Report

1. Introduction

This penetration testing report documents the security assessment conducted as part of my internship. The assessment focused on identifying vulnerabilities in the target system using various testing methodologies and tools, including OWASP ZAP, SQL injection attempts, XSS testing, security misconfiguration assessments, and analysis of plain text password storage.

2. Setup

2.1 Test Environment

- **Target System:** Web application running PHP and MySQL
- **Hosting Server:** Ubuntu-based Apache server
- **Database Management System:** phpMyAdmin
- **Testing Tools Used:**
    - OWASP ZAP
    - SQLMap
    - Burp Suite
    - Nikto
    - Manual Code Review

2.2 Scope

- **Allowed Tests:** Non-destructive testing of the web application, database, and server configurations.
- **Excluded Tests:** No stress testing or Denial-of-Service (DoS) testing was conducted.

3. Methodology

The penetration testing followed the OWASP Testing Guide and included:

1. **Reconnaissance:** Identified technologies and services using built-in scanners.
2. **Scanning & Enumeration:** Used OWASP ZAP and Nikto to discover potential attack surfaces.
3. **Exploitation Attempts:** Conducted SQL injection, XSS, and security misconfiguration assessments.
4. **Post-Exploitation Analysis:** Reviewed access gained, impact assessment, and mitigation strategies.

4. Findings and Vulnerability Analysis

4.1 SQL Injection (Mapped to OWASP A03:2021 – Injection)

Findings:

- Multiple input fields vulnerable to SQL injection, leading to unauthorized database access.
- Automated tools like SQLMap successfully extracted user credentials.

Impact:

- Attackers could manipulate SQL queries to extract sensitive data or bypass authentication.

Recommendations:

- Implement prepared statements and parameterized queries.
- Apply input validation and sanitization.

4.2 Plain Text Passwords in phpMyAdmin (Mapped to OWASP A02:2021 – Cryptographic Failures)

Findings:

- User credentials stored in plain text within phpMyAdmin configurations.

Impact:

- Credentials exposed to attackers in case of a system breach.

Recommendations:

- Encrypt stored passwords using hashing algorithms like bcrypt.
- Restrict access to database configuration files.

4.3 Cross-Site Scripting (XSS) (Mapped to OWASP A07:2021 – Identification and Authentication Failures)

Findings:

- Reflected XSS vulnerabilities found in search fields and comment sections.
- Successful payload injection executed JavaScript in users' browsers.

Impact:

- Attackers could steal session cookies or deface the website.

Recommendations:

- Implement input encoding using HTML escaping functions.
- Use Content Security Policy (CSP) headers.

4.4 Security Misconfigurations (Mapped to OWASP A05:2021 – Security Misconfiguration)

Findings:

- Unrestricted access to phpMyAdmin.
- Default credentials found on some services.
- Directory listing enabled on the web server.

Impact:

- Increased attack surface for brute force and information disclosure attacks.

Recommendations:

- Restrict access to phpMyAdmin with IP whitelisting.
- Remove default accounts and enforce strong authentication.
- Disable directory listing in Apache configurations.

5. Recommendations and Mitigation

1. **Harden Server Configurations:** Implement security best practices for Apache, PHP, and MySQL.
2. **Secure Authentication Mechanisms:** Enforce multi-factor authentication (MFA) and strong password policies.
3. **Conduct Regular Security Audits:** Perform automated and manual security assessments periodically.

6. Conclusion

The penetration test revealed multiple critical vulnerabilities that need immediate remediation. Implementing security best practices and OWASP-recommended mitigation strategies will enhance the overall security posture of the application.