Informe de Incidente de Seguridad

Vulnerabilidad de Inyección SQL en DVWA

1. Introducción

El presente informe documenta la explotación de una vulnerabilidad de **Inyección SQL (SQL Injection)** identificada en la aplicación **Damn Vulnerable Web Application (DVWA)**, utilizada con fines educativos para el análisis de fallos de seguridad en aplicaciones web.

El objetivo del ejercicio fue demostrar cómo una validación incorrecta de entradas permite a un atacante manipular consultas SQL y acceder a información sensible sin autorización.

---

2. Descripción del Incidente

La vulnerabilidad se encuentra en el módulo **SQL Injection** de DVWA.
La aplicación acepta un parámetro de entrada (*User ID*) que es concatenado directamente en una consulta SQL sin sanitización adecuada.

Esto permite que un atacante inserte código SQL malicioso y modifique el comportamiento original de la consulta.

---

3. Proceso de Reproducción

Se accedió a la aplicación DVWA desde la máquina virtual:

```
http://localhost/dvwa
```

1.
2. Se inició sesión con las credenciales por defecto:
    - **Usuario:** admin
    - **Contraseña:** password
3. Se configuró el nivel de seguridad en **Low**.
4. Se ingresó al módulo **SQL Injection**.

En el campo *User ID* se introdujo la siguiente entrada maliciosa:

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

5.
6. Se presionó el botón **Submit**.

---

4. Evidencia del Ataque

Al ejecutar la inyección SQL, la aplicación devolvió múltiples registros de usuarios, demostrando que la condición siempre se evalúa como verdadera y que la base de datos fue expuesta.

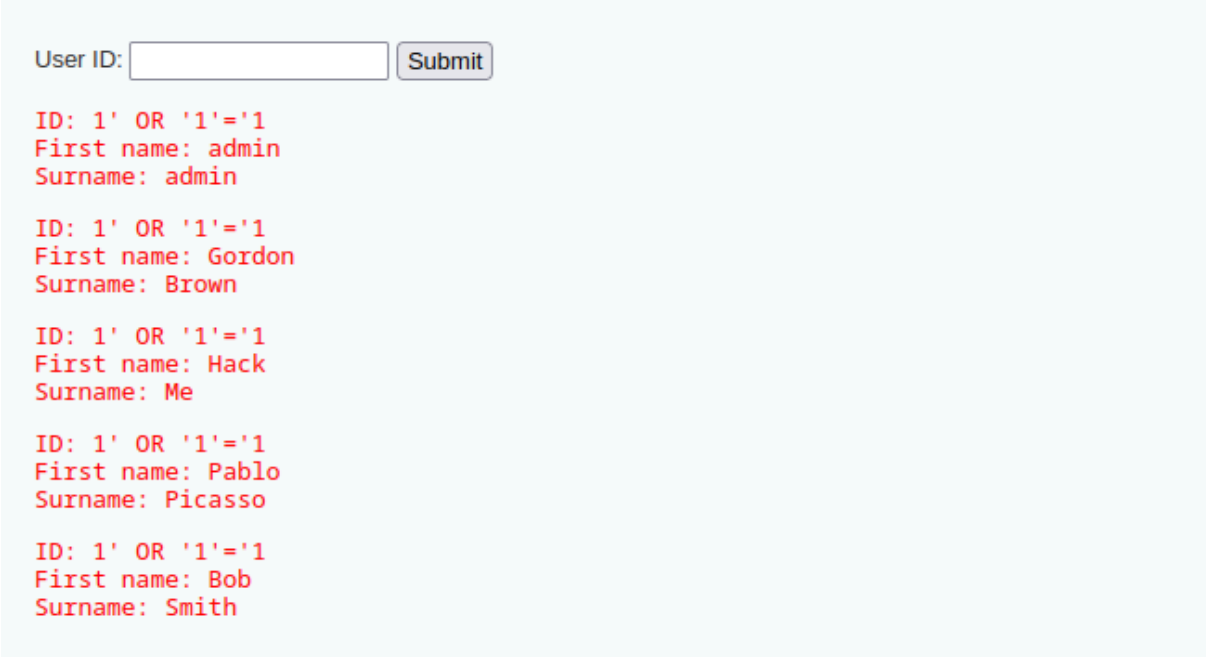

Figura 1. Resultado del ataque de SQL Injection mostrando la enumeración de usuarios.

---

5. Impacto del Incidente

La explotación de esta vulnerabilidad permite:

- Acceso no autorizado a información sensible.
- Enumeración de usuarios del sistema.
- Posible escalamiento a ataques más graves como extracción de contraseñas o modificación de datos.
  Este tipo de vulnerabilidad representa un riesgo crítico en aplicaciones reales.

---

6. Recomendaciones

Para mitigar este tipo de ataques se recomienda:

- Uso de **consultas preparadas (prepared statements)**.
- Validación y sanitización estricta de entradas del usuario.
- Implementación de controles de seguridad en capas.
- Uso de niveles de seguridad adecuados en entornos productivos.

---

7. Conclusión

El ejercicio demuestra cómo una mala gestión de entradas puede comprometer completamente la seguridad de una aplicación web.
La Inyección SQL sigue siendo una de las vulnerabilidades más peligrosas y comunes, por lo que su correcta mitigación es esencial en el desarrollo seguro de software.