

# SQL Injection en DVWA

## Introducción

Este informe detalla un ataque de SQL Injection llevado a cabo en la plataforma Damn Vulnerable Web Application (DVWA), configurada en una máquina virtual Debian. Se exploró una vulnerabilidad a nivel de base de datos que permitió la ejecución de consultas maliciosas, obteniendo acceso a información sensible.

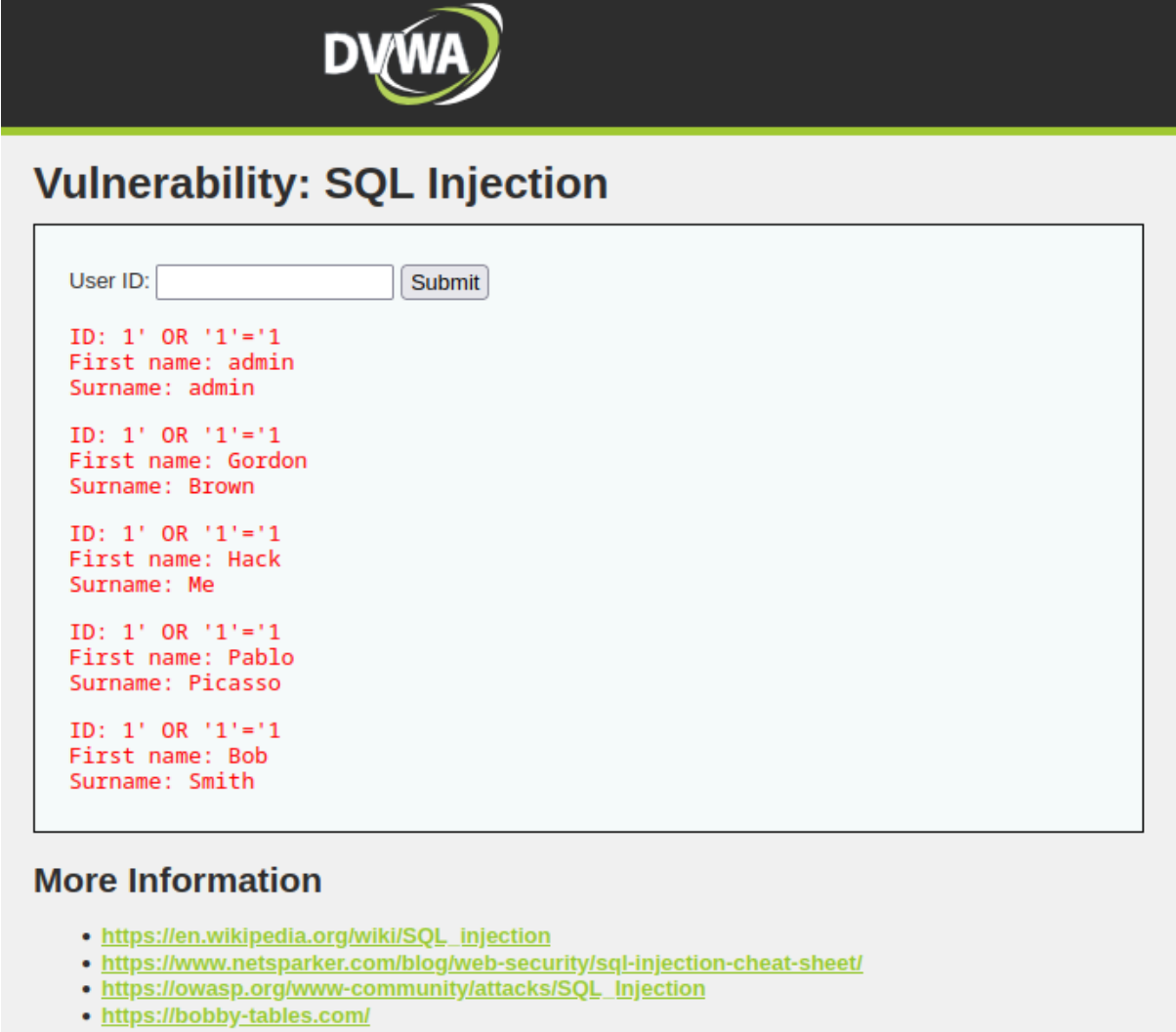
## Descripción del Incidente

El ataque se basa en una vulnerabilidad de inyección SQL, donde un atacante puede manipular las consultas enviadas al servidor mediante la inserción de código malicioso. En este caso, se utilizó la aplicación DVWA para ilustrar cómo es posible obtener resultados de la base de datos sin la autorización adecuada.

## Proceso de Reproducción

1. Configurar la red VM Debian en “Adaptador Puente” para que esté en la misma red que el host.
2. Verificar la correcta instalación de MySQL(MariaDB), Apache y PHP (LAMP Stack)
3. Descargar DVWA desde un enlace proporcionado:
  - a. `cd /var/www/html`
  - b. `2sudo apt-get install wget unzip`
  - c. `3sudo wget https://storage.googleapis.com/breathecode/virtualbox/DVWA.zip`  
`sudo unzip DVWA.zip`
  - d. `4sudo mv DVWA-master DVWA`
4. Configurar DVWA y renombrar el archivo de configuración:
  - a. `cd DVWA/config`
  - b. `sudo cp config.inc.php.dist config.inc.php`
5. Editar el archivo de configuración “config.inc.php” para configurar las credenciales correctas de MariaDB:
  - a. `sudo nano config.inc.php`
  - b. Credenciales:
    - i. `$_DVWA['db_user'] = 'root';`
    - ii. `$_DVWA['db_password'] = 'tu_contraseña_de_root';`
    - iii. `$_DVWA['db_database'] = 'dvwa';`
6. Crear la base de datos DVWA y ajustar los permisos:
  - a. `sudo mysql -u root -p`
  - b. `CREATE DATABASE dvwa;`
  - c. `EXIT;`
  - d. `sudo chown -R www-data:www-data /var/www/html/DVWA/`
  - e. `2sudo chmod -R 755 /var/www/html/DVWA/`
7. Abrir en el navegador de la VM Debian “<http://localhost/DVWA/setup.php>”, revisar la configuración y crearla con “**Create/Reset Database**”

8. Acceder a <http://localhost/DVWA>.
9. Iniciar sesión en la plataforma DVWA con el usuario **admin** y la contraseña **password**.
10. Cambiar el nivel de seguridad a "Low" en las opciones de configuración de DVWA.
11. En la sección SQL Injection, introducir el siguiente payload en el campo de entrada:  
**1' OR '1'='1**
12. Hacer clic en "Submit" y observar los resultados que el sistema devuelve desde la base de datos



The screenshot shows the DVWA (Damn Vulnerable Web Application) interface. At the top is the DVWA logo. Below it, the section is titled "Vulnerability: SQL Injection". There is a form with a "User ID:" label and a text input field, followed by a "Submit" button. Below the form, the results of the query are displayed in red text. The results show five rows of user data, all of which were retrieved due to the SQL injection payload. The payload used was "1' OR '1'='1".

User ID:

ID: 1' OR '1'='1  
First name: admin  
Surname: admin

ID: 1' OR '1'='1  
First name: Gordon  
Surname: Brown

ID: 1' OR '1'='1  
First name: Hack  
Surname: Me

ID: 1' OR '1'='1  
First name: Pablo  
Surname: Picasso

ID: 1' OR '1'='1  
First name: Bob  
Surname: Smith

**More Information**

- [https://en.wikipedia.org/wiki/SQL\\_injection](https://en.wikipedia.org/wiki/SQL_injection)
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- [https://owasp.org/www-community/attacks/SQL\\_injection](https://owasp.org/www-community/attacks/SQL_injection)
- <https://bobby-tables.com/>

### Impacto del Incidente

La explotación de esta vulnerabilidad permite al atacante obtener acceso no autorizado a información almacenada en la base de datos. Esto puede incluir nombres de usuarios, contraseñas y otros datos confidenciales, lo que compromete la integridad y confidencialidad del sistema.

### Recomendaciones

- Implementar una validación de entrada estricta para evitar que se procesen caracteres especiales en las consultas SQL.

- Utilizar consultas preparadas (prepared statements) para separar la lógica de la consulta de los datos ingresados por los usuarios.
- Aumentar el nivel de seguridad de la aplicación web y realizar pruebas periódicas de penetración para identificar vulnerabilidades.

### **Conclusión**

El ataque de inyección SQL realizado en DVWA demuestra cómo una falta de validación adecuada de las entradas puede resultar en la exposición de información sensible. Es crucial adoptar medidas de seguridad proactivas para mitigar este tipo de ataques y proteger la información almacenada en los sistemas.