INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

CRYPTOGRAPHY PROJECT-CS352

Report on Website Penetration Test of Zseano's Playground

GROUP MEMBERS

1. **ANNEM VENKATA KISHAN KUMAR REDDY(22BCS013)**
2. **NELLI VIVEK REDDY(22BCS076)**
3. **MODUGU YASHWANTH REDDY(22BCS072)**
4. **PEDISETTI TEJA VINAY KUMAR(22BCS084)**

Aim of the Project

The primary aim of this project is to **identify, analyze, and document vulnerabilities in a web application (Zseano's Playground)**, simulating a penetration testing scenario. The objective is to uncover potential security flaws that could be exploited by attackers, assess their impact, and propose mitigations to enhance the overall security posture of the application.

WEBSITE LINK:

Link

Key Objectives:

1. **Vulnerability Identification**:

   Detect weaknesses such as XSS (Cross-Site Scripting), IDOR (Insecure Direct Object References), credential leaks, and open URL redirection.

2. **Exploit Testing**:
   Demonstrate how attackers could exploit these vulnerabilities to compromise the system or access unauthorized information.
3. **Risk Assessment**:
   Evaluate the severity of the identified vulnerabilities and prioritize them based on their potential impact.

4. **Recommendations and Mitigation**:
   Provide actionable recommendations to secure the application, focusing on best practices like input validation, secure cookies, and access control mechanisms.
5. **Learning and Skill Development**:
   Gain hands-on experience with penetration testing tools, techniques, and methodologies. Understand common security pitfalls in web applications and ways to mitigate them.

6. **Documentation and Ethical Reporting**:
   Create a structured report to communicate findings effectively to stakeholders, ensuring ethical handling of sensitive information.

Tools Used

1 Burp Suite: Used for intercepting and analyzing HTTP requests and responses.
2 Browser DevTools: To inspect elements and find hidden information in the source code.
3 Custom Scripts: JavaScript payloads for XSS exploitation.
4 Curl/HTTP Clients: To send crafted requests and observe responses for IDOR and other vulnerabilities.
5 Manual Inspection: For URL endpoints, robots.txt, and other resources.

Major Vulnerabilities Found

1. **Reflected XSS (Cross-Site Scripting)**
   Location: act parameter during login.
   Exploit: Injecting JavaScript payloads by bypassing the HTML comment restrictions.
2. **IDOR (Insecure Direct Object References) and XSS**
   Location: order_id parameter in /fastfoodhackings/api/orders.php.
   Exploit: Access to other users' orders and injecting XSS payloads.
3. **Credential Leak**
   Location: f parameter in /fastfoodhackings/api/loader.php.
   Exploit: Accessing sensitive credentials by setting f=/generate-credentials.
4. **Information Leak**
   Location: adToken in /fastfoodhackings/api/admin.php.
   Exploit: Extracted sensitive tokens from the source code and used them to access administrative endpoints.
5. **Hidden Beta Access (Cookie Manipulation)**
   Exploit: Manipulating the beta cookie to gain access to beta features.
6. **Open URL Redirect & DOM-Based XSS**
   Location: custom-script.js with the redir parameter.
   Exploit: Redirecting users to malicious URLs and injecting JavaScript payloads.
7. **Another Open URL Redirect**
   Location: go.php with returnUrl parameter.
   Exploit: Redirecting to external or malicious URLs.
8. **Stored XSS**
   Location: Booking page date field.
   Exploit: Injecting XSS payloads that persisted and executed on subsequent page visits.
9. **IDOR on order_id**
   Exploit: Altering order_id values to view details of other users.

10. **XSS on Booking Page**
    Exploit: Injected payloads into the date parameter persisted as cookies, triggering on future visits.

Learnings

1 Understanding Web Application Vulnerabilities
Learned how common vulnerabilities such as XSS (Cross-Site Scripting), IDOR (Insecure Direct Object References), and Open URL Redirects can be exploited and their impact on web application security.

2 Use of Penetration Testing Tools
Gained proficiency in tools like Burp Suite for intercepting HTTP requests, analyzing server responses, and crafting malicious payloads for testing.

3 Parameter Manipulation
Discovered how attackers can manipulate parameters (e.g., act, order_id, f) to bypass security mechanisms or access unauthorized data.

4 Exploitation Techniques
Practiced crafting and injecting JavaScript payloads to exploit both reflected and stored XSS vulnerabilities, demonstrating their persistence and execution.