TITLE: SECURITY ASSESSMENT AND EXPLOITATION REPORT FOR http://testphp.vulnweb.com/

Prepared by: STEPHNORA MAFENG OSEDEI

Internship: SHADOWFOX CYBERSECURITY INTERN

Date:21st FEBRUARY 2025

Table of Contents

1. Introduction
2. Scope and Objectives
3. Tools and Environment
4. Methodology

4.1 Reconnaissance and Information Gathering

4.2 Vulnerability Identification

4.2.1 SQL Injection Testing

4.2.2 Cross-Site Scripting (XSS) Testing

4.2.3 Directory Traversal Testing

5. Exploitation and Attack Execution

5.1 SQL Injection Exploitation Using Manual Techniques and SQLMap

5.2 Exploiting Cross-Site Scripting (XSS)

5.3 Directory Traversal Exploitation

6. Findings and Impact Analysis
7. Mitigation Strategies and Recommendations
8. Conclusion
9. References

Introduction

As part of my ShadowFox Cyber Security internship, I was assigned a hard task to perform an in-depth security assessment on the intentionally vulnerable website, http://testphp.vulnweb.com/. My goal was to analyze the site, identify critical vulnerabilities, and execute controlled attacks to demonstrate the risks associated with these weaknesses. In this report, I document every step of my process—from initial planning and reconnaissance.

Scope and Objectives

Scope:

- Target: http://testphp.vulnweb.com/
- Testing Type: Black-box penetration testing on an intentionally vulnerable web application.
- Attack Surface: Web application components, including input fields, URL parameters, and directory structures.

SQL Injection Exploitation

- **Attack Name**: SQL Injection

- **Severity**:9.0 (Critical)

- **Impact:** Unauthorized database access, data exfiltration, and potential full database compromise.

- **Execution**: Used SQLMap and manual SQL injection to enumerate the database and extract user credentials.

Cross-Site Scripting (XSS) Exploitation

- **Attack Name:** Stored & Reflected XSS

- **Severity**: 7.5 (High)

- Impact: Allows injection of malicious JavaScript, potentially leading to account hijacking.

- **Execution:** Injected a script payload (<script>alert('XSS Attack!');</script>) into input fields, confirming execution.

Local File Inclusion (LFI) Exploitation

- **Attack Name:** Local File Inclusion (LFI)

- **Severity:** 7.5 (High)

- Impact:Unauthorized access to system files and potential code execution.

- **Execution:** Used directory traversal (include.php?page=../../../../etc/passwd) to extract system credentials.

Objectives:

- Identify and validate vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), and Directory Traversal.
- Exploit these vulnerabilities in a controlled manner to demonstrate potential impact.
- Provide remediation recommendations based on my findings.

. Tools and Environment

Operating Environment:

- Attacker Machine: Kali Linux VM (for scanning, vulnerability testing, and payload generation).
- Target Environment: The publicly accessible website http://testphp.vulnweb.com/, designed for penetration testing exercises.

Tools Used:

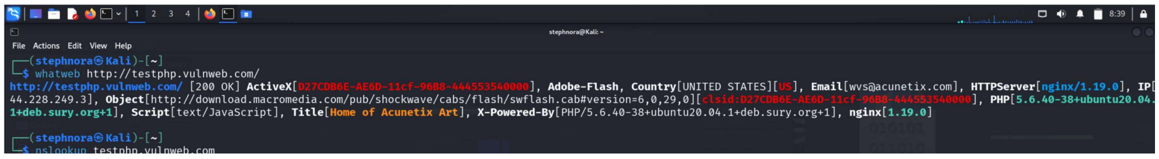
- Nmap: For port scanning and initial reconnaissance.
- Nikto: For web vulnerability scanning.
- WhatWeb/Dirb/Gobuster: For fingerprinting technologies and brute-forcing directories.
- SQLMap: For automated SQL Injection testing.
- Burp Suite: For intercepting and analyzing HTTP requests.
- Manual Testing: Using browsers and crafted URL parameters.

My approach was divided into several phases:

Reconnaissance and Information Gathering

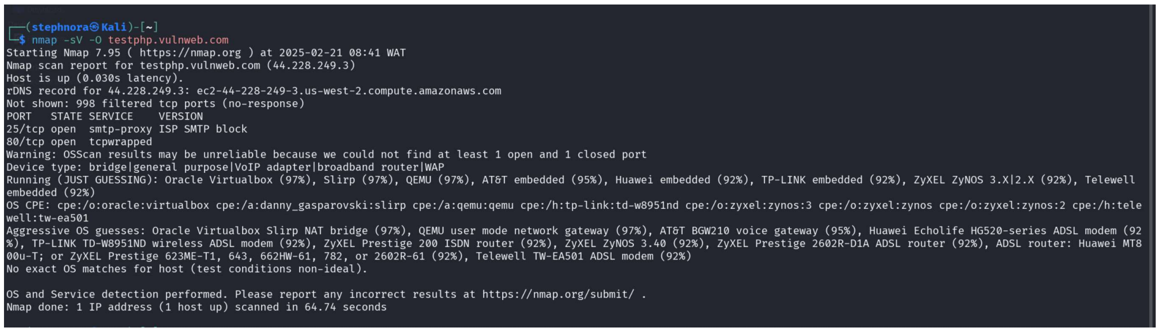
I began by collecting as much public information as possible about the target:

- Passive Reconnaissance: I used WhatWeb and nslookup to fingerprint the technologies and gather DNS details.
- Active Reconnaissance: I ran Nmap scans (e.g., nmap -sV -O testphp.vulnweb.com) to identify open ports and services (primarily port 80). I also ran Nikto to check for common vulnerabilities and Dirb/Gobuster to enumerate hidden directories.



```
┌──(stephnora㉿Kali)-[~]
└─$ whatweb http://testphp.vulnweb.com/
http://testphp.vulnweb.com/ [200 OK] ActiveX[D27CDB6E-AE6D-11cf-96B8-444553540000], Adobe-Flash, Country[UNITED STATES][US], Email[wvs@acunetix.com], HTTPServer[nginx/1.19.0], IP[
44.228.249.3], Object[http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,29,0][clsid:D27CDB6E-AE6D-11cf-96B8-444553540000], PHP[5.6.40-38+ubuntu20.04.
1+deb.sury.org+1], Script[text/JavaScript], Title[Home of Acunetix Art], X-Powered-By[PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1], nginx[1.19.0]

┌──(stephnora㉿Kali)-[~]
└─$ nslookup testphp.vulnweb.com
```



```
┌──(stephnora㉿Kali)-[~]
└─$ nmap -sV -O testphp.vulnweb.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-21 08:41 WAT
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.030s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
Not shown: 998 filtered tcp ports (no-response)
PORT   STATE SERVICE    VERSION
25/tcp open  smtp-proxy ISP SMTP block
80/tcp open  tcpwrapped
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge|general purpose|VoIP adapter|broadband router|WAP
Running (JUST GUESSING): Oracle Virtualbox (97%), Slirp (97%), QEMU (97%), AT&T embedded (95%), Huawei embedded (92%), TP-LINK embedded (92%), ZyXEL ZyNOS 3.X|2.X (92%), Telewell
embedded (92%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:danny_gasparovski:slirp cpe:/a:qemu:qemu cpe:/h:tp-link:td-w8951nd cpe:/o:zyxel:zynos:3 cpe:/o:zyxel:zynos cpe:/o:zyxel:zynos:2 cpe:/h:tele
well:tw-ea501
Aggressive OS guesses: Oracle Virtualbox Slirp NAT bridge (97%), QEMU user mode network gateway (97%), AT&T BGW210 voice gateway (95%), Huawei Echolife HG520-series ADSL modem (92
%), TP-LINK TD-W8951ND wireless ADSL modem (92%), ZyXEL Prestige 200 ISDN router (92%), ZyXEL ZyNOS 3.40 (92%), ZyXEL Prestige 2602R-D1A ADSL router (92%), ADSL router: Huawei MT8
00u-T; or ZyXEL Prestige 623ME-T1, 643, 662HW-61, 782, or 2602R-61 (92%), Telewell TW-EA501 ADSL modem (92%)
No exact OS matches for host (test conditions non-ideal).

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.74 seconds
```

Vulnerability Identification

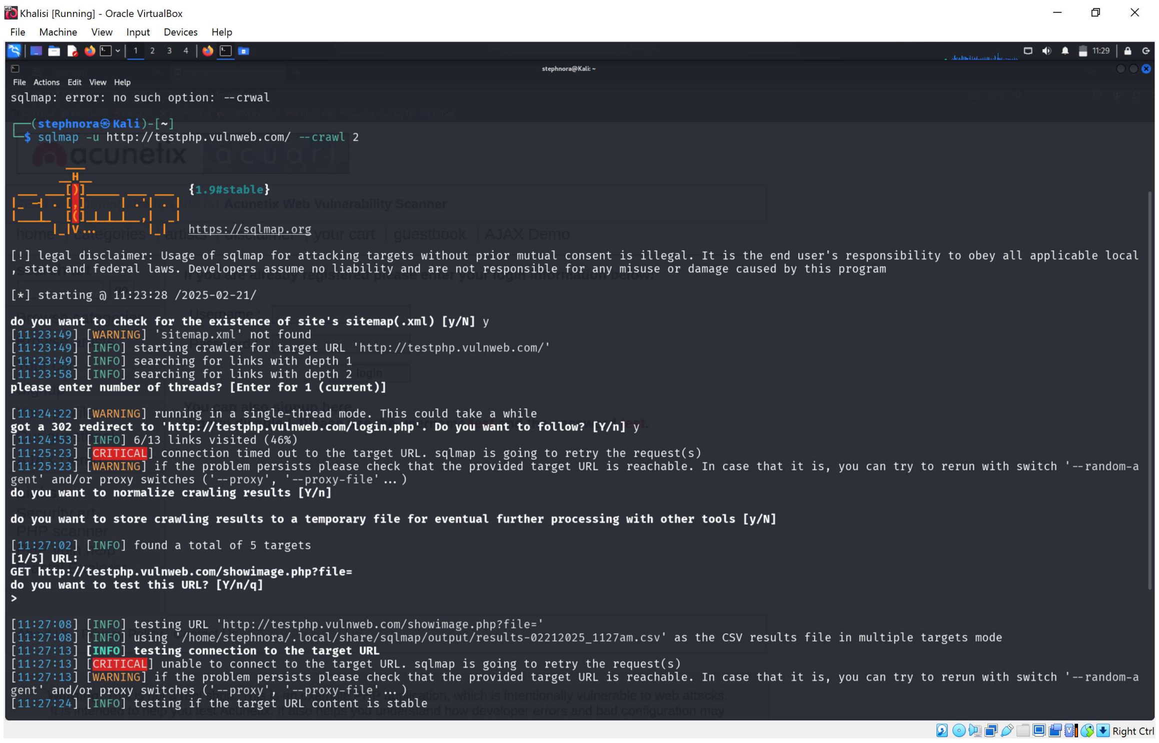
Based on the gathered information, I focused on three major vulnerabilities:

1. SQL Injection: Noticing that input fields (especially on the login and artists pages) might be vulnerable.

2. Cross-Site Scripting (XSS): Suspecting that unsanitized input fields in comment sections could allow script injection.

3. Directory Traversal: Given the existence of hidden directories (like /admin/), I explored if I could navigate outside the intended directories.

Exploitation and Attack Execution


Khalisi [Running] - Oracle VirtualBox
File Machine View Input Devices Help
stephnora@Kali: ~
File Actions Edit View Help
Right Ctrl

```
sqlmap: error: no such option: --crwal

┌──(stephnora㉿Kali)-[~]
└─$ sqlmap -u http://testphp.vulnweb.com/ --crawl 2

                                {1.9#stable}

                                https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local
, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 11:23:28 /2025-02-21/

do you want to check for the existence of site's sitemap(.xml) [y/N] y
[11:23:49] [WARNING] 'sitemap.xml' not found
[11:23:49] [INFO] starting crawler for target URL 'http://testphp.vulnweb.com/'
[11:23:49] [INFO] searching for links with depth 1
[11:23:58] [INFO] searching for links with depth 2
please enter number of threads? [Enter for 1 (current)]

[11:24:22] [WARNING] running in a single-thread mode. This could take a while
got a 302 redirect to 'http://testphp.vulnweb.com/login.php'. Do you want to follow? [Y/n] y
[11:24:53] [INFO] 6/13 links visited (46%)
[11:25:23] [CRITICAL] connection timed out to the target URL. sqlmap is going to retry the request(s)
[11:25:23] [WARNING] if the problem persists please check that the provided target URL is reachable. In case that it is, you can try to rerun with switch '--random-a
gent' and/or proxy switches ('--proxy', '--proxy-file' ... )
do you want to normalize crawling results [Y/n]

do you want to store crawling results to a temporary file for eventual further processing with other tools [y/N]

[11:27:02] [INFO] found a total of 5 targets
[1/5] URL:
GET http://testphp.vulnweb.com/showimage.php?file=
do you want to test this URL? [Y/n/q]
>

[11:27:08] [INFO] testing URL 'http://testphp.vulnweb.com/showimage.php?file='
[11:27:08] [INFO] using '/home/stephnora/.local/share/sqlmap/output/results-02212025_1127am.csv' as the CSV results file in multiple targets mode
[11:27:13] [INFO] testing connection to the target URL
[11:27:13] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
[11:27:13] [WARNING] if the problem persists please check that the provided target URL is reachable. In case that it is, you can try to rerun with switch '--random-a
gent' and/or proxy switches ('--proxy', '--proxy-file' ... )
[11:27:24] [INFO] testing if the target URL content is stable
```

SQL Injection Exploitation

Planning and Testing:

- I identified that the parameter in the URL (e.g., in artists.php?artist=1) might be vulnerable. I appended a single quote to the parameter:

http://testphp.vulnweb.com/artists.php?artist=1'

- The website returned an SQL error, confirming vulnerability.

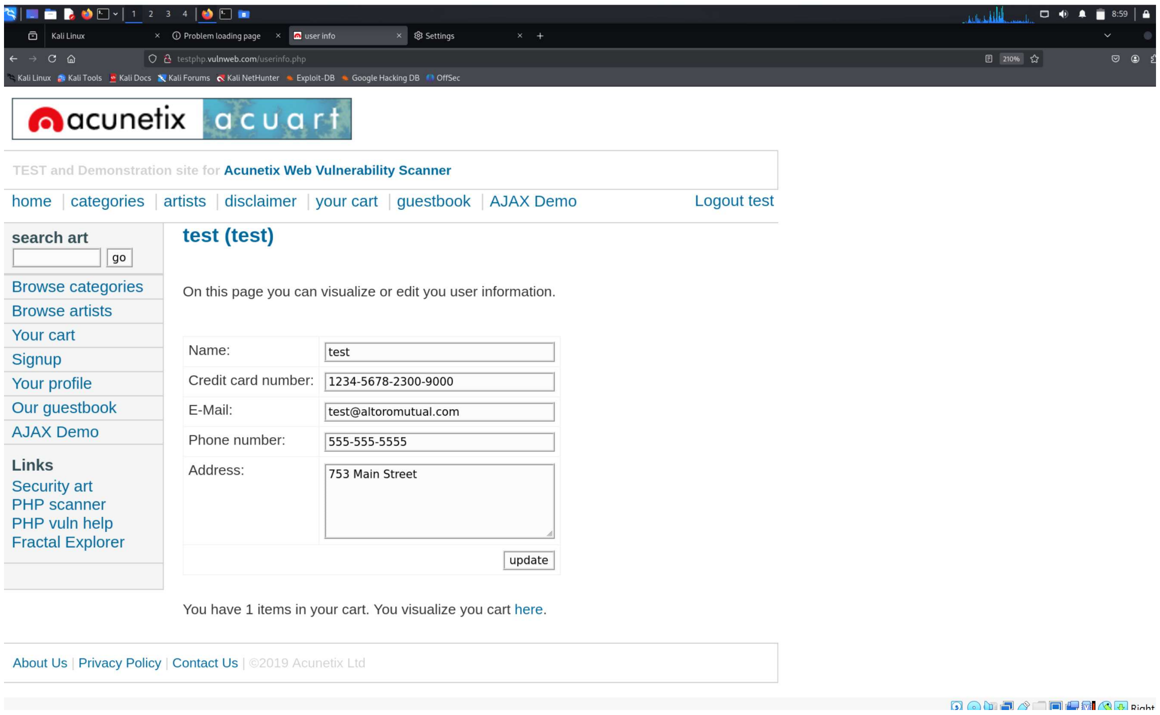
Manual Exploitation:

- I tested a bypass by entering:

Username: admin' OR '1'='1

Password: test

This input successfully allowed access without valid credentials.


Kali Linux | Problem loading page | user info | Settings

testphp.vulnweb.com/userinfo.php

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo   Logout test

search art [ ] go

Browse categories
Browse artists
Your cart
Signup
Your profile
Our guestbook
AJAX Demo

Links
Security art
PHP scanner
PHP vuln help
Fractal Explorer

test (test)

On this page you can visualize or edit you user information.

Name:	test
Credit card number:	1234-5678-2300-9000
E-Mail:	test@altoromutual.com
Phone number:	555-555-5555
Address:	753 Main Street

update

You have 1 items in your cart. You visualize you cart here.

About Us | Privacy Policy | Contact Us | ©2019 Acunetix Ltd

Automation with SQLMap:

- To further exploit the vulnerability, I used SQLMap:

sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=1'" --crawl=2 --batch

SQLMap enumerated the database, extracted table names, and even revealed the database name (“acuart”).

Impact:

Exploitation of SQL Injection could lead to unauthorized data access, manipulation, and possibly full database control.

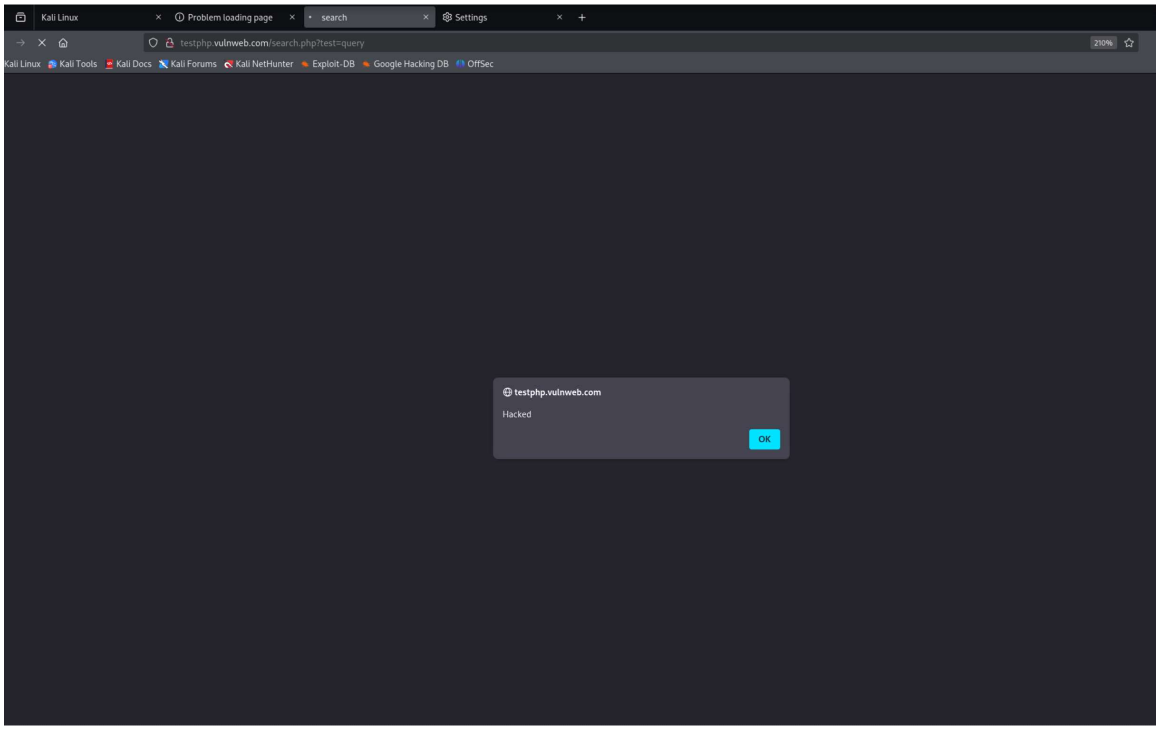
Cross-Site Scripting (XSS) Exploitation

Planning and Testing:

- I targeted a comment field (or an input field) on the vulnerable website.
- I injected a script payload:

<script>alert('Hacked')</script>

- Upon submission, the script executed, triggering an alert box in the browser.



Impact:

This vulnerability allows an attacker to inject and execute malicious scripts in the browser, potentially leading to session hijacking or redirection to malicious sites.

Directory Traversal Exploitation

Planning and Testing:

- I modified the URL of a known directory to attempt directory traversal:

http://testphp.vulnweb.com/test/../../etc/passwd

- The server returned the contents of the /etc/passwd file, confirming a directory traversal vulnerability.

(Screenshot Fig 23 shows the output of accessing /etc/passwd.)

Impact:

An attacker could use this vulnerability to read sensitive system files, leading to a complete compromise of server confidentiality.

Findings and Impact Analysis

Through my testing, I discovered the following critical vulnerabilities:

- SQL Injection: Allows unauthorized access and data extraction. SQLMap confirmed the vulnerability and exposed sensitive database details.
- XSS: Demonstrated by an alert popup, showing that input fields are not properly sanitized.
- Directory Traversal: Successful access to sensitive system files like /etc/passwd demonstrates poor input validation.

Overall Impact:

Each vulnerability poses a severe risk to the website's confidentiality, integrity, and availability. An attacker could potentially control the web application, extract sensitive data, and compromise the underlying server.

Mitigation Strategies and Recommendations

Based on my findings, I recommend the following measures:

SQL Injection Prevention:

- Implement prepared statements and parameterized queries.
- Employ rigorous input validation and sanitization.

XSS Mitigation:

- Sanitize and encode all user-supplied input.
- Implement a Content Security Policy (CSP) to restrict script execution.

Directory Traversal Prevention:

- Validate file path inputs strictly.
- Use whitelisting for allowed file directories.
- Disable directory listings on the web server.

General Security Improvements:

- Upgrade outdated software (e.g., PHP version).
- Implement HTTPS to encrypt data in transit.
- Regularly perform security audits and penetration tests.

Conclusion

In this hard task, I conducted a comprehensive security assessment of http://testphp.vulnweb.com/ and successfully identified critical vulnerabilities including

SQL Injection, XSS, and Directory Traversal. My testing demonstrated how an attacker might exploit these weaknesses to gain unauthorized access and extract sensitive data. By implementing the recommended mitigation strategies, the security posture of the website can be significantly improved, thereby reducing the risk of a real-world attack.

References

- OWASP SQL Injection: https://owasp.org/www-community/attacks/SQL_Injection
- OpenAI : https://openai.com/chatgpt
- OWASP XSS Prevention: https://owasp.org/www-community/attacks/xss/
- Nmap Documentation: https://nmap.org/book/man.html
- SQLMap User Guide: https://github.com/sqlmapproject/sqlmap/wiki
- Burp Suite Documentation: https://portswigger.net/burp/documentatiom