**Practical: 11**

**Aim:Use Burpsuit tool and refer SQL Injection lab. Implement at least two labs.Use necessary screenshot of it.Explain SQL injection and  how to prevent it.**

**1. What is SQL Injection?**

**SQL Injection (SQLi)** is a web security vulnerability that allows an attacker to interfere with a website’s database by injecting **malicious SQL queries**. It can lead to:  
Unauthorized access to sensitive data (usernames, passwords, credit cards, etc.)  
Modifying or deleting database records  
Gaining admin privileges

**2. Example of SQL Injection**

A vulnerable login query:

SELECT \* FROM users WHERE username = 'admin' AND password = 'password';

If an attacker enters:

Username: admin

Password: 12345

The query becomes:

SELECT \* FROM users WHERE username = 'admin' --' AND password = 'password';

Lab-1: SQL Injection to Retrieve Hidden Data

**📌 Steps to Perform SQL Injection**

1️ Open Burp Suite and turn Intercept ON.  
2️ Visit a vulnerable product listing page in Port Swigger SQL Injection Lab.  
3️ Capture the request using Burp Suite.  
4️ Modify the request by injecting SQL code:

?category=Gifts' OR '1'='1' –

5️ Forward the request.  
6️ Observe the hidden products appearing on the page.

Output : -

Previously hidden products **appear on the website** because SQL Injection bypassed access controls.

**Lab-2: SQL Injection to Bypass Authentication**

**Goal:** Login as an admin **without knowing the password**.

**📌 Steps to Perform SQL Injection**

1️ Open Burp Suite and turn Intercept ON.  
2️ Go to the login page of the vulnerable SQL Injection lab.  
3️ Enter the following credentials:

Username: admin , Password:12345

4️ Capture and modify the request if necessary.  
5️ Forward the request.  
6️ Observe that you are logged in as an admin!

Output : -

The website logs in as admin without a password.

**3. How to Prevent SQL Injection?**

Use Prepared Statements (Parameterized Queries)  
Instead of:

SELECT \* FROM users WHERE username = '"+user+"' AND password = '"+password+"';

Use:

cursor.execute("SELECT \* FROM users WHERE username = ? AND password = ?", (user, password))

Use ORM Frameworks (SQL Alchemy, Django ORM, etc.).  
Sanitize & Escape User Input before using it in SQL queries.  
Limit Database Permissions for web applications. Use Web Application Firewalls (WAFs) like Cloudflare, Mod Security.