auravant

Requisitos de Seguridad para Extensiones

DOC_TECH_201

Clasificación del documento: USO PÚBLICO

Tabla de contenido

Objetivo 2
Alcance 3
Referencias 3
Contenido 3
Registro de actualizaciones 6

Objetivo

El siguiente documento expresa los requisitos de seguridad **mínimos** que toda Extensión debe cumplir y que el equipo de seguridad analiza.

Alcance

Todos los desarrollos de Extensiones propios y de terceros.

Referencias

ISO/IEC 27001, cláusula 7.5 "Información documentada".

Contenido

Librerías y Frameworks actualizados

Todas las librerías y frameworks que el desarrollador utilice de tercero, deberá mantenerlas actualizadas a la última versión estable y disponible. Evitando de esta manera hacer uso de versiones sin soporte, con funcionalidades obsoletas y con vulnerabilidades reportadas.

Revisión y Depuración del Código Fuente

Evitar la utilización de funciones de depuración y debugger en las librerías que se utilizan, por ejemplo console.log(), console.info() o debugger;

No se recomienda el exceso de documentación y explicación de cada función dentro de la extensión.

Evitar las llamadas a direcciones de IP

Cuando una extensión requiera de información externa o consultas a servicios de terceros para realizar un cálculo o procesamiento, evitar realizar llamadas a direcciones de IP públicas o privadas, y hacerlo por medio de sus correspondientes APIs y web services, invocando los métodos y parámetros que correspondan.

Evitar la exposición de información sensible

Dentro del código fuente de la extensión, evitar divulgar información sensible, tales como números telefónicos, direcciones, credenciales, Token de acceso a un servicio, usuarios, contraseñas, archivos de configuraciones, entre otros.

Evitar añadir código ofuscado

Evitar ofuscar código de su aplicación, debido a que es un mecanismo que habitualmente se utiliza para ocultar código malicioso tales como virus, troyanos, rootkit, etc. Alentamos la creación de un código fuente limpio, donde sus funciones sean simples de entender.

Evitar añadir archivos que no son necesarios

Utilizar siempre la implementación del principio de Mínimo Punto de Exposición, para la extensión, por lo cual:

- Evitar publicar archivos y directorios ocultos.
- Evitar publicar imágenes que la extensión no utiliza.
- Evitar publicar documentación y archivos tales como README, Install, etc.
- Evitar publicar directorios con las diferentes versiones de una misma librería.
- Evitar publicar archivos de backup o resguardos.

Validación de Entradas y Salidas

Realizar la validación de todos los campos que ingresan datos a la base de datos, sobre todo si la extensión hace uso de formularios.

Realizar la validación y sanitización de los datos de ingreso, antes de realizar algún tipo de cálculo y procesamiento con datos obtenidos de una aplicación o almacenados.

Realizar la validación y sanitización de los datos de salida, antes de presentar, renderizar o mostrar la información, como resultado de un procesamiento o una consulta.

Revisión de vulnerabilidades más frecuentes

Siempre que sea posible, realizar las pruebas necesarias respecto a la existencia de las amenazas más frecuentes y mitigarlas, como son los casos de:

Cross-Site Scripting (XSS)

XSS es un término que se usa para describir una clase de ataque que permite al atacante inyectar código script malicioso del lado del cliente y de esta forma poder extraer cookie, sesión, o incluso la modificación del Document Object Model (DOM) de la web.
Más información: https://owasp.org/www-community/attacks/xss/

Inyección SQL

Las vulnerabilidades de inyección SQL permiten la manipulación de consultas con el objetivo de extraer o tomar control de los datos almacenados. Un ataque de Inyección SQL con éxito, podría falsificar identidades, acceder a todos los datos del servidor, destruir o modificar los datos.
Más información: https://owasp.org/www-community/attacks/SQL_Injection

Inyección de Comandos

La inyección de comandos es un ataque en el que el objetivo es la ejecución de comandos arbitrarios en el Sistema Operativo host a través de una aplicación vulnerable.
Más información: https://owasp.org/www-community/attacks/Command_Injection

Restricciones en los cambios a los paquetes de Software

Se debe evitar las modificaciones a los paquetes de software creado por los equipos de desarrollo de Auravant o aquellos que se encuentran con licencias libres u Open Source, entendiendo que se permiten las modificaciones únicamente cuando sean estrictamente necesarias, llevando un riguroso control sobre los cambios realizados, teniendo en cuenta:

- El riesgo de que los procesos de integridad y de control incorporados se vean comprometidos;
- Si es necesario obtener el consentimiento del propietario;
- La posibilidad de obtener los cambios requeridos del vendedor como un programa estándar de actualizaciones;
- El impacto, si la organización se hace responsable del mantenimiento futuro del software como resultado de los cambios.

Si los cambios son necesarios, el software original se deberá conservar y los cambios se aplicarán a una copia claramente identificada.

Se debería implementar un proceso de gestión de las actualizaciones del software para asegurarse de que los parches más actualizados aprobados y las actualizaciones de las aplicaciones están instalados en todo el software autorizado (véase el numeral 12.6).

Todos los cambios se prueban y documentan en su totalidad de manera que se puedan volver a aplicar, si es necesario, para mejoras futuras del software.

Registro de actualizaciones

Versión	Revisión Dd/mm/yyyy	Descripción del cambio	Responsable
1	29/06/2023	Creación del presente documento técnico	**Information Security Analyst** *Daniel Maldonado*
1	03/08/2023	Agregado del apartado “Restricciones en Restricciones en los cambios a los paquetes de Software”	**CDO - CISO** *Claudio Caracciolo*