🛠️ Ejercicio Práctico

📌 **Título:** Explotación Controlada de una Inyección SQL y Validación Manual de una Vulnerabilidad con Burp Suite

---

🎯 Objetivo del ejercicio:

Simular una prueba de penetración intermedia sobre una aplicación web vulnerable, utilizando **Burp Suite** para interceptar y manipular peticiones HTTP. El estudiante identificará manualmente una vulnerabilidad de tipo **SQL Injection**, validará el impacto real y documentará sus hallazgos bajo una estructura profesional.

📝 Escenario:

Estás participando en una auditoría de seguridad controlada sobre una aplicación web en entorno local (DVWA o similar). Se sospecha de una vulnerabilidad de tipo SQLi en la sección de búsqueda de usuarios. Se espera que valides manualmente si el fallo es explotable y determines el riesgo potencial.

🔍 Tu tarea:

✅ Paso 1 – Configuración y Reconocimiento

1. Asegúrate de que DVWA esté corriendo correctamente en `http://localhost/dvwa` con el nivel de seguridad configurado en **Low** o **Medium**.

2. Inicia **Burp Suite** y configura el navegador para redirigir todo el tráfico por `127.0.0.1:8080`.

3. Accede al módulo de **SQL Injection**.

✅ Paso 2 – Interceptar y analizar la solicitud vulnerable

1. Ingresa el valor 1 en el campo del parámetro “ID” y activa la opción “Intercept is on” en Burp.
2. Revisa los datos enviados al servidor.
3. Identifica el parámetro susceptible a inyección.

✅ Paso 3 – Probar la inyección SQL manual

1. Sustituye el valor original por un payload como:

```
1' AND 1=1 --
```

2. Reenvía la solicitud y observa el comportamiento de la aplicación.
3. Luego prueba con un payload falso:

```
1' AND 1=2 --
```

4. Compara ambos resultados para confirmar si el comportamiento varía entre una condición verdadera y una falsa.

✅ Paso 4 – Validar el impacto de la vulnerabilidad

1. Si la diferencia es visible (por ejemplo, se muestra o no se muestra información), indica que **la inyección es real y explotable**.
2. Agrega un tercer payload que extraiga la base de datos (si es posible):

```
1' UNION SELECT 1, database(), 3 --
```

✅ Paso 5 – Documentar hallazgos

1. Anota:
    - Parámetro vulnerable
    - Payloads utilizados
    - Resultados observados
    - Riesgo estimado (Bajo, Medio, Alto)
2. Escribe una recomendación profesional para mitigar este tipo de vulnerabilidad.

🧪 Resultado esperado:

- Identificación correcta de la inyección SQL manual
- Evidencia del comportamiento alterado según el payload
- Explicación clara del riesgo técnico asociado
- Recomendación técnica fundamentada

🧠 Reflexión Final:

¿Cómo cambia el comportamiento de la aplicación según los distintos payloads?
¿Qué tan fácil sería para un atacante obtener datos confidenciales?
¿Por qué es importante validar cada parámetro antes de darlo por seguro?
