Más sobre peticiones HTTP

Contenido

[Wireshark HTTP 2](#_Toc83319349)

[HTTP y documentos grandes 2](#_Toc83319350)

[HTTP con objetos embebidos 3](#_Toc83319351)

[Autenticación HTTP 3](#_Toc83319352)

# Wireshark HTTP

## HTTP y documentos grandes

En nuestros ejemplos hasta ahora, los documentos recuperados han sido archivos HTML simples y cortos. A continuación, veamos qué sucede cuando descargamos un archivo HTML largo. Haz lo siguiente:

- Inicie su navegador web y asegúrese de que la memoria caché de su navegador esté limpia, como se discutió anteriormente.

- Poner en marcha el paquete de Wireshark sniffer

- Ingrese la siguiente URL en su navegador

<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html>

Su navegador debe mostrar la extensa Declaración de Derechos de EE . UU.

Detenga la captura de paquetes de Wireshark e ingrese “http” en la ventana de especificación del filtro de visualización, de modo que solo se muestren los mensajes HTTP capturados.

En la ventana de listado de paquetes, debería ver su mensaje HTTP GET, seguido de una respuesta TCP de múltiples paquetes a su solicitud HTTP GET. El cuerpo de la entidad en la respuesta es todo el archivo HTML solicitado. En nuestro caso aquí , el archivo HTML es bastante largo y con 4500 bytes es demasiado grande para caber en un paquete TCP. Por lo tanto, el **mensaje de respuesta HTTP único se divide en varias partes mediante TCP**, conteniendo cada parte dentro de un segmento TCP separado. En versiones recientes de Wireshark, Wireshark indica cada segmento TCP como un paquete separado, y el hecho de que la respuesta HTTP única se fragmentó en múltiples paquetes TCP se indica mediante el "TCP segment of a reassembled PDU" en la columna Información de la pantalla Wireshark . Las versiones anteriores de Wireshark usaban la frase " Continuation " para indicar que todo el contenido de un mensaje HTTP se dividía en múltiples segmentos TCP. ¡Destacamos aquí que no hay un mensaje de " Continuation " en HTTP!

Responda las siguientes preguntas:

1. ¿Cuántos mensajes de solicitud HTTP GET envió su navegador? ¿Qué número de paquete contiene el mensaje GET para la Ley o Derechos?

**Un HTTP GET.**

**315860**

2. ¿Qué número de paquete contiene el código de estado y la frase asociados con la respuesta a la solicitud HTTP GET ?

**315865**

3. ¿Cuál es el código de estado y la frase en la respuesta?

**200 OK (texto/html)**

4. ¿Cuántos segmentos TCP que contienen datos fueron necesarios para llevar la respuesta HTTP única y el texto de la Declaración de Derechos?

**2 segmentos**

## HTTP con objetos embebidos

Pasamos a ver qué sucede cuando su navegador descarga un archivo con objetos incrustados, es decir, un archivo que incluye otros objetos (en el ejemplo a continuación, archivos de imagen ) que están almacenados en otro (s) servidor (es).

Haz lo siguiente:

- Inicie su navegador web y asegúrese de que la memoria caché de su navegador esté limpia, como se discutió anteriormente.

- Poner en marcha la captura de paquetes

- Ingrese la siguiente URL en su navegador

<http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html>

Su navegador debe mostrar un archivo HTML corto con dos imágenes. Se hace referencia a estas dos imágenes en el archivo HTML base. Es decir, las imágenes en sí mismas no están contenidas en el HTML; en cambio, las URL de las imágenes están contenidas en el archivo HTML descargado. Su navegador tendrá que recuperar estos logotipos de los sitios web indicados: uno de www.aw-bc.com y otra del servidor manic.cs.umass.edu.

Detenga la captura de paquetes de Wireshark e ingrese “http” en la ventana de especificación del filtro de visualización, de modo que solo se muestren los mensajes HTTP capturados.

Responde las siguientes preguntas:

1. ¿Cuántos mensajes de solicitud HTTP GET envió su navegador? ¿A qué direcciones de Internet se enviaron estas solicitudes GET ?

**4 GETs**

**/Pearson.png**

**/8E\_cover\_small.jpg**

**/fav\_icon.ico**

2. ¿Puede decir si su navegador descargó las dos imágenes en serie o si se descargaron de los dos sitios web en paralelo?   
**Se descargaron a diferente tiempo, una fue 200ms después de la otra**

## Autenticación HTTP

Finalmente, intentemos visitar un sitio web que esté protegido con contraseña y examinemos la secuencia del mensaje HTTP intercambiado por dicho sitio. La URL <http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html> está protegida con contraseña . El nombre de usuario es "wireshark-students" (sin las comillas), y la contraseña es "network" (de nuevo, sin las comillas). Así que accedamos a este sitio "seguro" protegido por contraseña. Haz lo siguiente:

1- Asegúrese de que la memoria caché de su navegador esté limpia, como se discutió anteriormente, y cierre su navegador. Luego, inicie su navegador

2 - Poner en marcha la captura de paquetes

3- Ingrese la URL mencionada anteriormente en su navegador

4 - Escriba el nombre de usuario y la contraseña solicitados en el cuadro emergente.

5- Detenga la captura de paquetes de Wireshark e ingrese “http” en la ventana de especificación del filtro de visualización, de modo que solo los mensajes HTTP capturados se muestren más adelante en la ventana de listado de paquetes.

Referencia (no es necesario consultarla para continuar la práctica)

“HTTP Access Authentication Framework” <http://frontier.userland.com/stories/storyReader$2159>

Responda las siguientes preguntas:

1. ¿Cuál es la respuesta del servidor (código de estado y mensaje) en respuesta al mensaje HTTP GET inicial de su navegador?

**401 - Unauthorized**

2. Cuando su navegador envía el mensaje de HTTP GET, por segunda vez, ¿cuál es el nuevo campo que está incluido en el HTTP GET?

**Authorization (Credentials: …)**

El nombre de usuario (wireshark-students) y la contraseña (network) que ingresó están codificados en la cadena de caracteres (d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms =) después del encabezado "Authorization: Basic" en el mensaje HTTP GET del cliente. Si bien puede parecer que su nombre de usuario y contraseña están encriptados, simplemente están codificados en un formato conocido como Base64

¡El nombre de usuario y la contraseña *no*están encriptados! Para ver esto, vaya a [http://www.motobit.com/util/base64-decoder-encoder.asp](https://translate.google.com/translate?hl=es&prev=_t&sl=en&tl=es&u=http://www.motobit.com/util/base64-decoder-encoder.asp) e ingrese la cadena codificada en base64 d2lyZXNoYXJrLXN0dWRlbnRz y decodifique. ¡Ha traducido de la codificación Base64 a la codificación ASCII y, por lo tanto, debería ver su nombre de usuario! Para ver la contraseña, ingrese el resto de la cadena Om5ldHdvcms = y presione decodificar. Dado que cualquiera puede descargar una herramienta como Wireshark y detectar paquetes (no solo los suyos) este tipo de sitios web no son seguros a menos que se tomen medidas adicionales.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.