

Hoja de ejercicios Tema 1

Introducción a las tecnologías web

El protocolo HTTP

▷ **1.** Descarga el proyecto `simple-server` del *Github* de la asignatura. Se trata de un proyecto *Node.js* que arranca un servidor en el puerto 3000. El objetivo de este ejercicio es utilizar las herramientas de desarrollador de Firefox (concretamente, la pestaña Red) para observar las peticiones HTTP que se producen al acceder a las distintas páginas de este servidor.

- (a). Introduce en el navegador la URL `http://localhost:3000/pagina1.html`. ¿Cuántas peticiones se realizan? ¿Qué código HTTP de respuesta devuelve cada una? Comprueba qué peticiones envía el navegador al hacer clic en el enlace de la página.
- (b). La URL `http://localhost:3000/pagina3.html` lleva a una página con un formulario de búsqueda. Rellena este formulario y haz clic en el botón *Buscar*. ¿Cómo se envía la información del formulario? ¿Qué pasa si no se selecciona ninguna de las opciones de búsqueda? ¿Y si se introduce "Me gustan los M&M's" como cadena de búsqueda? ¿Es posible enviar los datos al servidor sin necesidad de rellenar el formulario y sin utilizar las herramientas del desarrollador de Firefox? Repite el ejercicio con el formulario contenido en la URL `http://localhost:3000/pagina4.html`. ¿Qué diferencias observas? ¿En qué casos es más conveniente una petición de tipo GET para enviar formularios?
- (c). La página `http://localhost:3000/pagina5.html` contiene otro formulario en el que el usuario puede subir una imagen. Observa la petición HTTP realizada al enviar el formulario.
- (d). Accede a la URL `http://localhost:3000/pagina6.html` y pulsa el botón *Refresh* del navegador varias veces. ¿Qué observas? Analiza las peticiones y respuestas HTTP para buscar alguna cabecera que explique el funcionamiento de la página.

Utiliza la pestaña de *Almacenamiento*¹ de las herramientas de desarrollo de *Firefox* o la pestaña *Aplicación* de las herramientas de *Chrome* para visualizar las *cookies* guardadas en el navegador. Estas herramientas permiten modificar el contenido de las *cookies* y eliminarlas.

- ▷ **2.** En la mayoría de los sitios web que permiten algún tipo de identificación (Amazon, Gmail, UCM, etc.) no es necesario introducir el nombre de usuario y contraseña cada vez que se accede a una página de estos sitios. Por tanto, debe de haber alguna *cookie* que mantenga esta información y la proporcione al servidor cada vez que el usuario, desde su navegador, accede a una página de estos sitios. Imaginamos que la *cookie* almacenase directamente el nombre del usuario actualmente *logueado*. ¿Qué implicaciones, con respecto a la seguridad, tendría esto? Observa las *cookies* que se guardan en el navegador cada vez que te identificas en cada uno de estos sitios. ¿Eres capaz de hacer que el sitio web no te 'recuerde' manipulando estas *cookies*?
- ▷ **3.** Mediante un navegador web accede a la URL `http://dalila.sip.ucm.es/protected/`. El navegador te solicitará un nombre de usuario y contraseña. Pulsa el botón *Cancelar* y observa la respuesta HTTP devuelta. Haz lo mismo introduciendo un nombre de usuario y contraseña aleatorios. Por último, accede utilizando el nombre de usuario y contraseña correctos (respectivamente, `usuario` y `1234`) y analiza la petición HTTP enviada. ¿Cómo de seguro es este mecanismo?

¹ Si no te aparece esta pestaña en las herramientas de desarrollador, es posible que la tengas oculta por defecto. Puedes hacerla visible utilizando la configuración de las herramientas, a la que se accede mediante el icono con el dibujo de una rueda dentada en la barra superior de las herramientas de desarrollador.

- ▷ 4. Accede a la página de entrada al CV mediante la URL <https://cv4.ucm.es/moodle/>. Observa la respuesta HTTP devuelta. A continuación introduce la URL <https://cv4.ucm.es/moodle> (sin el símbolo '/' al final) y vuelve a observar la respuesta HTTP. ¿Qué diferencia observas?