

# Sistemas Telemáticos

## Práctica 5: HTTP

Departamento de Teoría de la Señal y Comunicaciones  
y Sistemas Telemáticos y Computación  
(GSyC)

Abril de 2018

Cuando un mensaje HTTP ocupa más de un segmento TCP, Wireshark muestra el siguiente mensaje por cada uno de los segmentos TCP que son parte de dicho mensaje HTTP:

[TCP segment of a reassembled PDU]

Cuando Wireshark interpreta que se ha recibido todo el mensaje HTTP, como resultado de haber recibido previamente un conjunto de segmentos TCP `segment of a reassembled PDU`, Wireshark concatena todos estos segmentos para mostrar el mensaje HTTP completo.

Por ejemplo, en la figura 1 se puede ver como en el segmento 8 se muestra todo el mensaje HTTP que en realidad viajaba en 3 segmentos TCP: segmento 4, 8 y 9.

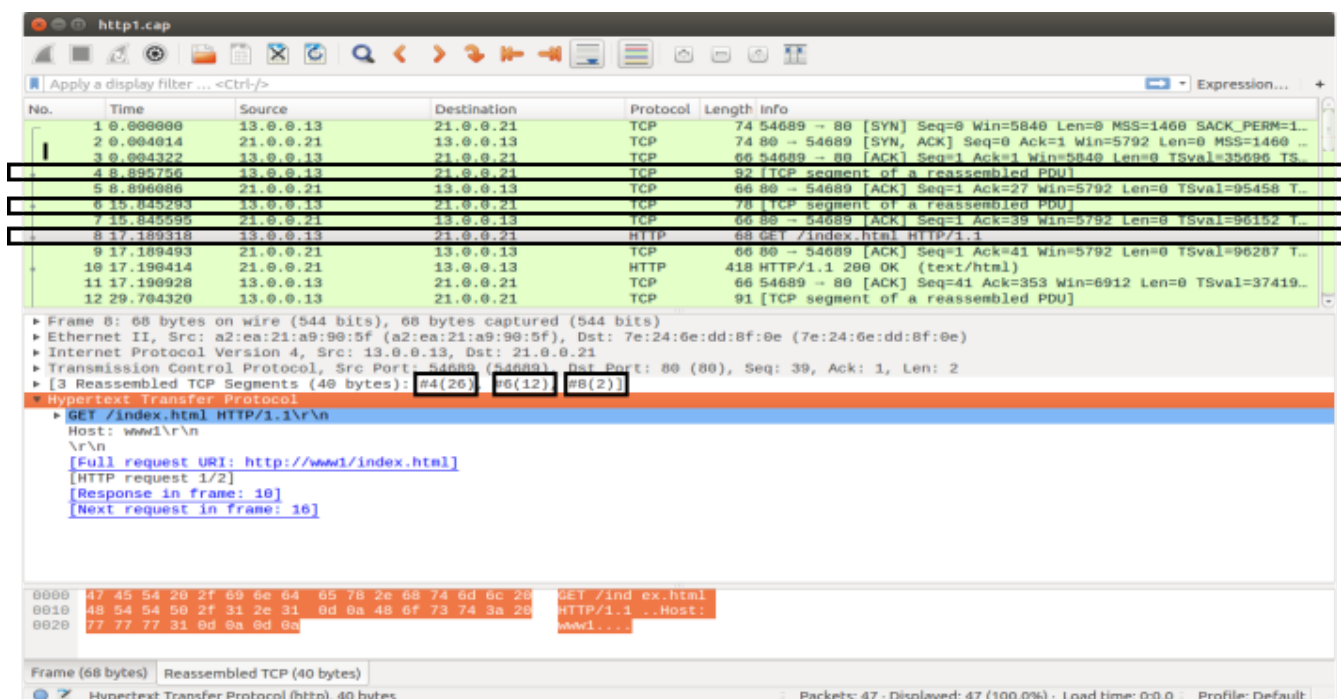


Figura 1: Mensaje HTTP compuesto de varios segmentos.

# 1. Comunicación cliente-servidor HTTP

Abre la captura `http1.cap` y responde a las siguientes preguntas:

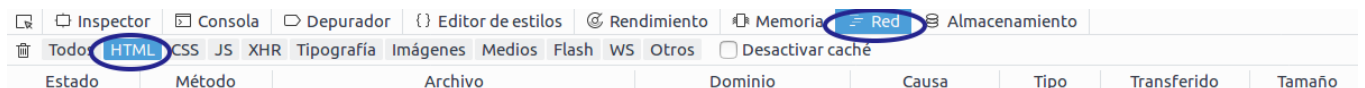
1. Indica qué dirección IP es la de la máquina cliente HTTP y cuál la del servidor.
2. Indica qué versión HTTP están utilizando cliente/servidor
3. Indica si están utilizando conexiones persistentes o no persistentes y el número de conexiones que se ven en el fichero de captura.
4. ¿Cuántas peticiones GET observas desde el cliente?
5. ¿Cuántas URLs crees que ha escrito el usuario en el navegador para obtener dicha captura? ¿Cuál/es? ¿Por qué?
6. Fíjate en el contenido de la página `index.html` que se ha descargado el cliente. ¿Qué crees que ocurrirá cuando el navegador se haya descargado `index.html`?

Abre la captura `http2.cap` y responde a las siguientes preguntas:

7. Indica qué versión HTTP están utilizando cliente/servidor
8. Indica si están utilizando conexiones persistentes o no persistentes y el número de conexiones que se ven en el fichero de captura. ¿Por qué?

# 2. Diferentes tipos de respuestas de un servidor

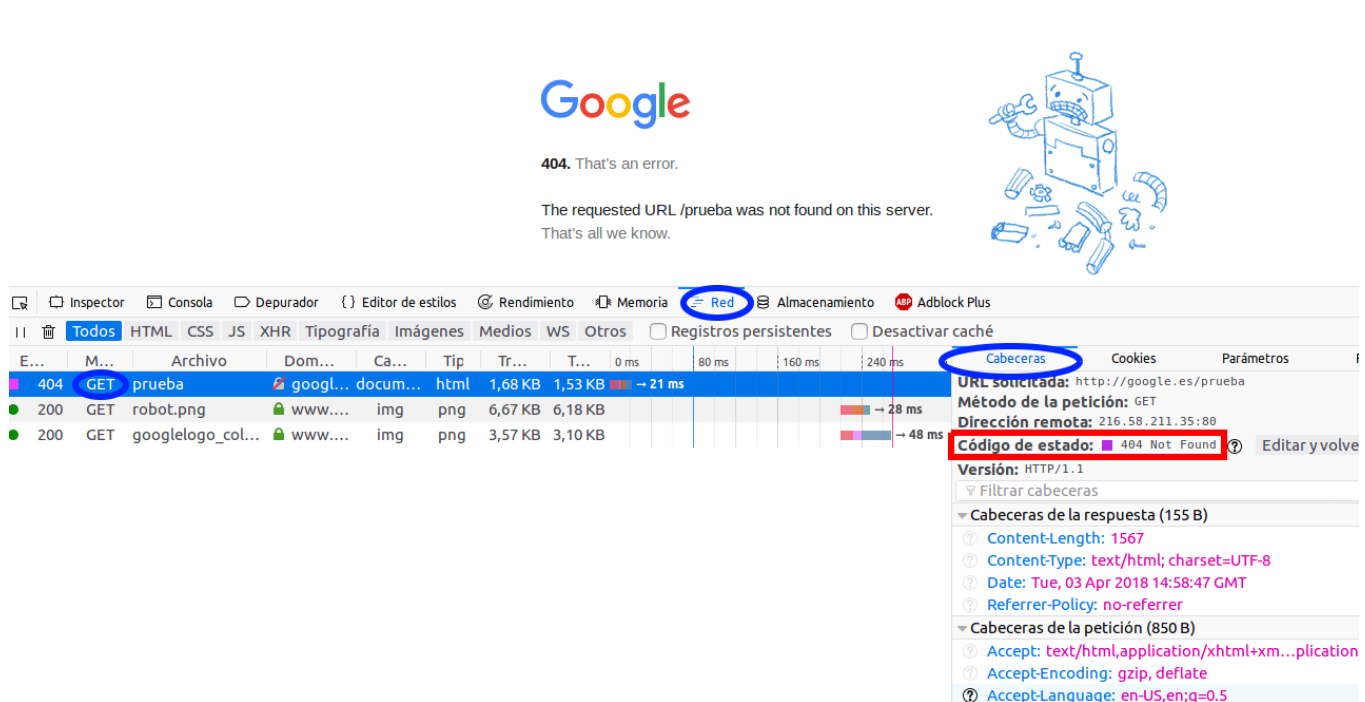
Abre una pestaña en el navegador y selecciona en el menú Herramientas → Desarrollador web → Alternar herramientas. Selecciona la pestaña Red y HTML:



Esta vista del navegador te permitirá observar todos los mensajes HTTP que se están intercambiando al cargar una página.

Explica lo que ocurre al cargar las siguientes URLs:

1. Dentro de esa pestaña carga la página `http://www.google.es/prueba`. Selecciona dentro de las herramientas del desarrollador la petición GET y la pestaña Cabeceras.



Fíjate en el campo Código de estado que indica el tipo de respuesta recibida y explica su contenido.

2. En esa misma pestaña carga la página <http://www.wikipedia.com>. Explica qué ocurre en la primera petición GET y a partir de las líneas de cabecera que ves en la respuesta, explica la segunda petición GET.

### 3. Formularios en HTTP

Abre la captura `http3.cap` y responde a las siguientes preguntas:

1. Indica qué tipo de conexiones HTTP utilizan el cliente y servidor e indica el número de conexiones entre cliente y servidor que aparecen en la captura.
2. Busca en la captura el segmento donde el servidor le envía al cliente un formulario. Indica los nombres de los campos del formulario que rellenará el usuario.
3. Indica si es el cliente o el servidor el que decide cómo debe enviar el cliente los datos del formulario (GET/POST) . ¿Qué método están usando en este caso? ¿Cómo lo sabes?
4. Busca en la captura el segmento donde el cliente le envía los datos del formulario al servidor y comprueba que se está realizando con el método GET
5. Fíjate cómo se llama el programa del servidor que va a recibir esos datos.
6. ¿Dónde viajan los datos que el cliente le envía al servidor? ¿Cuáles son esos datos?
7. Indica qué cabecera es la que representa el tipo de contenido del mensaje que el cliente envía al servidor con los datos del formulario.

Abre la captura `http4.cap` y responde a las siguientes preguntas:

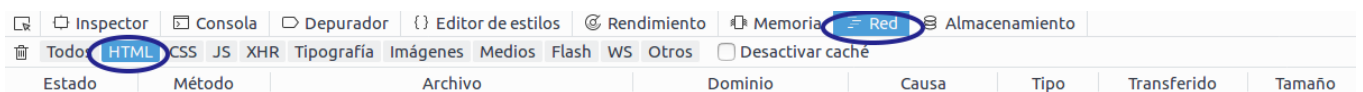
9. Busca en la captura el segmento donde el servidor le envía al cliente un formulario. Indica los nombres de los campos del formulario que rellenará el usuario.
10. Indica si es el cliente o el servidor el que decide cómo debe enviar el cliente los datos del formulario (GET/POST) . ¿Qué método están usando en este caso? ¿Cómo lo sabes?
11. Busca en la captura el segmento donde el cliente le envía los datos del formulario al servidor y comprueba que se está realizando con el método POST.
12. Fíjate cómo se llama el programa del servidor que va a recibir esos datos.
13. Indica qué cabecera es la que representa el tipo de contenido que el cliente envía al servidor y cuál es su valor.
14. Indica en qué parte del mensaje van los datos del formulario que el cliente le envía al servidor.
15. Explica si en este caso es necesario la cabecera **Content-Length** en el mensaje HTTP que el cliente envía al servidor con los datos del formulario. ¿Por qué?
16. Observa si el servidor le manda alguna respuesta cuando recibe los datos del formulario del cliente. En caso afirmativo localiza el número de segmento y observa en las cabeceras HTTP: tipo de contenido, longitud y cuerpo del mensaje

## 4. Cookies

### 4.1. Almacén de cookies en el navegador Firefox

Para ver las Cookies en el navegador Firefox, selecciona la opción de menú: Editar → Preferencias. En la zona de la izquierda selecciona la pestaña Privacidad y seguridad. Dentro de la sección Historial verás un enlace "eliminar cookies de forma individual". Pulsa sobre ese enlace para comprobar si tienes cookies del sitio web del Ayuntamiento de Fuenlabrada [ayto-fuenlabrada.es/](http://ayto-fuenlabrada.es/) (si las tienes, elimina sólo esas cookies). Cierra la ventana que te muestra las cookies, pero mantén esta pestaña con las preferencias del navegador.

Abre otra pestaña en el navegador y selecciona en el menú Herramientas → Desarrollador web → Alternar herramientas. Selecciona la pestaña Red y HTML:



Dentro de esa pestaña carga la página <http://www.ayto-fuenlabrada.es/>. Selecciona dentro de las herramientas del desarrollador la petición GET y la pestaña Cabeceras.

1. Fíjate en la línea de cabecera que el servidor le envía al cliente con el contenido de las cookies.
2. Pulsa sobre la pestaña "Cookies" para poder ver de forma más clara el contenido de las cookies. Copia los campos importantes. Fíjate que no hay fecha de expiración, eso quiere decir que la Cookie se eliminará cuando se cierre el navegador.
3. Vuelve a la pestaña donde tenías las preferencias del navegador y pulsa nuevamente sobre "eliminar cookies de forma individual". Indica qué cookies tienes almacenadas para el sitio `ayto-fuenlabrada.es`. Las cookies `__utmc`, `__utmt`, `__utma`, `__utmz` son debidas a que el sitio web usa Google Analytics, es decir, al descargar la página del Ayuntamiento de Fuenlabrada se ha descargado también una biblioteca en javascript que ha creado estas cookies para este sitio web dentro de nuestro navegador. Estas cookies permiten medir la interacción de los usuarios con el sitio web.
4. Vuelve a la pestaña donde tienes cargada la página `http://www.ayto-fuenlabrada.es/` y pulsa sobre la segunda petición GET que aparece, observarás que en esa petición web se envían todas las cookies almacenadas en el navegador. Usa también la pestaña Cookies para poder ver mejor los valores que se envían.

## 4.2. Envío de Cookies en mensajes HTTP

Abre la captura `http5.cap` y responde a las siguientes preguntas:

1. Indica qué cookies envía el servidor al cliente:
2. Indica qué cookies enviará el cliente al servidor cuando acceda a la página con la URL: `http://elcortebritanico/tienda/index.html`
3. ¿Y si el cliente accediera en el año 2025 a dicha URL?
4. ¿Y si el cliente accediera en el año 2035 a dicha URL?

Abre la captura `http6.cap` y responde a las siguientes preguntas:

- Indica qué cookies el cliente está enviando al servidor.
- ¿Por qué crees que le envía dichas cookies?
- Escribe un ejemplo de las posibles cabeceras que le habrá enviado dicho servidor al cliente previamente.
- A partir de la información de la captura ¿crees que si el cliente accede a otra página con la URL: `http://www2/dir1/dir2/index.html` enviaría esas cookies, más o menos?
- A partir de la información de la captura ¿crees que si el cliente accede a otra página con la URL: `http://www2/index.html` enviaría esas cookies, más o menos?
- A partir de la información de la captura ¿crees que si el cliente accede a otra página con la URL: `http://www/index.html` enviaría esas cookies, más o menos?

## 5. Comunicación a través de proxy HTTP

Abre la captura `http7.cap` y responde a las siguientes preguntas:

- Indica qué dirección IP es el cliente, el proxy y el servidor final.
- ¿Qué diferencia la petición HTTP que realiza el cliente de la petición que realiza el proxy?
- Identifica el nombre de la máquina donde se encuentra el servidor HTTP.
- ¿Se puede saber de la petición que realiza el proxy que dicho proxy tiene almacenada en su caché esa página?

Abre la captura `http8.cap` y responde a las siguientes preguntas:

- Indica qué tipo de conexiones HTTP utilizan el cliente y servidor e indica el número de conexiones entre cliente y servidor que aparecen en la captura.
- Explica qué es lo que se está descargando el cliente del servidor HTTP y cuantos objetos se descarga.
- Observa en las cabeceras HTTP el tipo de contenido de cada uno de los objetos.
- ¿Podrías saber si los paquetes capturados se corresponde a la comunicación entre un cliente y un proxy HTTP, entre un cliente y servidor final HTTP o entre un proxy y el servidor final HTTP? ¿Por qué?
- Sabiendo que la comunicación se ha realizado a través de un proxy HTTP, mira las cabeceras HTTP que envía dicho proxy para ver si en ellas existe alguna que muestre cuál es su nombre.

Abre la captura `http9.cap` y responde a las siguientes preguntas:

- ¿Podrías saber si los paquetes capturados se corresponde a la comunicación entre un cliente y un proxy HTTP, entre un cliente y servidor final HTTP o entre un proxy y el servidor final HTTP? ¿Por qué?

## 5.1. Caché en el proxy

Estudia las capturas `http10.cap` y `http11.cap` y responde a las siguientes preguntas:

- Indica cuáles son las direcciones IP del cliente, proxy y servidor web. ¿Cómo lo sabes?
- Explica qué es lo que ocurre en estas capturas.
- Localiza los campos relevantes con respecto al tratamiento de caché que incluye en las líneas de cabecera el servidor. ¿Qué significan?
- Explica qué ocurre en la segunda consulta que realiza el cliente.
- Viendo los paquetes 14 y 16 de la captura `http10.cap` indica cómo se puede saber que el contenido proviene de una caché.
- ¿Crees que el cliente tiene caché?

## 6. Caché en el navegador

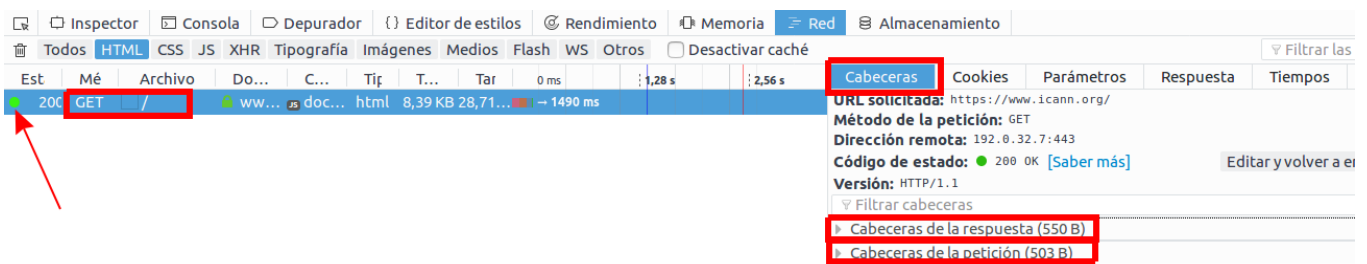
Busca en las preferencias del navegador la opción para eliminar el contenido de la caché.

Para ver cómo funciona la caché del navegador, despliega las herramientas del desarrollador tal y como lo hiciste en 4.1.

### 6.1. Sitio web ICANN

Carga en la URL la dirección de ICANN (Internet Corporation for Assigned Names and Numbers, organización que coordina la asignación de identificadores en Internet: direcciones IPs, números de ASs, etc): `http://www.icann.org`.

Selecciona la petición `GET /` y la pestaña Cabeceras (el punto verde al lado de la petición `GET` indica que se ha solicitado el recurso al servidor y se ha obtenido respuesta con el recurso):



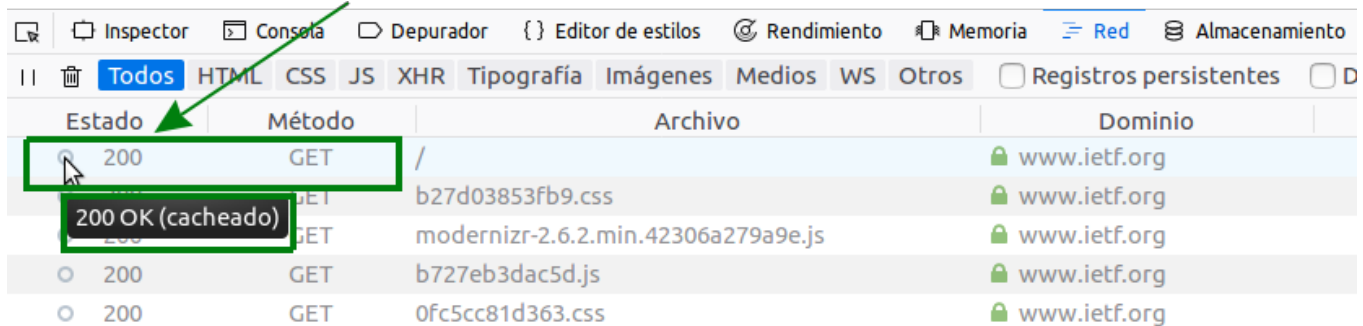
Al desplegar la pestaña Cabeceras de la respuesta y de la petición observarás las cabeceras que se han incluido en el mensaje de respuesta y petición para obtener el recurso `/`.

- Despliega la pestaña de las cabeceras de la respuesta e indica qué tipo de caché se permite.
- A la vista de los resultados, ¿qué crees que ocurrirá si se recarga la página en el navegador?
- Abre otra pestaña en el navegador, selecciona de nuevo en las herramientas → desarrollador web → alternar herramientas. Teclea en esta nueva pestaña la misma URL: `http://www.icann.org`, observarás que se muestra nuevamente el recurso solicitado y la respuesta con el mismo (junto a la solicitud se muestra el círculo verde).

## 6.2. Sitio web IETF

Carga en esa misma pestaña del navegador la dirección del IETF (Internet Engineering Task Force): <http://www.ietf.org>.

4. Selecciona la petición **GET /** y observa en las líneas de cabecera de respuesta qué tipo de caché se permite.
5. Indica cuándo expirará el recurso en la caché. Relaciona los valores de las cabeceras: **Date**, **Expires** y **Cache-Control**.
6. A la vista de los resultados ¿qué crees que ocurrirá si desde otra pestaña se solicita la misma URL?
7. Abre otra pestaña en el navegador, selecciona de nuevo en las herramientas → desarrollador web → alternar herramientas. Teclea en esta nueva pestaña la misma URL: <http://www.ietf.org>. Selecciona de nuevo la petición **GET /**. Al posicionar el ratón sobre el nombre de la petición **GET**, observarás que se muestra que el contenido estaba cacheado (fíjate que ahora ya no aparece un círculo verde al lado de **GET /**, en su lugar hay un círculo gris). Fíjate en las líneas de cabecera: **Date**, **Expires** y **Last-modified** y explica si han cambiado y por qué.



Estado	Método	Archivo	Dominio
200	GET	/	www.ietf.org
200 OK (cacheado)	GET	b27d03853fb9.css	www.ietf.org
200	GET	modernizr-2.6.2.min.42306a279a9e.js	www.ietf.org
200	GET	b727eb3dac5d.js	www.ietf.org
200	GET	0fc5cc81d363.css	www.ietf.org

## 6.3. Sitio web MECD

Carga en una de las pestañas donde tengas abiertas las herramientas del desarrollador, la dirección del Ministerio de Educación: [www.mecd.gob.es/portada-mecd/](http://www.mecd.gob.es/portada-mecd/), selecciona la petición **GET /portada-mecd/** y responde a las siguientes preguntas:

9. Observa las líneas de cabecera de la respuesta e indica qué tipo de caché se permite.
10. Indica cuándo expirará el recurso en la caché. Relaciona los valores de las cabeceras: **Date**, **Expires** y **Cache-Control**.
11. Abre una nueva pestaña del navegador con las herramientas del desarrollador visibles. Indica qué ocurre si tecleas la misma URL.
12. Espera el tiempo necesario para que la información caduque en la caché y abre una nueva pestaña con las herramientas del desarrollador visibles. Explica qué sucede cuando vuelves a teclear esta URL.
13. ¿Se está usando validación fuerte o débil?