REPORTE DE INCIDENTE — INYECCIÓN SQL (DVWA)

Proyecto: Instalación y explotación de DVWA en máquina virtual
Fecha: 22 de octubre de 2025
Autor: Eric Grossermann

---

1. Introduccion

Instalé DVWA en una máquina virtual y realicé una prueba práctica de inyección SQL con la aplicación puesta en nivel de seguridad **Low**. Use un payload sencillo (`' OR '1'='1`) en un campo vulnerable y, al enviar, la aplicación devolvió todos los registros de la tabla de usuarios.

Esto demuestra que la entrada del usuario se está insertando directamente en la consulta SQL sin protección: hubo una inyección SQL exitosa.

---

2. Entorno donde hice la prueba

- Máquina virtual: Debian/Ubuntu (local).
- Aplicación: DVWA (Damn Vulnerable Web Application).
- URL de acceso: `http://127.0.0.1/dvwa/`
- Base de datos: MySQL/MariaDB local, puerto `3306`.
- Credenciales DVWA: usuario `admin` / contraseña `password`.
- Configuración en `dvwa/config/config.inc.php`:

    - `db_server = 127.0.0.1`
    - `db_database = dvwa`
    - `db_user = dvwa`
      `db_password = pass`
    - `db_port = 3306`

3. Pasos que seguí (instalación y preparación)

1. `cd /var/www/html/`
2. `sudo apt-get install wget unzip git`
3. `sudo git clone https://github.com/digininja/DVWA.git /var/www/html/dvwa`
4. `sudo chmod -R 755 /var/www/html/dvwa`
5. `cd /var/www/html/dvwa/config/`
6. `sudo mv config.inc.php.dist config.inc.php` y edité el archivo con las credenciales indicadas arriba.

7. En MySQL:

    - `CREATE DATABASE dvwa;`
    - `CREATE USER 'dvwa'@'127.0.0.1' IDENTIFIED BY 'pass';`
    - `GRANT ALL PRIVILEGES ON dvwa.* TO 'dvwa'@'127.0.0.1';`

8. Abri `http://127.0.0.1/dvwa/setup.php` y pulsé **Crear/Restablecer base de datos**.

9. Entré en `http://127.0.0.1/dvwa/login.php` con `admin / password`.

10. En DVWA -> **DVWA Security** seleccioné **Low**.

---

4. Explotación (que hice y que paso)

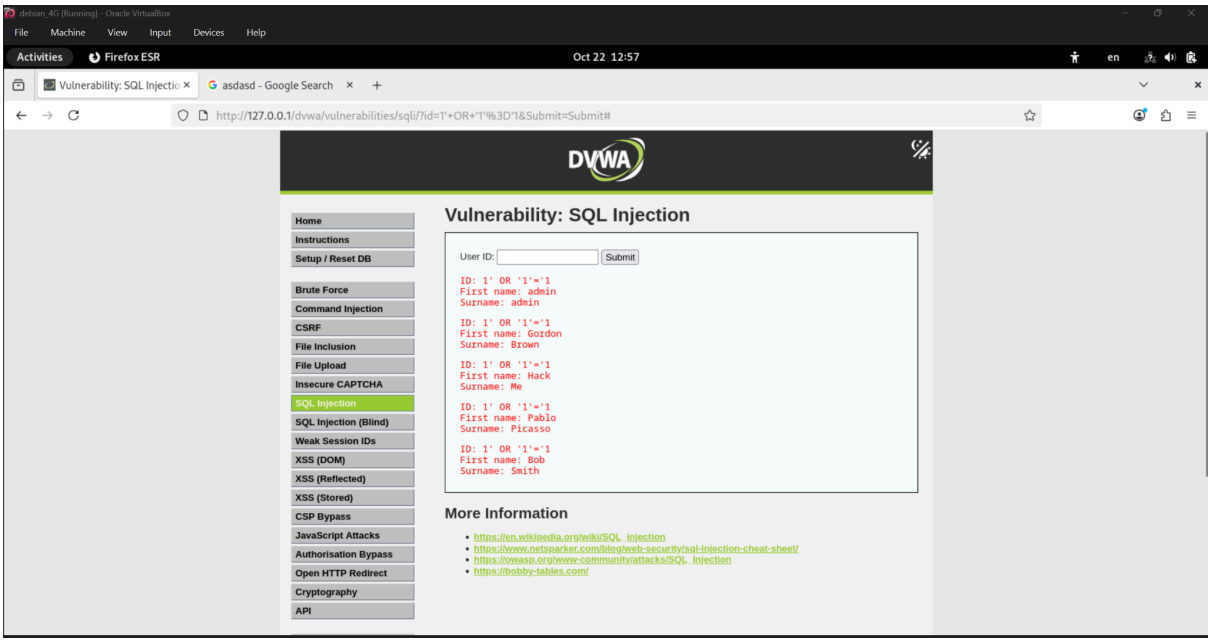
- **objetivo:** formulario vulnerable.

Payload usado:

```
' OR '1'='1
```

- **Qué ocurrió:** la aplicación construyo una consulta SQL concatenando mi entrada. La condición `OR '1'='1'` es siempre verdadera, por eso la consulta devolvio todas las filas de la tabla `users`. En pantalla vi la lista

completa de usuarios — prueba clara de inyección SQL.



---

6. Impacto y clasificacion

- **Impacto técnico:** divulgación de datos en la base de datos → confidencialidad comprometida.
- **Alcance:** filas de la tabla `users` de DVWA (credenciales de prueba).
- **Riesgo real:** en una aplicación productiva esto seria crítico — robo de credenciales, acceso no autorizado, escalamiento.
- **Clasificación (ISO/27001):** incidente de seguridad que afecta confidencialidad/integridad; requiere respuesta y mitigación.

---

7. Conclusion

La prueba confirmó una vulnerabilidad de **inyeccion SQL** en la instalación de DVWA con nivel de seguridad *Low*: el payload `' OR '1'='1` permitió extraer datos de la base de datos al no aplicarse saneamiento ni consultas preparadas.

Recomendaciones: **-Aplicar Validaciones de Entradas**