objetos alojados en el mismo, como por ejemplo, la obtención del código fuente de la página, imágenes, videos, vínculos a otros sitios del servidor, etc. He aquí algunos ejemplos:

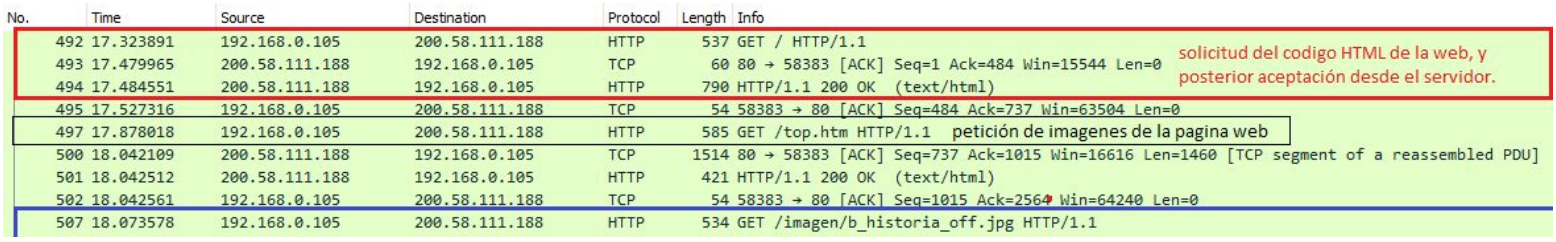
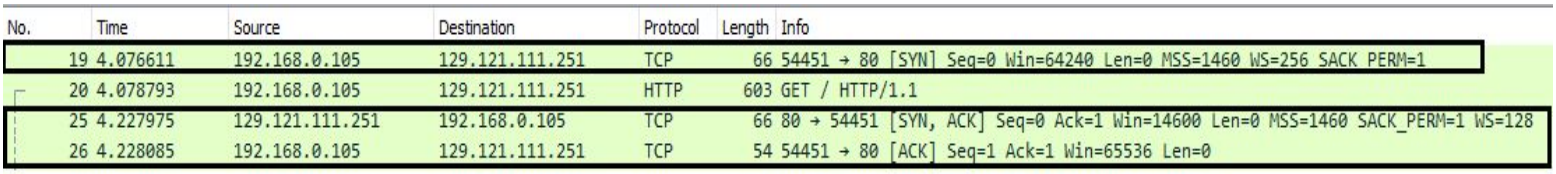


Figura 4

Para la siguiente página web el HandShake ocurre de la misma manera. Realizado el típico envío de los tres paquetes fundamentales, se podrá continuar con una comunicación en la que el cliente será el que mande peticiones/solicitudes al servidor.

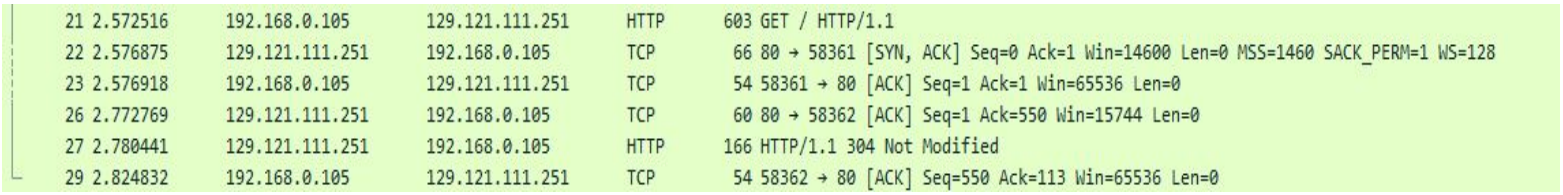


No.	Time	Source	Destination	Protocol	Length	Info
19	4.076611	192.168.0.105	129.121.111.251	TCP	66	54451 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
20	4.078793	192.168.0.105	129.121.111.251	HTTP	603	GET / HTTP/1.1
25	4.227975	129.121.111.251	192.168.0.105	TCP	66	80 → 54451 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1 WS=128
26	4.228085	192.168.0.105	129.121.111.251	TCP	54	54451 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0

HandShake realizado entre cliente y página web www.diazydiaz.cl (IP de destino 129.121.111.251)

Figura 5

De esta página también se obtienen paquetes estilo Request–response.



21	2.572516	192.168.0.105	129.121.111.251	HTTP	603	GET / HTTP/1.1
22	2.576875	129.121.111.251	192.168.0.105	TCP	66	80 → 58361 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1 WS=128
23	2.576918	192.168.0.105	129.121.111.251	TCP	54	58361 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
26	2.772769	129.121.111.251	192.168.0.105	TCP	60	80 → 58362 [ACK] Seq=1 Ack=550 Win=15744 Len=0
27	2.780441	129.121.111.251	192.168.0.105	HTTP	166	HTTP/1.1 304 Not Modified
29	2.824832	192.168.0.105	129.121.111.251	TCP	54	58362 → 80 [ACK] Seq=550 Ack=113 Win=65536 Len=0

Figura 6