UANL

FCFM

Doo Tarea 3

Adrián Campos Treviño

1547530

Aspectos de seguridad en html y javascript.

Ya que en la actualidad el crecimiento del internet ha impactado en gran manera en la seguridad de la información manejada diariamente, mantener todos estos datos protegidos resulta demasiado importante. Así mismo, uno de los puntos más críticos de la seguridad en Internet son las herramientas que trabajan directamente con los usuarios, como los servidores web; sin embargo, la mayoría de los problemas detectados en servicios web no son a causa de los servidores o del lenguaje utilizado, sino por malas prácticas de parte de los programadores. Las amenazas a la seguridad evolucionan tan rápido como la tecnología que intentan comprometer. La base de datos CVE (Vulnerabilidades y Exposiciones Comunes) incluye de por sí más de 59.000 amenazas conocidas contra la seguridad de la información, y una búsqueda en la base de datos de apache devuelve una lista de más de 500 vulnerabilidades conocidas. Así, cuanto más valiosa sea la información agrupada dentro de los datos de las páginas web y su base de datos, mayores serán las amenazas posibles de perderlas o de ser un blanco de ataque. Si sus registros consisten en información financiera o confidencial que podría facilitar el robo monetario o un fraude, su base de datos será más atractiva para hackers que puedan utilizar las vulnerabilidades encontradas para robar y utilizar o vender esta información y así conseguirse sus propios beneficios. JavaScript es un lenguaje de programación que fue desarrollado justamente para hacer que páginas estáticas tengan un “comportamiento dinámico”. Permite ejecutar códigos directa y automáticamente gracias a su interacción con el navegador utilizado para visitar la página. De esta forma, la estructura de la página sufre alteraciones más allá de estar programada en HTML (lo cual la haría estática). Y, aunque esta forma es más atractiva respecto a los aspectos visuales y la rapidez y facilidad para el programador, es también el lenguaje ideal para hacer ataques, saltando herramientas de seguridad. Otra ventaja que encuentran los cibercriminales en la utilización de JavaScript es que sus códigos pueden explotar múltiples vectores al mismo tiempo, según las características del navegador utilizado y su interacción con la página web. En otras palabras, no hace falta propagar el código a varios sitios; con tan solo estar presente en un sitio de uso masivo, es posible atacar a muchas víctimas y robar nombres de usuarios y contraseñas o capturar contraseñas ingresadas en el teclado, entre otras cosas. Junto con todo lo anterior citado, se incluyen las librerías obsoletas de JavaScript que utilizan los programadores, aún teniendo en cuenta la gran cantidad de vulnerabilidades encontradas anteriormente en estas librerías. Según lo que se ha publicado en ZDNet, de 133.000 webs escaneadas, al menos un 37% de ellas están dotadas de una librería JavaScript con vulnerabilidad bastante conocida. De este mismo hecho La Northwestern University ya había alertado al mundo en un estudio. Dado que se hizo caso omiso de tal hecho, la Universidad y su grupo de investigadores decidieron publicar otro estudio, en el que señalan que las librerías vulnerables pueden ser “muy peligrosas” bajo las condiciones adecuadas, como ejemplo se tiene un antiguo bug de JQuery que se podía explotar usando un ataque de secuencia de comandos entre páginas o XSS. Un reciente estudio realizado por la College of Computer and Information Science ha demostrado que más de un tercio de las páginas web disponibles utilizan librerías JavaScript desactivadas que tienen, al menos, una o más vulnerabilidades conocidas y que están poniendo en peligro tanto a los usuarios

como al propio servidor. Además, el estudio también ha demostrado que muchas páginas web utilizan librerías como SWFObject y YUI que, desde hace tiempo, ya no tienen mantenimiento y cuentan con vulnerabilidades conocidas. Además de las librerías instaladas en el propio servidor, las librerías cargadas por terceras aplicaciones, como extensiones, plugins o widgets, utilizan también versiones obsoletas de JavaScript que ponen en peligro a los usuarios. Por lo mismo, existen ciertas prácticas básicas de seguridad web para permitir a los programadores seguir el fin de proteger los datos en Internet de una manera más práctica y protocolizada, así como vulnerabilidades comunes en páginas web:

Balancear riesgo y usabilidad

Normalmente siempre se debe pensar en las maneras en que usuarios ilegítimos nos pueden atacar y la facilidad de uso para los usuarios legítimos. Es conveniente emplear medidas de seguridad que sean transparentes a los usuarios y que no resulten engorrosas en su empleo. Por ejemplo, el uso de un login que solicita el nombre de usuario y contraseña, permite controlar el acceso de los usuarios hacia secciones restringidas de la aplicación. Este paso adicional, es una característica que impacta en la rapidez de acceso a la información por parte del usuario, pero que proporciona un elemento adicional de protección.

Rastrear el paso de los datos

Es muy importante mantener conocimiento de los pasos que ha recorrido la información en todo momento. Conocer de dónde vienen los datos y hacia dónde van. En muchas ocasiones lograr esto puede ser complicado, especialmente sin un conocimiento profundo de cómo funcionan las páginas web.

Filtrar entradas

Existen muchos puntos de vista diferentes sobre cómo realizar el filtrado o proceso de limpieza. Lo que usualmente se recomienda es ver al filtrado como un proceso de inspección, no debemos tratar de corregir los datos, es mejor forzar a los usuarios a jugar con las reglas válidas. Si llegamos a utilizar algún framework se debe tener especial cuidado, ya que estos brindan tantas comodidades que muchos desarrolladores inexpertos los utilizan sin preocuparse en entender el código que están observando y por lo tanto implementan medidas de validación en entradas, variables, entre otros, sin entender exactamente el funcionamiento de la solución empleada.

Exposición de datos

Una de las preocupaciones más comunes relacionadas con las bases de datos es la exposición de datos sensibles. Al almacenar números de tarjetas de crédito, por ejemplo,

es preferible asegurarse que los datos almacenados en la base de datos se encuentran seguros e inaccesibles incluso para los administradores de la base.

Páginas privadas y los sistemas de autenticación

La autenticación consiste en verificar la identidad de un usuario. Comúnmente el procedimiento involucra un nombre de usuario y una contraseña a revisar. Muchas aplicaciones tienen recursos que son accesibles sólo para los usuarios autenticados, así como recursos totalmente públicos.

Sesiones y Cookies

Este tipo de vulnerabilidad se debe a la utilización indebida de las sesiones en cuando el usuario utiliza logueos o autenticación para acceder a alguna sección de la web. Debemos tener cuidado de encriptar la información que guardamos en secciones y cookies para evitar que puedan ser leídas y utilizadas por un atacante.

Inyección de Código

Esta técnica consiste en enviar código por medio de las url y que esos datos no esten validados cuando los ejecute la web, las inyecciones sql y php las más frecuentes. Para encontrar posibles sitio es buscar en Google web que contengan una variable en su url por ejemplo. Con todo lo anterior presentado, por tanto, hay que mantener bien detectado, como programador, todas estas vulnerabilidades en las páginas web que aumentan la amenaza de sufrir un ataque. Existen herramientas de detección de amenazas y vulnerabilidades como VirusTotal, que es una herramienta de escaneo online proporcionado por Google, proporciona e inspecciona de forma rápida los archivos de un sitio web que este antivirus encuentre en el servidor, aunque solamente sirve como escánero y no como desinfectador de la web. O MXToolbox, que ofrece un conjunto de herramientas online para comprobar el desempeño, funcionamiento y reputación de un dominio o ip. Si con esta herramienta también verificamos que aparecemos en alguna lista negra deberemos evitar mediante firewall, antispam o suspendiendo el dominio para poder detener el envío de correos spam. Para solucionar el problema debemos asegurarnos corregir la vulnerabilidad y asegurarnos de que no quede ninguno email ni archivo que haya sigo utilizado para este fin malicioso, de lo contrario podremos volver a tener problemas, además es importante anular el puerto 25 muy utilizado para el envío de SPAM y ataques por mail.