HTTP

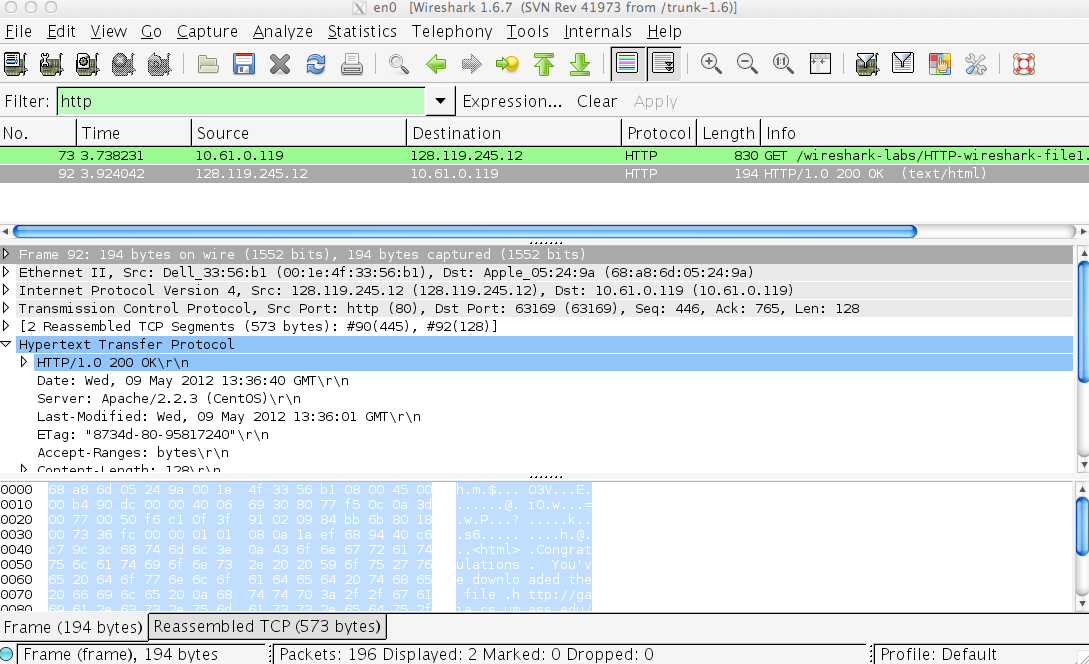
*Traducción basada en el material suplementario del libro Redes de computadores: un enfoque de arriba hacia abajo, 7ª ed, J.F. Kurose y K.W. Ross.*

*“Dime y lo olvido. Muéstrame y recuerdo. Implicarme y entiendo “.* Proverbio chino

En esta actividad, vamos a explorar varios aspectos del protocolo HTTP: la interacción básica GET/respuesta, los formatos de mensaje HTTP, recuperar grandes archivos HTML y recuperar archivos HTML con objetos incrustados. Para ello, nos basaremos en archivos con capturas previas de Wireshark acorde a cada una de los ejercicios previstos. Ests capturas pueden descargarse desde <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>. Las capturas en este archivo zip se recogieron por Wireshark en uno de los ordenadores de los autores, mientras realizaba los pasos indicados en cada ejercicio. Una vez descomprimido, puedes cargarlos en Wireshark a través del menú, opción Abrir y, a continuación, seleccionar el archivo de captura correspondiente.

# La interacción básica HTTP GET / respuesta

Vamos a comenzar nuestra exploración de HTTP examinando la descarga de un archivo HTML muy simple (corto que no contiene objetos incrustados) desde el servidor <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html>. Para ver los resultados de esta captura, abre en Wireshark el archivo **http-ethereal-trace-1**. La ventana mostrada por Wireshark debe ser similar a la ventana que se muestra en la Figura 1.



**Figura 1:** Ventana con la captura de página Web solicitada por su navegador

El ejemplo de la figura 1 muestra los dos paquetes que fueron capturados con dos mensajes HTTP: el mensaje GET (desde su navegador al servidor Web gaia.cs.umass.edu) y el mensaje de respuesta desde el servidor a su navegador. La ventana de contenido de paquetes muestra los detalles del mensaje seleccionado (en este caso, el mensaje de respuesta HTTP/1.0 200 OK, que se destaca en la ventana de lista de paquete). Recordemos que, puesto que el mensaje HTTP se llevó dentro de un segmento TCP, que se llevó dentro de un datagrama IP, que se llevó dentro de una trama Ethernet, Wireshark también mostrará esta información. En esta actividad sólo nos interesa conocer con detalle la información relativa al protocolo HTTP. Para el resto de protocolos, la información debe estar colapsada para no interferir en la interpretación de los datos.

Observa la información contenida en los mensajes HTTP GET y respuesta para responder a las siguientes preguntas:

1. ¿Está ejecutando el navegador la versión HTTP 1.0 ó 1.1? ¿Qué versión de HTTP se ejecuta el servidor? ¿Cuáles son los números de paquete en los que puedes ver esta información respectivamente?
2. ¿Qué idiomas (si los hay) dice su navegador que puede aceptar para las respuestas del servidor?
3. ¿Cuál es la dirección IP de su ordenador? ¿Y la del servidor gaia.cs.umass.edu?
4. ¿Cuál es el código de estado devuelto desde el servidor a su navegador en la respuesta?
5. ¿Cuál es la fecha de la última actualización del código HTML del archivo recuperado desde el servidor?

# La interacción HTTP CONDICIONAL GET / respuesta

La mayoría de los navegadores Web disponen de una caché de objetos y por lo tanto realizan un GET condicional al recuperar un objeto HTTP. Antes de realizar los siguientes pasos, asegúrese de **vaciar la** **caché del navegador** (para hacer esto en Firefox, seleccione Herramientas> Limpiar el historial reciente y marca la casilla de caché, o para Internet Explorer, seleccione Herramientas> Opciones de Internet> Eliminar archivos; estas acciones eliminarán los archivos almacenados de la caché del navegador).

Para la siguiente exploración de HTTP examinaremos la descarga de un archivo HTML de nuevo muy simple (corto que no contiene objetos incrustados) desde el servidor <http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html>. Para ver los resultados de esta captura, abre en Wireshark el archivo **http-ethereal-trace-2**. Responde las siguientes preguntas:

1. Inspecciona el contenido de la **primera solicitud** HTTP GET desde el navegador al servidor. ¿Encuentras una línea de cabecera “If-Modified-Since” en el HTTP GET? ¿Cuál es el número de paquete correspondiente?
2. Inspecciona el contenido de la respuesta del servidor. ¿El servidor ha devuelto de manera explícita el contenido del archivo? ¿Cuál es el número de paquete dónde podría encontrarse esta información?
3. Ahora inspecciona el contenido de la **segunda petición** HTTP GET desde el navegador al servidor. ¿Encuentras una línea de cabecera “If-Modified-Since” en el HTTP GET? Si es así, ¿qué información completa la cabecera “If-Modified-Since: …”?
4. ¿Cuál es el código de estado HTTP y la frase devuelta por el servidor en respuesta a esta segunda HTTP GET? ¿El servidor ha devuelto de manera explícita el contenido del archivo? Explícalo.