

# ¿Qué es un ataque de inyección SQL y cómo protegerte de él?

En un servidor web, la seguridad es muy importante. Uno de los peligros más críticos que afectan a los servidores y a las aplicaciones web es el llamado "ataque de inyección SQL", una táctica utilizada por hackers para comprometer sistemas y acceder a información confidencial.

## ¿Qué es la inyección SQL?

Imagina que una aplicación web es como una puerta de entrada a una base de datos donde se almacena información valiosa como datos de clientes, facturas, etc. Normalmente, esta puerta está protegida por una serie de cerraduras, pero si un hacker encuentra una forma de "inyectar" código malicioso a través de un formulario de búsqueda o un campo de inicio de sesión, puede "romper" esta cerradura, abrir la puerta y acceder a datos sensibles.

## Cómo Funciona

El hacker identifica un campo de entrada en la aplicación web, como un cuadro de búsqueda. Luego, introduce datos manipulados, como un fragmento de código SQL, en ese campo. Cuando la aplicación procesa esta entrada, en lugar de simplemente buscar lo que el usuario solicitó, o devolver un código de error, puede ejecutar también un código malicioso, que puede dar al hacker acceso no autorizado a la base de datos.

### Ejemplos

Supongamos que tienes un sitio web WordPress con un formulario de búsqueda que permite a los usuarios buscar publicaciones por título. El campo de búsqueda envía una consulta SQL a la base de datos para recuperar las publicaciones que coincidan con el término de búsqueda.

Ahora, un atacante malintencionado podría intentar explotar una posible vulnerabilidad de inyección SQL en este formulario de búsqueda. Por ejemplo, el atacante podría ingresar lo siguiente en el campo de búsqueda:

```
' ; DROP TABLE wp_users; --
```

Cuando se envía esta cadena como consulta de búsqueda, la consulta SQL resultante podría ser algo así:

```
SELECT * FROM wp_posts WHERE post_title LIKE '%'; DROP TABLE wp_users; -- %'
```

Aquí está el análisis de cómo funciona esta cadena:

- `'`: Cierra la cadena de búsqueda actual.
- `DROP TABLE wp_users;`: Agrega una nueva consulta SQL maliciosa para eliminar la tabla de usuarios de la base de datos.
- `--`: Comentario en SQL para ignorar el resto de la consulta original y cualquier consulta adicional.

Si la aplicación WordPress no está protegida adecuadamente contra la inyección SQL, ejecutaría esta consulta sin validarla correctamente. Como resultado, la tabla de usuarios `wp_users` se eliminaría de la base de datos, lo que eliminaría todos los usuarios del sitio, esto como puedes suponer sería catastrófico en términos de seguridad y funcionalidad del sitio, sobre todo si en es Wordpress tenemos por ejemplo un WooCommerce con cientos de clientes de nuestro e-commerce almacenado en la tabla de usuarios.

## Los Riesgos

Un ataque de inyección SQL que tenga acceso a los datos de nuestra base de datos, puede tener consecuencias graves. Los hackers pueden obtener información delicada, como nombres de usuario, contraseñas, números de tarjetas de crédito o datos personales. Además, podrían manipular o eliminar datos, causando estragos en la integridad de los datos almacenados en la base de datos de la aplicación con los peligros que ello conlleva amén de las sanciones económicas que pueden derivarse del acceso no autorizado a datos confidenciales.

## Cómo Protegerte

Una de las mejores formas, es establecer una serie de criterios de buenas prácticas para el desarrollo de aplicaciones web para protegerse contra los ataques de inyección SQL:

1. **Validación de Entradas:** Cuando se desarrolla cualquier aplicación web, los programadores deben implementar medidas para asegurarse de que los datos que los usuarios introducen sean seguros y estén correctamente formateados y validados antes de procesarlos.
2. **Consultas Parametrizadas:** En lugar de concatenar cadenas de texto directamente en las consultas SQL, los programadores pueden utilizar consultas parametrizadas, que separan los datos de las instrucciones SQL, haciendo más difícil para los hackers inyectar código malicioso.

3. **Actualizaciones y Parches:** Mantén tus aplicaciones web y sistemas actualizados con los últimos parches de seguridad para mitigar las vulnerabilidades conocidas.
4. **Auditorías de Seguridad:** Realiza auditorías de seguridad regulares para identificar y corregir posibles vulnerabilidades antes de que los hackers las exploren.

## ¿Y si no puedo? WAF es tu amigo

En muchos casos lo que tenemos es una aplicación desarrollada por terceros como pueden ser Wordpress, Joomla, Prestashop, Moodle, etc.

En estos casos, debido a que no podemos modificar estos desarrollos, no hay otra forma para realizar esto que es contar con un WAF que permita realizar el filtrado de estos tipos de ataques en la mayor medida posible.

## Protección de ataques de inyección SQL mediante WAF

1. Un WAF puede inspeccionar el tráfico entrante a tu sitio o a tu aplicación web y detectar patrones de entrada que podrían indicar un intento de inyección SQL. Puede examinar los parámetros de las solicitudes HTTP en busca de cadenas de texto sospechosas que coincidan con los patrones utilizados en ataques de inyección SQL.
2. El WAF puede aplicar reglas específicas para validar y limpiar los datos de entrada antes de pasarlos a tu aplicación. Esto incluye la eliminación o el escape de caracteres especiales que podrían ser utilizados en un ataque de inyección SQL. Por ejemplo, puede bloquear consultas SQL que contengan palabras clave como "SELECT" o "DROP TABLE", o puede escapar automáticamente comillas y otros caracteres que podrían ser parte de un ataque.
3. Los WAFs a menudo incluyen una base de datos de firmas y reglas específicas, que identifican patrones de tráfico malicioso conocidos, incluidos los ataques de inyección SQL. Estas reglas pueden ser actualizadas regularmente para mantenerse al día con las últimas amenazas.
4. Un WAF puede implementar políticas de control de acceso que limiten el acceso a ciertos recursos o funcionalidades de tu aplicación o sitio web, lo que puede ayudar a prevenir ataques de inyección SQL al restringir el acceso a áreas sensibles de tu aplicación.
5. Algunos WAFs pueden realizar un seguimiento del comportamiento normal de tu aplicación web y detectar desviaciones significativas que podrían indicar un ataque de inyección SQL u otras actividades maliciosas.

En resumen, los ataques de inyección SQL representan una amenaza seria para la seguridad de tu web y de los datos que se almacenan en ella, pero con las medidas adecuadas, puedes proteger tus aplicaciones web y datos sensibles contra estos ataques maliciosos.

---

Revision #2

Created 1 February 2024 17:00:05 by etaboda

Updated 1 February 2024 17:21:56 by etaboda