NAQASH ABBAS

DHC-3776

CYBERSECURITY

My repository :

https://github.com/Naqash14/cyber-security-assessment-internship

---

1. Introduction

This report summarizes the tasks completed during my cybersecurity internship, focusing on assessing and strengthening the security measures of a web application. The project involved performing a vulnerability assessment, implementing security measures, and conducting penetration testing to enhance the security of the application.

---

2. Security Assessment

Understanding the Application

- Cloned the repository:

https://github.com/goshurarah/best-login-signup-form-using-nodejs.git

- Installed dependencies and set up MongoDB
- Launched the application locally for security testing

Basic Vulnerability Assessment

- **Tools Used:** OWASP ZAP, Browser Developer Tools, Manual Testing
- **Vulnerabilities Identified:**
    - **Reflected XSS:** Injecting <script>alert('XSS');</script> in input fields
    - **SQL Injection:** Using admin' OR '1'='1 to bypass authentication
    - **Weak Password Storage:** Passwords stored in plaintext
    - **Security Misconfigurations:** Missing Content-Security-Policy header
    - **Insecure Cookies:** Lack of HttpOnly and Secure flags
- **Documentation:** Findings were recorded, and a vulnerability report was generated.

---

3. Implementing Security Measures

Fixing Identified Vulnerabilities

1. **Input Validation & Sanitization:**
    - Implemented validation using validator.js to prevent XSS and SQL injection.
2. **Password Hashing:**
    - Used bcrypt to hash passwords before storing them in the database.
3. **Authentication Security:**
    - Implemented JWT-based authentication for secure session management.
4. **Enhancing Security Headers:**
    - Used helmet.js to set Content-Security-Policy and other security headers.
5. **Secure Cookies:**
    - Enabled HttpOnly and Secure flags to prevent session hijacking.

---

4. Penetration Testing & Final Security Checks

Basic Penetration Testing

- **Tools Used:** OWASP ZAP, Nmap, Browser Developer Tools
- **Simulated Attacks:**
    - XSS and SQL Injection tests
    - CSRF attack simulation
    - Network scanning with Nmap

Logging & Monitoring Implementation

- Configured winston logging for tracking application activity and detecting threats.

Security Checklist Review

- Verified that security best practices were followed, including:
    - Input validation
    - Secure authentication and session management
    - Proper error handling
    - Regular security audits

---

5. Conclusion & Recommendations

- **Final Security Posture:**
    - Application security significantly improved with the fixes applied.
    - Reduced risks of XSS, SQL Injection, and insecure authentication.
- **Recommendations for Future Security Enhancements:**

0. Conduct periodic security audits.
1. Regularly update dependencies and apply security patches.
2. Implement advanced security features like Multi-Factor Authentication (MFA).

Educate developers on secure coding practice.