IT241 Project 2
v1.0
Score: _____/100

Specifications

Students will conduct a focused research project on one of three major web security topics: SQL Injection, Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF), or Session Management and Security. Each student (or group, depending on your decision) will complete a research paper and a presentation, along with practical coding examples in PHP and MySQL (if applicable). All code must be pushed to the assignment GitHub repository for review.

Research Topics

1. SQL Injection: Understand and demonstrate the vulnerability, show an insecure PHP example, and implement a secure solution using MySQL and PDO.
2. Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF): Explain both vulnerabilities, show insecure PHP examples, and implement secure solutions, such as input sanitization for XSS and token-based protection for CSRF.
3. Session Management and Security: Discuss session management vulnerabilities (e.g., session hijacking, session fixation), explain how PHP handles sessions, and demonstrate securing session management by using HttpOnly and Secure flags, regenerating session IDs, and handling session data in MySQL.

Project Deliverables

1. **Research Paper (5-7 pages):**
    a. Double-spaced, Times New Roman, 12-point font, 1-inch margins.
    b. Title page (not included in the page count).
    c. No in-text citations required, but a reference page must be included at the end (also not included in the page count).
    d. A maximum of 1.5 cumulative pages of code, screenshots, and diagrams is allowed.
    e. The paper should discuss:
        - Explanation of the chosen security vulnerability (SQL Injection, XSS/CSRF, or HTTP insecurities).
        - Real-world examples of attacks that exploited this vulnerability.
        - Methods for preventing these vulnerabilities using PHP and MySQL (if applicable).
        - Code examples (on GitHub) that demonstrate insecure code and the secure version of it.
    f. The paper should be uploaded to the assignment GitHub repository in docx, doc, odt, or pdf format.
2. **Presentation**
    a. A 10-15 minute presentation summarizing the research findings.

b. Must include at least one demonstration of insecure vs. secure PHP code or HTTP requests.

c. Visual aids (slides, diagrams) should be used to enhance the explanation of technical content.

d. The presentation should be uploaded to the assignment GitHub repository in docx, doc, odt, or pdf format.

e. Note: Students must be prepared to ask and receive challenging questions from the instructor and classmates regarding their topic. Mastery of the subject matter will be assessed based on the student's ability to respond thoughtfully.

3. **Code on GitHub:**

   a. One or more PHP web applications that demonstrates:

      - For SQL Injection: An insecure PHP script with a form that interacts with MySQL, followed by the same form with secure, parameterized queries using PDO.
      - For Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF): An example of a vulnerable PHP page that displays user input without sanitization, followed by a secure version using htmlspecialchars() and strip_tags(). Additionally, a PHP form that lacks CSRF protection, followed by the implementation of token-based protection to prevent attacks.
      - For Session Management and Security: Demonstrate PHP session vulnerabilities such as session hijacking and fixation. Create a PHP script that handles user authentication and session management. Show an insecure implementation and a secure one using best practices such as session ID regeneration, HttpOnly and Secure cookie flags, and optionally, session storage in MySQL.

   b. GitHub repository must include:

      - A README.md file explaining the purpose of the code, with clear instructions on how to run it.
      - A .gitignore file in the root of the repository to exclude sensitive or unnecessary files (e.g., .idea/, node_modules/, composer.lock, and environment configuration files like .env).
      - Separate folders for insecure and secure versions of each example (if applicable).

Submission

You will submit the commit ID for the commit you want graded. Submit the commit ID on the Canvas assignment.

Academic Honesty Policy

All code submissions must be original and authored by the student. Any code sourced from another student, falsely presented as one's own, or derived from third-party websites or generated by AI, will constitute a breach of the school's academic honesty policy.

AI tools may be used for SQL query help *only* if clearly documented in the README.md file.