🛡 Cybersecurity Internship Final Report

Submitted by: ***Rayyan Chaaran***
Date: 26 June 2025

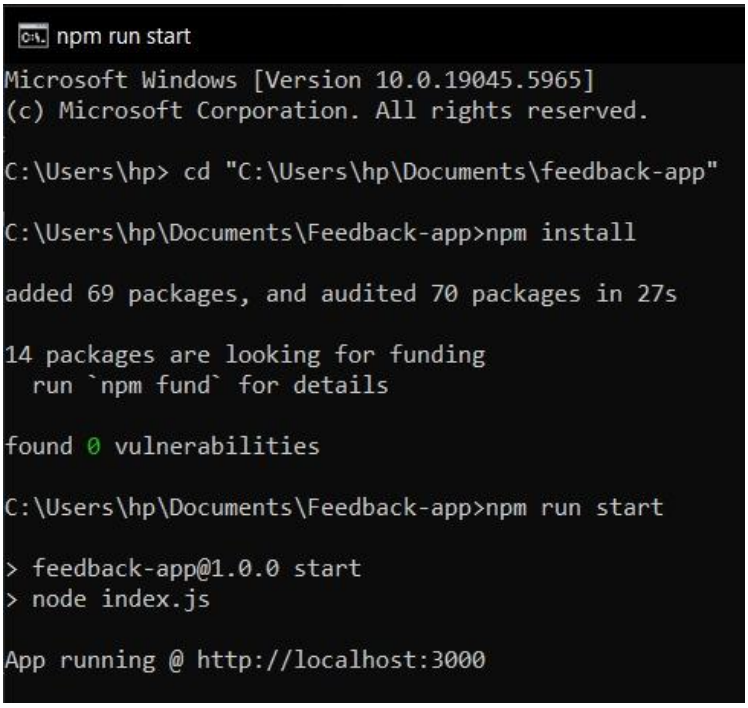
🕮 Introduction:

This report outlines the security testing and improvements conducted on a mock **User Management Web Application** as part of my cybersecurity internship.
The objective was to identify common web application vulnerabilities (XSS, SQL Injection, weak passwords), apply security best practices (input validation, password hashing, secure authentication), and finally implement logging and penetration testing measures.

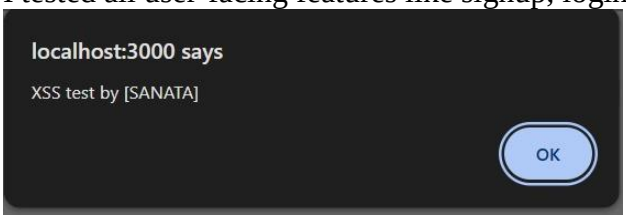
🗓 Week 1 — Application Setup & Vulnerability Assessments

🔑 Application Setup

First, I set up the web application using:



This ran the app at `http://localhost:3000`.
I tested all user-facing features like signup, login, and profile viewing.



Goal: Get familiar with the app and check for visible security flaws.

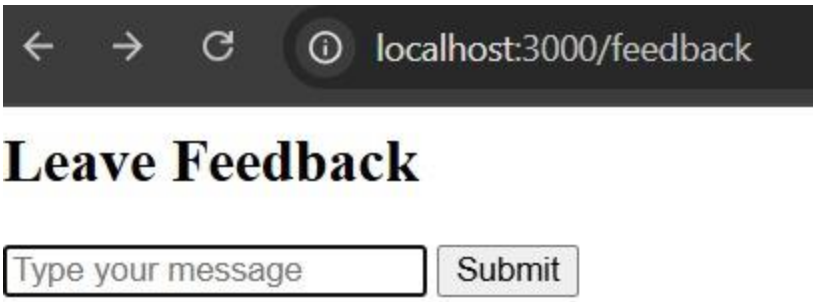
(Application signup/login screen as initially set up)

Vulnerability Testing:

Cross-Site Scripting (XSS):

Cross-Site Scripting (XSS) allows attackers to inject malicious JavaScript into web pages viewed by other users.
Test Performed:
Submitted `<script>alert('XSS');</script>` into the profile input fields.

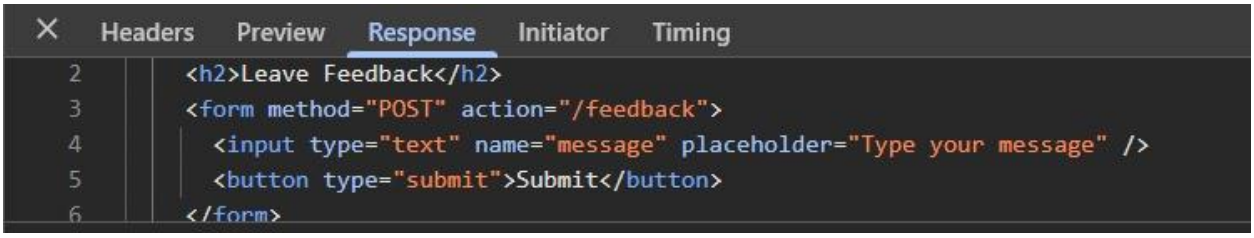
Result:
XSS was successful — alert pop-up displayed.



SQL Injection:

SQL Injection allows an attacker to manipulate SQL queries via form inputs — often bypassing authentication.
Test Performed:
Submitted `admin' OR '1'='1` as username and password.

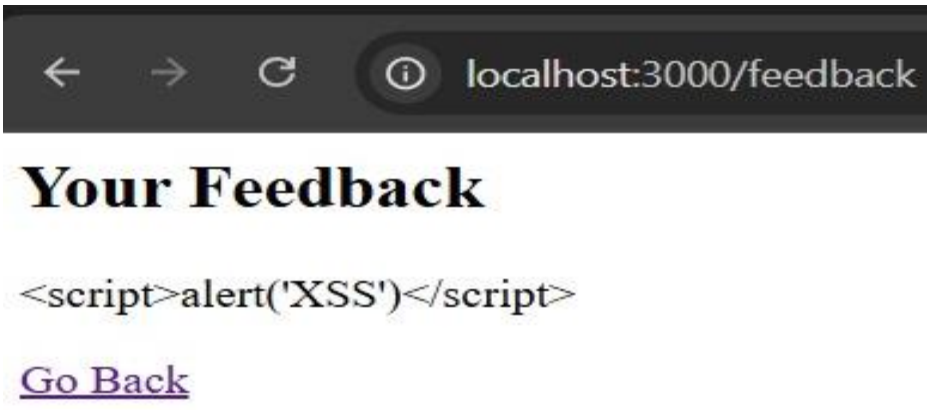
Result:
Access was granted without a real password — SQL Injection present.



Vulnerability Scan:

Using automated tools like **OWASP ZAP** identifies common web app misconfigurations and vulnerabilities.
Test Performed:
Scanned the app with ZAP and browser tools.

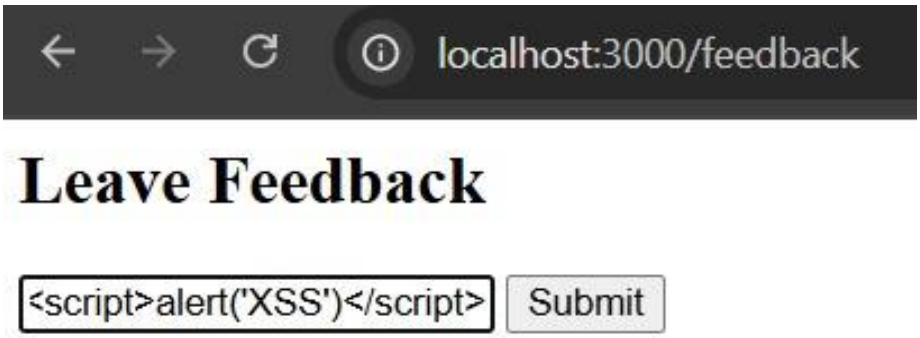
Findings:
Misconfigured security headers, missing input validation, plain-text password storage.



Weak Password Storage:

Storing passwords as plain text is extremely dangerous — attackers can read all passwords if the database is leaked.

Result:
Password was visible as plain text.


localhost:3000/feedback
Leave Feedback
<script>alert('XSS')</script>
Submit