Caso de Prueba de Seguridad: Prevención de Inyección de SQL en un Sistema de Comercio Electrónico

Objetivo:

Evaluar la resistencia del sistema de comercio electrónico ante **intentos de inyección de SQL, una vulnerabilidad** común que podría permitir a un atacante manipular la base de datos subyacente.

Pasos:

Descripción del Caso de Prueba:

- El objetivo de este caso de prueba es verificar que el sistema de comercio electrónico es resistente a ataques de inyección de SQL en el formulario de búsqueda.

Tareas a Realizar:

- Intentar ingresar consultas SQL maliciosas en el campo de búsqueda del sitio web para verificar si el sistema filtra y sanitiza correctamente las entradas.

Ambiente de Prueba:

- Este caso de prueba se llevará a cabo en un entorno de prueba que replica de cerca el entorno de producción del sistema de comercio electrónico.

Datos de Prueba:

- Utilizar consultas SQL maliciosas, como intentos de inyección de SQL, para evaluar la respuesta del sistema.

Acciones Específicas:

- Intentar ingresar consultas SQL en el campo de búsqueda, incluyendo caracteres especiales y declaraciones que podrían comprometer la integridad de la base de datos.

Caso de prueba ejemplo:

```
<form action="/buscar" method="get">

  <input type="text" name="producto" placeholder="Buscar
productos">

  <input type="submit" value="Buscar">

</form>
```

Cuando el usuario realiza una búsqueda, el sistema utiliza el valor proporcionado en el parámetro "producto" para realizar la consulta en la base de datos.

Consulta SQL (Pseudocódigo):

SELECT * FROM productos WHERE nombre = 'valor_del_campo_producto';

Cuando el usuario realiza una búsqueda, el sistema utiliza el valor proporcionado en el parámetro "producto" para realizar la consulta en la base de datos.

Consulta SQL (Pseudocódigo):

' OR '1'='1'; --

Después de la inyección, la consulta SQL resultante podría ser algo así:

SELECT * FROM productos WHERE nombre = '' OR '1'='1'; -- ';

En este caso, la parte `' OR '1'='1'; -- '` se ha insertado en el campo de búsqueda. La cláusula `OR '1'='1'` siempre es verdadera, lo que significa que la condición de búsqueda será verdadera para todos los productos, devolviendo potencialmente todos los registros de la tabla.

Este es solo un ejemplo simple de inyección de SQL. En la realidad, los atacantes pueden utilizar técnicas más avanzadas y sutiles**. La prevención de la inyección de SQL implica el uso de consultas parametrizadas o la validación adecuada de las entradas del usuario para evitar la ejecución no autorizada de código SQL.**

Resultado Esperado:

- El sistema debe filtrar y sanitizar adecuadamente las entradas del campo de búsqueda, evitando que las consultas SQL maliciosas tengan un impacto en la base de datos.

Criterios de Éxito:

- La aplicación web no debe mostrar resultados inesperados o mensajes de error que indiquen una posible inyección de SQL.
- El sistema debe manejar adecuadamente las entradas maliciosas y responder de manera controlada sin afectar la integridad de la base de datos.

Condiciones Previas:

- El sistema de comercio electrónico está en un estado funcional y se ha asegurado de que todas las entradas del usuario se validen y filtren correctamente.

Finalización de la Prueba:

- Se debe documentar si el sistema ha superado o no la prueba de inyección de SQL, y proporcionar recomendaciones para mejorar la seguridad si es necesario.

Notas Adicionales:

- Durante la prueba, se pueden intentar diferentes variantes de inyección de SQL, como UNION-based SQL injection, Blind SQL injection, o Time-based Blind SQL injection, para evaluar la robustez del sistema.
- El objetivo final es asegurarse de que las consultas SQL ingresadas maliciosamente no tengan un impacto negativo en la base de datos y que el sistema responda de manera segura y controlada.

Este caso de prueba se centra en evaluar la seguridad del sistema frente a una vulnerabilidad específica (inyección de SQL) que es común en aplicaciones web. Es importante realizar pruebas de seguridad regulares para identificar y abordar posibles riesgos de seguridad en un sistema de comercio electrónico