Informe de Análisis de Ciberseguridad - Inyección SQL

Proyecto: Análisis de Vulnerabilidad en DVWA (Damn Vulnerable Web Application)

Realizado por: Santiago Rivero

Introducción

Este informe documenta la explotación exitosa de una vulnerabilidad de **Inyección SQL** en la aplicación **Damn Vulnerable Web Application (DVWA)**. El objetivo de este ejercicio es demostrar la capacidad de un atacante para comprometer la base de datos de una aplicación web que no aplica una sanitización de datos adecuada, siguiendo los procedimientos de documentación de incidentes bajo el estándar **ISO 27001**.

Descripción del Incidente y Explotación

1. Naturaleza y Localización de la Falla

Se identificó una falla de seguridad de **Inyección SQL** en el campo de entrada "User ID" del módulo "SQL Injection".

- **Causa:** El código de la aplicación no diferencia entre los **datos del usuario** y el **código SQL ejecutable**. Los caracteres introducidos por el usuario son concatenados directamente en la consulta de la base de datos.

2. Método de Explotación (Payload)

Se ejecutó un ataque de **Bypass de Lógica Booleana** (lógica de autenticación) en el servidor.

Parámetro	Valor Inyectado
User ID	`1' OR '1'='1`

Análisis de la Consulta Inyectada:

1. **Consulta Original (Estimada):** `SELECT * FROM users WHERE id = '1' LIMIT 1;`
2. **Consulta Modificada:** La consulta se convierte en: `SELECT * FROM users WHERE id = '1' OR '1'='1' --' LIMIT 1;`
3. **Resultado:** La condición `'1'='1'` siempre es **verdadera**. Esto anula la condición original del ID, haciendo que la cláusula `WHERE` devuelva **todos los registros** de la base de datos, no solo el registro de un usuario específico.

3. Evidencia del Compromiso

La aplicación respondió exitosamente a la inyección, revelando la lista completa de usuarios de la base de datos: **admin, Gordon, Hack Me, Pablo Picasso, y Bob Smith**. Esto confirma la **fuga de información** por medio del **bypass de la lógica de autenticación y consulta**.

Impacto y Riesgos

La explotación de esta vulnerabilidad presenta un **riesgo significativo** para la confidencialidad, integridad y disponibilidad (C-I-D) de los activos.

- **Confidencialidad:** Se reveló información sensible de la base de datos (nombres de usuarios y estructura). En un escenario real, esto incluiría *hashes* de contraseñas, correos electrónicos y datos personales.
- **Integridad:** Un atacante con *payloads* más complejos podría modificar o eliminar datos críticos del sistema (p. ej., otorgar privilegios de administrador o borrar la tabla de usuarios).
- **Disponibilidad:** El servidor de la base de datos podría ser sobrecargado o corrompido, llevando a la denegación de servicio (DoS) de la aplicación web.

Recomendaciones para Mitigación

Para subsanar esta vulnerabilidad y fortalecer la postura de seguridad de la aplicación, se recomiendan las siguientes medidas correctivas y preventivas:

1. **Principio de Mínimo Privilegio:** La cuenta de base de datos que utiliza la aplicación web **no debe tener permisos excesivos** (como `DROP TABLE` o `DELETE ALL`). Debe limitarse estrictamente a las operaciones necesarias (p. ej., `SELECT` en la tabla de usuarios).

2. **Validación de Entradas (Whitelist):** Aplicar una **validación estricta del lado del servidor**. Si el campo espera un ID numérico, la aplicación debe rechazar cualquier carácter que no sea un número.

Conclusión

La identificación y explotación de la inyección SQL en DVWA demuestra que la **validación de entradas a nivel de código** es crítica para la seguridad de la aplicación. Implementar controles robustos en la capa de datos es esencial para proteger la integridad del sistema y **asegurar la continuidad del negocio**.