UANL

FCFM

Doo Tarea 4

Adrián Campos Treviño

1547530

Introducción

HTML cuyo significado es Lenguaje de Marcado para Hipertextos (Hypertext Markup Language), ojo, no confundirlo como un lenguaje de programación, es el elemento de construcción más básico de una página web y se usa para crearlas y representarlas visualmente. Estas determinan el contenido de la página web, pero no determinan su funcionalidad.

También existen otras tecnologías distintas a HTML que son usadas por lo general para describir la apariencia de una página web en el caso de CSS o su funcionalidad que sería JavaScript.

Ahora, hablando más fondo acerca de JavaScript, este es un lenguaje de programación que permite a los desarrolladores a crear acciones en sus páginas web, aunque esta no requiere de compilación ya que el lenguaje funciona por medio del lado del cliente ya que los navegadores son los encargados de interpretar estos códigos.

La razón por la que JavaScript fue creado fue porque Netscape (unos de los navegadores más importantes de su época) decidió que se necesitaban dos leguajes para sus navegadores en lugar de tener solo Java el cual era visto solo un componente de lenguaje usado por programadores hábiles, aunque ellos buscaban un lenguaje más simple para aquellos programadores casuales quienes pudieran convertirse gracias a esto en desarrolladores web creando contenido como imágenes, Java applets, plugins, ect. Debido a la simplicidad de este, los programadores que usaran JavaScript podrían integrar componentes y automatizar sus interacciones. Gracias esto, JS se encuentra dentro de documentos HTML y permite la interacción con las páginas web en formas que nunca hubieran sido posibles con solo HTML.

Los problemas emergentes

Los hackers utilizan herramientas de explotación de JavaScript para poder atacar páginas web, organizaciones como individuos.

Las vulnerabilidades en JavaScript pueden ser tanto problemas dentro del cliente-servidor como la pesadilla de cualquier empresa ya que los hackers pueden robar datos del lado del servidor e infectar a los usuarios con malware.

El uso más común de vulnerabilidad en JavaScript es el conocido Ataque de Cross-Site Scripting (XSS), donde a través de la manipulación de los scripts de JS y HTML, los hackers pueden ejecutar scripts maliciosos conocidos como “payloads

maliciosos” donde utilizan navegadores “inofensivos” los cuales pueden resultar en que el script termine incrustándose en la página web que están visitando, así dando a que cada vez que un usuario visite la página web o cierta acción predefinida sea ejecutada, el script malicioso es liberado y ejecutado.

Estos ataques tienen el potencial de causar serios problemas a las cuentas de compañías y empresas los cuales pueden resultar en robo de identidad y robo de información. También existen los ataques de solicitud falsa (Cross-Site Request Forgery “CSRF”) la cual es una forma de exploit que se produce cuando los comandos no autorizados, que normalmente se rechazan, resultan en que el sitio web crea que el atacante e un usuario autorizado. Tras una explotación exitosa de esta vulnerabilidad, el hacker puede acceder a las funciones de la aplicación web que normalmente no tendría.

Al igual que con los ataques XSS, los riesgos asociados a los ataques CSRF incluyen el robo de identidad, modificación de la aplicación utilizando la información de la víctima utilizando sus permisos y credenciales.

Una de las cosas que pueden hacerse para evitar estos ataques es realizando validaciones en la página web que son una parte importantísima en lo que seguridad se refiere, si los desarrolladores tienen en cuenta algo tan básico estos tipos de ataques se pueden mitigar de una forma considerable ya que de nuevo, esto es tan solo una de las formas en las que se puede comprometer la seguridad de la página web aunque así que o se pueden evitar todos los peligros que conlleva el hecho de elaborar una página mediante estos 2 lenguajes muy útiles para desarrollo web.