Lab 03: SQL Injection Retrieve Database Version (Oracle)

Lab URL (solved instance):

https://portswigger.net/web-security/sql-injection/union-attacks/lab-determine-database-version

Objective

This lab demonstrates how to exploit a SQL injection vulnerability in a product category filter to retrieve the Oracle database version.

Oracle requires that all SELECT statements include a FROM clause. When using UNION SELECT on Oracle, queries must include a FROM dual clause (a built-in one-row table). Version information can be extracted from the v$version table.

Key Concepts:

- UNION SELECT allows us to inject custom data into the query results
- The v$version table stores Oracle version strings
- dual is a required dummy table in Oracle

Payloads Used:

1. Column count and text test:

```
' UNION SELECT 'abc','def' FROM dual--
```

2. Version retrieval:

```
' UNION SELECT BANNER,NULL FROM v$version--
```

Exploitation Steps with Screenshots

Step 1: Accessed the lab page (1-access-lab.png)

Step 2: Clicked on a product category (2-click-category.png)

Step 3: Intercepted the original category request in Burp Suite (3-intercepted-request.png)

Step 4: Tested column count and types with UNION SELECT (4-column-test.png)

Step 5: Sent final payload to retrieve version from v$version (5-db-version-request.png)

Step 6: Verified that Oracle version info was displayed (6-db-version-revealed.png)

Step 7: Lab marked as solved with confirmation banner (7-lab-solved.png)

Repository Structure

Cybersecurity-Portfolio/

SQL-Injection/

Lab-03_SQL-injection-version-on-Oracle/

1-access-lab.png

2-click-category.png

3-intercepted-request.png

4-column-test.png

5-db-version-request.png

6-db-version-revealed.png

7-lab-solved.png

#bugbounty #sqlinjection #oracle #cybersecurity #portswigger #infosec #ethicalhacking #australianinfosec