CodeAlpha Task3 : Rushikesh Mahalle

Internship Report: Identifying and Analyzing a Login Vulnerability

Internship Title: Bug Bounty Software Analysis

Internship Organization: CodeAlpha

Intern: Rushikesh Mahalle

**Introduction**

This report details the identification and analysis of a login vulnerability discovered on a test site using ZAProxy as part of my third internship. The primary objective was to enhance my skills in security testing and bug bounty software.

**Vulnerability Overview**

Type of Vulnerability: Authentication Bypass

Affected URL: `http://testphp.vulnweb.com/`

Description: The login page accepted the generic credentials "test" and "test," which granted unauthorized access to the site. This vulnerability could potentially allow attackers to gain unauthorized access to the application.

**Technical Analysis**

Scope Tab Configuration

Objective: Focus ZAProxy’s testing efforts on the relevant domain to filter out unrelated traffic.

Steps:

1. Added the test site `http://testphp.vulnweb.com/` to the scope.

2. Defined include and exclude rules to refine the target scope, ensuring precise and efficient testing.

Request Tab Analysis

Objective: Intercept and examine the HTTP request sent during the login attempt.

Steps:

1. Used ZAProxy to intercept the HTTP POST request generated by entering the credentials "test" and "test".

2. Analyzed the request headers and body to understand the data being sent to the server.

Response Tab Analysis

Objective: Observe and analyze the server's response to the intercepted request.

Steps:

1. Examined the HTTP response status code and content.

2. Identified that the server accepted the weak credentials, granting access to the application.

3. Confirmed the presence of the authentication bypass vulnerability.

**Findings and Impact**

Findings: The application accepted weak, hardcoded credentials, allowing unauthorized access. This vulnerability poses significant risks, including potential data breaches and unauthorized actions within the application.

Impact: Unauthorized access could lead to data exposure, loss of data integrity, and compromise of sensitive information. In a real world scenario, such vulnerabilities could be exploited to gain unauthorized control over the application and its resources.

**Recommendations**

1. Implement Strong Password Policies:

Enforce complex password requirements.

Prevent the use of default or easily guessable passwords.

2. Enhance Authentication Mechanisms:

Implement multi factor authentication (MFA) to add an extra layer of security.

Regularly review and update authentication processes.

3. Regular Security Audits:

Conduct periodic security assessments to identify and address vulnerabilities.

Implement automated security tools to continuously monitor and protect the application.

4. User Education and Awareness:

Educate users on the importance of strong passwords and secure practices.

Provide guidelines for creating and managing secure passwords.

**Conclusion**

The discovery and analysis of the login vulnerability provided valuable insights into the importance of robust authentication mechanisms and the risks associated with weak credentials. This internship has significantly enhanced my skills in security testing and vulnerability analysis, reinforcing the critical role of secure coding practices in software development.