



**InternsElite**  
**MAJOR PROJECT**  
**Cyber Security program**

**Name:- Dharmendra Kumar**  
**Program Name:-CYBER SECURITY PROGRAM**  
**Roll no:- 24/CS/J1456**  
**Submission Date:- 21/07/2024**

Dharmendra kumar

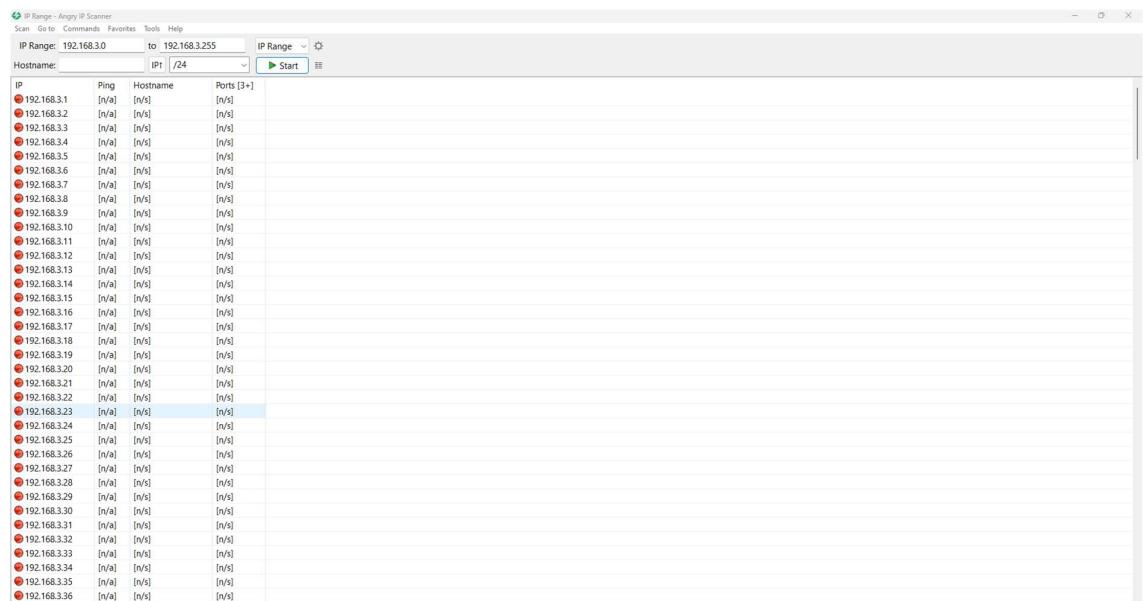
# CYBER SECURITY – MAJOR PROJECT

## **TASK:-1 Use Angry IP Scanner for Network Scanning.**

Network scanning refers to the process of identifying active devices on a network and gathering information about them. This activity is crucial for network administrators to manage and secure their networks effectively.

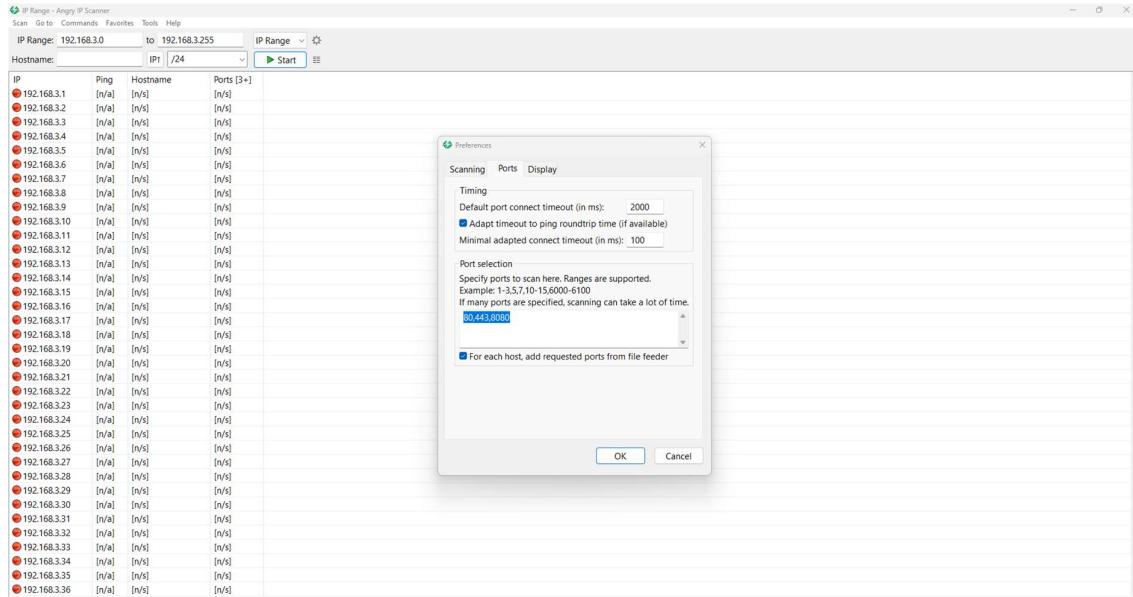
**Angry IP Scanner:** A lightweight tool for fast IP address and port scanning.

**Step:- 1** Firstly, I use a Network IP address and set an IP range in the network, then I set the netmask. then click to start button.



The second step is to click on "Ports" and then navigate to "Ports Preferences."

Then If many ports are specified, scanning can take a lot of time



Finally, I conduct a scan of all ports.

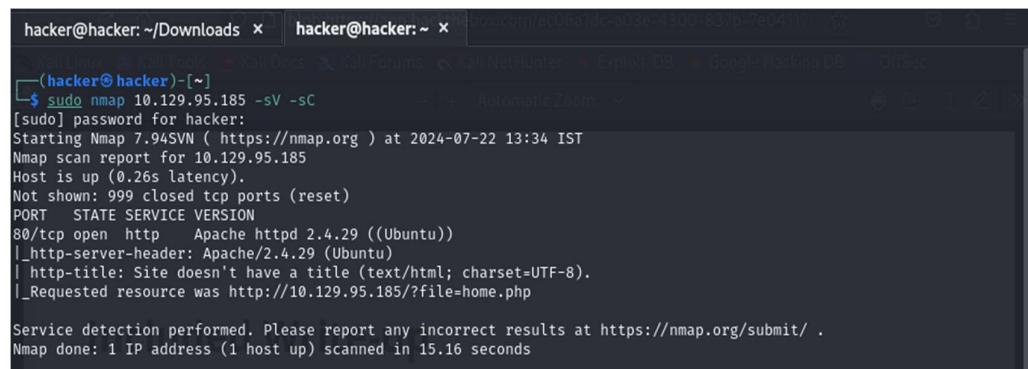
The first task is end

## **TASK:-2 Use Nmap/Zenmap Tool for Network Reconnaissance**

One of the most renowned open-source network security scanning tools among pentesters is Network Mapper (Nmap). Nmap, a command-line utility, leverages multiple network protocols and advanced functionalities to scan hosts for open TCP and UDP ports, identify operating systems, extract service banners, and perform other detailed assessments. Enumeration involves actively establishing connections to target systems to uncover potential attack vectors, including discovering hosts, services, domains, URLs, and valid user accounts for potential exploitation.

### **Step:-1 For Examples :-**

This command, sudo nmap 10.129.95.185 -sV -sC , conducts a comprehensive scan that identifies live hosts within the specified IP range, while also performing service version detection (-sV) and running default scripts (-sC).

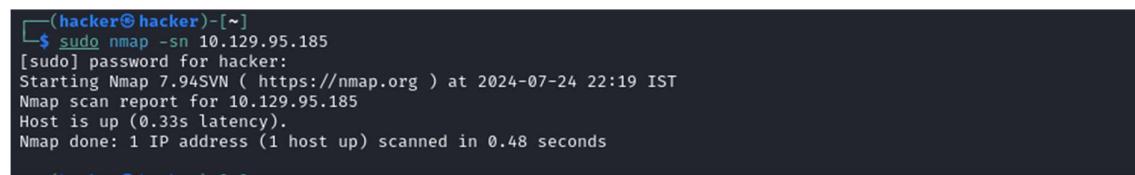


```
hacker@hacker:~/Downloads x  hacker@hacker:~ x [root] kali-tools kali-tools kali-distro kali-forums kali-netHunter exploit-db google-hacking-db offset
└── (hacker㉿hacker)-[~]
$ sudo nmap 10.129.95.185 -sV -sC
[sudo] password for hacker:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-22 13:34 IST
Nmap scan report for 10.129.95.185
Host is up (0.26s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http   Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
| http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_Requested resource was http://10.129.95.185/?file=home.php

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.16 seconds
```

Then the next uses is

The command sudo nmap -sn 10.129.95.185 is shortened to perform a ping scan (-sn) on the IP address 10.129.95.185. This scan is used to identify which hosts are up on the network without performing detailed port scanning.



```
└── (hacker㉿hacker)-[~]
$ sudo nmap -sn 10.129.95.185
[sudo] password for hacker:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-24 22:19 IST
Nmap scan report for 10.129.95.185
Host is up (0.33s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.48 seconds
```

Then the next use is

The command sudo nmap -sF 10.129.95.185 is used to conduct a FIN scan (-sF) on the IP address 10.129.95.185.

```
(hacker㉿hacker)-[~]
$ sudo nmap -sF 10.129.95.185
[sudo] password for hacker:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-24 22:19 IST
Nmap scan report for 10.129.95.185
Host is up (0.24s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE            SERVICE
80/tcp    open|filtered  http

Nmap done: 1 IP address (1 host up) scanned in 9.72 seconds
```

Then the next use is

The command sudo nmap -sU 10.129.95.185 is used to perform a UDP scan on the specified IP address (10.129.95.185)

```
(hacker㉿hacker)-[~]
$ sudo nmap -sU 10.129.95.185
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-24 22:18 IST
Nmap scan report for 10.129.95.185
Host is up (0.32s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT      STATE            SERVICE
68/udp   open|filtered  dhcpc
69/udp   open|filtered  tftp

Nmap done: 1 IP address (1 host up) scanned in 1022.78 seconds
```

Then the different command is nmap tool like as uses :-

```
$ nmap -h
Nmap 7.94SVN ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] [target specification]
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, 192.168.0.1, 10.0.0-255.1-254
  -l file: read targets from file
  -iR <num hosts>: Choose random targets
  -e <exclude>: <host1[,host2[,host3]]...>; Exclude hosts/networks
  -r <exclude_file>: <Exclude list from file>
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sP: Ping Scan - disable port scan
  -sH: Host Discovery - skip host discovery
  -PS/PV/VU/V[erbose]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[rotocol list]: IP Protocol Ping
  -A: Aggressive mode - performs OS detection/Allexploit resolve [default: sometimes]
  -n: no DNS resolution - specify custom DNS servers
  -sS: Use OS's DNS resolver
  -T: Trace hop path to each host
SCAN METHODS:
  -sS/-sT/-sW/-sM: TCP SYN/Connect() /ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/-sF/-sK: TCP Null, FIN, and Xmas scans
  -sM: Monte Carlo scan: Customized TCP scan Flags
  -sT: TCP connect(2) scan: Idle scan
  -sV/-sZ: SCTP INIT/COOKIE-ECHO scan
  -sO: IP protocol scan
  -sB: TCP SYN/ACK bounces scan
PORT SPECIFICATION AND SCAN ORDER:
  -p<port ranges>: Only specify ports
  Ex: -p22,-55555-55556,U:53,113,T:21-25,80,139,8000,S:59
  -r<port range>: Randomize ports from specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports sequentially - don't randomize
  -top-ports <numbers>: Scan <numbers> most common ports
  -script-args <args>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  -version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  -version-all: Try every single probe (intensity 9)
  -version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
  -sC: equivalent to --script=<default>
  -script<lua scripts>: <lua scripts> is a comma separated list of
  directories, script-files or script-categories
  -script-args=<n>=v1,[n2=v2,...]: provide arguments to scripts
  -script-args-file=<filename>: provide NSE script args in a file
```

```

OS DETECTION:
  -O: Enable OS detection
  --oscan-limit: Limit OS detection to promising targets
  --oscan-guess: Guess OS more aggressively

TIMING AND PERFORMANCE:
  Options which take <time> are in seconds, or append 'ms' (milliseconds),
  's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
  -T<0-5>: Set timing template (higher is faster)
  --min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
  --min-parallelism/max-parallelism <numprobes>: Probe parallelization
  --min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>: Specifies
    probe round trip time.
  --max-retries <tries>: Caps number of port scan probe retransmissions.
  --host-timeout <time>: Give up on target after this long
  --scan-delay/-max-scan-delay <time>: Adjust delay between probes
  --min-rate <number>: Send packets no slower than <number> per second
  --max-rate <number>: Send packets no faster than <number> per second

FIREWALL/IDS EVASION AND SPOOFING:
  -f; --mtu <val>: fragment packets (optionally w/given MTU)
  -D <decoy1[,decoy2[,ME], ...]>: Cloak a scan with decoys
  -S <IP_Address>: Spoof source address
  -e <iface>: Use specified interface
  -g/<source-port <portnum>: Use given port number
  --proxies <url1,[url2], ...>: Relay connections through HTTP/SOCKS4 proxies
  --data <hex string>: Append a custom payload to sent packets
  --data-string <string>: Append a custom ASCII string to sent packets
  --data-length <num>: Append random data to sent packets
  --ip-options <options>: Send packets with specified ip options
  --ttl <val>: Set IP time-to-live field
  --spoof-mac <mac address/prefix/vendor name>: Spoof your MAC address
  --badsum: Send packets with a bogus TCP/UDP/SCTP checksum

OUTPUT:
  -oN/-oX/-oS/-oG <file>: Output scan in normal, XML, slrIpt KIddi3,
    and Grepable format, respectively, to the given filename.
  -oA <basename>: Output in the three major formats at once
  -v: Increase verbosity level (use -vv or more for greater effect)
  -d: Increase debugging level (use -dd or more for greater effect)
  --reason: Display the reason a port is in a particular state
  --open: Only show open (or possibly open) ports
  --packet-trace: Show all packets sent and received
  --iflist: Print host interfaces and routes (for debugging)
  --append-output: Append to rather than clobber specified output files
  --resume <filename>: Resume an aborted scan
  --noninteractive: Disable runtime interactions via keyboard
  --stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML
  --webxml: Reference stylesheet from Nmap.Org for more portable XML
  --no-stylesheet: Prevent associating of XSL stylesheet w/XML output

MISC:
  -6: Enable IPv6 scanning
  -A: Enable OS detection, version detection, script scanning, and traceroute
  --datadir <dirname>: Specify custom Nmap data file location

```

```

MISC:
  -6: Enable IPv6 scanning
  -A: Enable OS detection, version detection, script scanning, and traceroute
  --datadir <dirname>: Specify custom Nmap data file location
  --send-eth/--send-ip: Send using raw ethernet frames or IP packets
  --privileged: Assume that the user is fully privileged
  --unprivileged: Assume the user lacks raw socket privileges
  -V: Print version number
  -h: Print this help summary page.

EXAMPLES:
  nmap -v -A scanme.nmap.org
  nmap -v -sn 192.168.0.0/16 10.0.0.0/8
  nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES

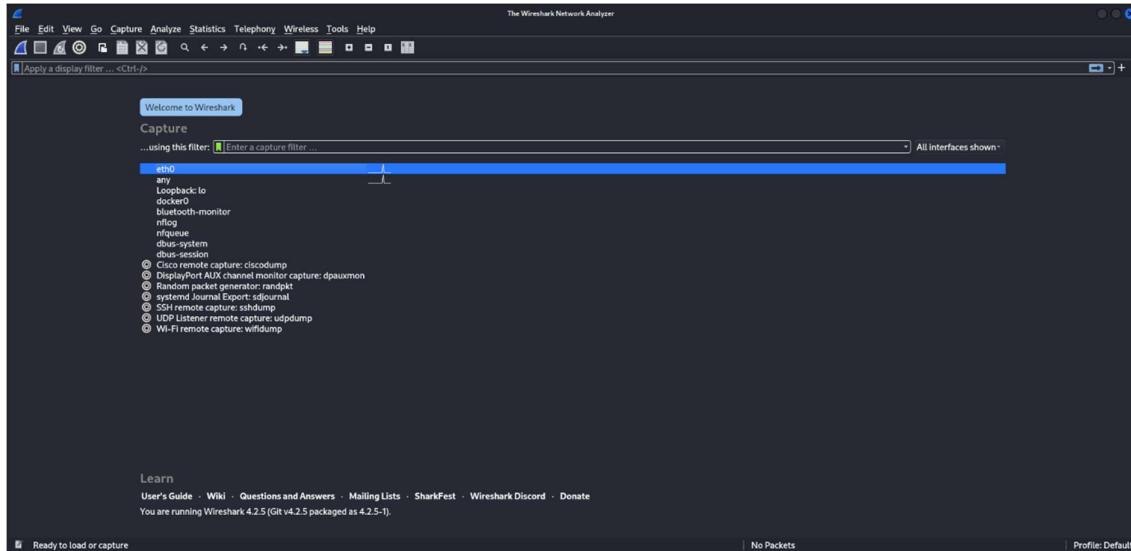
```

## The Second Task is end

**TASK:-3 Perform Sniffing for capturing the Passwords by wireshark**

I am going to wireshark tool

**Step:-1 After opening Wireshark, the interface appears as follows:**



**Step:- 2 I am see the internet IP and then copying it to the internet IP." and then go to wireshark.**

```
hacker@hacker: ~
File Actions Edit View Help
(hacker@hacker)-[~]
$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:1e:31:a4:aa txqueuelen 0 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.3.78 netmask 255.255.255.0 broadcast 192.168.3.255
        inet6 2409:40e2:1022:7f90::d4f:5122:9df5:0fb prefixlen 64 scopeid 0x0<global>
        inet6 2409:40e2:1022:7f90::a00:27ff:fea2:9fde prefixlen 64 scopeid 0x0<global>
        inet6 fe80::a00:27ff:fea2:9fde prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:a2:9f:de txqueuelen 1000 (Ethernet)
        RX packets 149 bytes 37333 (36.4 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 182 bytes 33268 (32.4 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 90 bytes 4580 (4.4 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 90 bytes 4580 (4.4 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

**Step:-3** I am going to the website, filling in the username and password, and then clicking the login button. And Then go to the wireshark.

**Step:-4** Then, filter for HTTP and proceed to the POST method in Hypertext Transfer Protocol.

And then click to the HTML Form URL Encoded.

The Third Task is End

#### TASK:-4 Perform Data Auditing with Hashdeep.

Then follow the steps

```
root@hacker: ~/hashdeep
File Actions Edit View Help
└─(root@hacker)-[~]
  └─# cd hashdeep
    └─(root@hacker)-[~/hashdeep]
      └─# ls
        └─(root@hacker)-[~/hashdeep]
          └─# cat >>test1.txt
            testing file A
            ^Z
            zsh: suspended  cat >> test1.txt
      └─(root@hacker)-[~/hashdeep]
        └─# cat >>test2.txt
          testing file B
          ^Z
          zsh: suspended  cat >> test2.txt
    └─(root@hacker)-[~/hashdeep]
      └─# cat *
        testing file A
        testing file B
    └─(root@hacker)-[~/hashdeep]
      └─# hashdeep --help
hashdeep version 4.4 by Jesse Kornblum and Simson Garfinkel.
$ hashdeep [OPTION] ... [FILES] ...
-c <alg1,[alg2]> - Compute hashes only. Defaults are MD5 and SHA-256
                  legal values: md5,sha1,sha256,tiger,whirlpool,
-p <size> - piecewise mode. Files are broken into blocks for hashing
-r - recursive mode. All subdirectories are traversed
-d - output in DFXML (Digital Forensics XML)
-k <file> - add a file of known hashes
-a - audit mode. Validates FILES against known hashes. Requires -k
-m - matching mode. Requires -k
-x - negative matching mode. Requires -k
-w - in -m mode, displays which known file was matched
-M and -X act like -m and -x, but display hashes of matching files
-e - compute estimated time remaining for each file
-s - silent mode. Suppress all error messages
-b - prints only the bare name of files; all path information is omitted
-l - print relative paths for filenames
-i/-I - only process files smaller than the given threshold
-o - only process certain types of files. See README/manpage
-v - verbose mode. Use again to be more verbose
-d - output in DFXML; -W FILE - write to FILE.
-j <num> - use num threads (default 4)
hashdeep: option requires an argument -- 'p'
Try `hashdeep -h` for more information.
```

```
[root@hacker]~[~/dharm2]
# hashdeep test1.txt
%%%% HASHDEEP-1.0
%%% size,md5,sha256,filename
## Invoked from: /root/dharm2
## # hashdeep test1.txt
##
14,997a9f00e25f85b73ffb1898376b7186,76d793c71e4f153583c2687898468b4068db96782fcc793a2282abdf684a52c3,/root/dharm2/test1.txt

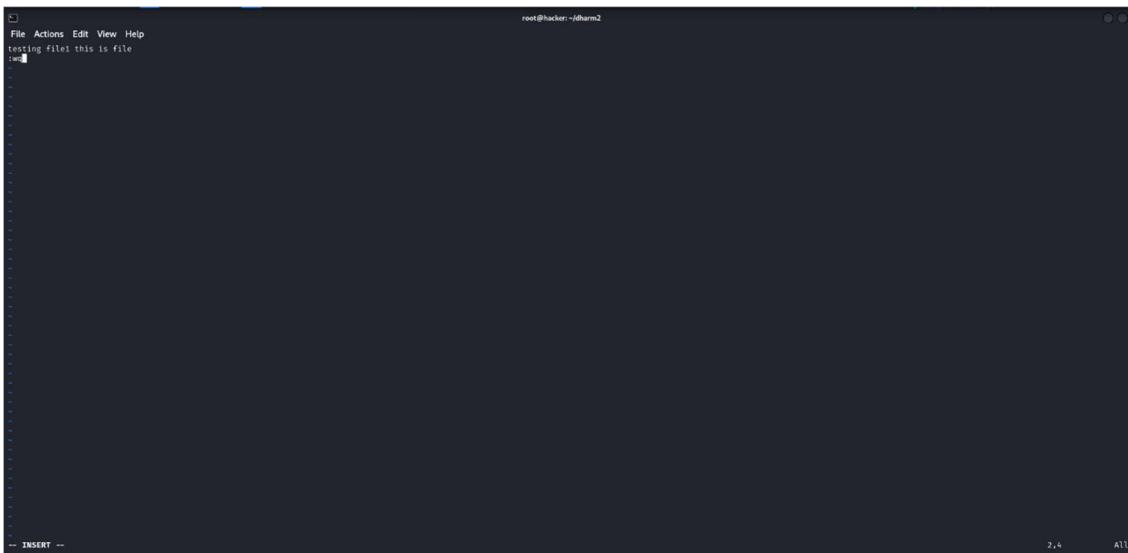
[root@hacker]~[~/dharm2]
# hashdeep test1.txt>test1_hash.txt

[root@hacker]~[~/dharm2]
# ls
test1.txt test1_hash.txt test2.txt

[root@hacker]~[~/dharm2]
# cat test1_hash.txt
%%%% HASHDEEP-1.0
%%% size,md5,sha256,filename
## Invoked from: /root/dharm2
## # hashdeep test1.txt
##
14,997a9f00e25f85b73ffb1898376b7186,76d793c71e4f153583c2687898468b4068db96782fcc793a2282abdf684a52c3,/root/dharm2/test1.txt
```

```
[root@hacker]~[~/dharm2]
# hashdeep -a test1.txt -k test1_hash.txt
hashdeep: Audit passed

[root@hacker]~[~/dharm2]
# vi test1.txt
```



```
File Actions Edit View Help
testing file1 this is file
:wq
```

```
[root@hacker]~[~/dharm2]
# cat test1.txt
testing file1 this is file
```

THE END TASK

## 6. Give command of windows and Linux for Network/Server Details gathering

### Windows Commands

#### 1. Get IP Configuration:

ipconfig

```
Microsoft Windows [Version 10.0.22631.3958]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dharm>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet 2:

  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::d58c:4482:a0eb:21c8%5
  IPv4 Address. . . . . : 192.168.56.1
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection* 9:

  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 10:

  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:

  Connection-specific DNS Suffix . :
  IPv6 Address. . . . . : 2409:40e2:1022:7f90:a2e2:f480:1fb0:9c43
  Temporary IPv6 Address. . . . . : 2409:40e2:1022:7f90:b099:46a:2d7:bc68
  Link-local IPv6 Address . . . . . : fe80::521:38d4:6be3:9516%20
  IPv4 Address. . . . . : 192.168.3.39
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : fe80::50f9:23ff:fedc:7554%20
                                         192.168.3.165

Ethernet adapter Ethernet:

  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . : bbrouter
```

### Windows Commands

#### 2. View Routing Table:

route print

```
C:\Users\dharm>route print
=====
Interface List
5...0a 00 27 00 00 05 ....VirtualBox Host-Only Ethernet Adapter
12...28 c5 d2 cc 24 79 ....Microsoft Wi-Fi Direct Virtual Adapter
8...2a c5 d2 cc 24 78 ....Microsoft Wi-Fi Direct Virtual Adapter #2
20...28 c5 d2 cc 24 78 ....Intel(R) Wi-Fi 6E AX211 160MHz
13...d0 ad 08 a7 18 e1 ....Realtek Gaming GbE Family Controller
1.....Software Loopback Interface 1
=====

IPv4 Route Table
=====
Active Routes:
Network Destination      Netmask        Gateway       Interface Metric
          0.0.0.0        0.0.0.0    192.168.3.165  192.168.3.39    35
         127.0.0.0      255.0.0.0        On-link     127.0.0.1    331
        127.0.0.1      255.255.255.255        On-link     127.0.0.1    331
      127.255.255.255  255.255.255.255        On-link     127.0.0.1    331
       192.168.3.0      255.255.255.0        On-link   192.168.3.39    291
     192.168.3.39      255.255.255.255        On-link   192.168.3.39    291
    192.168.3.255      255.255.255.255        On-link   192.168.3.39    291
     192.168.56.0      255.255.255.0        On-link   192.168.56.1    281
    192.168.56.1      255.255.255.255        On-link   192.168.56.1    281
   192.168.56.255      255.255.255.255        On-link   192.168.56.1    281
     224.0.0.0        240.0.0.0        On-link     127.0.0.1    331
    224.0.0.0        240.0.0.0        On-link   192.168.56.1    281
     224.0.0.0        240.0.0.0        On-link   192.168.3.39    291
  255.255.255.255      255.255.255.255        On-link     127.0.0.1    331
 255.255.255.255      255.255.255.255        On-link   192.168.56.1    281
 255.255.255.255      255.255.255.255        On-link   192.168.3.39    291
=====
Persistent Routes:
  None

IPv6 Route Table
=====
Active Routes:
If Metric Network Destination      Gateway
 20      51 ::/0           fe80::50f9:23ff:fedc:7554
  1      331 ::1/128        On-link
 20      51 2409:40e2:1022:7f90::/64 On-link
 20      291 2409:40e2:1022:7f90:a2e2:f480:1fb0:9c43/128
          On-link
 20      291 2409:40e2:1022:7f90:b099:46a:2d7:bc68/128
          On-link
  5      281 fe80::/64        On-link
 20      291 fe80::/64        On-link
 20      291 fe80::521:38d4:6be3:9516/128
          On-link
=====
```

```
5      281 fe80::d58c:4482:a0eb:21c8/128
          On-link
 1      331 ff00::/8        On-link
 5      281 ff00::/8        On-link
 20      291 ff00::/8        On-link
=====
```

```
Persistent Routes:
  None
```

### 3. Display Active Network Connections:

```
netstat -an
```

```
C:\Users\dharm>netstat -an

Active Connections

  Proto  Local Address        Foreign Address      State
  TCP    0.0.0.0:135          0.0.0.0:0          LISTENING
  TCP    0.0.0.0:445          0.0.0.0:0          LISTENING
  TCP    0.0.0.0:5040         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49664         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49665         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49666         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49667         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49668         0.0.0.0:0          LISTENING
  TCP    0.0.0.0:49670         0.0.0.0:0          LISTENING
  TCP    127.0.0.1:49694       127.0.0.1:49695     ESTABLISHED
  TCP    127.0.0.1:49695       127.0.0.1:49694     ESTABLISHED
  TCP    127.0.0.1:49696       127.0.0.1:49697     ESTABLISHED
  TCP    127.0.0.1:49697       127.0.0.1:49696     ESTABLISHED
  TCP    192.168.3.39:139      0.0.0.0:0          LISTENING
  TCP    192.168.3.39:60853    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60854    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60855    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60856    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60857    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60858    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60875    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60876    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60877    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60878    192.168.3.165:53    TIME_WAIT
  TCP    192.168.3.39:60879    192.168.3.165:53    TIME_WAIT
  TCP    192.168.56.1:139      0.0.0.0:0          LISTENING
  TCP    [::]:135              [::]:0            LISTENING
  TCP    [::]:445              [::]:0            LISTENING
  TCP    [::]:49664             [::]:0            LISTENING
  TCP    [::]:49665             [::]:0            LISTENING
  TCP    [::]:49666             [::]:0            LISTENING
  TCP    [::]:49667             [::]:0            LISTENING
  TCP    [::]:49668             [::]:0            LISTENING
  TCP    [::]:49670             [::]:0            LISTENING
  TCP    [::]:49669             [::]:0            LISTENING
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:49409  [2603:1040:a06:6::]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:57712  [2603:1063:2202:14::3]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:58819  [64:ff9b::14d4:5875]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:58822  [2404:6800:4003:c03::bc]:5228 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:58844  [2405:200:1630:18a::1011]:443 CLOSE_WAIT
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:58946  [2603:1063:2202:14::3]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:59199  [2a03:2880:f26b:c8:face:b00c:0:7260]:80 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60585  [2620:1ec:21::14]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60827  [64:ff9b::14bd:ad1a]:443 TIME_WAIT
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60828  [64:ff9b::d6b:2a0c]:443 ESTABLISHED
  TCP    [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60852  [2603:1046:1406::5]:443 TIME_WAIT
```

```

TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60827 [64:ff9b::14bd:ad1a]:443 TIME_WAIT
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60828 [64:ff9b::d6b:2a0c]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60852 [2603:1046:1406::5]:443 TIME_WAIT
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60859 [2603:1046:1400::7]:443 TIME_WAIT
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60860 [2603:1046:1400::1]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60861 [2405:200:160b:1731::312c:b752]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60863 [2405:200:160b:1731::312c:b74b]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60868 [2603:1046:900:54::2]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60869 [2620:1ec:33::254]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60870 [2a01:111:202c::254]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60871 [2606:2800:147:120f:30c:1ba0:fc6:265a]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60872 [64:ff9b::98c3:264c]:80 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60873 [64:ff9b::cc4f:c5de]:443 ESTABLISHED
TCP [2409:40e2:1022:7f90:b099:46a:2d7:bc68]:60874 [2603:1046:1400::7]:443 TIME_WAIT

UDP 0.0.0.0:500 *:*
UDP 0.0.0.0:4500 *:*
UDP 0.0.0.0:5050 *:*
UDP 0.0.0.0:5353 *:*
UDP 0.0.0.0:49244 0.0.32.10:443
UDP 0.0.0.0:54108 0.0.32.14:443
UDP 0.0.0.0:56586 *:*
UDP 0.0.0.0:57486 *:*
UDP 127.0.0.1:1900 *:*
UDP 127.0.0.1:51705 *:*
UDP 127.0.0.1:62435 127.0.0.1:62435
UDP 192.168.3.39:137 *:*
UDP 192.168.3.39:138 *:*
UDP 192.168.3.39:1900 *:*
UDP 192.168.3.39:51704 *:*
UDP 192.168.56.1:137 *:*
UDP 192.168.56.1:138 *:*
UDP 192.168.56.1:1900 *:*
UDP 192.168.56.1:51703 *:*
UDP [:]:500 *:*
UDP [:]:4500 *:*
UDP [:]:5353 *:*
UDP [:]:5353 *:*
UDP [:]:5353 *:*

```

```

UDP [:]:5353 *:*
UDP [:]:5355 *:*
UDP [:]:49244 [2404:6800:4002:80a::200a]:443
UDP [:]:54108 [2404:6800:4002:813::200e]:443
UDP [:]:56586 *:*
UDP [:]:57486 *:*
UDP [:]:1900 *:*
UDP [:]:51702 *:*
UDP [fe80::521:38d4:6be3:9516%20]:1900 *:*
UDP [fe80::521:38d4:6be3:9516%20]:51701 *:*
UDP [fe80::d58c:4482:a0eb:21c8%5]:1900 *:*
UDP [fe80::d58c:4482:a0eb:21c8%5]:51700 *:*

```

#### 4. Get Network Adapter Details:

wmic nic get /all

Name	TimeOfLastReset	PowerManagementCapabilities	PowerManagementSupported	ProductName	NetConnectionID	NetConnectionStatus	NetEnabled	NetworkAddresses	PermanentAddress	PhysicalDeviceID	DeviceID	Index	InstallDate	Installed	InterfaceIndex	LastErrorCode	MACAddress	Manufacturer	MaxNumberCo	SystemStatus	StatusInfo	SystemCreationClassName	
rk Adapter	0	3	[00000000] Microsoft Kernel Debug Network Adapter	0		TRUE	16											Win32_NetworkAdapter	Microsoft Kernel Debug Netwo	0	ROOT\KDNIC\00000000	Win32_ComputerSystem	HACKER
Ethernet 892.3	0	2	Microsoft Kernel Debug Network Adapter	FALSE	Microsoft Kernel Debug Network Adapter	kdnic												Win32_NetworkAdapter	Realtek Gaming GBE Family Co	0	D8:AD:08:17:18:E1	Win32_ComputerSystem	HACKER
Ethernet Controller	1	3	[00000001] Realtek Gaming GBE Family Controller	0		TRUE	13											PCI\VEN_10E2\DEV_01\SUBSYS_8C3F\1035\VEN_1					
6\4K11F7E7E3D060607	2	3	Realtek Gaming GBE Family Controller	FALSE	Realtek Gaming GBE Family Controller	rt68cx21	0											Win32_NetworkAdapter	Realtek Gaming GBE Family Co	0	D8:AD:08:17:18:E1	Win32_ComputerSystem	HACKER
2024\0726202620.500000+330	3	2	[00000002] WAN Miniport (SSTP)	0		TRUE	15										Win32_NetworkAdapter	WAN Miniport (SSTP)	0	SW\MSRRAS\MS_SSTPINPORT	Win32_ComputerSystem	HACKER	
2024\0726202620.500000+330	4	3	WAN Miniport (SSTP)	FALSE	WAN Miniport (SSTP)	RasSstp											Win32_NetworkAdapter	WAN Miniport (IKEv2)	0	SW\MSRRAS\MS_IKEV2INPORT	Win32_ComputerSystem	HACKER	
2024\0726202620.500000+330	5	3	[00000003] WAN Miniport (IKEv2)	0		TRUE	18										Win32_NetworkAdapter	WAN Miniport (IKEv2)	0	SW\MSRRAS\MS_IKEV2INPORT	Win32_ComputerSystem	HACKER	
2024\0726202620.500000+330	6	3	WAN Miniport (IKEv2)	FALSE	WAN Miniport (IKEv2)	RasAgileVpn										Win32_NetworkAdapter	WAN Miniport (L2TP)	0	SW\MSRRAS\MS_L2TPINPORT	Win32_ComputerSystem	HACKER		
2024\0726202620.500000+330	7	3	[00000004] WAN Miniport (L2TP)	0		TRUE	6									Win32_NetworkAdapter	WAN Miniport (L2TP)	0	SW\MSRRAS\MS_L2TPINPORT	Win32_ComputerSystem	HACKER		
2024\0726202620.500000+330	8	3	WAN Miniport (L2TP)	FALSE	WAN Miniport (L2TP)	RasL2tp										Win32_NetworkAdapter	WAN Miniport (PPPoE)	0	SW\MSRRAS\MS_PPPOEINPORT	Win32_ComputerSystem	HACKER		
Ethernet 892.3	9	3	[00000005] WAN Miniport (PPPoE)	0		TRUE	19									Win32_NetworkAdapter	WAN Miniport (PPPoE)	0	SW\MSRRAS\MS_PPPOEINPORT	Win32_ComputerSystem	HACKER		
2024\0726202620.500000+330	10	3	WAN Miniport (PPPoE)	FALSE	WAN Miniport (PPPoE)	PptpMiniport										Win32_NetworkAdapter	WAN Miniport (PPPoE)	0	SW\MSRRAS\MS_PPPOEINPORT	Win32_ComputerSystem	HACKER		
Ethernet 892.3	11	3	[00000006] WAN Miniport (IP)	0		TRUE	14									Win32_NetworkAdapter	WAN Miniport (IP)	0	62:3B:08:52:41:53	Win32_ComputerSystem	HACKER		
2024\0726202620.500000+330	12	3	WAN Miniport (IP)	FALSE	WAN Miniport (IP)	NdisWan										Win32_NetworkAdapter	WAN Miniport (IP)	0	SW\MSRRAS\MS_NDISWANIP	Win32_ComputerSystem	HACKER		
Ethernet 892.3	13	3	[00000007] WAN Miniport (IPv6)	0		TRUE	9									Win32_NetworkAdapter	WAN Miniport (IPv6)	0	64:79:20:52:41:53	Microsoft	HACKER		
2024\0726202620.500000+330	14	3	WAN Miniport (IPv6)	FALSE	WAN Miniport (IPv6)	NdisWan										SW\MSRRAS\MS_NDISWANIPV6	Win32_ComputerSystem	HACKER					
Ethernet 892.3	15	3	[00000008] WAN Miniport (IPv6)	0		TRUE	8									Win32_NetworkAdapter	WAN Miniport (Network Monitor)	0	64:DE:28:52:41:53	Microsoft	HACKER		
2024\0726202620.500000+330	16	3	WAN Miniport (Network Monitor)	FALSE	WAN Miniport (Network Monitor)	NdisWan										SW\MSRRAS\MS_NDISWANB	Win32_ComputerSystem	HACKER					
Ethernet 892.3	17	3	[00000009] WAN Miniport (Network Monitor)	0		TRUE	3									Win32_NetworkAdapter	WAN Miniport (Network Monitor)	0	64:DE:28:52:41:53	Microsoft	HACKER		
2024\0726202620.500000+330	18	3	WAN Miniport (Network Monitor)	FALSE	WAN Miniport (Network Monitor)	NdisWan										SW\MSRRAS\MS_NDISWANB	Win32_ComputerSystem	HACKER					
Ethernet 892.3	19	3	[00000010] Intel(R) Wi-Fi 6E AX211 160MHz	0		TRUE	20									Win32_NetworkAdapter	Intel(R) Wi-Fi 6E AX211 160M	0	28:C5:D2:CC:24:78	Intel Corporation	HACKER		
1\3G1158369606A3	20	3	[EC2507F9-3HAI-4F37-BUAB-5D63458E2F88]	10		TRUE	2									PICIVEN_008866DEV_51F6GCSUBSYS_00988686&REV_0	Win32_ComputerSystem	HACKER					
2024\0726202620.500000+330	21	3	Intel(R) Wi-Fi 6E AX211 160MHz	FALSE	Intel(R) Wi-Fi 6E AX211 160MHz	Netww14	628358000									Win32_NetworkAdapter	Intel(R) Wi-Fi 6E AX211 160MHz	0	64:DE:28:52:41:53	Microsoft	HACKER		
Ethernet 892.3	22	3	[00000011] Microsoft Wi-Fi Direct Virtual Adapter	0		TRUE	11									Win32_NetworkAdapter	Microsoft Wi-Fi Direct Virtu	0	28:C5:D2:CC:24:79	Microsoft	HACKER		
al Adapter	23	3	Microsoft Wi-Fi Direct Virtual Adapter	FALSE	Microsoft Wi-Fi Direct Virtual Adapter	vwifimp	922372036854775807									{5D624F94-8858-4C83-A3FA-AFD2888BAF3}\WIF	Win32_ComputerSystem	HACKER					
IMP_WFD\46288HC906G611	24	3	[00000012] Bluetooth Device (Personal Area Network)	0		TRUE	12									Win32_NetworkAdapter	Bluetooth Device (Personal A	0	0				
rea Network	25	3	Bluetooth Device (Personal Area Network)	FALSE	Bluetooth Device (Personal Area Network)	NdisWan										Win32_NetworkAdapter	Bluetooth Device (Personal Area Network)	0	Win32_ComputerSystem	HACKER			
2024\0726202620.500000+330	26	3	[00000013] Microsoft Wi-Fi Direct Virtual Adapter	0		TRUE	8									Win32_NetworkAdapter	Microsoft Wi-Fi Direct Virtu	0	2A:C5:D2:CC:24:78	Microsoft	HACKER		
Ethernet 892.3	27	3	Microsoft Wi-Fi Direct Virtual Adapter	FALSE	Microsoft Wi-Fi Direct Virtual Adapter	vwifimp	922372036854775807									{5D624F94-8858-4C83-A3FA-AFD2888BAF3}\WIF	Win32_ComputerSystem	HACKER					
2024\0726202620.500000+330	28	3	[00000014] VirtualBox Host-Only Ethernet Adapter	0		TRUE	5									Win32_NetworkAdapter	VirtualBox Host-Only Etherne	0	0A:00:27:00:00:05	Oracle Corporation	HACKER		
t Adapter	29	3	[39F9493D-F9C4-4B37-B1C-399618A19000]	14		TRUE	2									ROOT\NET\0000	VirtualBox Host-Only Ethernet Adapter	0	Win32_ComputerSystem	HACKER			
2024\0726202620.500000+330	30	3	VirtualBox Host-Only Ethernet Adapter	FALSE	VirtualBox Host-Only Ethernet Adapter	VBoxNetAdp	1000000000									Win32_NetworkAdapter	VirtualBox Host-Only Ethernet Adapter	0	Win32_ComputerSystem	HACKER			
haring Device	31	3	[00000015] Remote NDIS based Internet Sharing Device	0		TRUE	17									Win32_NetworkAdapter	Remote NDIS based Internet S	0	0				
net share	32	3	Remote NDIS based Internet Sharing Device	FALSE	Remote NDIS based Internet Sharing Device											Win32_NetworkAdapter	Remote NDIS based Internet S	0	0				

## 5. View Network Shares:

net share

## 6. Check DNS Configuration:

nslookup

```
C:\Users\dharm>net share  
  
Share name     Resource          Remark  
-----  
C$             C:\                Default share  
D$             D:\                Default share  
IPC$             
ADMIN$          C:\Windows       Remote IPC  
The command completed successfully.
```

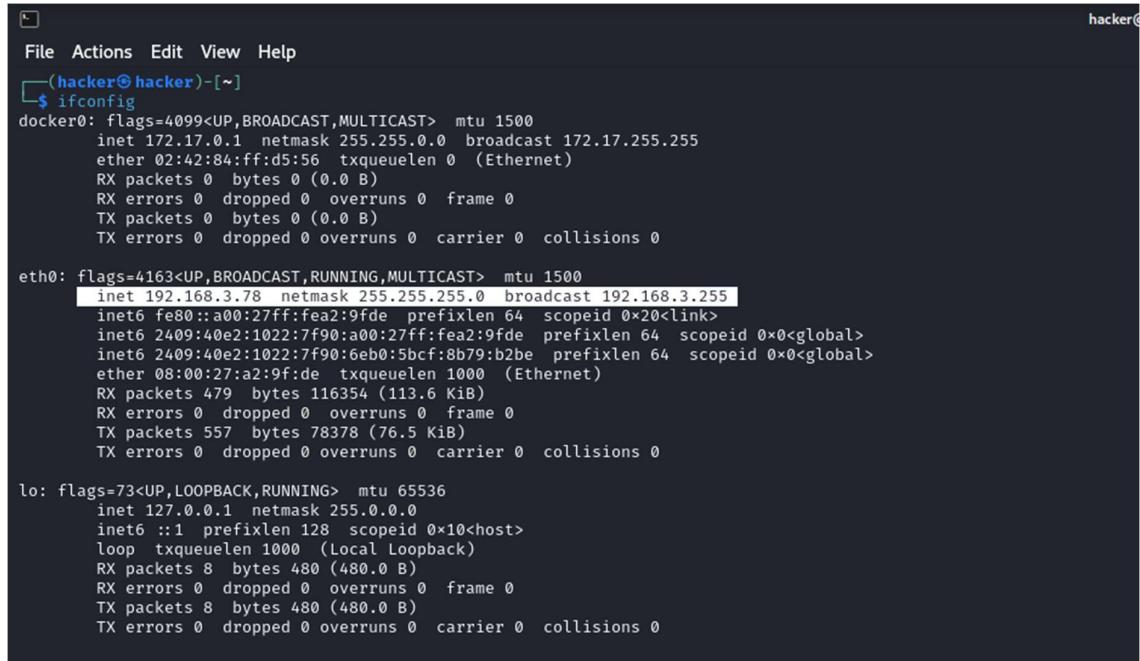
```
C:\Users\dharm>nslookup  
Default Server: Unknown  
Address: 192.168.3.165
```

```
> |
```

### Give command of Linux for Network/Server Details gathering

#### 1. Get IP Configuration:

```
ifconfig
```



```
File Actions Edit View Help  
(hacker@hacker)-[~]  
$ ifconfig  
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
      inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255  
        ether 02:42:84:ff:d5:56 txqueuelen 0 (Ethernet)  
          RX packets 0 bytes 0 (0.0 B)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 0 bytes 0 (0.0 B)  
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
      inet 192.168.3.78 netmask 255.255.255.0 broadcast 192.168.3.255  
        inet6 fe80::a00:27ff:fea2:9fde prefixlen 64 scopeid 0x20<link>  
          inet6 2409:40e2:1022:7f90:a00:27ff:fea2:9fde prefixlen 64 scopeid 0x0<global>  
          inet6 2409:40e2:1022:7f90:6eb0:5bcf:8b79:b2be prefixlen 64 scopeid 0x0<global>  
        ether 08:00:27:a2:9f:de txqueuelen 1000 (Ethernet)  
          RX packets 479 bytes 116354 (113.6 KiB)  
          RX errors 0 dropped 0 overruns 0 frame 0  
          TX packets 557 bytes 78378 (76.5 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
```

or, on newer systems:

```

hacker@hacker: ~
File Actions Edit View Help
(hacker@hacker)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:a2:9f:de brd ff:ff:ff:ff:ff:ff
        inet 192.168.3.7/24 brd 192.168.3.255 scope global dynamic noprefixroute eth0
            valid_lft 120sec preferred_lft 120sec
            inet6 2a09:40e2:1022:7f90:6eb0:5bcf:8b79:b2fe/64 scope global temporary dynamic
                valid_lft 6843sec preferred_lft 6843sec
                inet6 2a09:40e2:1022:7f90:a00:27ff:fea2:9fde/64 scope global dynamic mngtmpaddr noprefixroute
                    valid_lft 6843sec preferred_lft 6843sec
                    inet6 fe80::a00:27ff:fea2:9fde/64 scope link noprefixroute
                        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:04:ff:d5:56 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever

```

## View Routing Table:

route -n

or, on newer systems:

ip route

## Display Active Network Connections:

netstat -tuln

or, on newer systems:

ss -tuln

## Check DNS Configuration:

cat /etc/resolv.conf

```

        valid_lft forever preferred_lft forever
(hacker@hacker)-[~]
$ route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref  Use Iface
0.0.0.0          192.168.3.165   0.0.0.0         UG    100    0      0 eth0
172.17.0.0        0.0.0.0        255.255.0.0     U     0      0      0 docker0
192.168.3.0       0.0.0.0        255.255.255.0   U     100    0      0 eth0

(hacker@hacker)-[~]
$ ip route
default via 192.168.3.165 dev eth0 proto dhcp src 192.168.3.78 metric 100
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
192.168.3.0/24 dev eth0 proto kernel scope link src 192.168.3.78 metric 100

(hacker@hacker)-[~]
$ netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp     0      0 127.0.0.1:36375          0.0.0.0:*              LISTEN
udp     0      0 0.0.0.0:69                0.0.0.0:*
udp6    0      0 :::69                          :::*

(hacker@hacker)-[~]
$ cat /etc/resolv.conf
# Generated by NetworkManager
nameserver 192.168.3.165
nameserver 2409:40e2:1022:7f90::1c

```

or, on newer systems:

```
ss -tuln
```

### Get Network Adapter Details:

```
bash
```

```
Co
```

```
lshw -C network
```

```
or:
```

```
nmcli device
```

### View Network Shares (Samba):

```
smbclient -L localhost
```

```
(hacker㉿hacker) ~]$ ncclient -l 172.17.0.1
do_connect: Connection to 172.17.0.1 failed (Error NT_STATUS_CONNECTION_REFUSED)

(hacker㉿hacker) ~]$ ncml device
└─$ ncml device
[~]
DEVICE  TYPE      STATE           CONNECTION
eth0    ethernet  connected       standard connection 1
        linklayer connected (externally) to
docker0  bridge    connected (externally) docker0

(hacker㉿hacker) ~]$ ss -talin
Netid   State      Recv-Q      Send-Q      Local Address:Port          Peer Address:Port
udp     UNKNOWN    0           0           0.0.0.0:*
```

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port
udp	UNKNOWN	0	0	0.0.0.0:*	172.17.0.1:569
tcp	LISTEN	0	4096	127.0.0.1:36375	0.0.0.0:*

```
(hacker㉿hacker) ~]$ lshw -C network
WARNING: you should run this program as super-user.
*-network
      description: Ethernet interface
      product: 82540EM Gigabit Ethernet Controller
      vendor: Intel Corporation
      physical id: 3
      bus info: pci@0000:00:03.0
      logical name: eth0
      version: 02
      serial: 08:00:27:a2:9f:de
      size: 10Gbit/s
      capacity: 10Gbit/s
      width: 32 bits
      clock: 66MHz
      capabilities: bus_master cap_list ethernet physical tp 10bt 10bt-fd 100bt 100bt-fd 1000bt-fd autonegotiation
      configuration: autonegotiation=on broadcast=yes driver=e1000e driverversion=6.8.11-amd64 duplex-full ip=192.168.3.78 latency=64 link=yes mingnt=255 multicast=yes port=twisted pair speed=1Gbit/s
      resources: irq:19 memory:fa200000-fa21ffff iomemory:fa202000-fa21ffff
WARNING: output may be incomplete or inaccurate, you should run this program as super-user.
```

These commands will help you gather detailed information about the network and server configuration on both Windows and Linux systems.

THE TASK IS END

**TASK:-6 Reconnaissance of any website/IP with the help of these tools (IP LOOKUP WEBSITE/WHOIS/WHAT IS MY IP ADDRESS)**

### Step 1: Find the IP address

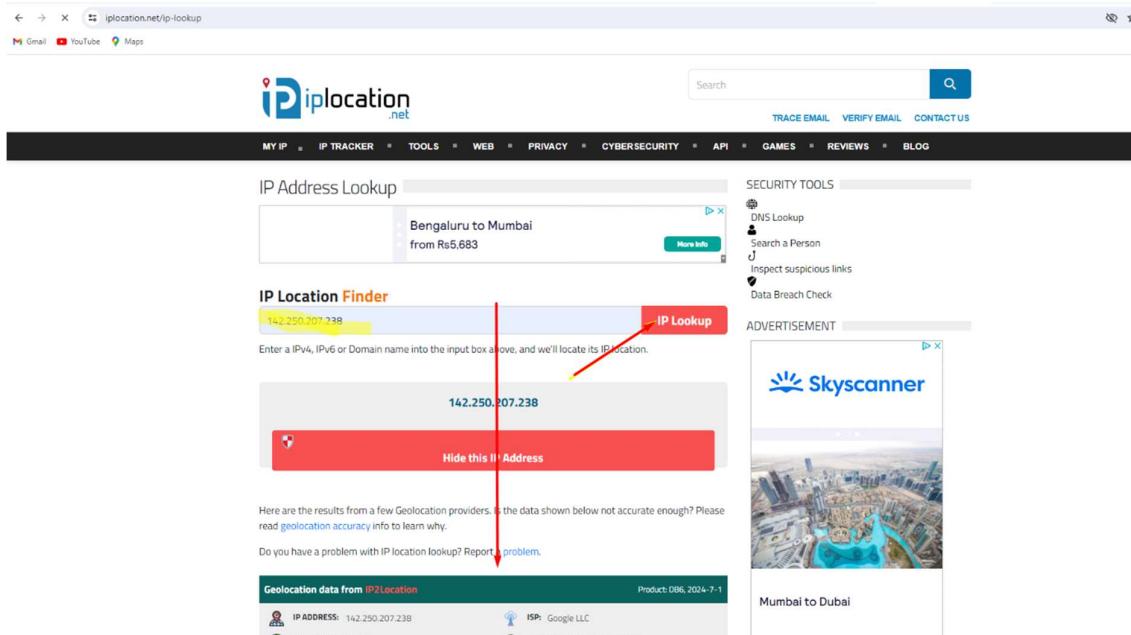
This step I am used command is nslookup: and then copy to the ip Address

```
File Actions Edit View Help

(hacker㉿hacker)-[~]
$ nslookup google.com
Server:      192.168.3.165
Address:     192.168.3.165#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.207.238
Name:   google.com
Address: 2404:6800:4002:81e::200e
```

**Step:-2 I'll go to the iplocation.net website, enter the IP address, and click the "IP Lookup" button**



The screenshot shows the iplocation.net website. At the top, there's a navigation bar with links like 'MY IP', 'IP TRACKER', 'TOOLS', 'WEB', 'PRIVACY', 'CYBERSECURITY', 'API', 'GAMES', 'REVIEWS', and 'BLOG'. Below the navigation, there's a search bar and a 'Search' button. The main content area has a heading 'IP Address Lookup' and a sub-section 'IP Location Finder' where the IP address '142.250.207.238' is entered. A red arrow points to the 'IP Lookup' button. To the right of the search bar, there's a 'SECURITY TOOLS' section with links to 'DNS Lookup', 'Search a Person', 'Inspect suspicious links', and 'Data Breach Check'. Below the search bar, there's an 'ADVERTISEMENT' for Skyscanner showing a cityscape.

I would go to the iplocation.net website, enter the IP address, click the "IP Lookup" button, scroll down the website, and get to see the data in various formats

This task is end

**TASK:-7 Copy the website pages with the tools (HT-Track / site sucker / Get left)**

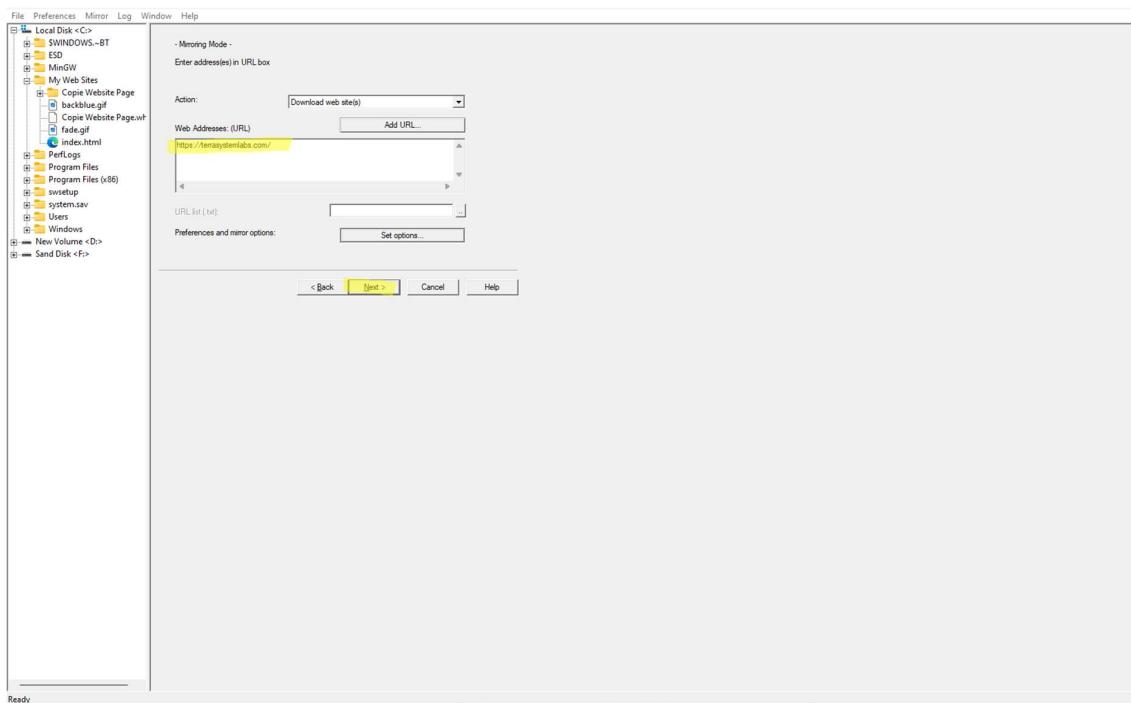
**Step:- 1. Open the HTTrack application, enter the project name, and then click the Next button.**



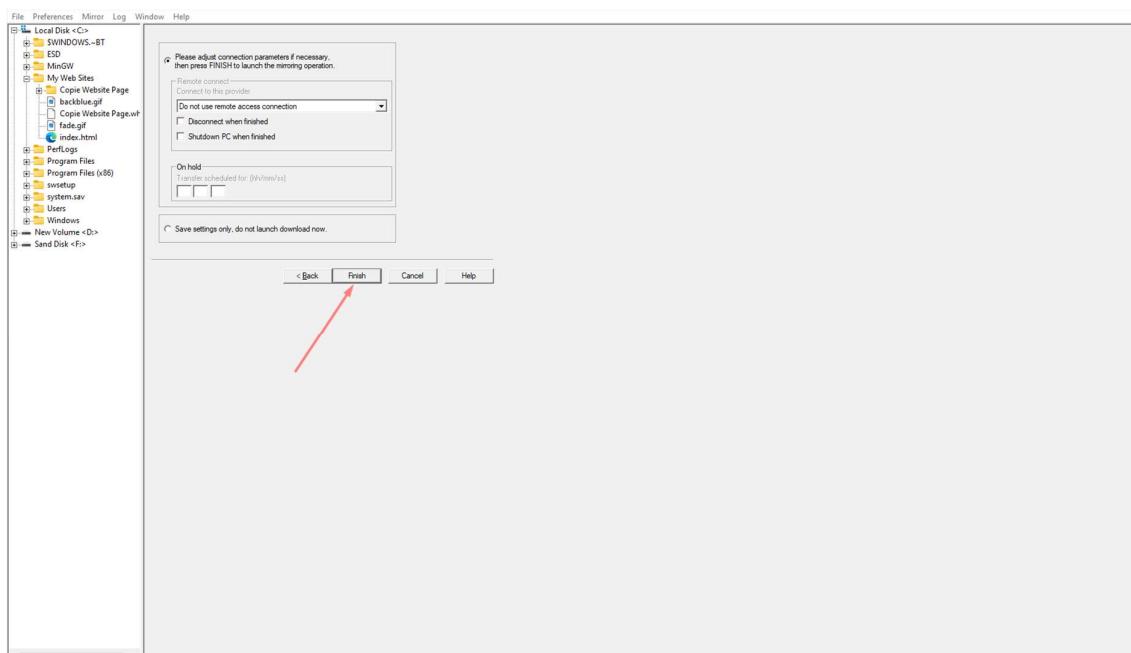
**Step:- 2 Then, go to the target website and copy the link. Then go into the httrack application**



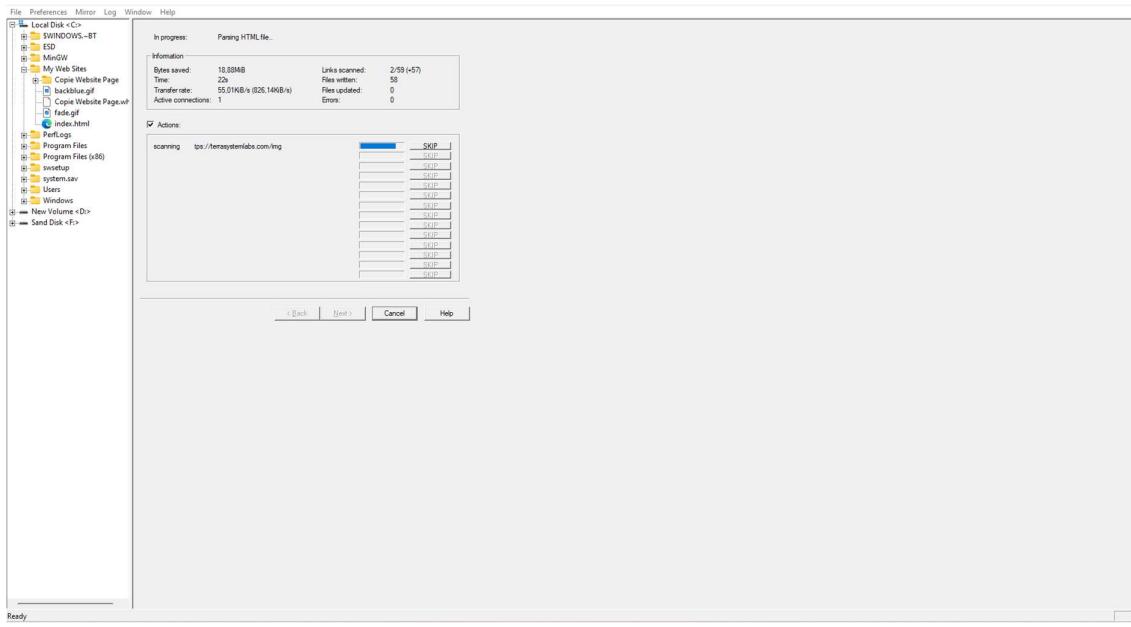
**Step:-3** Next, paste the website link into the URL section and click the Next button.



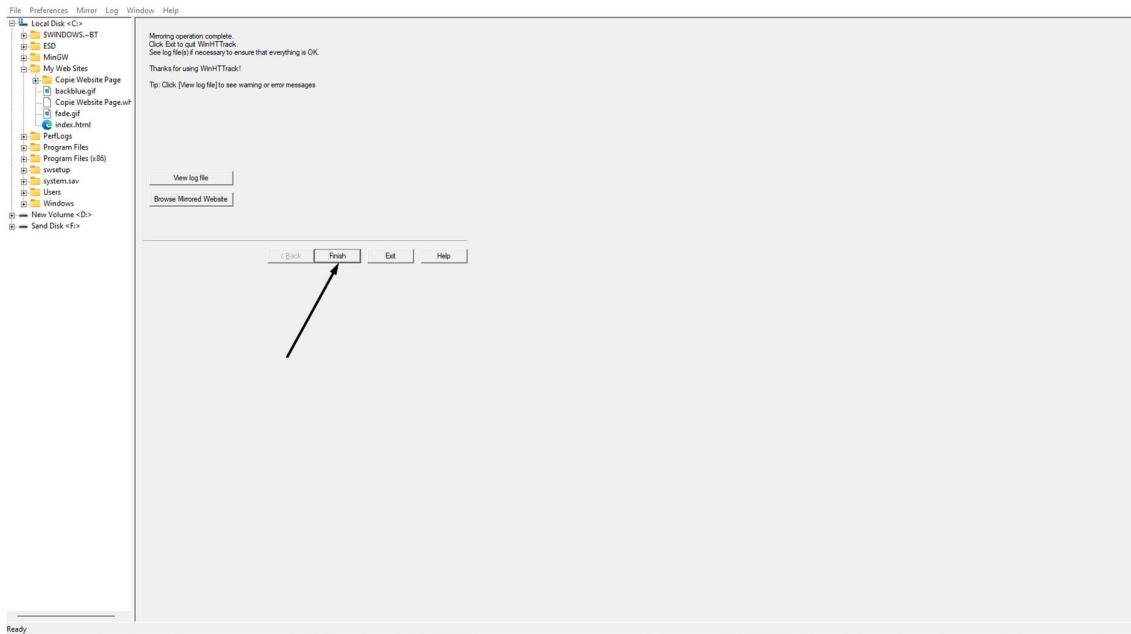
**Step:- 4 Then Click to next button**



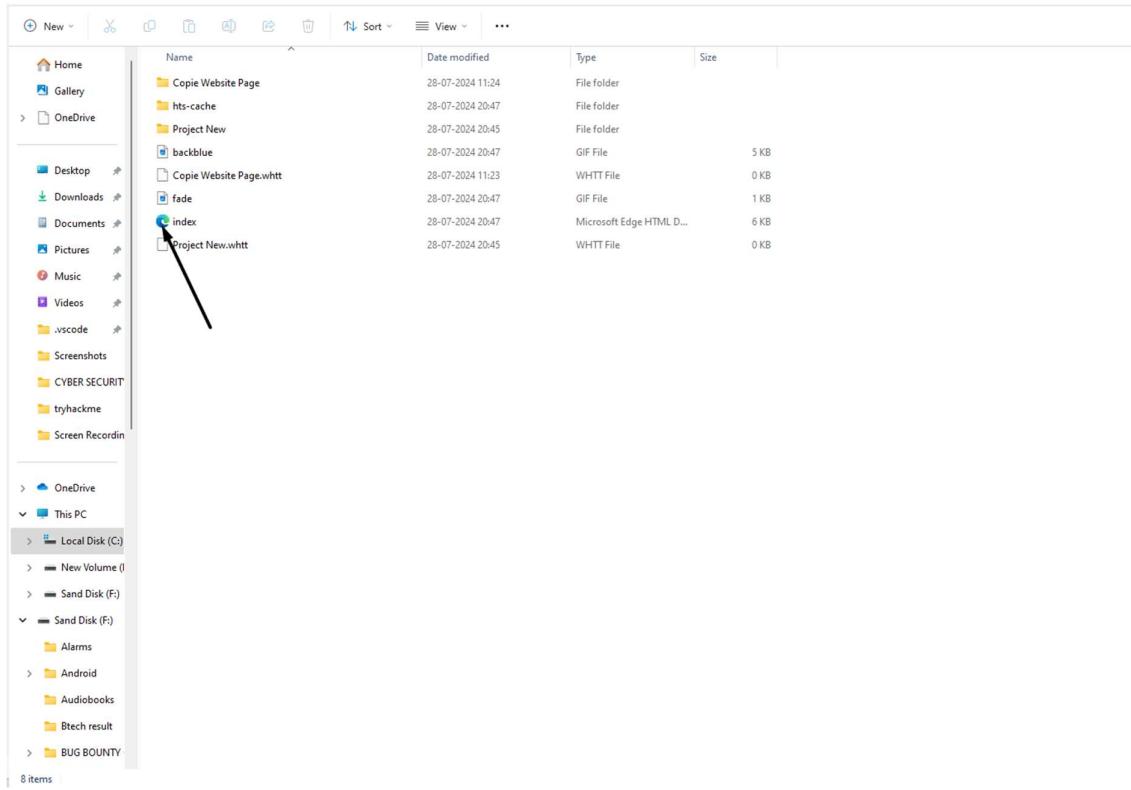
**Step:- 5** The process will start. Please wait until it finishes



**Step:- 6** Then, click the Finish button and go to the file manager.



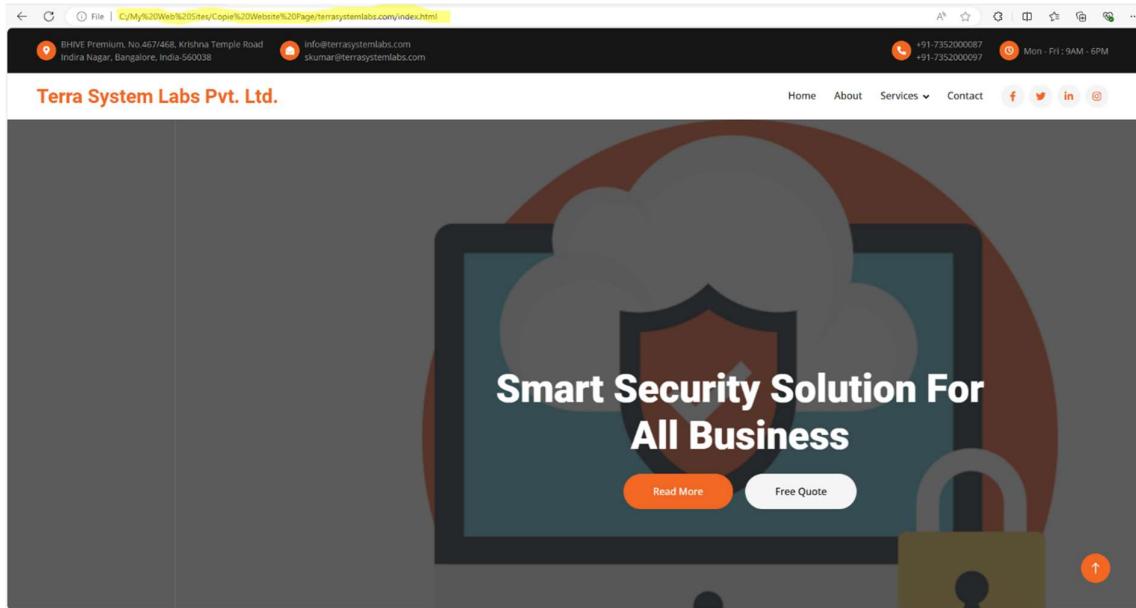
## Step:-7 then click to the Index



## Step:- 8 Then click the Copie Website Page



**Step:-9 Then finally cope the website page**



**Task is end**

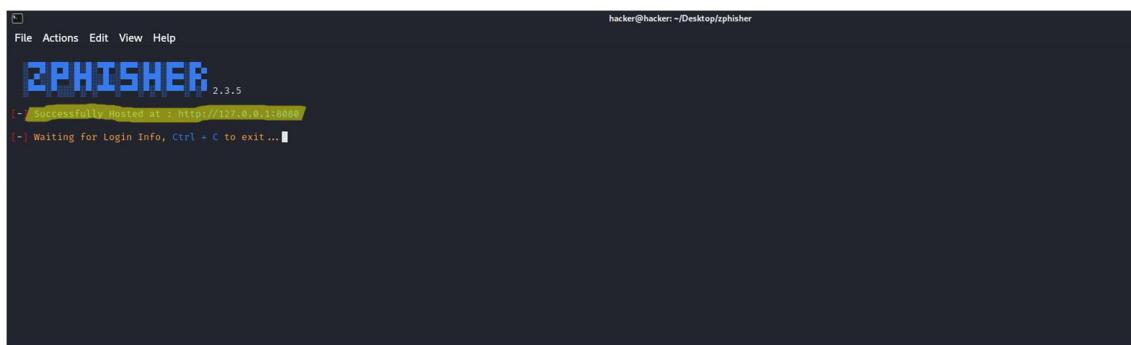
### **TASK:-8 Perform Phishing of any websites:**

Phishing is a malicious activity that involves attempting to acquire sensitive information such as usernames, passwords, and credit card details by pretending to be a trustworthy entity in electronic communications. It is illegal and unethical to engage in or support phishing activities.

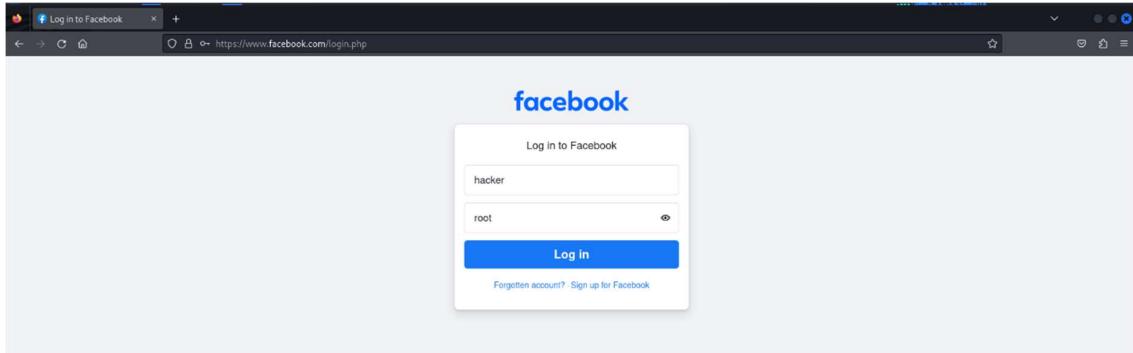
I am perform kali linux I am already setup the Zphisher



"Then, copy the link and send it to any user via email or messages." Then the user you are sending it to will enter his username and password and then he can view his details.



"Then the facebook login page will look like this." Then every person fill the data and then click to login page



"I checked the terminal and found the username and password." This is called as phishing attack

```
hacker@hacker: ~/Desktop/zphisher
File Actions Edit View Help
ZPHISHER 2.3.5
[+] Successfully Hosted at : http://127.0.0.1:8080
[+] Waiting for Login Info, Ctrl + C to exit ...
[+] Victim IP Found !
[+] Victim's IP : 127.0.0.1
[+] Saved in : auth/ip.txt
[+] Login info Found !!
[+] Account [REDACTED]
[+] Password [REDACTED] [+] Password
[+] Saved in : auth/usernames.dat
[+] Waiting for Next Login Info, Ctrl + C to exit.
```

"All tasks are completed."

THANKYOU

