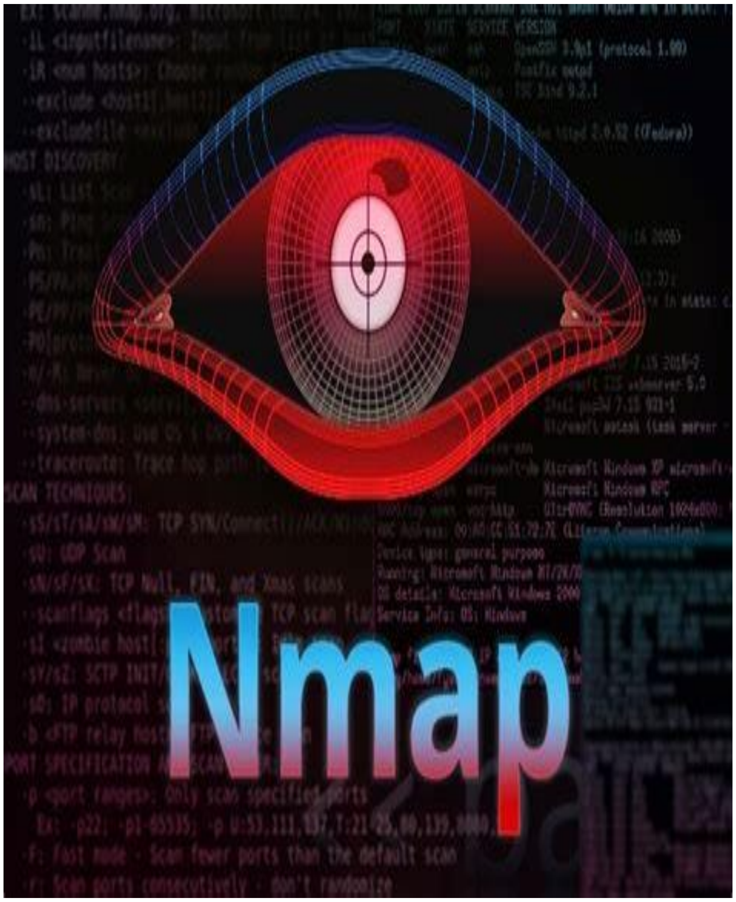


# NETWORK ENUMERATION WITH NMAP

**A PRACTICAL HANDOUT ON NMAP**

DISCLAIMER “The content of this article is meant to share information and experience on the best practices in understanding. The basic step of network penetration testing that’s is Enumeration of network information using the NMAP tool. simple language with practical meaning is used to enhance understanding to meet the desired goal of this program. Whenever shared editing as per trainer scope are allowed with the citation from the first author made, for training proposes only, meanwhile the author is open for any enquiry relating to presented information in this document. Any error or misunderstanding of the concepts are retained as human error and the author shall no guarantee charge or compromise: “

*Godwin S. Aruga (Dar es Salaam Tanzania, BEng Telecommunication (DIT), Ethical hacker (Cisco academy), & cyber security analyst (CC Certified). Data Science associate (DataCamp).*

# NETWORK ENUMERATION WITH NMAP

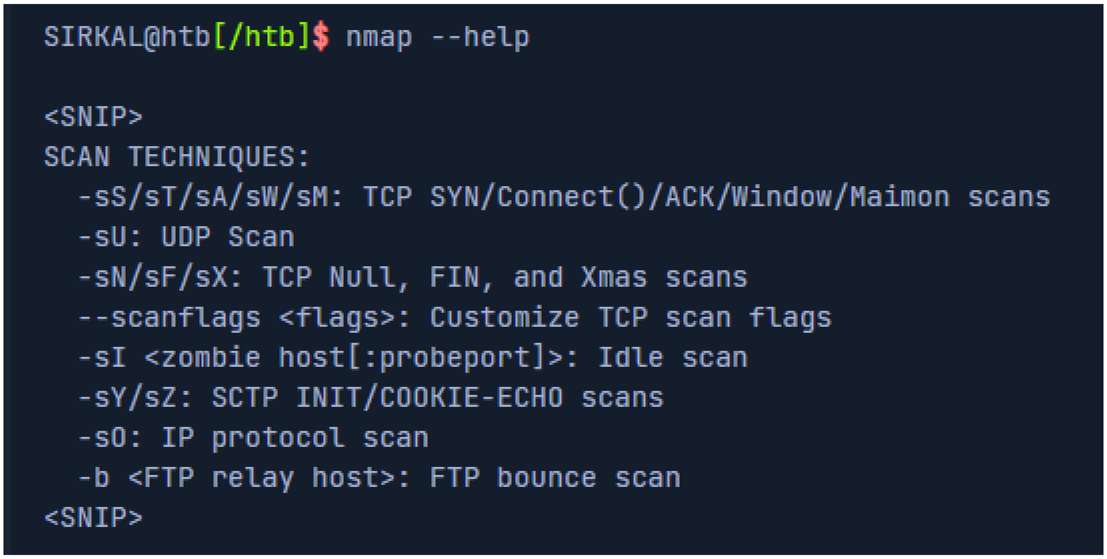
POWERED BY.

EA SPORTS intel AKERVA Deloitte. Synack Booz Allen

GODWIN ARUGA (SIRKAL).

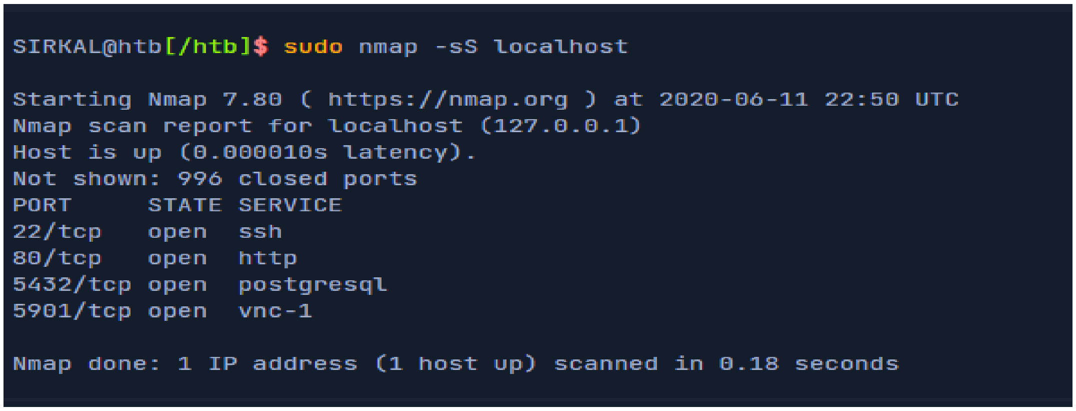
## PART 1: HOST DISCOVERY

1. VIEW POSSIBLE OPTIONS



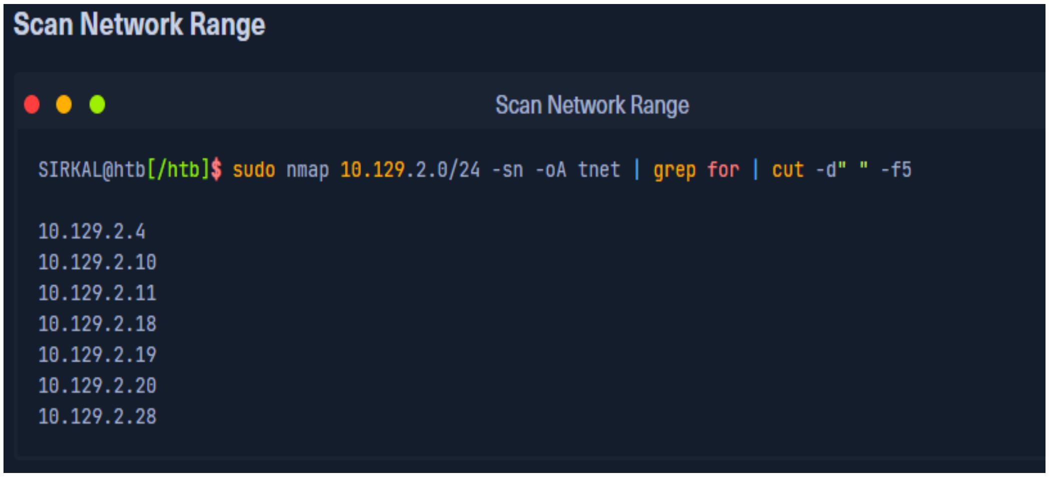
```
SIRKAL@htb[/htb]$ nmap --help

<SNIP>
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
<SNIP>
```

2. Scanning the local host



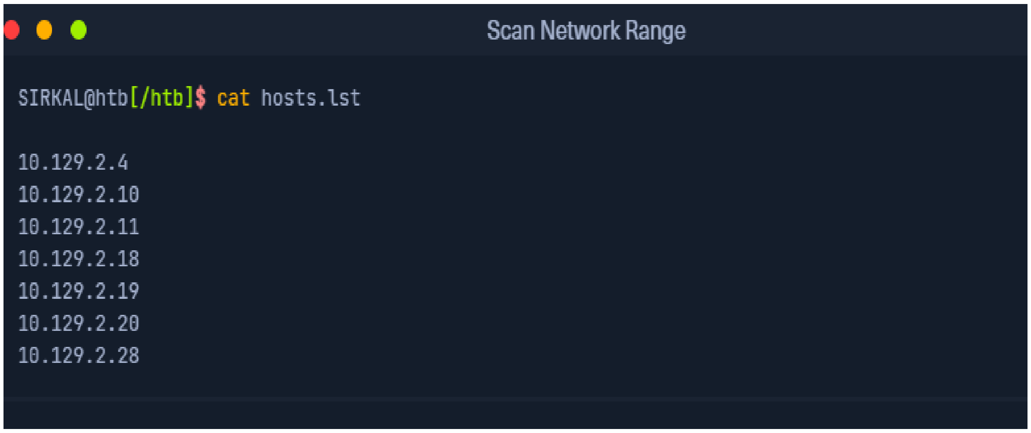
```
SIRKAL@htb[/htb]$ sudo nmap -sS localhost

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-11 22:50 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000010s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
5432/tcp open  postgresql
5901/tcp open  vnc-1

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
```

3. Scanning the network range



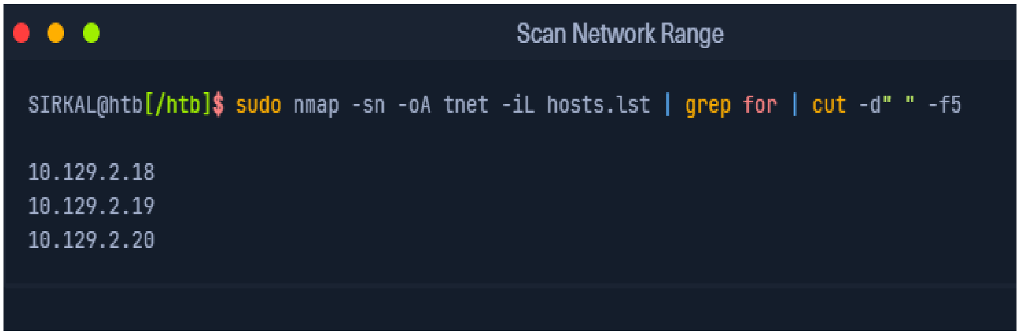
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.0/24 -sn -oA tnet | grep for | cut -d" " -f5

10.129.2.4
10.129.2.10
10.129.2.11
10.129.2.18
10.129.2.19
10.129.2.20
10.129.2.28
```

4. Scanning range by listing IP

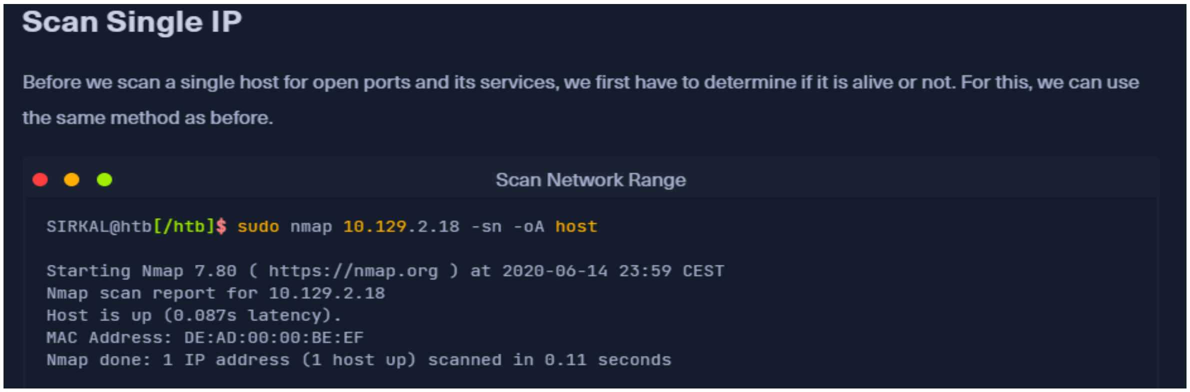


```
SIRKAL@htb[/htb]$ cat hosts.lst

10.129.2.4
10.129.2.10
10.129.2.11
10.129.2.18
10.129.2.19
10.129.2.20
10.129.2.28
```

5. Other network range scanning techniques



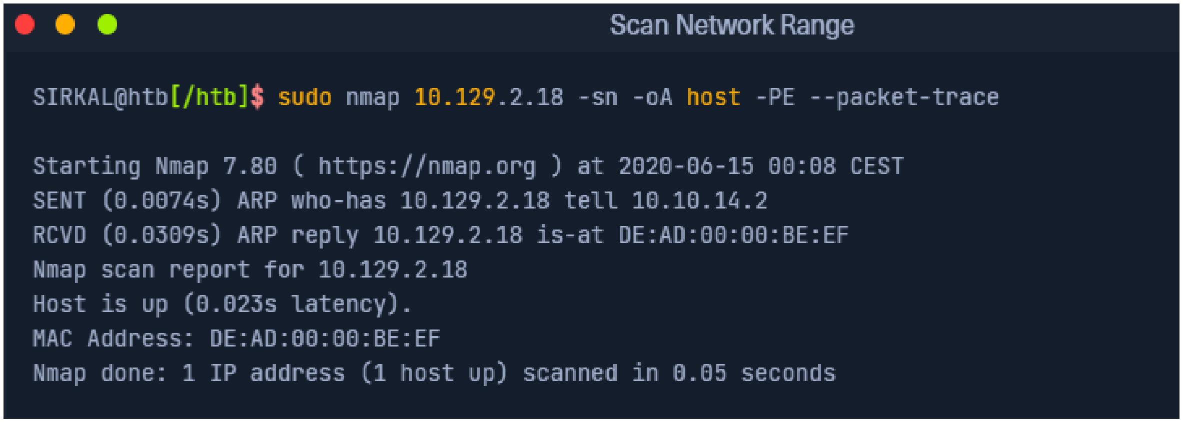
```
SIRKAL@htb[/htb]$ sudo nmap -sn -oA tnet -iL hosts.lst | grep for | cut -d" " -f5

10.129.2.18
10.129.2.19
10.129.2.20
```



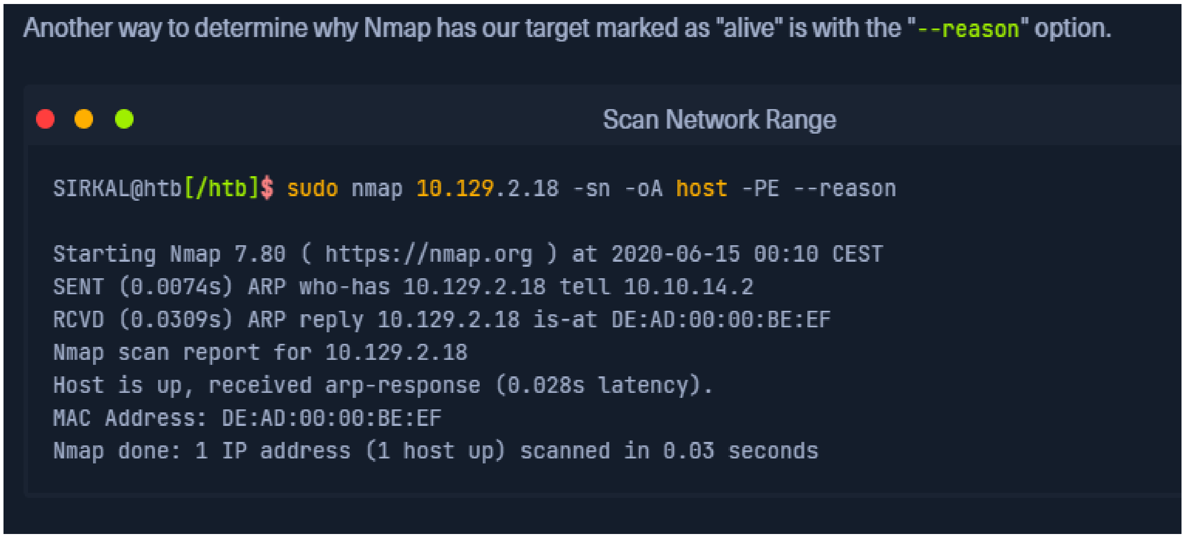
## Scan Single IP

Before we scan a single host for open ports and its services, we first have to determine if it is alive or not. For this, we can use the same method as before.

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.18 -sn -oA host

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-14 23:59 CEST
Nmap scan report for 10.129.2.18
Host is up (0.087s latency).
MAC Address: DE:AD:00:00:BE:EF
Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
```

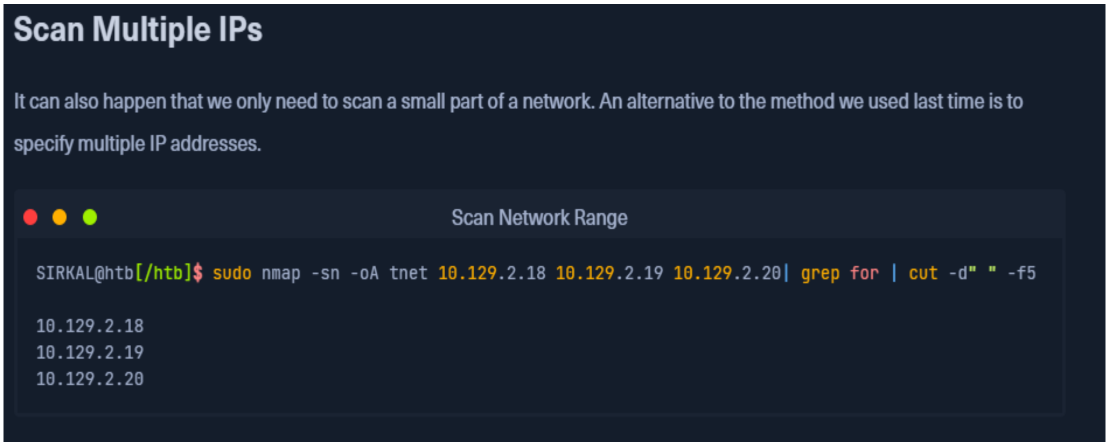


```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.18 -sn -oA host -PE --packet-trace

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 00:08 CEST
SENT (0.0074s) ARP who-has 10.129.2.18 tell 10.10.14.2
RCVD (0.0309s) ARP reply 10.129.2.18 is-at DE:AD:00:00:BE:EF
Nmap scan report for 10.129.2.18
Host is up (0.023s latency).
MAC Address: DE:AD:00:00:BE:EF
Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds
```



Another way to determine why Nmap has our target marked as "alive" is with the "`--reason`" option.

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.18 -sn -oA host -PE --reason

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 00:10 CEST
SENT (0.0074s) ARP who-has 10.129.2.18 tell 10.10.14.2
RCVD (0.0309s) ARP reply 10.129.2.18 is-at DE:AD:00:00:BE:EF
Nmap scan report for 10.129.2.18
Host is up, received arp-response (0.028s latency).
MAC Address: DE:AD:00:00:BE:EF
Nmap done: 1 IP address (1 host up) scanned in 0.03 seconds
```



## Scan Multiple IPs

It can also happen that we only need to scan a small part of a network. An alternative to the method we used last time is to specify multiple IP addresses.

Scan Network Range

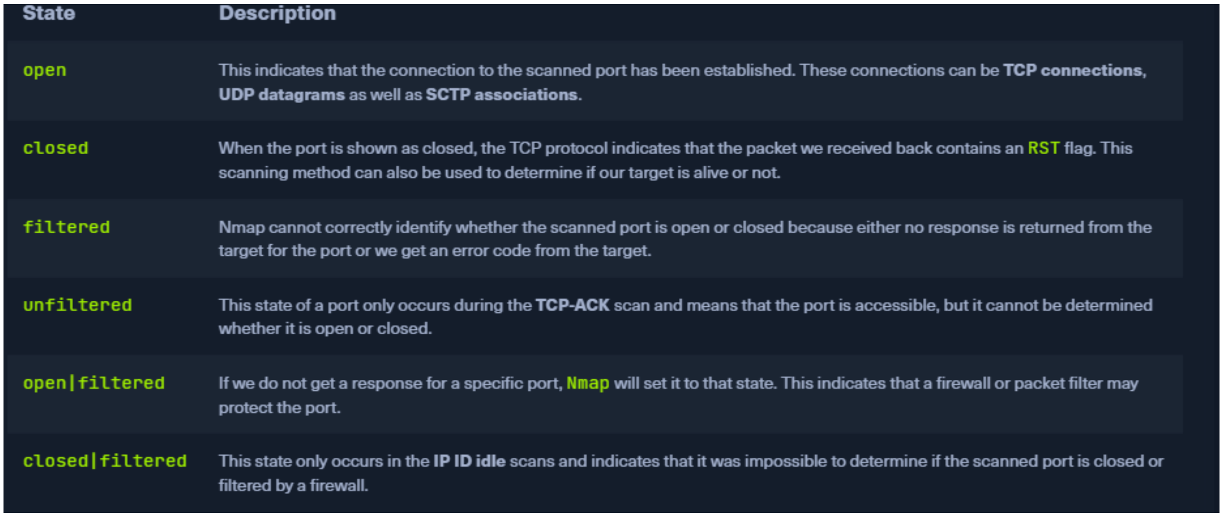
```
SIRKAL@htb[/htb]$ sudo nmap -sn -oA tnet 10.129.2.18 10.129.2.19 10.129.2.20| grep for | cut -d" " -f5

10.129.2.18
10.129.2.19
10.129.2.20
```

PART 2: HOST AND PORT SCANNING

It is essential to understand how the tool we use works and how it performs and processes the different functions. We will only understand the results if we know what they mean and how they are obtained. Therefore we will take a closer look at and analyze some of the scanning methods. After we have found out that our target is alive, we want to get a more accurate picture of the system. The information we need includes:

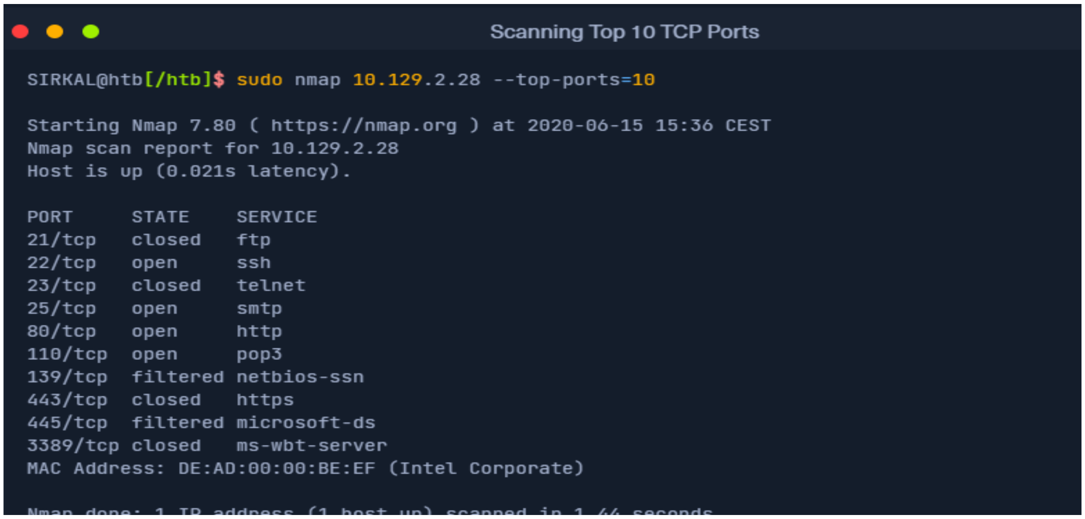
- Open ports and its services
- Service versions
- Information that the services provided
- Operating system

THE CHEAT SHEET



| State | Description |
|---|---|
| `open` | This indicates that the connection to the scanned port has been established. These connections can be **TCP connections**, **UDP datagrams** as well as **SCTP associations**. |
| `closed` | When the port is shown as closed, the TCP protocol indicates that the packet we received back contains an **RST** flag. This scanning method can also be used to determine if our target is alive or not. |
| `filtered` | Nmap cannot correctly identify whether the scanned port is open or closed because either no response is returned from the target for the port or we get an error code from the target. |
| `unfiltered` | This state of a port only occurs during the **TCP-ACK** scan and means that the port is accessible, but it cannot be determined whether it is open or closed. |
| `open\|filtered` | If we do not get a response for a specific port, **Nmap** will set it to that state. This indicates that a firewall or packet filter may protect the port. |
| `closed\|filtered` | This state only occurs in the **IP ID idle** scans and indicates that it was impossible to determine if the scanned port is closed or filtered by a firewall. |

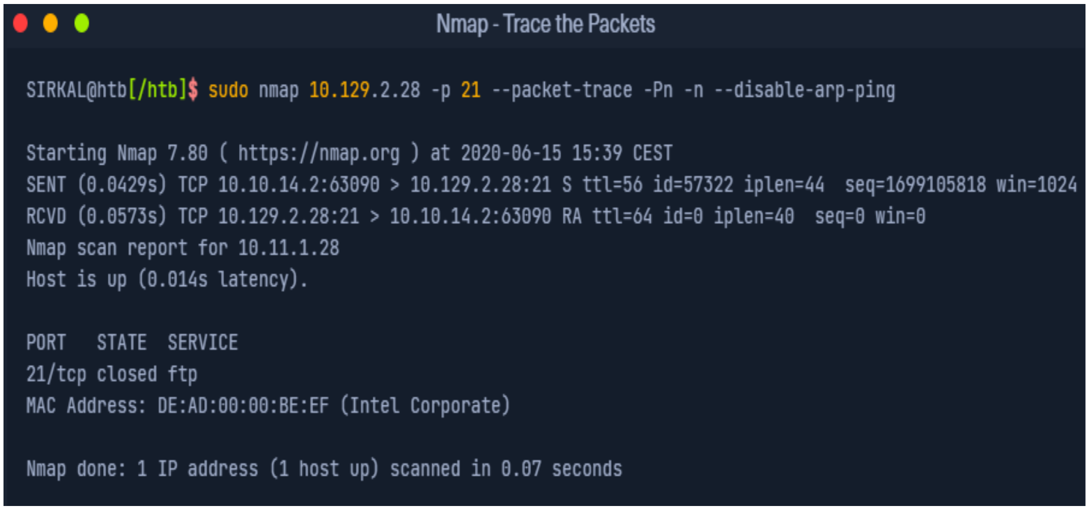
1. Scanning the TCP port



Scanning Top 10 TCP Ports

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 --top-ports=10

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 15:36 CEST
Nmap scan report for 10.129.2.28
Host is up (0.021s latency).

PORT     STATE    SERVICE
21/tcp   closed   ftp
22/tcp   open     ssh
23/tcp   closed   telnet
25/tcp   open     smtp
80/tcp   open     http
110/tcp  open     pop3
139/tcp  filtered netbios-ssn
443/tcp  closed   https
445/tcp  filtered microsoft-ds
3389/tcp closed   ms-wbt-server
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 1.44 seconds
```

2. Tracing the packet from specific host



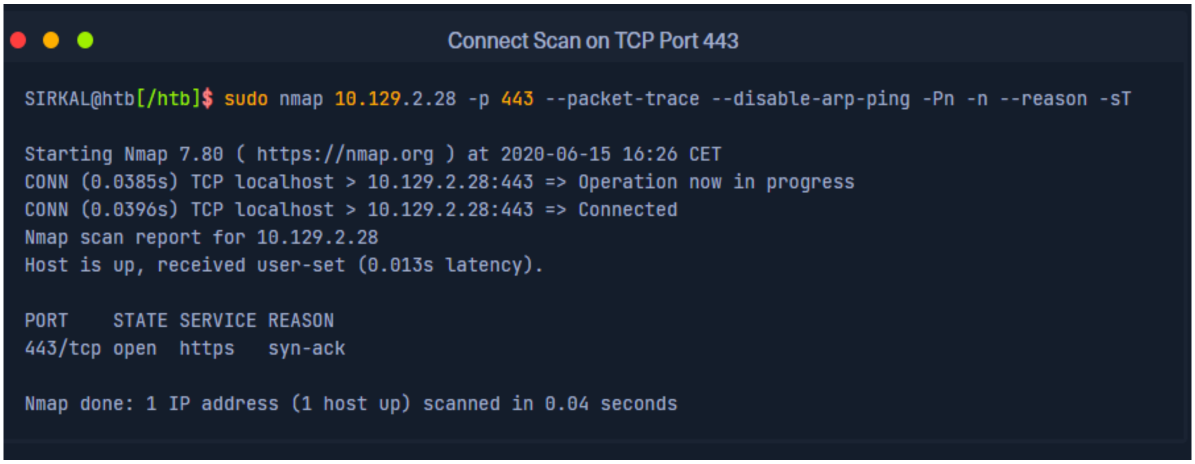
Nmap - Trace the Packets

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 21 --packet-trace -Pn -n --disable-arp-ping

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 15:39 CEST
SENT (0.0429s) TCP 10.10.14.2:63090 > 10.129.2.28:21 S ttl=56 id=57322 iplen=44  seq=1699105818 win=1024
RCVD (0.0573s) TCP 10.129.2.28:21 > 10.10.14.2:63090 RA ttl=64 id=0 iplen=40  seq=0 win=0
Nmap scan report for 10.11.1.28
Host is up (0.014s latency).

PORT   STATE  SERVICE
21/tcp closed ftp
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

3. Connect Scan

The Nmap TCP Connect Scan (-sT) uses the TCP three-way handshake to determine if a specific port on a target host is open or closed. The scan sends an SYN packet to the target port and waits for a response. It is considered open if the target port responds with an SYN-ACK packet and closed if it responds with an RST packet.

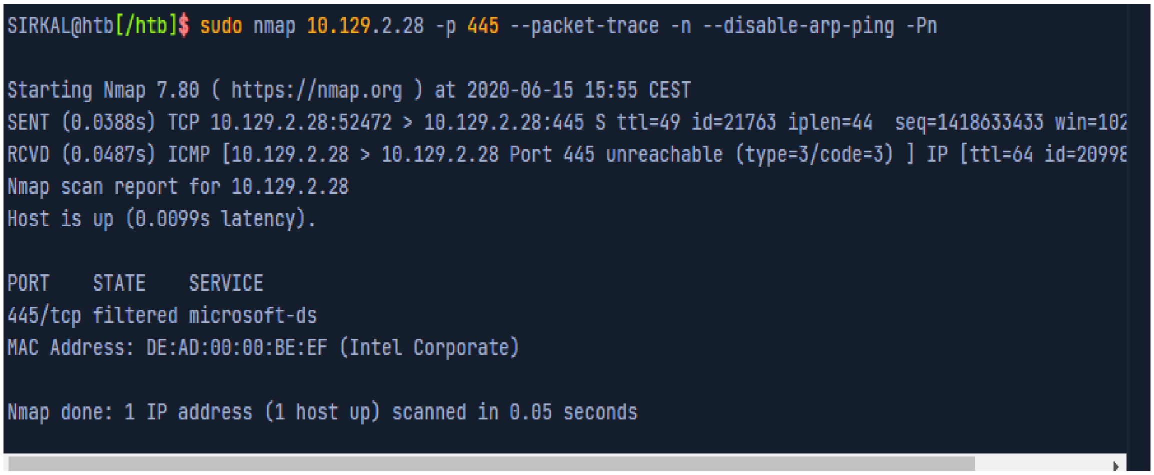
The Connect scan is useful because it is the most accurate way to determine the state of a port, and it is also the stealthiest. Unlike other types of scans, such as the SYN scan, the Connect scan does not leave any unfinished connections or unsent packets on the target host, which makes it less likely to be detected by intrusion detection systems (IDS) or intrusion prevention systems (IPS).


Connect Scan on TCP Port 443

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 443 --packet-trace --disable-arp-ping -Pn -n --reason -sT

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 16:26 CET
CONN (0.0385s) TCP localhost > 10.129.2.28:443 => Operation now in progress
CONN (0.0396s) TCP localhost > 10.129.2.28:443 => Connected
Nmap scan report for 10.129.2.28
Host is up, received user-set (0.013s latency).

PORT    STATE SERVICE REASON
443/tcp open  https   syn-ack

Nmap done: 1 IP address (1 host up) scanned in 0.04 seconds
```

4. Filtered Ports

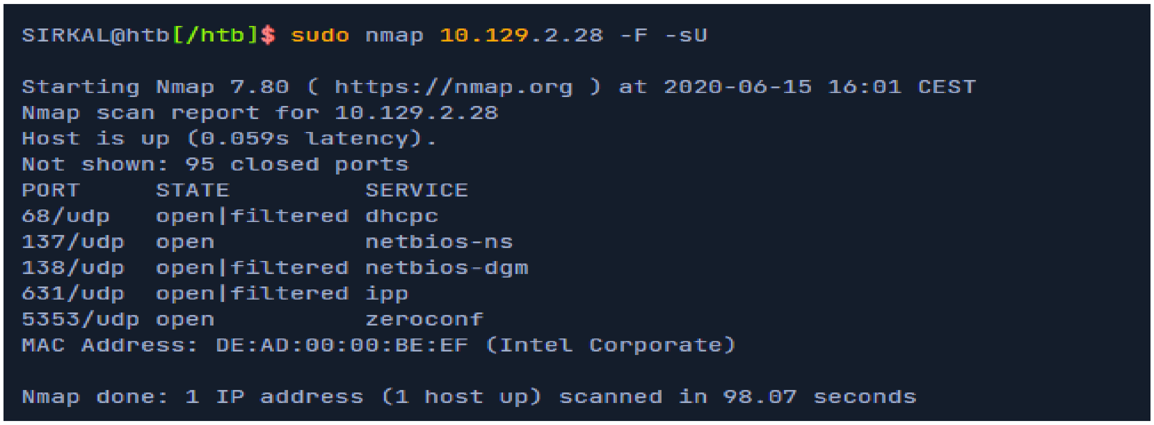
When a port is shown as filtered, it can have several reasons. In most cases, firewalls have certain rules set to handle specific connections. The packets can either be dropped, or rejected. When a packet gets dropped, Nmap receives no response from our target, and by default, the retry rate (--max-retries) is set to 1. This means Nmap will resend the request to the target port to determine if the previous packet was not accidentally mishandled. Let us look at an example where the firewall drops the TCP packets, we send for the port scan. Therefore, we scan the TCP port 139, which was already shown as filtered. To be able to track how our sent packets are handled, we deactivate the ICMP echo requests (-Pn), DNS resolution (-n), and ARP ping scan (--disable-arp-ping) again.



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 445 --packet-trace -n --disable-arp-ping -Pn

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 15:55 CEST
SENT (0.0388s) TCP 10.129.2.28:52472 > 10.129.2.28:445 S ttl=49 id=21763 iplen=44  seq=1418633433 win=102
RCVD (0.0487s) ICMP [10.129.2.28 > 10.129.2.28 Port 445 unreachable (type=3/code=3) ] IP [ttl=64 id=2099
Nmap scan report for 10.129.2.28
Host is up (0.0099s latency).

PORT    STATE    SERVICE
445/tcp filtered microsoft-ds
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds
```

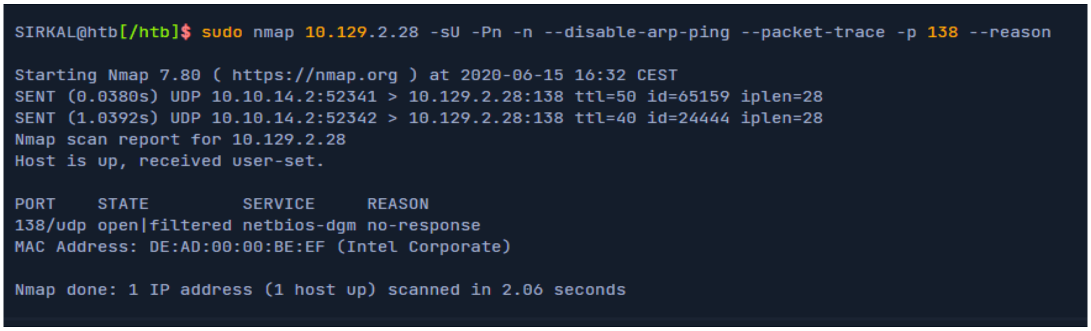
5. Discovering Open UDP Ports

Some system administrators sometimes forget to filter the UDP ports in addition to the TCP ones. Since UDP is a stateless protocol and does not require a three-way handshake like TCP. We do not receive any acknowledgment. Consequently, the timeout is much longer, making the whole UDP scan (-sU) much slower than the TCP scan (-sS).



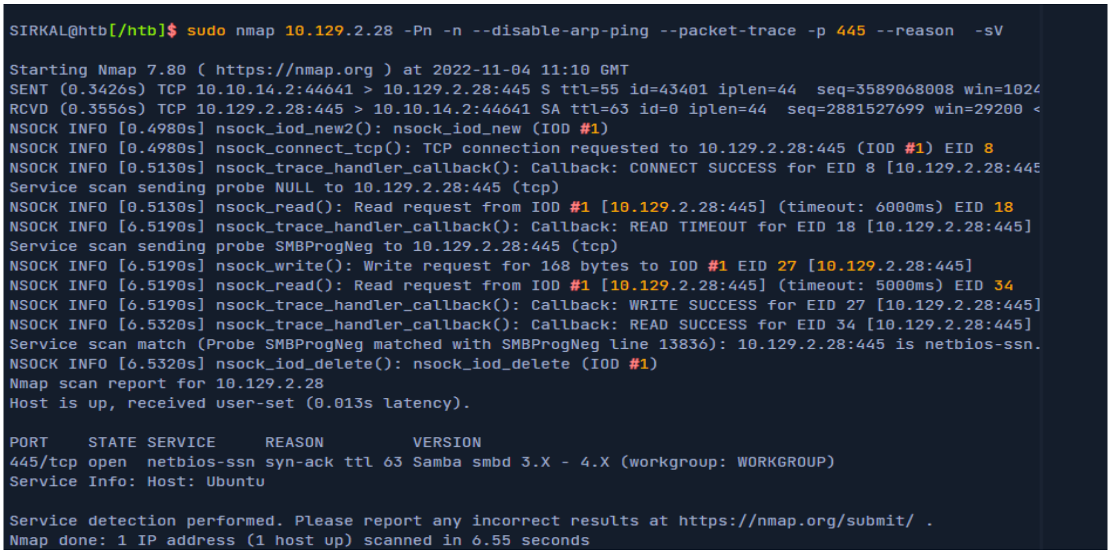
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -F -sU

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 16:01 CEST
Nmap scan report for 10.129.2.28
Host is up (0.059s latency).
Not shown: 95 closed ports
PORT      STATE         SERVICE
68/udp    open|filtered dhcpc
137/udp   open          netbios-ns
138/udp   open|filtered netbios-dgm
631/udp   open|filtered ipp
5353/udp open          zeroconf
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 98.07 seconds
```

The UDP effective scan



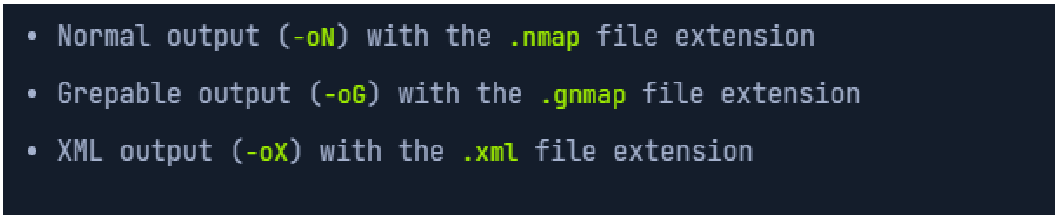
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -sU -Pn -n --disable-arp-ping --packet-trace -p 138 --reason

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 16:32 CEST
SENT (0.0380s) UDP 10.10.14.2:52341 > 10.129.2.28:138 ttl=50 id=65159 iplen=28
SENT (1.0392s) UDP 10.10.14.2:52342 > 10.129.2.28:138 ttl=40 id=24444 iplen=28
Nmap scan report for 10.129.2.28
Host is up, received user-set.

PORT    STATE         SERVICE     REASON
138/udp open|filtered netbios-dgm no-response
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 2.06 seconds
```

Version scanning



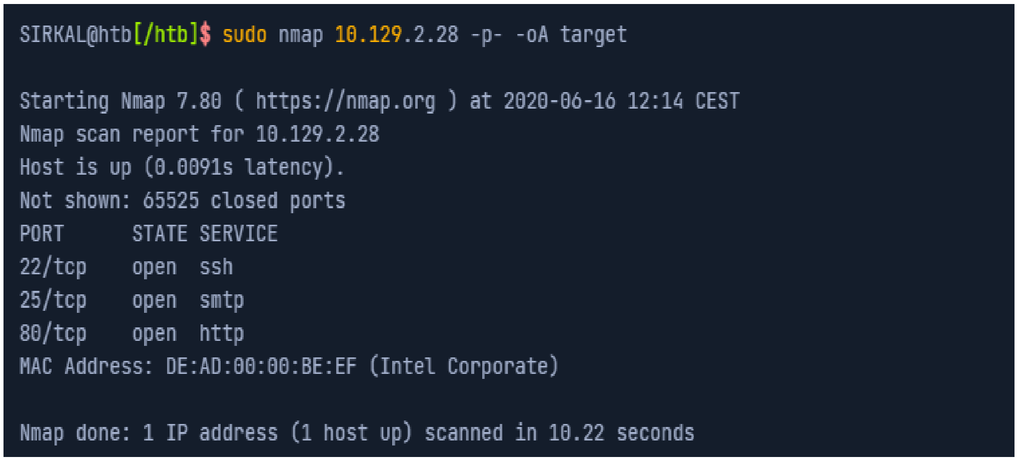
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -Pn -n --disable-arp-ping --packet-trace -p 445 --reason  -sV

Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-04 11:10 GMT
SENT (0.3426s) TCP 10.10.14.2:44641 > 10.129.2.28:445 S ttl=55 id=43401 iplen=44  seq=3589068008 win=1024
RCVD (0.3556s) TCP 10.129.2.28:445 > 10.10.14.2:44641 SA ttl=63 id=0 iplen=44  seq=2881527699 win=29200 <
NSOCK INFO [0.4980s] nsock_iod_new2(): nsock_iod_new (IOD #1)
NSOCK INFO [0.4980s] nsock_connect_tcp(): TCP connection requested to 10.129.2.28:445 (IOD #1) EID 8
NSOCK INFO [0.5130s] nsock_trace_handler_callback(): Callback: CONNECT SUCCESS for EID 8 [10.129.2.28:445
Service scan sending probe NULL to 10.129.2.28:445 (tcp)
NSOCK INFO [0.5130s] nsock_read(): Read request from IOD #1 [10.129.2.28:445] (timeout: 6000ms) EID 18
NSOCK INFO [6.5190s] nsock_trace_handler_callback(): Callback: READ TIMEOUT for EID 18 [10.129.2.28:445]
Service scan sending probe SMBProgNeg to 10.129.2.28:445 (tcp)
NSOCK INFO [6.5190s] nsock_write(): Write request for 168 bytes to IOD #1 EID 27 [10.129.2.28:445]
NSOCK INFO [6.5190s] nsock_read(): Read request from IOD #1 [10.129.2.28:445] (timeout: 5000ms) EID 34
NSOCK INFO [6.5190s] nsock_trace_handler_callback(): Callback: WRITE SUCCESS for EID 27 [10.129.2.28:445]
NSOCK INFO [6.5320s] nsock_trace_handler_callback(): Callback: READ SUCCESS for EID 34 [10.129.2.28:445]
Service scan match (Probe SMBProgNeg matched with SMBProgNeg line 13836): 10.129.2.28:445 is netbios-ssn.
NSOCK INFO [6.5320s] nsock_iod_delete(): nsock_iod_delete (IOD #1)
Nmap scan report for 10.129.2.28
Host is up, received user-set (0.013s latency).

PORT    STATE SERVICE     REASON         VERSION
445/tcp open  netbios-ssn syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: Ubuntu

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.55 seconds
```

SAVING THE RESULT



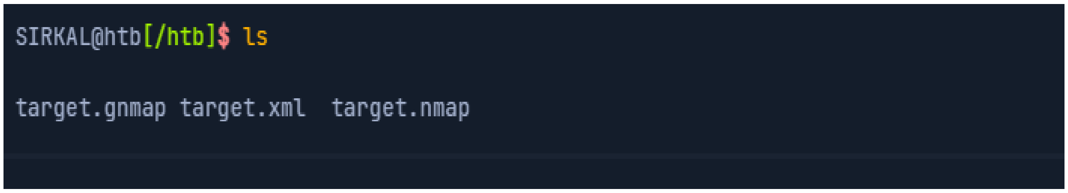
```
• Normal output (-oN) with the .nmap file extension
• Grepable output (-oG) with the .gnmap file extension
• XML output (-oX) with the .xml file extension
```

a. NORMAL O/P



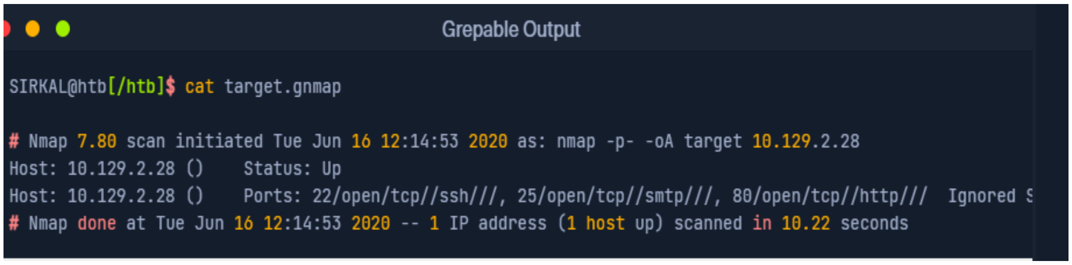
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p- -oA target

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 12:14 CEST
Nmap scan report for 10.129.2.28
Host is up (0.0091s latency).
Not shown: 65525 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp
80/tcp    open  http
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 10.22 seconds
```

VIEW STORED FILES



```
SIRKAL@htb[/htb]$ ls

target.gnmap target.xml  target.nmap
```

b. Grapable o/p



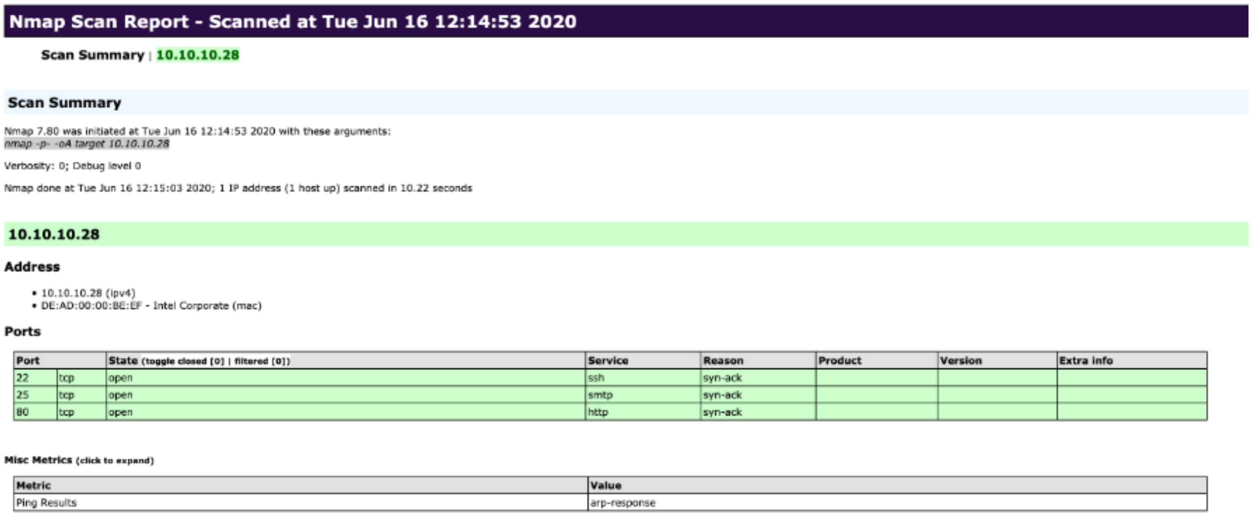
Grepable Output

```
SIRKAL@htb[/htb]$ cat target.gnmap

# Nmap 7.80 scan initiated Tue Jun 16 12:14:53 2020 as: nmap -p- -oA target 10.129.2.28
Host: 10.129.2.28 ()    Status: Up
Host: 10.129.2.28 ()    Ports: 22/open/tcp//ssh///, 25/open/tcp//smtp///, 80/open/tcp//http///  Ignored S
# Nmap done at Tue Jun 16 12:14:53 2020 -- 1 IP address (1 host up) scanned in 10.22 seconds
```

c. Style sheets- the common approach

With the XML output, we can easily create HTML reports that are easy to read, even for non-technical people. This is later very useful for documentation, as it presents our results in a detailed and clear way. To convert the stored results from XML format to HTML, we can use the tool xsltproc.

Note: -ST is used for full TCP port scanning



**Nmap Scan Report - Scanned at Tue Jun 16 12:14:53 2020**

**Scan Summary** | **10.10.10.28**

**Scan Summary**

Nmap 7.80 was initiated at Tue Jun 16 12:14:53 2020 with these arguments:
*nmap -p- -oA target 10.10.10.28*

Verbosity: 0; Debug level 0

Nmap done at Tue Jun 16 12:15:03 2020; 1 IP address (1 host up) scanned in 10.22 seconds

**10.10.10.28**

**Address**

- 10.10.10.28 (ipv4)
- DE:AD:00:00:BE:EF - Intel Corporate (mac)

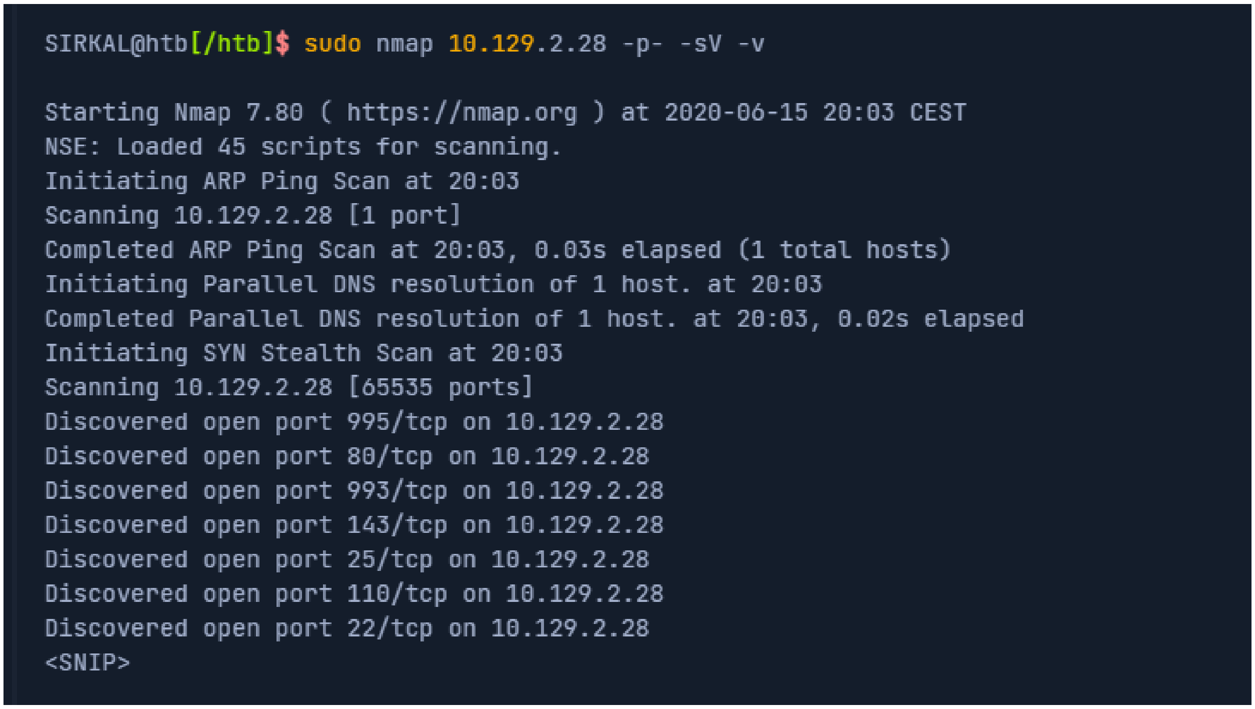
**Ports**

| Port | | State (toggle closed [0] \| filtered [0]) | Service | Reason | Product | Version | Extra info |
|---|---|---|---|---|---|---|---|
| 22 | tcp | open | ssh | syn-ack | | | |
| 25 | tcp | open | smtp | syn-ack | | | |
| 80 | tcp | open | http | syn-ack | | | |

**Misc Metrics** (click to expand)

| Metric | Value |
|---|---|
| Ping Results | arp-response |

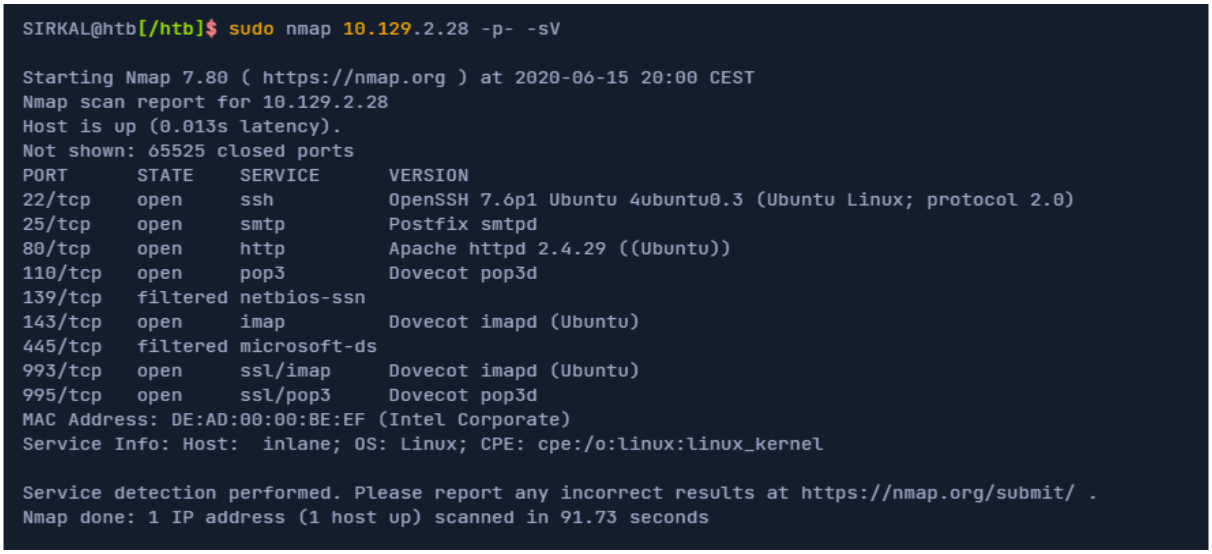
6. SERVICE VERSION DETECTION
   a. Normal scanning



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p- -sV -v

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 20:03 CEST
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 20:03
Scanning 10.129.2.28 [1 port]
Completed ARP Ping Scan at 20:03, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 20:03
Completed Parallel DNS resolution of 1 host. at 20:03, 0.02s elapsed
Initiating SYN Stealth Scan at 20:03
Scanning 10.129.2.28 [65535 ports]
Discovered open port 995/tcp on 10.129.2.28
Discovered open port 80/tcp on 10.129.2.28
Discovered open port 993/tcp on 10.129.2.28
Discovered open port 143/tcp on 10.129.2.28
Discovered open port 25/tcp on 10.129.2.28
Discovered open port 110/tcp on 10.129.2.28
Discovered open port 22/tcp on 10.129.2.28
<SNIP>
```

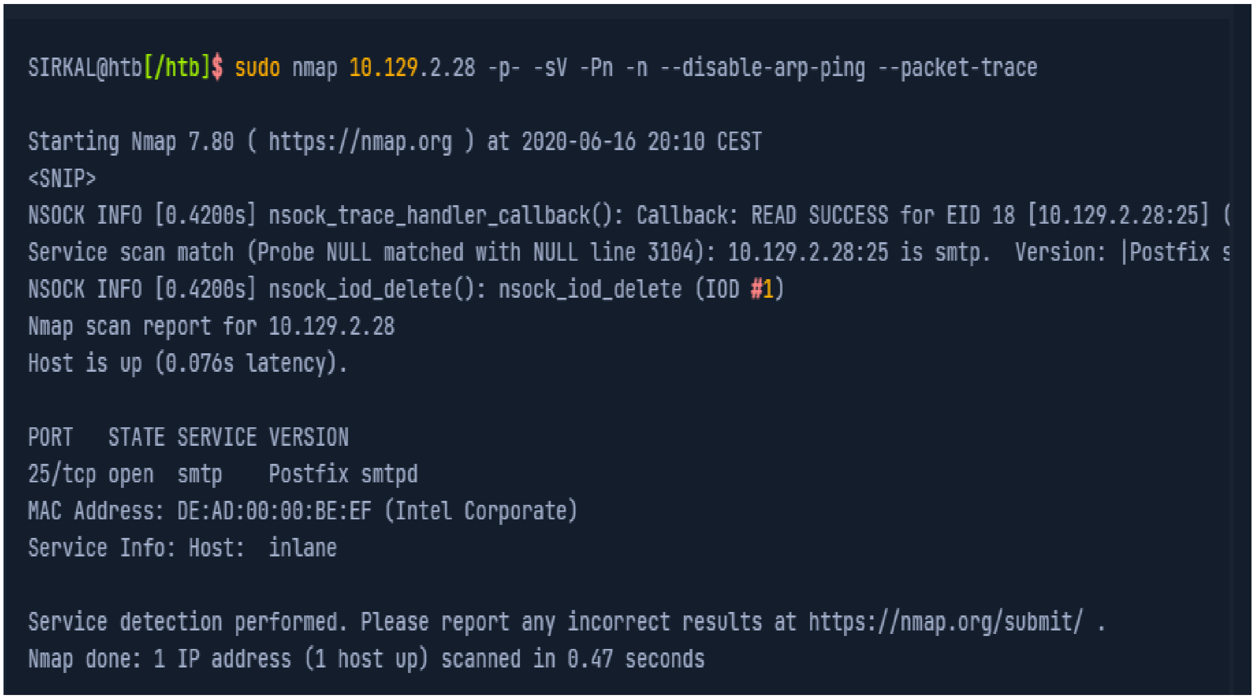
   b. Banner Grabbing

Once the scan is complete, we will see all TCP ports with the corresponding service and their versions that are active on the system.



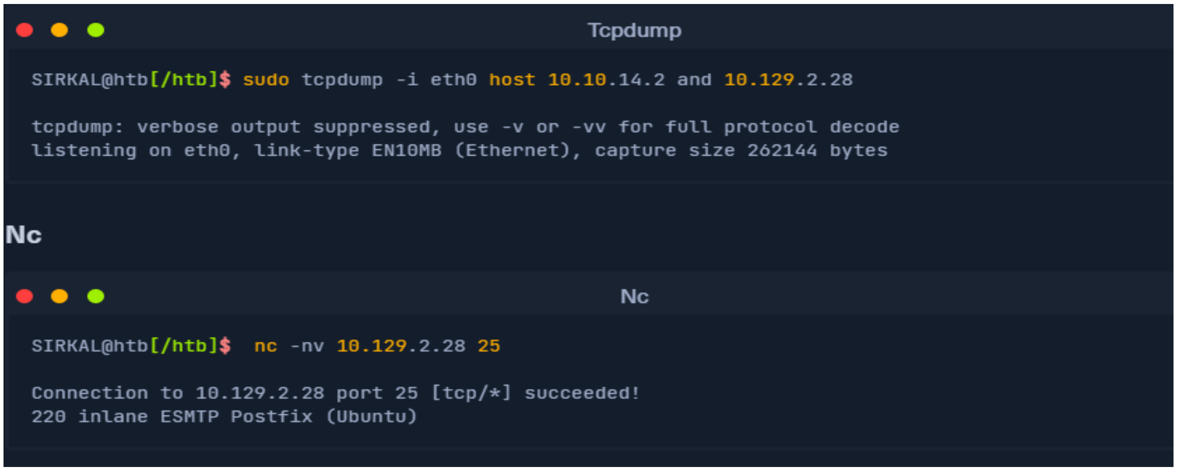
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p- -sV

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-15 20:00 CEST
Nmap scan report for 10.129.2.28
Host is up (0.013s latency).
Not shown: 65525 closed ports
PORT      STATE    SERVICE      VERSION
22/tcp    open     ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
25/tcp    open     smtp         Postfix smtpd
80/tcp    open     http         Apache httpd 2.4.29 ((Ubuntu))
110/tcp   open     pop3         Dovecot pop3d
139/tcp   filtered netbios-ssn
143/tcp   open     imap         Dovecot imapd (Ubuntu)
445/tcp   filtered microsoft-ds
993/tcp   open     ssl/imap     Dovecot imapd (Ubuntu)
995/tcp   open     ssl/pop3     Dovecot pop3d
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)
Service Info: Host:  inlane; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 91.73 seconds
```

Primarily, Nmap looks at the banners of the scanned ports and prints them out. If it cannot identify versions through the banners, Nmap attempts to identify them through a signature-based matching system, but this significantly increases the scan's duration. One disadvantage to Nmap's presented results is that the automatic scan can miss some information because sometimes Nmap does not know how to handle it. Let us look at an example of this.



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p- -sV -Pn -n --disable-arp-ping --packet-trace

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 20:10 CEST
<SNIP>
NSOCK INFO [0.4200s] nsock_trace_handler_callback(): Callback: READ SUCCESS for EID 18 [10.129.2.28:25] (
Service scan match (Probe NULL matched with NULL line 3104): 10.129.2.28:25 is smtp.  Version: |Postfix s
NSOCK INFO [0.4200s] nsock_iod_delete(): nsock_iod_delete (IOD #1)
Nmap scan report for 10.129.2.28
Host is up (0.076s latency).

PORT   STATE SERVICE VERSION
25/tcp open  smtp    Postfix smtpd
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)
Service Info: Host:  inlane

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
```

c. TCPDUMPING AND NC



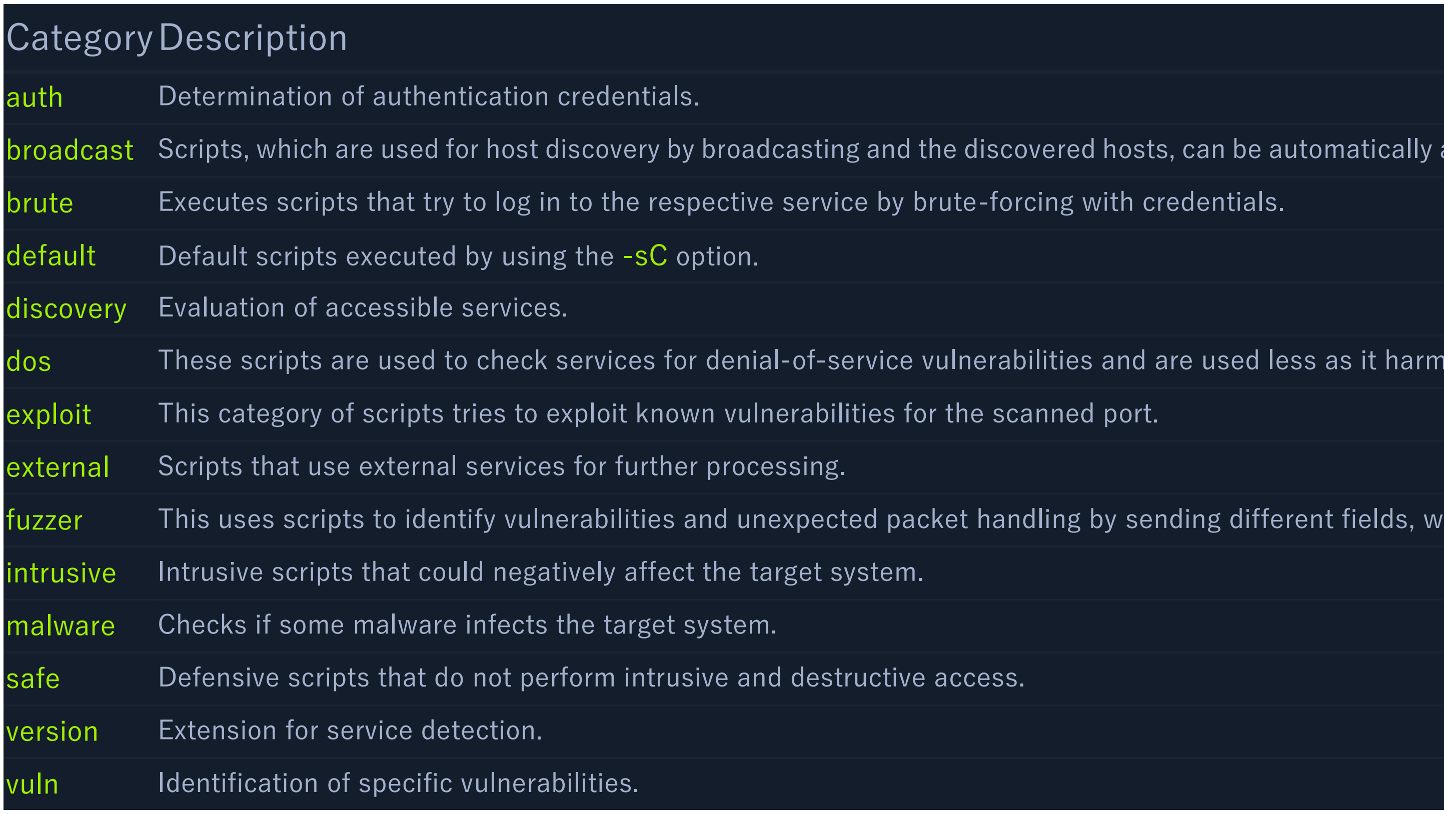
```
SIRKAL@htb[/htb]$ sudo tcpdump -i eth0 host 10.10.14.2 and 10.129.2.28

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Nc

```
SIRKAL@htb[/htb]$  nc -nv 10.129.2.28 25

Connection to 10.129.2.28 port 25 [tcp/*] succeeded!
220 inlane ESMTP Postfix (Ubuntu)
```

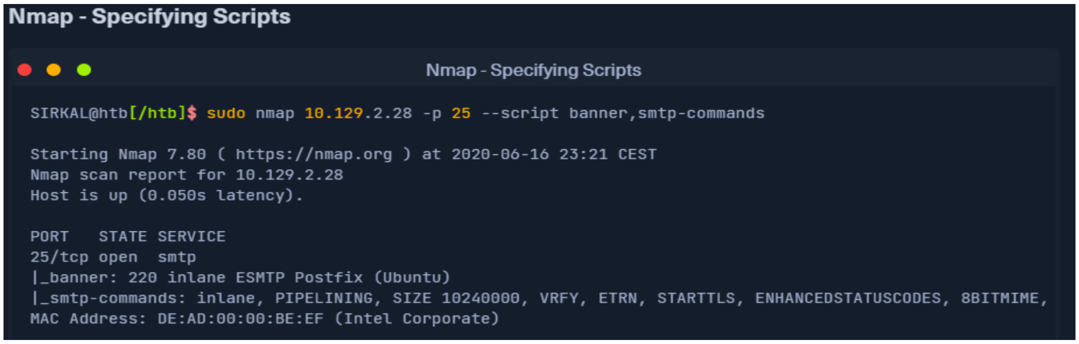
## PART 3. Nmap Scripting Engine

Nmap Scripting Engine (NSE) is another handy feature of Nmap. It provides us with the possibility to create scripts in Lua for interaction with certain services. There are a total of 14 categories into which these scripts can be divided:



| Category | Description |
|---|---|
| auth | Determination of authentication credentials. |
| broadcast | Scripts, which are used for host discovery by broadcasting and the discovered hosts, can be automatically a |
| brute | Executes scripts that try to log in to the respective service by brute-forcing with credentials. |
| default | Default scripts executed by using the -sC option. |
| discovery | Evaluation of accessible services. |
| dos | These scripts are used to check services for denial-of-service vulnerabilities and are used less as it harm |
| exploit | This category of scripts tries to exploit known vulnerabilities for the scanned port. |
| external | Scripts that use external services for further processing. |
| fuzzer | This uses scripts to identify vulnerabilities and unexpected packet handling by sending different fields, w |
| intrusive | Intrusive scripts that could negatively affect the target system. |
| malware | Checks if some malware infects the target system. |
| safe | Defensive scripts that do not perform intrusive and destructive access. |
| version | Extension for service detection. |
| vuln | Identification of specific vulnerabilities. |

Specifying the scripting

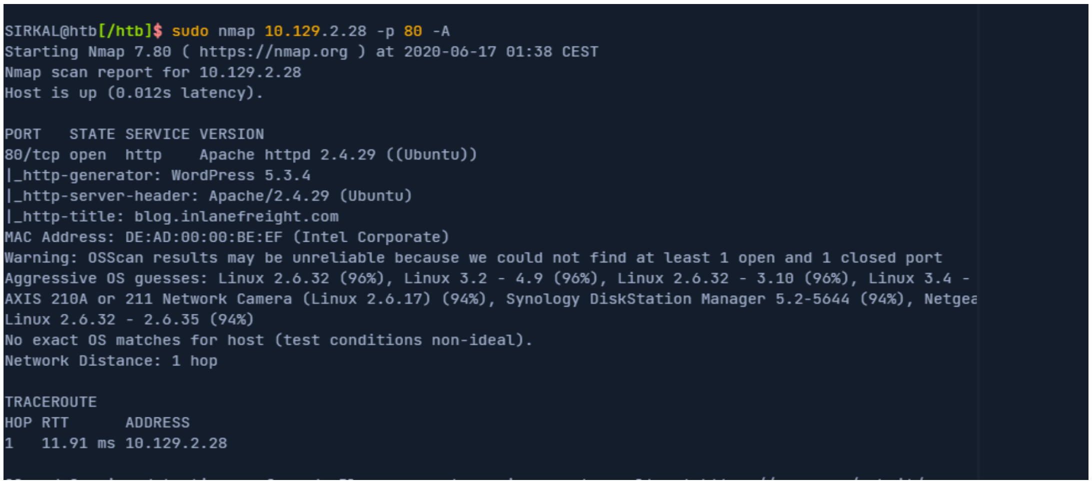


Nmap - Specifying Scripts

Nmap - Specifying Scripts

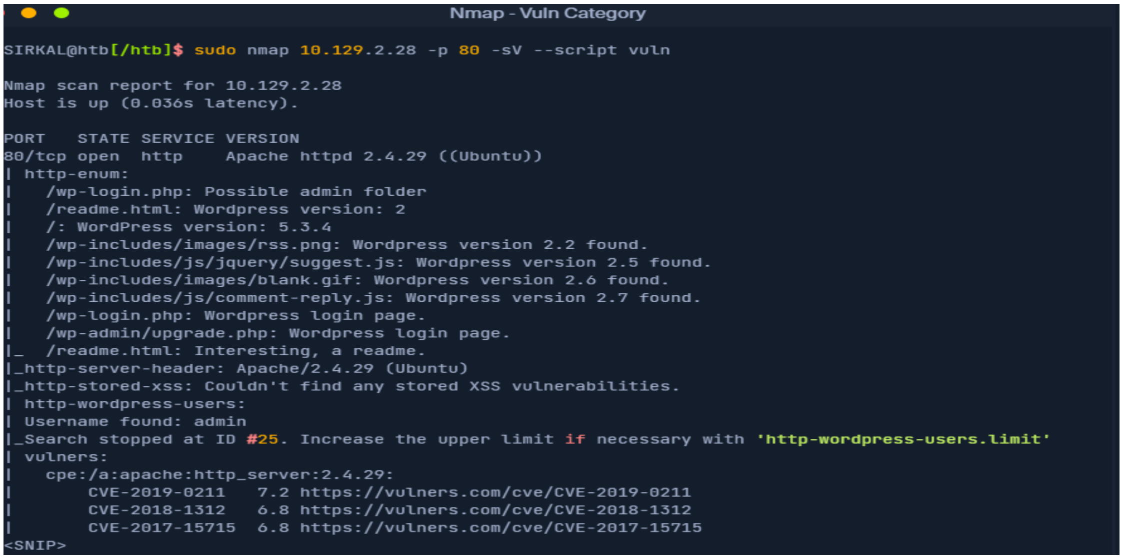
```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 25 --script banner,smtp-commands

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-16 23:21 CEST
Nmap scan report for 10.129.2.28
Host is up (0.050s latency).

PORT   STATE SERVICE
25/tcp open  smtp
|_banner: 220 inlane ESMTP Postfix (Ubuntu)
|_smtp-commands: inlane, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME,
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)
```

When the issue is critical aggressive scan can be used



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 80 -A
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-17 01:38 CEST
Nmap scan report for 10.129.2.28
Host is up (0.012s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-generator: WordPress 5.3.4
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: blog.inlanefreight.com
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 2.6.32 (96%), Linux 3.2 - 4.9 (96%), Linux 2.6.32 - 3.10 (96%), Linux 3.4 -
AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), Synology DiskStation Manager 5.2-5644 (94%), Netgea
Linux 2.6.32 - 2.6.35 (94%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

TRACEROUTE
HOP RTT      ADDRESS
1   11.91 ms 10.129.2.28
```

Vulnerability script running



Nmap - Vuln Category

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 80 -sV --script vuln

Nmap scan report for 10.129.2.28
Host is up (0.036s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
| http-enum:
|   /wp-login.php: Possible admin folder
|   /readme.html: Wordpress version: 2
|   /: WordPress version: 5.3.4
|   /wp-includes/images/rss.png: Wordpress version 2.2 found.
|   /wp-includes/js/jquery/suggest.js: Wordpress version 2.5 found.
|   /wp-includes/images/blank.gif: Wordpress version 2.6 found.
|   /wp-includes/js/comment-reply.js: Wordpress version 2.7 found.
|   /wp-login.php: Wordpress login page.
|   /wp-admin/upgrade.php: Wordpress login page.
|_  /readme.html: Interesting, a readme.
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-wordpress-users:
| Username found: admin
|_Search stopped at ID #25. Increase the upper limit if necessary with 'http-wordpress-users.limit'
| vulners:
|   cpe:/a:apache:http_server:2.4.29:
|       CVE-2019-0211   7.2 https://vulners.com/cve/CVE-2019-0211
|       CVE-2018-1312   6.8 https://vulners.com/cve/CVE-2018-1312
|       CVE-2017-15715  6.8 https://vulners.com/cve/CVE-2017-15715
<SNIP>
```

Performance

Scanning performance plays a significant role when we need to scan an extensive network or are dealing with low network bandwidth. We can use various options to tell Nmap how fast (-T <0-5>), with which frequency (--min-parallelism <number>), which timeouts (--max-rtt-timeout <time>) the test packets should have, how many packets should be sent simultaneously (--min-rate <number>), and with the number of retries (--max-retries <number>) for the scanned ports the targets should be scanned.

## PART 4: BYPASSING SECURITY INFRASTRUCTURES

Nmap gives us many different ways to bypass firewalls rules and IDS/IPS. These methods include the fragmentation of packets, the use of decoys, and others that we will discuss in this section

### Firewalls

A firewall is a security measure against unauthorized connection attempts from external networks. Every firewall security system is based on a software component that monitors network traffic between the firewall and incoming data connections and decides how to handle the connection based on the rules that have been set. It checks whether individual network packets are being passed, ignored, or blocked. This mechanism is designed to prevent unwanted connections that could be potentially dangerous.

### IDS/IPS

Like the firewall, the intrusion detection system (IDS) and intrusion prevention system (IPS) are also software-based components. IDS scans the network for potential attacks, analyzes them, and reports any detected attacks. IPS complements IDS by taking specific defensive measures if a potential attack should have been detected. The analysis of such attacks is based on pattern matching and signatures. If specific patterns are detected, such as a service detection scan, IPS may prevent the pending connection attempts.

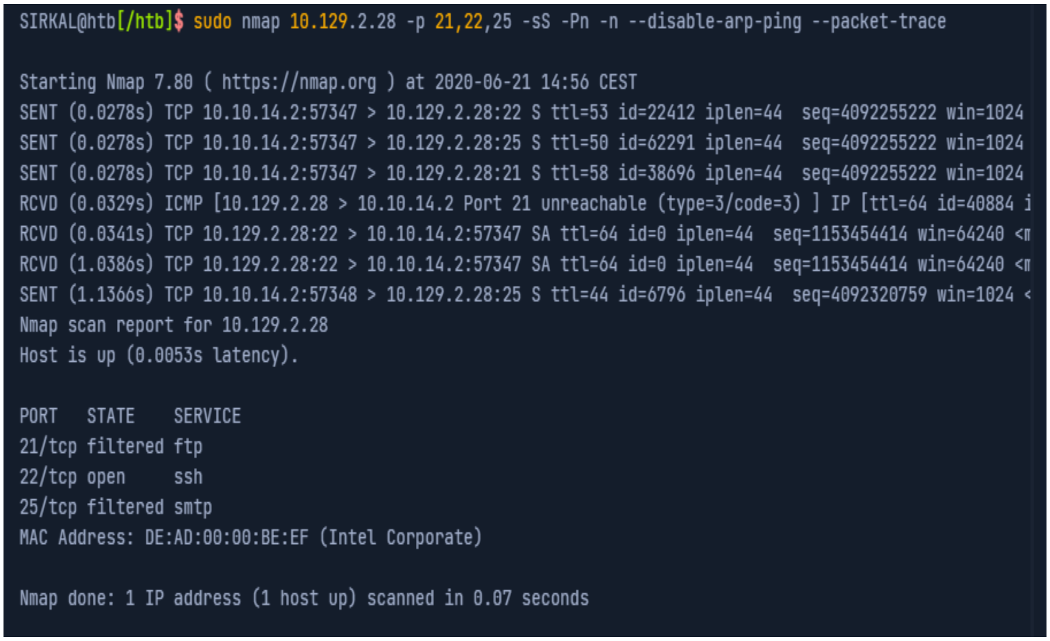
### Determine Firewalls and Their Rules

We already know that when a port is shown as filtered, it can have several reasons. In most cases, firewalls have certain rules set to handle specific connections. The packets can either be dropped, or rejected. The dropped packets are ignored, and no response is returned from the host. This is different for rejected packets that are returned with an RST flag. These packets contain different types of ICMP error codes or contain nothing at all.

Such errors can be:
Net Unreachable
Net Prohibited
Host Unreachable
Host Prohibited
Port Unreachable
Proto Unreachable

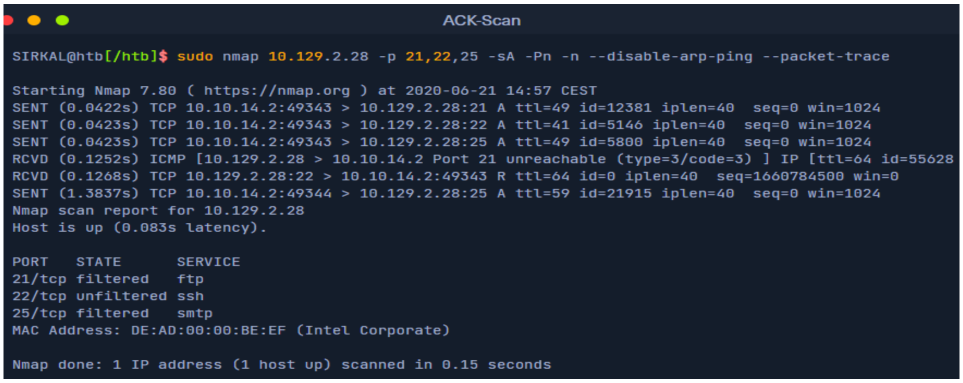
Nmap's TCP ACK scan (-sA) method is much harder to filter for firewalls and IDS/IPS systems than regular SYN (-sS) or Connect scans (sT) because they only send a TCP packet with only the ACK flag. When a port is closed or open, the host must respond with an RST flag. Unlike outgoing connections, all connection attempts (with the SYN flag) from external networks are usually blocked by firewalls. However, the packets with the ACK flag are often passed by the firewall because the firewall cannot determine whether the connection was first established from the external network or the internal network.

1. SYK SCAN



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 21,22,25 -sS -Pn -n --disable-arp-ping --packet-trace

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-21 14:56 CEST
SENT (0.0278s) TCP 10.10.14.2:57347 > 10.129.2.28:22 S ttl=53 id=22412 iplen=44  seq=4092255222 win=1024
SENT (0.0278s) TCP 10.10.14.2:57347 > 10.129.2.28:25 S ttl=50 id=62291 iplen=44  seq=4092255222 win=1024
SENT (0.0278s) TCP 10.10.14.2:57347 > 10.129.2.28:21 S ttl=58 id=38696 iplen=44  seq=4092255222 win=1024
RCVD (0.0329s) ICMP [10.129.2.28 > 10.10.14.2 Port 21 unreachable (type=3/code=3) ] IP [ttl=64 id=40884 i
RCVD (0.0341s) TCP 10.129.2.28:22 > 10.10.14.2:57347 SA ttl=64 id=0 iplen=44  seq=1153454414 win=64240 <m
RCVD (1.0386s) TCP 10.129.2.28:22 > 10.10.14.2:57347 SA ttl=64 id=0 iplen=44  seq=1153454414 win=64240 <m
SENT (1.1366s) TCP 10.10.14.2:57348 > 10.129.2.28:25 S ttl=44 id=6796 iplen=44  seq=4092320759 win=1024 <
Nmap scan report for 10.129.2.28
Host is up (0.0053s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
22/tcp open     ssh
25/tcp filtered smtp
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

2. ACK SCAN



```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 21,22,25 -sA -Pn -n --disable-arp-ping --packet-trace

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-21 14:57 CEST
SENT (0.0422s) TCP 10.10.14.2:49343 > 10.129.2.28:21 A ttl=49 id=12381 iplen=40  seq=0 win=1024
SENT (0.0423s) TCP 10.10.14.2:49343 > 10.129.2.28:22 A ttl=41 id=5146 iplen=40  seq=0 win=1024
SENT (0.0423s) TCP 10.10.14.2:49343 > 10.129.2.28:25 A ttl=49 id=5800 iplen=40  seq=0 win=1024
RCVD (0.1252s) ICMP [10.129.2.28 > 10.10.14.2 Port 21 unreachable (type=3/code=3) ] IP [ttl=64 id=55628
RCVD (0.1268s) TCP 10.129.2.28:22 > 10.10.14.2:49343 R ttl=64 id=0 iplen=40  seq=1660784500 win=0
SENT (1.3837s) TCP 10.10.14.2:49344 > 10.129.2.28:25 A ttl=59 id=21915 iplen=40  seq=0 win=1024
Nmap scan report for 10.129.2.28
Host is up (0.083s latency).

PORT   STATE      SERVICE
21/tcp filtered   ftp
22/tcp unfiltered ssh
25/tcp filtered   smtp
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.15 seconds
```

### Detecting IDS/IPS

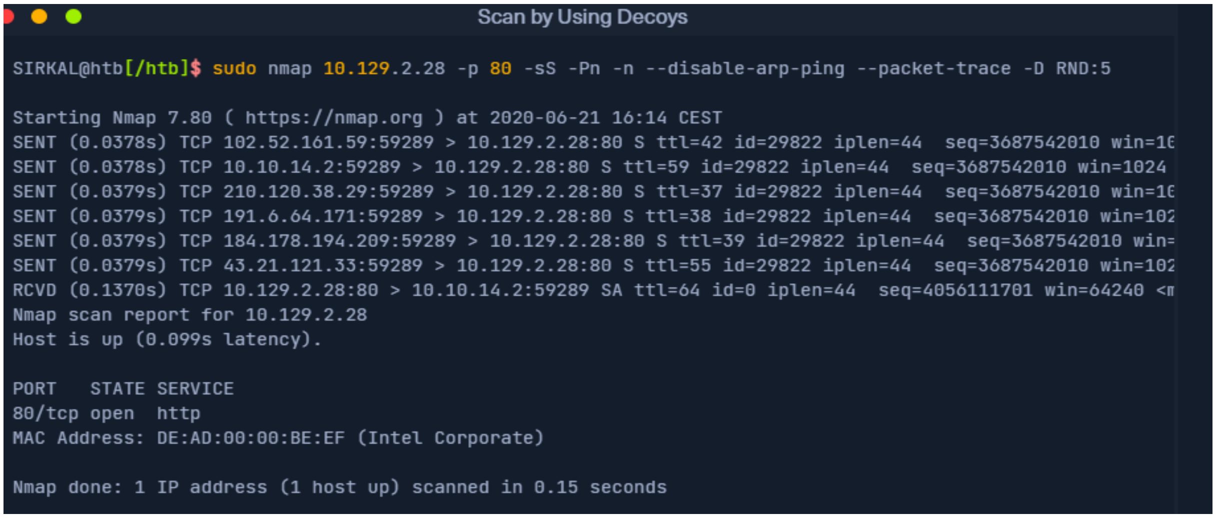
Unlike firewalls and their rules, the detection of IDS/IPS systems is much more difficult because these are passive traffic monitoring systems. IDS systems examine all connections between hosts. If the IDS finds packets containing the defined contents or specifications, the administrator is notified and takes appropriate action in the worst case.

IPS systems take measures configured by the administrator independently to prevent potential attacks automatically. It is essential to know that IDS and IPS are different applications and that IPS serves as a complement to IDS.

One method to determine whether such IPS system is present in the target network is to scan from a single host (VPS). If at any time this host is blocked and has no access to the target network, we know that the administrator has taken some security measures. Accordingly, we can continue our penetration test with another VPS.

### Scanning with Decoys

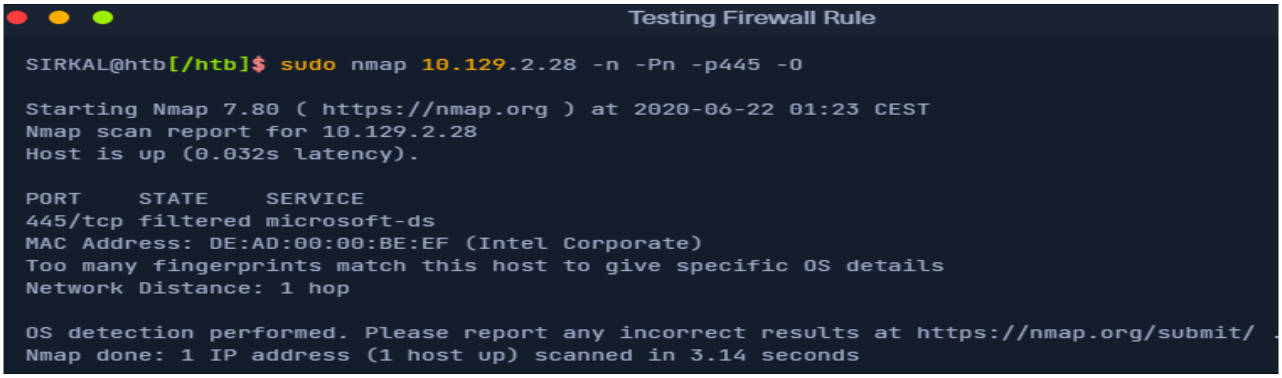
There are cases in which administrators block specific subnets from different regions in principle. This prevents any access to the target network. Another example is when IPS should block us. For this reason, the Decoy scanning method (-D) is the right choice. With this method, Nmap generates various random IP addresses inserted into the IP header to disguise the origin of the packet sent. With this method, we can generate random (RND) a specific number (for example: 5) of IP addresses separated by a colon (:). Our real IP address is then randomly placed between the generated IP addresses. In the next example, our real IP address is therefore placed in the second position. Another critical point is that the decoys must be alive. Otherwise, the service on the target may be unreachable due to SYN-flooding security mechanisms.



Scan by Using Decoys

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -p 80 -sS -Pn -n --disable-arp-ping --packet-trace -D RND:5

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-21 16:14 CEST
SENT (0.0378s) TCP 102.52.161.59:59289 > 10.129.2.28:80 S ttl=42 id=29822 iplen=44  seq=3687542010 win=10
SENT (0.0378s) TCP 10.10.14.2:59289 > 10.129.2.28:80 S ttl=59 id=29822 iplen=44  seq=3687542010 win=1024
SENT (0.0379s) TCP 210.120.38.29:59289 > 10.129.2.28:80 S ttl=37 id=29822 iplen=44  seq=3687542010 win=10
SENT (0.0379s) TCP 191.6.64.171:59289 > 10.129.2.28:80 S ttl=38 id=29822 iplen=44  seq=3687542010 win=102
SENT (0.0379s) TCP 184.178.194.209:59289 > 10.129.2.28:80 S ttl=39 id=29822 iplen=44  seq=3687542010 win=
SENT (0.0379s) TCP 43.21.121.33:59289 > 10.129.2.28:80 S ttl=55 id=29822 iplen=44  seq=3687542010 win=102
RCVD (0.1370s) TCP 10.129.2.28:80 > 10.10.14.2:59289 SA ttl=64 id=0 iplen=44  seq=4056111701 win=64240 <m
Nmap scan report for 10.129.2.28
Host is up (0.099s latency).

PORT   STATE SERVICE
80/tcp open  http
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.15 seconds
```

Another scenario would be that only individual subnets would not have access to the server's specific services. So, we can also manually specify the source IP address (-S) to test if we get better results with this one. Decoys can be used for SYN, ACK, ICMP scans, and OS detection scans. So let us look at such an example and determine which operating system it is most likely to be.



Testing Firewall Rule

```
SIRKAL@htb[/htb]$ sudo nmap 10.129.2.28 -n -Pn -p445 -O

Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-22 01:23 CEST
Nmap scan report for 10.129.2.28
Host is up (0.032s latency).

PORT    STATE    SERVICE
445/tcp filtered microsoft-ds
MAC Address: DE:AD:00:00:BE:EF (Intel Corporate)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.14 seconds
```