

Cybersecurity Internship Report: Week 1

Intern Name: Mahnoor Khurram

Project: Security Assessment of NodeGoat Application

Date: December 30, 2025

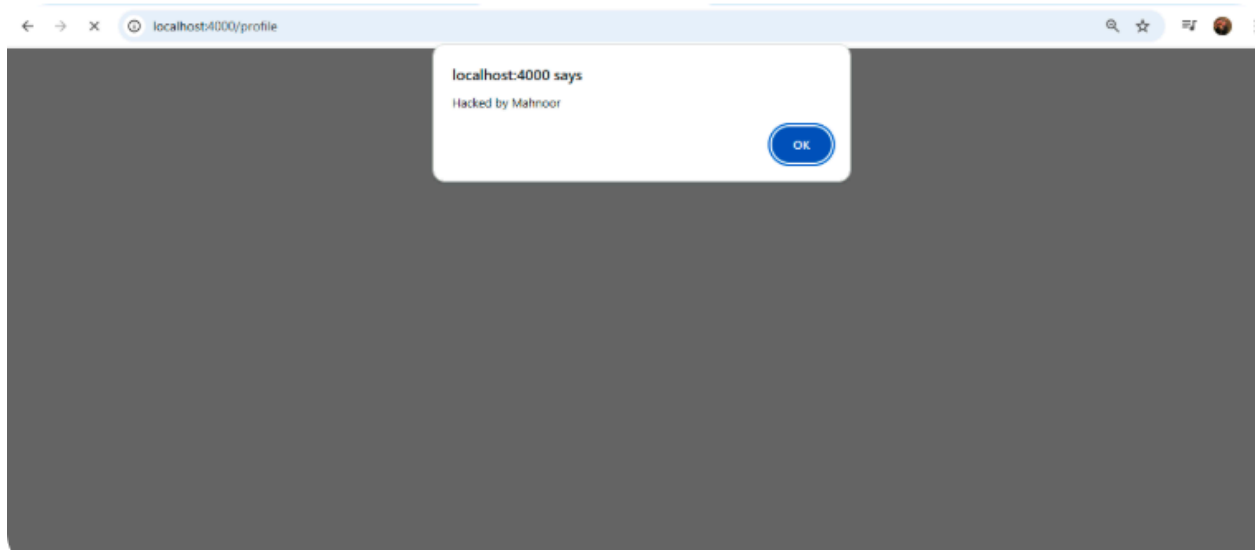
1. Introduction

The goal of this week's task was to perform a manual security assessment on a local web application (NodeGoat). I focused on identifying common vulnerabilities such as **Cross-Site Scripting (XSS)** and **SQL Injection (SQLi)** using manual techniques and Browser Developer Tools.

2. Security Findings

A. Stored Cross-Site Scripting (XSS)

- **Method:** I entered a malicious JavaScript payload `<script>alert('Hacked by Mahnoor')</script>` into the "First Name" field on the Profile page.
- **Result:** The application failed to sanitize the input, and the script executed automatically, displaying an alert box.
- **Impact:** This is a **High** severity issue because an attacker could use this to steal user cookies or session data.



B. SQL Injection (SQLi) Attempt


- **Method:** I attempted to bypass the login screen by entering a common SQL payload: `admin' OR '1'='1` in the username and password fields.
- **Result:** The application returned an "Invalid username" error message.
- **Conclusion:** This indicates that the login form has basic input validation or protection against simple SQL injection attacks.



 Tutorial Guide: Learn OWASP Top 10

 **RetireEasy**

Employee Retirement Savings Management

Invalid password 

User Name

admin

Password

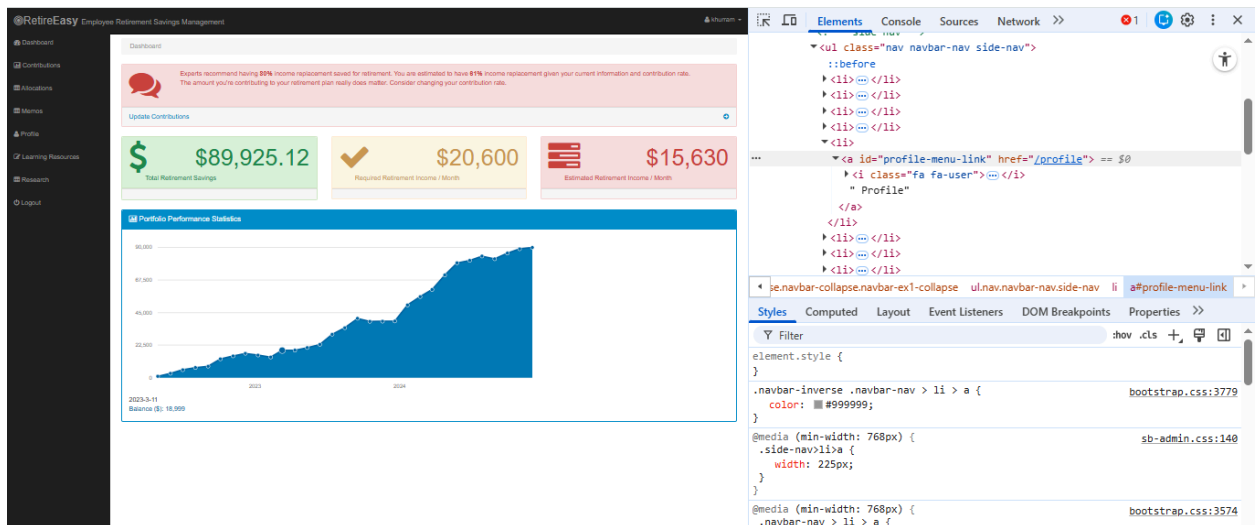
Enter Password

[New user? Sign Up](#)

Submit

C. Browser Developer Tools Analysis

- **Method:** I used the browser's "Inspect Element" feature to analyze the website's frontend code and DOM structure.
- **Result:** I was able to see how the application handles data and identify specific input fields for testing.



The screenshot displays the RetireEasy Employee Retirement Savings Management dashboard and the browser's developer tools. The dashboard features a sidebar with navigation links (Dashboard, Contributions, Educations, Metrics, Profile, Learning Resources, Research, Logout) and a main content area with a dashboard overview, a portfolio performance chart, and a sidebar with navigation links. The dashboard overview includes a message about 80% income replacement, a table with three rows (Total Retirement Savings, Required Retirement Income / Month, Estimated Retirement Income / Month), and a portfolio performance chart showing a line graph of performance over time. The browser developer tools are open on the right, showing the 'Elements' panel with the DOM tree. The selected element is a link with the id 'profile-menu-link' and href '/profile'. The 'Styles' panel shows the computed styles for the selected element, including a color of #999999 and a width of 225px.

Category	Value
Total Retirement Savings	\$89,925.12
Required Retirement Income / Month	\$20,600
Estimated Retirement Income / Month	\$15,630

Portfolio Performance Statistics

2023-3-11
Balance (\$): 10,000

3. Recommendations

- **Fix XSS:** Implement strict input validation and output encoding to prevent scripts from running in the browser.
- **Improve Security:** Ensure the application uses HTTPS to encrypt data between the user and the server.