Ejercicio Formativo #1

CC5002 - Desarrollo de Aplicaciones Web - Otoño 2021

Profesor: José Urzua - Auxiliares: Pablo Pizarro - Gabriel Iturra
Alumno: Matías Vergara

A.

¿Qué dificultades presenta implementar una aplicación de este estilo en términos de seguridad?

Una aplicación como esta es compleja en términos de seguridad por distintos motivos:

- Primeramente, las retricciones impuestas sobre las entradas son fácilmente vulnerables. Cualquier usuario malintencionado podría utilizar la herramienta de inspección que los navegadores ofrecen para modificar el documento y enviar algún formulario desacorde a ellas, para lo cual el servidor debería realizar una verificación previa antes de realizar cualquier cambio en la base de datos.
- Lo anterior también aplica para la posibilidad de enviar formularios conforme a las restricciones, pero con un contenido construido para ser interpretado de cierta forma no deseada por el servidor/base de datos y así lograr accesos indebidos (por ejemplo, acceder a archivos privados, a una lista de contraseñas, o incluso borrar todo). Para ello sería necesario **sanitizar** las entradas.
- Otro aspecto importante es que la aplicación actualmente no limita el tamaño de los archivos subidos. Aún si el objetivo es superar el limite de Dropbox, debería existir algún límite, de lo contrario una persona malintencionada podría saturar rápidamente el sistema y volverlo inutilizable.
- Por otro lado, manejar archivos con contraseña requiere prestar atención a ciertos aspectos: dichas contraseñas deberían guardarse encriptadas, y la desencriptación debería ser imposible incluso para los desarrolladores (cada intento de contraseña debería encriptarse y permitir el acceso solo si los hashcode resultantes son iguales).
- Dar privacidad a un archivo únicamente mediante un link oculto también es peligroso, pues podría ser adivinado o filtrarse. Una primera capa de seguridad es que el link no sea trivial (hackbox.com/pauta_examen.pdf), si no un código (hackbox.com/123u28u3iqje182j).
- Finalmente, la autodestrucción de un archivo debe implementarse con precaución: si para identificar el archivo a destruir se utiliza su nombre, un usuario malintencionado podría subir un archivo con el mismo nombre de otro - no esta planeado para autodestruirse - con la opción de autodestrucción activada y conseguir que se destruyan ambos.

Desde luego, es probable que existan otras vulnerabilidades adicionales a las ya mencionadas. Identificarlas suele ser un proceso iterativo complejo.

B.

¿Cuál es la diferencia entre HTML y CSS?

HTML (HyperText Mark-up Language) es un *lenguaje de marcado de hipertexto*, usado para entregarle al navegador web cierto contenido y que éste entienda como desplegarlo, "creando" el sitio. CSS (Cascading Style Sheets), por otro lado, es un *lenguaje de diseño gráfico,* el cual se utiliza para modificar la forma en que se ve el contenido en la página - colores, tamaños, grosores, etc - más sin modificar su estructura. Se podría decir que HTML trabaja sobre la lógica de la página - qué rol cumple cada pieza, jerarquías entre los elementos, etc. - y CSS trabaja sobre la visualización de la página - cómo mostrar cada pieza -.

C.

¿Cual es la diferencia entre HTML y HTML5?

Como ya se mencionó en la pregunta anterior, HTML es un lenguaje de marcado de hipertexto , mientras que HTML5 es la quinta versión de dicho lenguaje, la cual incorpora algunas características nuevas con respecto a sus versiones anteriores (características entre las cuales destaca la compatibiidad con CSS3, la última versión de CSS - además de la incorporación de nuevas etiquetas, tales como <header> y <footer> -). HTML5.2 es la versión actualmente recomendada por la W3C, estándar desde el año 2014.

D.

¿Todos los tags en HTML son del estilo <tag>, <\tag>?

Si por dicho estilo nos referimos a que requieren ser abiertos y cerrados, la respuesta es no. Existen tags para los cuales basta "abrirlos" para que cumplan su propósito (tales como
) - y cerrarlos es redundante -.