Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico Matemáticas

Diseño Orientado a Objetos

Riesgos y Vulnerabilidades HTML y JavaScript

Profesor: Miguel Salazar
Alumno: Héctor Iván Arrieta Jaime
Matrícula: 1604738

Riesgos De JavaScript

Velocidad

Temas relacionados con la velocidad y la velocidad han estado plagando JavaScript desde sus inicios. La situación ha mejorado significativamente, pero la velocidad sigue siendo un problema grave para ciertos dominios y plataformas. Esto es particularmente cierto para los juegos. Su juego nuevo y trascendental puede trabajar maravillosamente en su PC de escritorio de doble núcleo, pero intente cargar en tu iPhone o dispositivo Android. ¿Aviso cualquier problema? Es probable que las animaciones que trabajaste tan duro son muy por debajo de los 30 frames por segundo te necesitan tener una experiencia suave para sus usuarios.

Diferencias de motor

Hay un motor de JavaScript. Google, Apple y otras organizaciones tienen sus motores preferidos. Son similares, pero no idénticas, y puede haber diferencias de rendimiento. Esto es especialmente sensible en dispositivos móviles donde Apple y Google están enzarzados en una lucha para producir el motor más rápido y menos intensivo de la batería.

Plagio

Los usuarios pueden acceder el código fuente de los navegadores más comunes simplemente haciendo clic en el botón "Ver código fuente". Los visitantes del sitio pueden, sin su conocimiento, copia su código y pasar como propios. Es poco lo que se puede hacer para combatir esto que no sean de ofuscar el código, o intencionalmente escribir el código de una manera que es difícil de leer y entender. Por supuesto, eso no impide a nadie de robar su código por mayor, pero puede disuadir a alguien que quiere modificar el código. Cabe señalar que este problema no existe cuando se trabaja con JavaScript embebido en dispositivos móviles.

Seguridad

Seguridad sigue siendo un problema con JavaScript, aunque la situación ha mejorado desde el lenguaje temprano días. Algunos de los problemas de seguridad más comunes relacionados con la lengua caen bajo la amplia categoría de "vulnerabilidades cross-site." Esto es cuando un atacante es capaz de obtener una página Web confiable, como un sitio de banca en línea, para incluir un script malintencionado con sus propios scripts benignos; el script malicioso generalmente registrará su credencial

de registro y enviarla al atacante para ser utilizado en un momento posterior.

Riesgos en HTML

Gran parte de los problemas de seguridad en las aplicaciones web son causados por la falta de seguimiento en dos rubros muy importantes de los que depende cualquier aplicación, las entradas y salidas del sistema. es importante considerar la exposición accidental de datos que pueden ser empleados en un posible ataque sobre el sistema. Los mensajes de error enviados por el servidor, que suelen ser de gran utilidad durante el proceso de desarrollo de la aplicación, pueden ser empleados maliciosamente cuando siguen apareciendo en un entorno de producción, por lo que es necesario deshabilitar todos estos mensajes y editar algunos otros (como los que se envían cuando el servidor no encuentra algún archivo en particular) los cuales también pueden ser utilizados por los atacantes para obtener información sobre nuestro sistema.

Vulnerabilidades de JavaScript

Los hackers utilizan herramientas de explotación de JavaScript para atacar sitios web, organizaciones e individuos. Las vulnerabilidades de JavaScript pueden ser tanto problemas del cliente como pesadillas empresariales, ya que los hackers pueden robar datos del lado del servidor e infectar a los usuarios con malware. Ataques Cross-Site Scripting (XSS) El uso más común de la vulnerabilidad de las aplicaciones en las aplicaciones web es el cross-site scripting (XSS). A través de la manipulación de scripts JavaScript y HTML, los hackers ejecutan scripts maliciosos (también conocidos como "maliciosos payloads") utilizando un navegador web de usuario desprevenido que puede resultar en el script que se incrusta en la página web que están visitando. Cada vez que el usuario visita la página web o se realiza una acción predefinida, se desencadena y ejecuta la secuencia de comandos malintencionada. Los ataques XSS tienen el potencial de causar serias amenazas a empresas y cuentas empresariales que pueden resultar en robo de identidad y robo de datos. Al ejecutar ataques XSS, los hackers pueden inyectarse y propagar virus y gusanos en toda la red de la compañía, acceder a los datos del portapapeles e historias de

navegación e incluso obtener el control remoto del navegador, lo que les ayuda a buscar e identificar otras posibles vulnerabilidades que pueden utilizarse para más información. XSS ataques. Debido a la presencia de JavaScript en casi todos los elementos de la experiencia de navegación web, las aplicaciones escritas con JavaScript son las víctimas más comunes de los ataques XSS. Cross-Site Solicitud de Falsificación Cross-Site Request Forgery (CSRF) es una forma de exploit que se produce cuando los comandos no autorizados, que normalmente se rechazan, lo que resulta en el sitio web que se cree que el usuario malicioso es un usuario autorizado a través de una autorización falsificada. Tras una explotación exitosa de esta vulnerabilidad, el hacker puede acceder a las funciones de la aplicación web que normalmente se negaría. Los riesgos asociados a los ataques CSRF incluyen la suplantación de identidad y la identidad, la modificación de los datos de la aplicación utilizando las credenciales y permisos de la víctima, el lanzamiento de ataques organizados contra todos los usuarios de la aplicación, la explotación de routers DSL vulnerables y más. CSRF es a menudo pronunciado "mar-surf" y es alternativamente abreviado como XSRF.

Vulnerabilidades en HTML

La inyección de HTML es un ataque similar al de Cross-site Scripting (XSS). Mientras que en la vulnerabilidad XSS el atacante puede inyectar y ejecutar código JavaScript, el ataque de inyección HTML sólo permite la inyección de ciertas etiquetas HTML. Cuando una aplicación no maneja correctamente los datos suministrados por el usuario, un atacante puede proporcionar código HTML válido, normalmente a través de un valor de parámetro, e inyectar su propio contenido en la página. Este ataque suele utilizarse en combinación con alguna forma de ingeniería social, ya que el ataque está explotando una vulnerabilidad basada en código y la confianza de un usuario.