TLP:CLEAR

Alerta de seguridad desde el diseño

Eliminación de vulnerabilidades de inyección de SQL en software

Los agentes cibernéticos malintencionados utilizan vulnerabilidades de inyección de SQL para comprometer los sistemas

La industria del software ha sabido eliminar estos defectos a gran escala durante décadas. MySQL introdujo las declaraciones preparadas, que pueden eliminar las vulnerabilidades de inyección de SQL, en 2004.[2]

Las vulnerabilidades de inyección de SQL (o SQLi) siguen siendo una clase persistente de defecto en los productos de software comerciales.[1] A pesar del conocimiento y la documentación generalizados de las vulnerabilidades de SQLi durante las últimas dos décadas, junto con la disponibilidad de medidas de mitigación efectivas, los fabricantes de software han seguido desarrollando productos con este defecto, lo que pone en riesgo a muchos clientes.[2] La CISA y la FBI están publicando esta Alerta de seguridad desde el diseño en respuesta a una reciente campaña de agentes de amenazas maliciosas muy publicitada que explotó los defectos de SQLi en una aplicación de transferencia administrada de archivos para atacar y comprometer a los usuarios de esa aplicación, lo que afectó a miles de organizaciones. La CISA y la FBI instan a los altos ejecutivos de los fabricantes de tecnología a realizar una revisión formal de su código para determinar su susceptibilidad a los ataques SQLi y animan a todos los clientes de tecnología a preguntar a sus proveedores si han realizado dicha revisión. Si descubren que su código tiene vulnerabilidades, los altos ejecutivos deben asegurarse de que los desarrolladores de software de sus organizaciones comiencen de inmediato a implementar medidas de mitigación para eliminar este tipo de defectos de todos los productos de software actuales y futuros.[3] Incorporar seguridad en los productos desde el principio puede eliminar las vulnerabilidades de SQLi.

Lecciones por aprender de Seguridad desde el diseño

Seguro desde el diseño significa que los fabricantes diseñan y construyen sus productos de una manera que los proteja de forma razonable contra agentes cibernéticos maliciosos que explotan con éxito los defectos del producto. Incorporar esta medida de mitigación desde el principio (comenzando en la fase de diseño y continuando durante el desarrollo, el lanzamiento y las actualizaciones) reduce la carga de la ciberseguridad para los clientes y el riesgo para el público. Vulnerabilidades como SQLi han sido consideradas por otros como vulnerabilidades "imperdonables" desde, al menos, 2007. A pesar de este hallazgo, las vulnerabilidades de SQL (como CWE-89) siguen siendo una clase predominante. Por ejemplo, CWE-89 se encuentra entre las 25 principales debilidades de software más peligrosas y persistentes en 2023.[4]

¿Qué son las vulnerabilidades de inyección de SQL?

Las vulnerabilidades de inyección de SQL implican la inserción de una entrada proporcionada por el usuario directamente en un comando SQL, lo que permite a los agentes de amenazas ejecutar consultas arbitrarias (consulte CWE-89: Neutralización inadecuada de elementos especiales utilizados en un comando SQL ["SQL Injection"]). Las vulnerabilidades de SQLi son causadas por la falta de atención de los desarrolladores de software a las prácticas recomendadas de seguridad, lo que resulta en la mezcla de consultas de bases de datos y datos proporcionados por el usuario. El impacto de una explotación exitosa de SQLi puede ser devastador, ya que pone en peligro la confidencialidad, integridad y disponibilidad de una base de datos y su información. Específicamente, las vulnerabilidades de SQLi pueden permitir que agentes cibernéticos maliciosos roben información confidencial, alteren, eliminen o hagan que la información no esté disponible en una base de datos. Las inyecciones de SQL tienen éxito porque **los desarrolladores de software no tratan el contenido proporcionado por el usuario como potencialmente malicioso**.

---

[1] OWASP Foundation. "SQL Injection", sin fecha, https://owasp.org/www-community/attacks/SQL_Injection.
[2] En 2004, MySQL introdujo una técnica llamada "declaraciones preparadas" que separan los comandos de base de datos de los datos no confiables, eliminando así las vulnerabilidades de inyección de SQL. Consultar: MySQL. "Changes in release 4.1.3 (28 Jun 2004: Beta)". 28 de junio de 2004. https://web.archive.org/web/20060422175612/http://dev.mysql.com/doc/refman/4.1/en/news-4-1-3.html.
[3] OWASP Foundation. "SQL Injection Prevention Cheat Sheet", sin fecha, https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html.
[4] "2023 CWE Top 25 Most Dangerous and Stubborn Software Weaknesses in the CWE Top 25." MITRE's CWE Top 25, 2023. https://cwe.mitre.org/top25/archive/2023/2023_top25_list.html,https://cwe.mitre.org/top25/archive/2023/2023_stubborn_weaknesses.html

¿Cómo pueden los fabricantes de software prevenir las inyecciones de SQL?

Durante el diseño y desarrollo de un producto de software, los desarrolladores deben utilizar consultas parametrizadas con declaraciones preparadas para separar el código SQL de los datos proporcionados por el usuario para evitar esta clase de vulnerabilidad. Esta separación garantiza que el sistema trate la entrada del usuario como datos y no como código ejecutable, eliminando así el riesgo de que la entrada malintencionada del usuario se interprete como una declaración SQL. Los fabricantes de software deberían eliminar sistemáticamente las vulnerabilidades de SQLi mediante la aplicación del uso de consultas parametrizadas en sus aplicaciones. **Nota:** Algunos desarrolladores intentan utilizar técnicas de desinfección de entradas para evitar vulnerabilidades de SQLi. Si bien la desinfección de entradas puede prevenir algunos ataques, esas técnicas son frágiles, difíciles de aplicar a gran escala y, con frecuencia, se pueden eludir. Por lo tanto, las consultas parametrizadas incorporan mejor un enfoque seguro desde el diseño.

La CISA y la FBI recomiendan que los fabricantes de software investiguen las causas y las soluciones ampliamente conocidas para esta vulnerabilidad predecible y explotada de forma habitual. Además, la CISA y la FBI animan a los fabricantes a revisar los siguientes tres principios de la guía conjunta, Cambiar el equilibrio del riesgo de la ciberseguridad: principios y enfoques para un software seguro desde el diseño.

Principio 1: Asumir los resultados del cliente en materia de seguridad

Hay áreas de seguridad clave en las que los fabricantes deberían invertir para proteger a sus clientes y al público. Esto incluye proporcionar bloques de construcción seguros para sus desarrolladores de software para garantizar que un solo error de desarrollador no comprometa los datos de millones de usuarios. Los fabricantes de software deberían adoptar el uso de declaraciones preparadas con **consultas parametrizadas** como una práctica estándar en el desarrollo de software. Esto debería implementarse en sus entornos de desarrollo a través de, por ejemplo, bibliotecas de desarrollo que hagan que la ruta segura sea la predeterminada para los desarrolladores y verificaciones en el momento de las solicitudes de extracción.

Además, los altos ejecutivos de los fabricantes de software deben asumir la responsabilidad de la seguridad de sus clientes, empezando por realizar revisiones formales de su código para determinar su susceptibilidad a los ataques. Una simple revisión del código revelaría la prevalencia de esta conocida clase de vulnerabilidad, con medidas de mitigación claras y efectivas disponibles de forma fácil y pública. Los fabricantes y desarrolladores deberían tomar responsabilidad de proteger sus productos mediante el uso predeterminado de consultas parametrizadas, eliminando así toda una clase de amenazas.

Principio 2: Adoptar métodos radicales de transparencia y rendición de cuentas

Los fabricantes deben actuar con transparencia al revelar las vulnerabilidades de sus productos. Para tal fin, los fabricantes deberían rastrear las clases de vulnerabilidad asociadas con su software y revelarlas a sus clientes a través del programa de CVE. Los fabricantes deben asegurarse de que sus registros de CVE sean correctos y completos. Es especialmente importante que los fabricantes proporcionen una CWE precisa para que la industria pueda rastrear clases de defectos de software, no solo CVE individuales, y los clientes puedan comprender áreas en las que las prácticas de desarrollo de un proveedor determinado pueden requerir mejoras.[5] También deben identificar y documentar las causas fundamentales de esas vulnerabilidades y declarar como objetivo comercial trabajar para eliminar clases enteras de vulnerabilidad.

Principio 3: Construir estructura organizativa y liderazgo para lograr estos objetivos

Así como los ejecutivos de fabricación de software y hardware se preocupan por el costo, las características y la experiencia del cliente, también deberían priorizar la seguridad de sus productos. Los líderes deben considerar el panorama completo: que los clientes, nuestra economía y nuestra seguridad nacional actualmente soportan el peso de las decisiones comerciales de no incorporar seguridad a sus productos, como lo refleja con claridad la campaña de agentes de amenazas descrita anteriormente en esta Alerta. Además, orientar la empresa hacia el desarrollo de software seguro desde el diseño a menudo reduce los costos financieros y de productividad, así como la complejidad. Los líderes deben realizar las inversiones apropiadas y desarrollar las estructuras de incentivos correctas que promuevan la seguridad como un objetivo comercial declarado.

[5] La clasificación de enumeración de debilidades comunes (CWE, por sus siglas en inglés) identifica clases de debilidades de software y hardware (incluidas vulnerabilidades y defectos); la clasificación de vulnerabilidades y exposiciones comunes (CVE, por sus siglas en inglés) identifica y etiqueta vulnerabilidades únicas en productos de software y hardware específicos.

Los líderes deberían destacar la importancia de erradicar clases enteras de vulnerabilidades en lugar de abordarlas caso por caso. Además, los líderes deben establecer estructuras organizativas que prioricen medidas proactivas, como la adopción de prácticas de codificación segura como consultas parametrizadas, para crear una seguridad duradera y reducir la dependencia de respuestas reactivas. Los altos ejecutivos también deben asegurarse de que su organización realice revisiones para detectar vulnerabilidades comunes y conocidas, como SQLi, para determinar su susceptibilidad e implementar las medidas de mitigación existentes, efectivas y documentadas. Estas revisiones deben realizarse de forma continua para erradicar las clases de vulnerabilidad, ya que algunas pueden cambiar o desarrollarse con el tiempo.

Punto de acción para los fabricantes de software

Si bien esta Alerta de seguridad desde el diseño se centra en enfoques para mitigar las inyecciones de SQL como una clase de defecto, es solo una parte de un conjunto más completo de prácticas de seguridad desde el diseño. Para proteger a sus clientes de una amplia gama de actividades cibernéticas maliciosas, los fabricantes deben implementar por completo los principios y las prácticas abordados en esta alerta mediante la revisión de Cambiar el equilibrio del riesgo de la ciberseguridad: principios y enfoques para un software seguro desde el diseño. Además, la CISA y la FBI instan a los fabricantes a publicar su propia hoja de ruta de seguridad desde el diseño para demostrar que no están simplemente implementando controles tácticos, sino que están repensando de forma estratégica su responsabilidad de mantener seguros a los clientes.