UANL
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FCFM
FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

Materia: Diseño Orientado a Objetos.

Tema: Riesgos de seguridad en JS.

Alumno: Arodi Fuentes Montelongo.

Matricula: 1837486.

Carrera: LSTI

Grupo: 006.

Profesor(a): Lic. Miguel Salazar Santillán.

San Nicolás de los Garza, N.L.

Introduccion.

El internet nació como una evolución de las redes de comunicación entre los dispositivos fijos permitiendo la comunicación desde diferentes lugares del mundo a través de la computadora. En los noventa se introdujo la World Wide Web, que se hizo común y con ello se crearon las bases de protocolo de transmisión HTTP, el lenguaje de documentos HTML y el concepto de los URL. Internet tuvo un grandísimo impacto en el mundo laboral, por lo que fue necesario mejorar las páginas estáticas desarrolladas en HTML y se comenzó a crear el HTML dinámico capaz de actualizar los formularios y las bases de datos, en el momento de enviar una petición. Conforme a estos cambios, actualmente, Internet ha cambiado la vida de las personas; con el uso de un ordenador y acceso a la red. Es posible hacer casi de todo, que aún un usuario se le pueda ocurrir. Se desarrollan sistemas Web para la mayoría de las actividades que se realicen. Estos sistemas se han convertido en las herramientas más utilizadas para el desarrollo económico, educativo y social. Además de esto, los sistemas Web son necesarios en el desarrollo laboral tanto en el ámbito empresarial privado como en administraciones públicas. Por este motivo es necesario definir mecanismos de protección, con la finalidad de proteger los datos de cada individuo y se mantengan tanto íntegros, como disponibles con los niveles de confidencialidad adecuada.

Java Script es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas, se ejecuta en el ordenador del usuario y actualmente también se ejecuta en el servidor. Sin embargo, existen diversos riesgos asociados a este, como lo es el plagio ya que usuarios pueden acceder al código fuente de la mayoría de los navegadores comunes, por lo que cualquiera podría copiar su código y hacerlo pasar por propio.

Ataques.

Los ataques URL de tipo semántico involucran a un usuario modificando la URL para descubrir acciones a realizar que originalmente no se plantean para ser manejadas por el servidor. Enviar ciertos parámetros con el método GET, se agregan directamente en la URL, lo que produce que los atacantes puedan utilizarlos, ya que son fáciles de capturar y modificar.

Los Ataques de Cross-Site Scripting son un tipo de vulnerabilidad de seguridad informática encontrada en las aplicaciones web que permiten la inyección de código por usuarios maliciosos en páginas web. Los atacantes de valen de código HTML y de scripts ejecutados en el cliente. Este se subdivide en tres tipos, ataque basado en el DOM o local, en donde si un código de JavaScript accede a una URL como un parámetro de una petición al servidor y utiliza esta información para escribir HTML en la misma página sin ser codificada empleando entidades HTML.

El segundo de estos tipos es el ataque no persistente, en el cual si los datos no validados por el usuario son incluidos en la página resultante sin codificación HTML, se le permite al cliente inyectar código en la página dinámica. A diferencia del ataque persistente donde la información proporcionada por el usuario es almacenada en la base de datos, en el sistema de archivos o algún otro lugar, después es mostrada a otros usuarios que visiten la página.

Los Ataques de Cross-Site Request Forgery permiten al atacante enviar peticiones HTTP a voluntad desde la máquina de la víctima. Difícilmente es posible determinar cuándo una petición HTML se ha originado por un ataque de este tipo. Un recurso que se utiliza comúnmente para realizar este tipo de ataques suelen tener incluida la petición dentro de una imagen.

Además de este tipo de riesgos existen las peticiones HTTP falsificadas empleando herramientas especiales para este propósito. Se emplean herramientas de línea de comandos o plugins agregados a los navegadores, con estos se pone a la escucha de los servicios web que típicamente se conectan a través del puerto 80.

¿Como protegerse?

Como protección frente a los diversos riesgos y ataques se recomendaría no enviar datos sensibles a través del cliente y si no queda más remedio, cifrarlos o firmarlos, además de validar en el servidor todos los datos procedentes del cliente. En el caso de los ataques cross-site scripting es necesario validad la entrada del usuario aplicando restricciones de longitud, conjunto de caracteres, expresiones regulares, etc. Así como validar salida remplazando caracteres reservados de HTML por referencias a entidades, eliminando puntos peligrosos de inserción, donde el usuario pueda editar HTML, limitar las macas que pueda utilizar o utilizar lenguajes de marcas alternativos.

Conclusión

Los riesgos en las tecnologías siempre van a existir y mientras sigan evolucionando las tecnologías, de igual manera lo harán los ataques y riesgos que puedan vulnerar algun sistema, ningún sistema será 100% seguro, la seguridad es lo primordial en los sitios y aplicaciones web, ya que sin ella, serán muy vulnerables a diferentes tipos de ataques u robos de información que pudiese ser muy valiosa.