St. Francis Institute of Technology, Mumbai-400 103
Department Of Information Technology

A.Y. 2025-2026
Class: BE-IT A/B, Semester: VII

Subject: Secure Application Development Lab

Student Name: Student Roll No:

Experiment – 6: SQL injection vulnerability that allows the login page to bypass.

Aim: To apply SQL injection vulnerability that allows the login page to bypass.

Objectives: Aim of this experiment is that the students will be able understand

- To study the different types of Vulnerabilities
- To apply SQL injection Attack.

Lab objective mapped: ITL703.4 To apply Data Validation and Authentication for application

Requirements: Personal Computer, Windows operating system browser, Internet Connection

Pre-Experiment Theory:

SQL injection (SQLi) is a type of an injection attack that makes it possible to execute malicious SQL statements. These statements control a database server behind a web application. Attackers can use SQL Injection vulnerabilities to bypass application security measures. They can go around authentication and authorization of a web page or web application and retrieve the content of the entire SQL database. They also use SQL injection for adding, modifying and deleting records from the database.

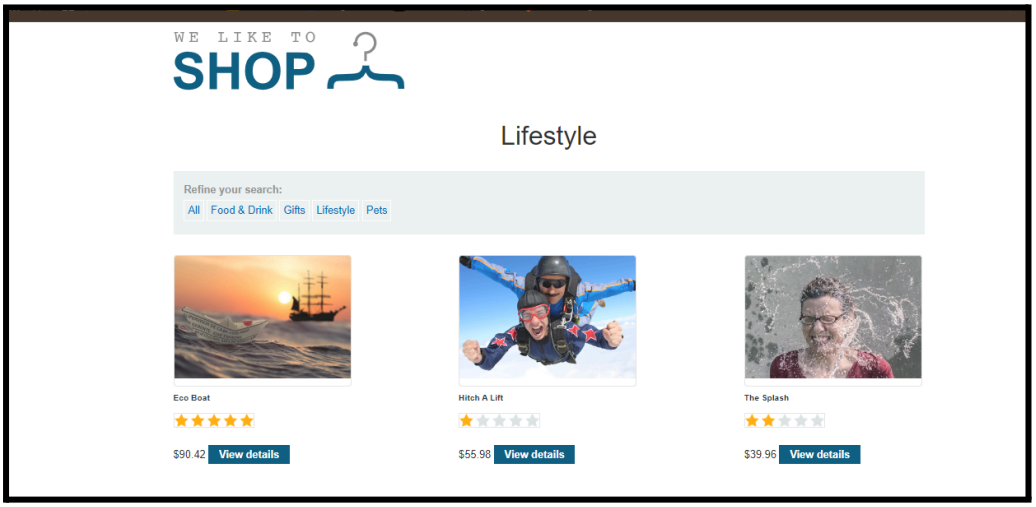
A hacker executes an SQL injection with an SQL statement that is always true. For instance, 1=1; instead of just entering the “wrong” input, the hacker uses a statement that will always be true.

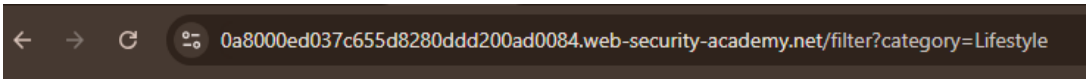
Entering “117 OR 1=1” in the query input box will return a response with the details of a table.

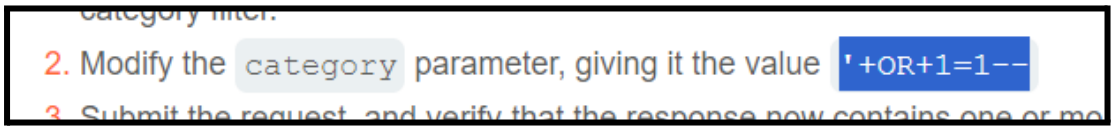
This SQL injection approach is similar to the above. A hacker needs to enter "OR ""=" into the query input box. These two signs serve as the malicious code to break into the application. Consider the following example. An attacker seeks to retrieve user data from an application and can simply type “OR=” in the user ID or password. As this SQL statement is valid and true, it will return the data of the user table in the database.

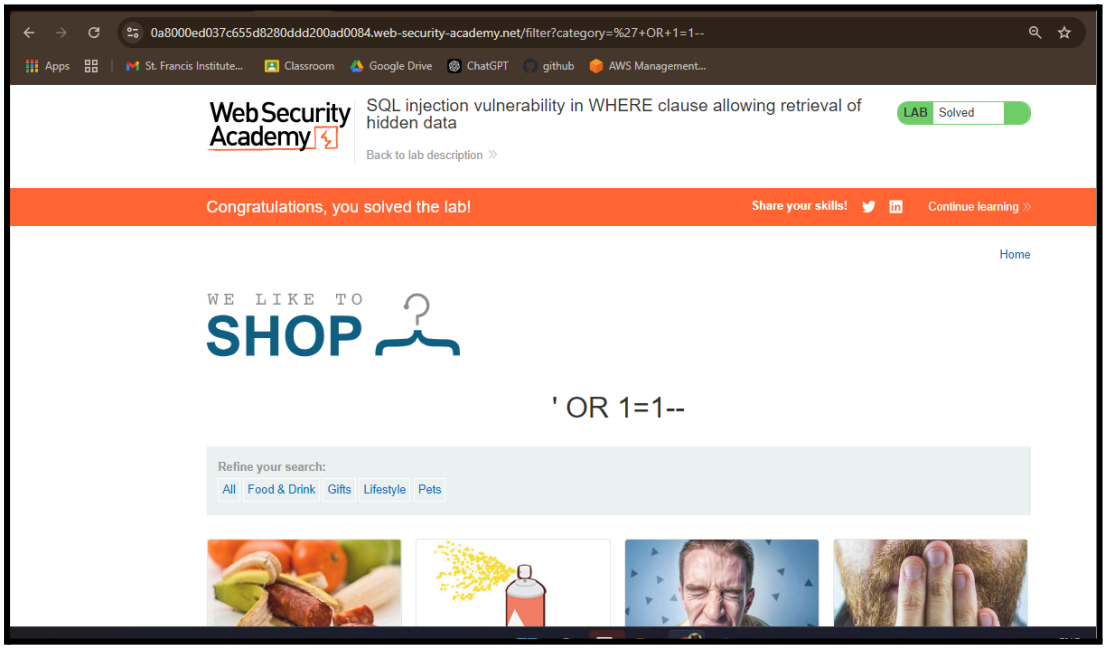
Procedure:

1. **Perform an SQL injection vulnerability in WHERE clause allowing retrieval of hidden data**
   a. Use Burp Suite to intercept and modify the request that sets the product category filter.
   b. Modify the category parameter, giving it the value '+OR+1=1–
   c. Submit the request, and verify that the response now contains additional items.





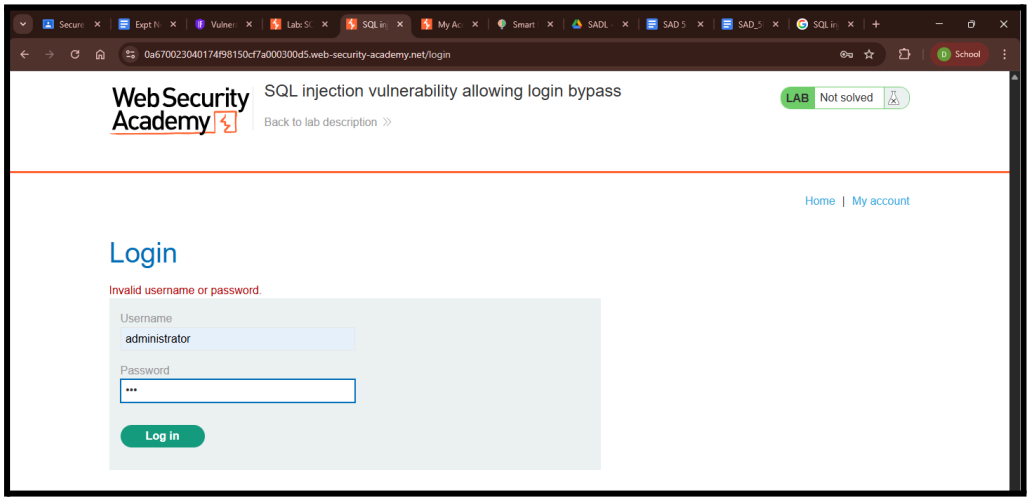


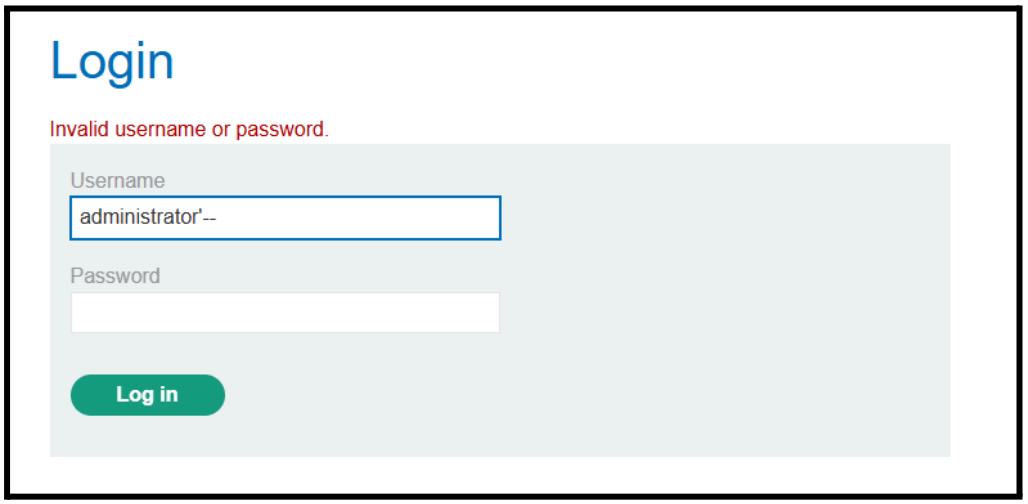


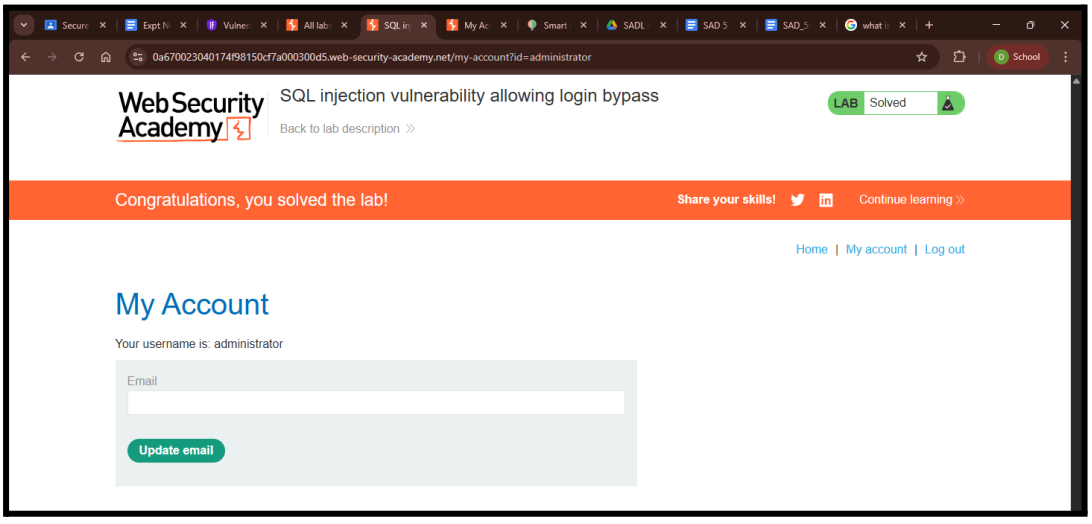
2. **SQL injection vulnerability allowing login bypass**

a. Use Burp Suite to intercept and modify the login request.

b. Modify the username parameter, giving it the value: administrator'--







Post-Experiments Exercise:

Extended Theory: Nil

Post Experimental Exercise:

Questions:

- How to prevent SQL injection Attack?
- How can organizations ensure their databases and web application are secure against SQL injection threat?
- What are some real-world examples of security breaches caused by SQL injection

Conclusion:

- Write what was performed in the experiment.
- Write the significance of the topic studied in the experiment.

References:

1. https://portswigger.net/web-security/sql-injection
2. https://portswigger.net/web-security/sql-injection/lab-login-bypass
3. https://www.indusface.com/blog/explore-vulnerability-assessment-types-and-methodology/
