

Penetration Test Report: Fastfoodhackings

Report Information

Report Date: August 30, 2025

Target: Fastfoodhackings Application

URL: <https://www.bugbountytraining.com/fastfoodhackings/>

Status:  In Progress - Crawling for Endpoints Phase

Tester: Security Assessment Team

Table of Contents

1. [Executive Summary](#)
2. [Scope and Objectives](#)
3. [Vulnerability Findings](#)
 - [Vulnerability Summary](#)
 - [FFHK-001: Information Disclosure - Origin IP Address Exposed](#)
 - [FFHK-002: Information Disclosure - Sensitive Panels Indexed](#)
 - [FFHK-003: Cross-Site Scripting \(XSS\) Vulnerabilities](#)
 - [FFHK-004: Open Redirect Vulnerability](#)
4. [URL Enumeration Results](#)
5. [Next Steps](#)

Executive Summary

This report details the results of a penetration test conducted on the **Fastfoodhackings** web application. The assessment has progressed through initial reconnaissance, subdomain enumeration, and comprehensive URL discovery phases.

Key Findings

The assessment has identified **four significant vulnerabilities** across multiple severity levels:

Origin IP Address Exposure

The server's real IP address and its specific technology stack are exposed, allowing attackers to bypass Cloudflare security protections and customize attacks for the identified software.

Sensitive Page Indexing

Critical pages, including an administrative panel, have been indexed by Google and are publicly discoverable, providing a direct target for attackers.

Cross-Site Scripting (XSS) Vulnerabilities

Multiple XSS injection points discovered in the main application, allowing for client-side code execution and potential session hijacking.

Open Redirect Vulnerability

The application redirects users to external domains without proper validation, enabling phishing and credential theft attacks.

Current Status: Assessment is proceeding to Visual Reconnaissance phase for deeper application analysis.

Scope and Objectives

Primary Objective

The objective of this penetration test is to **identify security vulnerabilities** in the Fastfoodhackings application for educational and assessment purposes.

Test Scope

- **Target Application:** Fastfoodhackings
- **Primary URL:** <https://www.bugbountytraining.com/fastfoodhackings/>
- **Test Type:** Black-box Penetration Testing
- **Methodology:** OWASP Testing Guide

Limitations

- ⚠ Scope is **limited** to the application hosted at the specified URL
- 🎓 Test conducted for **educational purposes** exclusively

Vulnerability Findings

This section contains a detailed description of each identified vulnerability, its potential impact, and recommended remediation steps.

Vulnerability Summary

ID	Vulnerability	Severity	Status
FFHK-001	Information Disclosure - Origin IP Exposed	🟡 Medium	🟢 Active
FFHK-002	Information Disclosure - Sensitive Panels Indexed	🔴 High	🟢 Active
FFHK-003	Cross-Site Scripting (XSS) Vulnerabilities	🔴 High	🟢 Active
FFHK-004	Open Redirect Vulnerability	🔴 High	🟢 Active

FFHK-001: Information Disclosure - Origin IP Address Exposed

ID: FFHK-001

Severity: 🟡 Medium

Category: Information Disclosure

CVSS Score: 5.3 (AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N)

Description

A passive DNS enumeration check successfully identified the web server's origin IP address and the specific technologies it uses. The domain's DNS records point directly to this IP instead of being proxied through Cloudflare.

Technical Details

IDENTIFIED INFRASTRUCTURE:

- |— Hosting Provider: DigitalOcean (ASN 14061)
- |— DNS Provider: Cloudflare
- |— Origin IP Address: 134.209.18.185
- |— Technology Stack:
 - |— Web Server: Nginx
 - |— Operating System: Ubuntu
 - |— Frontend Libraries: Bootstrap, Popper, Ionicons

Impact

- **Protection Bypass:** Completely bypasses security protections offered by Cloudflare (WAF, DDoS mitigation)
- **Targeted Attacks:** Technology stack exposure allows attackers to research and implement specific exploits
- **Direct Access:** Enables direct server access, avoiding protection layers

Recommended Remediation

1. **Enable Cloudflare Proxy:** Enable Cloudflare proxy (the "orange cloud") for all relevant DNS records
2. **Restrict Direct Access:** Configure server to accept only traffic from Cloudflare IP ranges
3. **Minimize Exposure:** Reduce verbose headers and error messages that reveal underlying technologies

FFHK-002: Information Disclosure - Sensitive Panels Indexed

ID: FFHK-002

Severity: ● High

Category: Information Disclosure

CVSS Score: 7.5 (AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N)

Description

Google dorking techniques revealed that sensitive pages, including an administrative panel and a login page, are indexed by search engines. This allows attackers to bypass typical discovery phases and directly target high-value areas of the application.

Discovered URLs

INDEXED SENSITIVE PAGES:

- |— Admin Panel:
 - |— <https://www.bugbountytraining.com/challenges/AdminPanel/>
- |— Login Challenge:
 - |— <https://www.bugbountytraining.com/challenges/loginchallenge/>

Impact

- **Direct Target:** Publicly indexed administrative panels are prime targets for attacks
- **Effort Reduction:** Significantly reduces the effort needed to find critical entry points
- **Attack Vectors:** Facilitates brute force attacks, credential stuffing, and exploitation of panel-specific vulnerabilities

Recommended Remediation

Immediate Action:

1. **Implement Robust Authentication:** Ensure endpoints are not publicly accessible, implement proper authentication and authorization


Search Engine De-indexing: 2. **Google Search Console:** Request immediate removal of these URLs from the search index 3. **Prevent Re-indexing:**

```
# robots.txt
Disallow: /challenges/

# HTTP Header
X-Robots-Tag: noindex
```

FFHK-003: Cross-Site Scripting (XSS) Vulnerabilities

ID: FFHK-003

Severity:  High

Category: Cross-Site Scripting

CVSS Score: 8.8 (AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H)

Description

Multiple Cross-Site Scripting (XSS) vulnerabilities were identified in the FastFoodHackings application during URL enumeration. These vulnerabilities allow attackers to inject malicious JavaScript code that executes in victims' browsers.

Vulnerable Endpoints

```
XSS INJECTION POINT:
└─ index.php Parameter Injection:
    └─ https://www.bugbountytraining.com/fastfoodhackings/index.php?act=--
    %3E%3Cimg%20src=x%20onerror=alert(2) [200 OK]
```

Impact

- **Session Hijacking:** Steal authentication cookies and session tokens

- **Credential Theft:** Capture user credentials through fake forms
- **Malware Distribution:** Redirect users to malicious downloads
- **Data Exfiltration:** Access sensitive user information

Recommended Remediation

1. Input Sanitization:

```
// Example for index.php
$safe_input = htmlspecialchars($_GET['act'], ENT_QUOTES, 'UTF-8');

// Test URL: https://www.bugbountytraining.com/fastfoodhackings/index.php?
act=<script>alert('XSS')</script>
```

2. Content Security Policy:

```
Content-Security-Policy: default-src 'self'; script-src 'self'
```

3. **Output Encoding:** Properly encode all user-controlled data before rendering
4. **Parameter Validation:** Validate and sanitize all GET/POST parameters before processing

FFHK-004: Open Redirect Vulnerability

ID: FFHK-004

Severity:  High

Category: Open Redirect

CVSS Score: 7.4 (AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:N/A:N)

Description

The `go.php` endpoint accepts arbitrary URLs in the `returnUrl` parameter and redirects users to external domains without proper validation. This enables phishing attacks and credential theft.

Proof of Concept

```
CONFIRMED EXTERNAL REDIRECTS:
├─ https://www.bugbountytraining.com/fastfoodhackings/go.php
│   └─ ?returnUrl=https://batmanapollo.ru/ [302 Found]
├─ https://www.bugbountytraining.com/fastfoodhackings/go.php
│   └─ ?returnUrl=https://gysn.ru/ [302 Found]
└─ https://www.bugbountytraining.com/fastfoodhackings/go.php
    └─ ?returnUrl=https://www.windowsanddoors-r-us.co.uk/ [302 Found]
```

Impact

- **Phishing Attacks:** Redirect users to fake login pages
- **Malware Distribution:** Redirect to malicious file downloads
- **SEO Poisoning:** Abuse domain reputation for malicious redirects
- **Social Engineering:** Leverage trusted domain for malicious purposes

Recommended Remediation

1. URL Validation:

```
// Example validation for go.php
$allowed_domains = ['bugbountytraining.com'];
$parsed_url = parse_url($_GET['returnUrl']);
if (!in_array($parsed_url['host'], $allowed_domains)) {
    // Block redirect - Test with:
    // https://www.bugbountytraining.com/fastfoodhackings/go.php?
    returnUrl=https://malicious.com
}
```

- 2. **Whitelist Approach:** Only allow predefined redirect destinations
- 3. **User Confirmation:** Display warning for external redirects

URL Enumeration Results

Discovery Summary

During the comprehensive URL enumeration phase using Dirsearch, the following attack surface was mapped:

Category	Count	Key Findings
Accessible Endpoints	67+	Main application and API endpoints
Redirect Responses	20+	HTTPS enforcement and application redirects
Missing Resources	25+	Potential for information gathering
Challenge Applications	16+	Additional testing targets discovered

Key Endpoints Discovered

Main Application

- /fastfoodhackings/index.php - Main entry point (XSS vulnerable)
- /fastfoodhackings/menu.php - Menu functionality
- /fastfoodhackings/locations.php - Location services
- /fastfoodhackings/book.php - Booking system

API Endpoints

- /fastfoodhackings/api/book.php - Booking API
- /fastfoodhackings/api/invites.php - Invitation system

Administrative Areas

- [/challenges/AdminPanel/](#) - Administrative interface
- [/challenges/loginchallenge/](#) - Login testing area
- [/dev/](#) - Development directory (301 redirect)

Technology Stack Confirmed

- **Web Server:** Nginx on Ubuntu
- **Application:** PHP-based
- **Frontend:** Bootstrap, Ionicons, Google Fonts API
- **Server IP:** 134.209.18.185 (DigitalOcean)

Next Steps

Pending Actions


Completed Phases

- ☒ 1. SUBDOMAIN ENUMERATION
- ☒ 2. PORT SCANNING
- ☒ 3. DIRECTORY ENUMERATION
- ☒ 4. PARAMETER DISCOVERY
- ☒ 5. WAYBACK MACHINE
- ☒ 6. COMBINING & DE-DUPLICATING URLS

Completed Phase

- ☒ 7. VISUAL RECONNAISSANCE

Current Phase

- ☐ 8. CRAWLING FOR ENDPOINTS  IN PROGRESS
- ☐ 9. FINDING SECRETS IN JAVASCRIPT FILES
- ☐ 10. NETWORK & SERVICE SCANNING
- ☐ 11. ENDPOINT & PARAMETER DISCOVERY
- ☐ 12. CMS DETECTION & SCANNING

Upcoming Phases - Active Reconnaissance

- ☐ 13. AUTOMATED VULNERABILITY SCANNING
- ☐ 14. SQL INJECTION TESTING
- ☐ 15. CROSS-SITE SCRIPTING (XSS) TESTING
- ☐ 16. SPECIALIZED VULNERABILITY TESTING

Upcoming Phases - Post-Discovery

- ☐ 17. FINDING PUBLIC EXPLOITS
- ☐ 18. PAYLOAD TESTING & VALIDATION

Validation and Reports

- ☐ **Verify fixes** for identified vulnerabilities
- ☐ **Execute regression testing**
- ☐ **Document new discoveries**
- ☐ **Update risk classifications**

Next Phases

1. **Visual Reconnaissance:** Complete manual application review and screenshot analysis using EyeWitness/aquatone
2. **Endpoint Crawling:** Discover additional functionality through automated crawling with katana
3. **JavaScript Analysis:** Extract secrets and API endpoints from client-side code using linkfinder, gf, and SecretFinder
4. **Network & Service Scanning:** Identify additional network services and potential attack vectors using nmap and masscan
5. **Parameter Discovery:** Use paramspider and arjun to discover hidden parameters and endpoints
6. **CMS Detection:** Identify and scan content management systems with CMSeeK and wpscan
7. **Automated Vulnerability Scanning:** Deploy Nuclei templates and Nikto for comprehensive vulnerability detection
8. **SQL Injection Testing:** Test identified parameters for SQL injection vulnerabilities using sqlmap
9. **XSS Testing:** Systematic cross-site scripting testing using Dalfox and XSSStrike
10. **Specialized Testing:** File upload testing (Fuxploader), S3 bucket enumeration (AWSBucketDump), Git repository discovery (GitDumper)
11. **Exploit Research:** Search for public exploits using searchsploit for identified software versions
12. **Payload Validation:** Test and validate XSS payloads and other injection techniques
13. **Impact Analysis:** Evaluate the combined impact of vulnerabilities and exploit chaining potential
14. **Final Report:** Prepare comprehensive executive report with remediation priorities

Contacts

For questions about this report:

- **Email:** security-team@example.com
- **Next Update Date:** [TBD]

⚠ Legal Notice: This document contains confidential information and must be handled according to the organization's security policies.