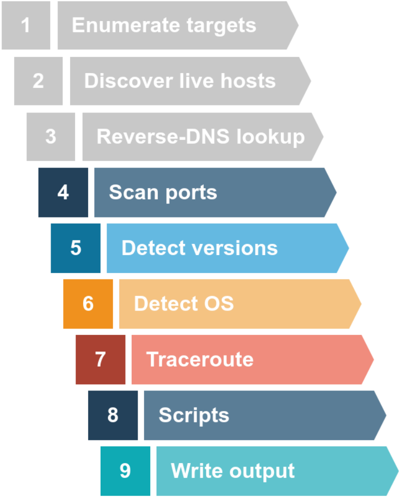
Nmap Basic Port Scans

Introduction

In the previous room, we focused on discovering online systems. So far, we have covered three steps of a Nmap scan:

1. Enumerate targets
2. Discover live hosts
3. Reverse-DNS lookup



The next step would be checking which ports are open and listening and which ports are closed. Therefore, in this room and the next one, we focus on port scanning and the different types of port scans used by nmap. This room explains:

- TCP connect port scan
- TCP SYN port scan
- UDP port scan

Moreover, we discuss the different options to specify the ports, the scan rate, and the number of parallel probes.

TCP and UDP Ports

In the same sense that an IP address specifies a host on a network among many others, a TCP port or UDP port is used to identify a network service running on that host. A server provides the network service, and it adheres to a specific network protocol. Examples include providing time, responding to DNS queries, and serving web pages. A port is usually linked to a service using that specific port number. For instance, an HTTP server would bind to TCP port 80 by default; moreover, if the HTTP server supports SSL/TLS, it would listen on TCP port 443. (TCP ports 80 and 443 are the default ports for HTTP and HTTPS; however, the webserver administrator might choose other port numbers if necessary.) Furthermore, no more than one service can listen on any TCP or UDP port (on the same IP address).

At the risk of oversimplification, we can classify ports in two states:

1. Open port indicates that there is some service listening on that port.
2. Closed port indicates that there is no service listening on that port.

However, in practical situations, we need to consider the impact of firewalls. For instance, a port might be open, but a firewall might be blocking the packets. Therefore, Nmap considers the following six states:

1. Open: indicates that a service is listening on the specified port.
2. Closed: indicates that no service is listening on the specified port, although the port is accessible. By accessible, we mean that it is reachable and is not blocked by a firewall or other security appliances/programs.
3. Filtered: means that Nmap cannot determine if the port is open or closed because the port is not accessible. This state is usually due to a firewall preventing Nmap from reaching that port. Nmap's packets may be blocked from reaching the port; alternatively, the responses are blocked from reaching Nmap's host.
4. Unfiltered: means that Nmap cannot determine if the port is open or closed, although the port is accessible. This state is encountered when using an ACK scan -sA.
5. Open|Filtered: This means that Nmap cannot determine whether the port is open or filtered.
6. Closed|Filtered: This means that Nmap cannot decide whether a port is closed or filtered.

****

Answer the questions below:

Which service uses UDP port 53 by default?
Answer: DNS

Which service uses TCP port 22 by default?
Answer: SSH

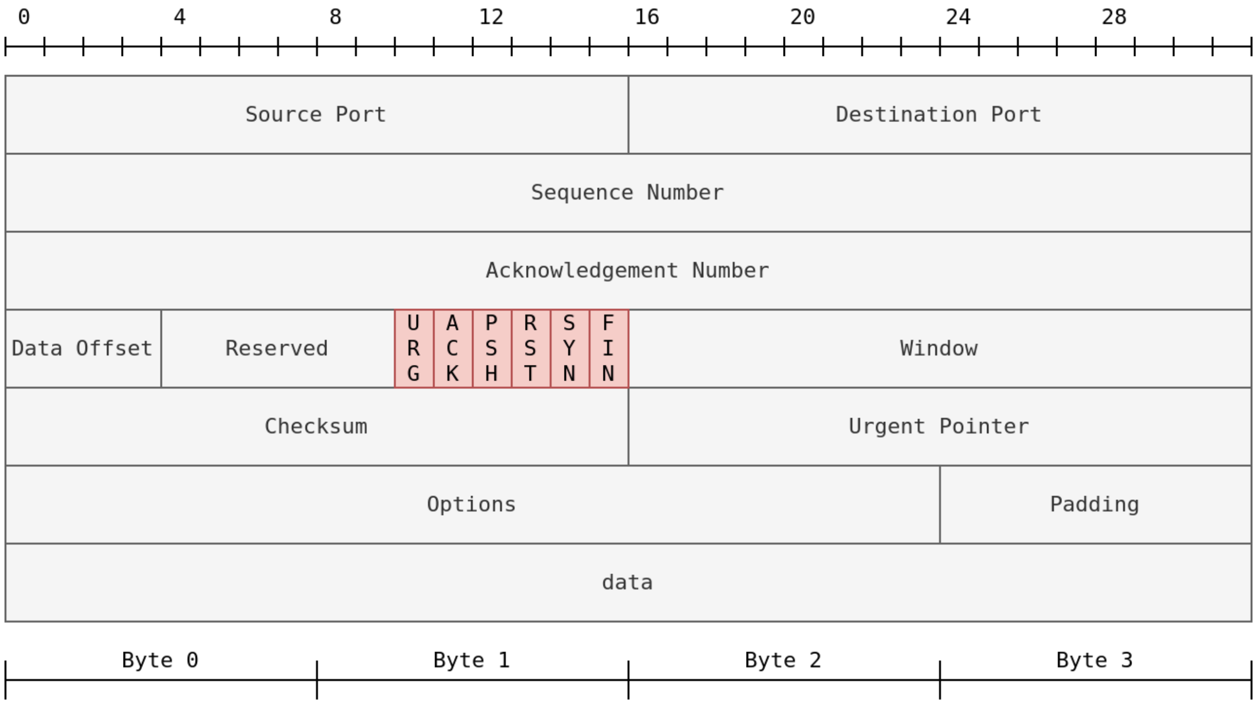
How many port states does Nmap consider?
Answer: 6

Which port state is the most interesting to discover as a pentester?
Answer: Open
****

TCP Flags

Nmap supports different types of TCP port scans. To understand the difference between these port scans, we need to review the TCP header. The TCP header is the first 24 bytes of a TCP segment. The following figure shows the TCP header as defined in RFC 793. This figure looks sophisticated at first; however, it is pretty simple to understand. In the first row, we have the source TCP port number and the destination port number. We can see that the port number is allocated 16 bits (2 bytes). In the second and third rows, we have the sequence number and the acknowledgement number. Each row has 32 bits (4 bytes) allocated, with six rows total, making up 24 bytes.

TCP Header (RFC793)



In particular, we need to focus on the flags that Nmap can set or unset. We have highlighted the TCP flags in red. Setting a flag bit means setting its value to 1. From left to right, the TCP header flags are:

1. URG: Urgent flag indicates that the urgent pointer filed is significant. The urgent pointer indicates that the incoming data is urgent, and that a TCP segment with the URG flag set is processed immediately without consideration of having to wait on previously sent TCP segments.
2. ACK: Acknowledgement flag indicates that the acknowledgement number is significant. It is used to acknowledge the receipt of a TCP segment.
3. PSH: Push flag asking TCP to pass the data to the application promptly.
4. RST: Reset flag is used to reset the connection. Another device, such as a firewall, might send it to tear a TCP connection. This flag is also used when data is sent to a host and there is no service on the receiving end to answer.
5. SYN: Synchronize flag is used to initiate a TCP 3-way handshake and synchronize sequence numbers with the other host. The sequence number should be set randomly during TCP connection establishment.
6. FIN: The sender has no more data to send.

****

Answer the questions below:

What 3 letters represent the Reset flag?

Answer: RST

Which flag needs to be set when you initiate a TCP connection (first packet of TCP 3-way handshake)?

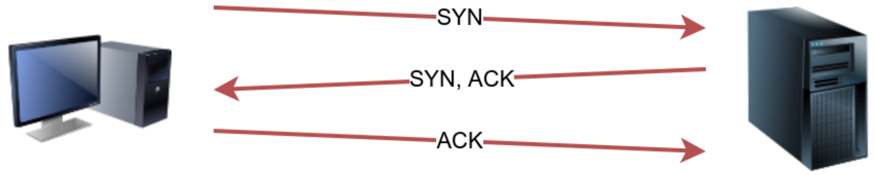
Answer: SYN

****

TCP Connect Scan

TCP connect scan works by completing the TCP 3-way handshake. In standard TCP connection establishment, the client sends a TCP packet with SYN flag set, and the server responds with SYN/ACK if the port is open; finally, the client completes the 3-way handshake by sending an ACK

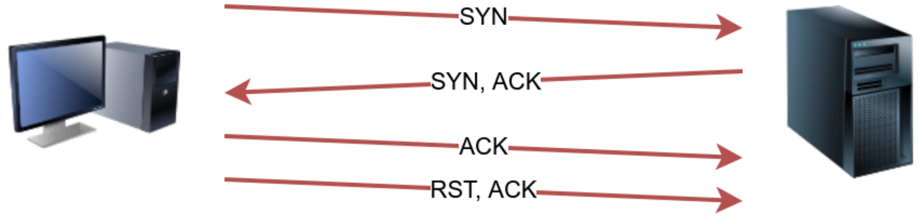
`TCP 3-Way Handshake`



Case: TCP port is open.

We are interested in learning whether the TCP port is open, not establishing a TCP connection. Hence the connection is torn as soon as its state is confirmed by sending a RST/ACK. You can choose to run TCP connect scan using -sT.

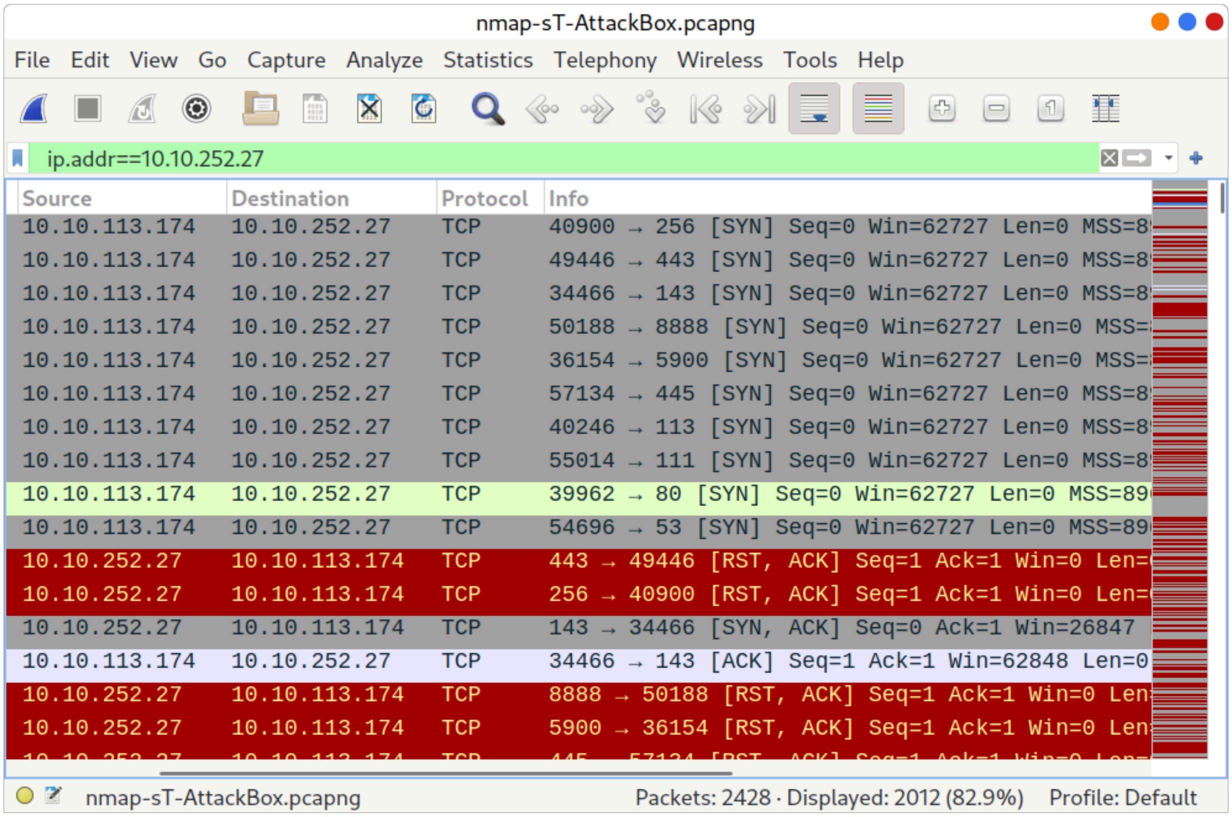
```
nmap -sT TARGET
```



Case: TCP port is open.

It is important to note that if you are not a privileged user (root or sudoer), a TCP connect scan is the only possible option to discover open TCP ports.

In the following Wireshark packet capture window, we see Nmap sending TCP packets with SYN flag set to various ports, 256, 443, 143, and so on. By default, Nmap will attempt to connect to the 1000 most common ports. A closed TCP port responds to a SYN packet with RST/ACK to indicate that it is not open. This pattern will repeat for all the closed ports as we attempt to initiate a TCP 3-way handshake with them.

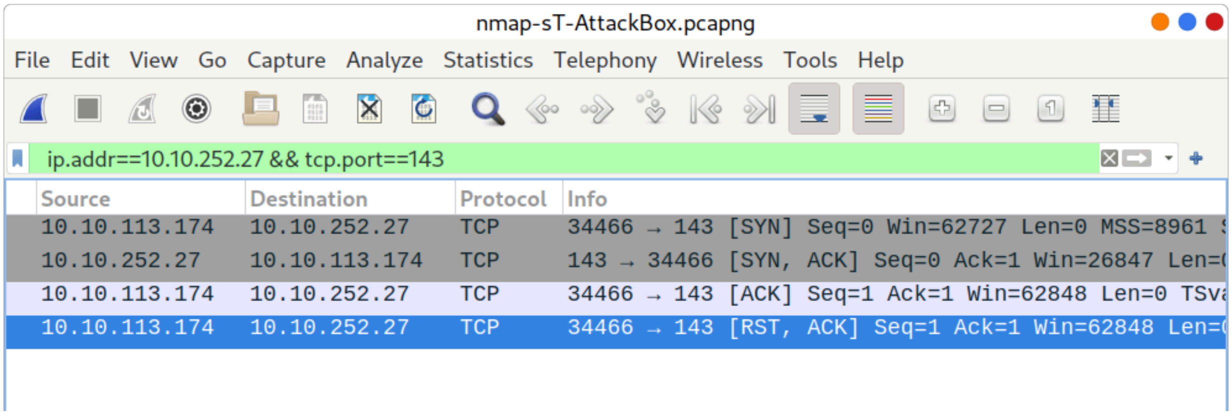

nmap-sT-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==10.10.252.27

```
Source          Destination     Protocol  Info
10.10.113.174   10.10.252.27    TCP       40900 → 256 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       49446 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       34466 → 143 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       50188 → 8888 [SYN] Seq=0 Win=62727 Len=0 MSS=
10.10.113.174   10.10.252.27    TCP       36154 → 5900 [SYN] Seq=0 Win=62727 Len=0 MSS=
10.10.113.174   10.10.252.27    TCP       57134 → 445 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       40246 → 113 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       55014 → 111 [SYN] Seq=0 Win=62727 Len=0 MSS=8
10.10.113.174   10.10.252.27    TCP       39962 → 80 [SYN] Seq=0 Win=62727 Len=0 MSS=89
10.10.113.174   10.10.252.27    TCP       54696 → 53 [SYN] Seq=0 Win=62727 Len=0 MSS=89
10.10.252.27    10.10.113.174   TCP       443 → 49446 [RST, ACK] Seq=1 Ack=1 Win=0 Len=
10.10.252.27    10.10.113.174   TCP       256 → 40900 [RST, ACK] Seq=1 Ack=1 Win=0 Len=
10.10.252.27    10.10.113.174   TCP       143 → 34466 [SYN, ACK] Seq=0 Ack=1 Win=26847
10.10.113.174   10.10.252.27    TCP       34466 → 143 [ACK] Seq=1 Ack=1 Win=62848 Len=0
10.10.252.27    10.10.113.174   TCP       8888 → 50188 [RST, ACK] Seq=1 Ack=1 Win=0 Len
10.10.252.27    10.10.113.174   TCP       5900 → 36154 [RST, ACK] Seq=1 Ack=1 Win=0 Len
10.10.252.27    10.10.113.174   TCP       445 → 57134 [RST, ACK] Seq=1 Ack=1 Win=0 Len=
```

nmap-sT-AttackBox.pcapng Packets: 2428 · Displayed: 2012 (82.9%) Profile: Default

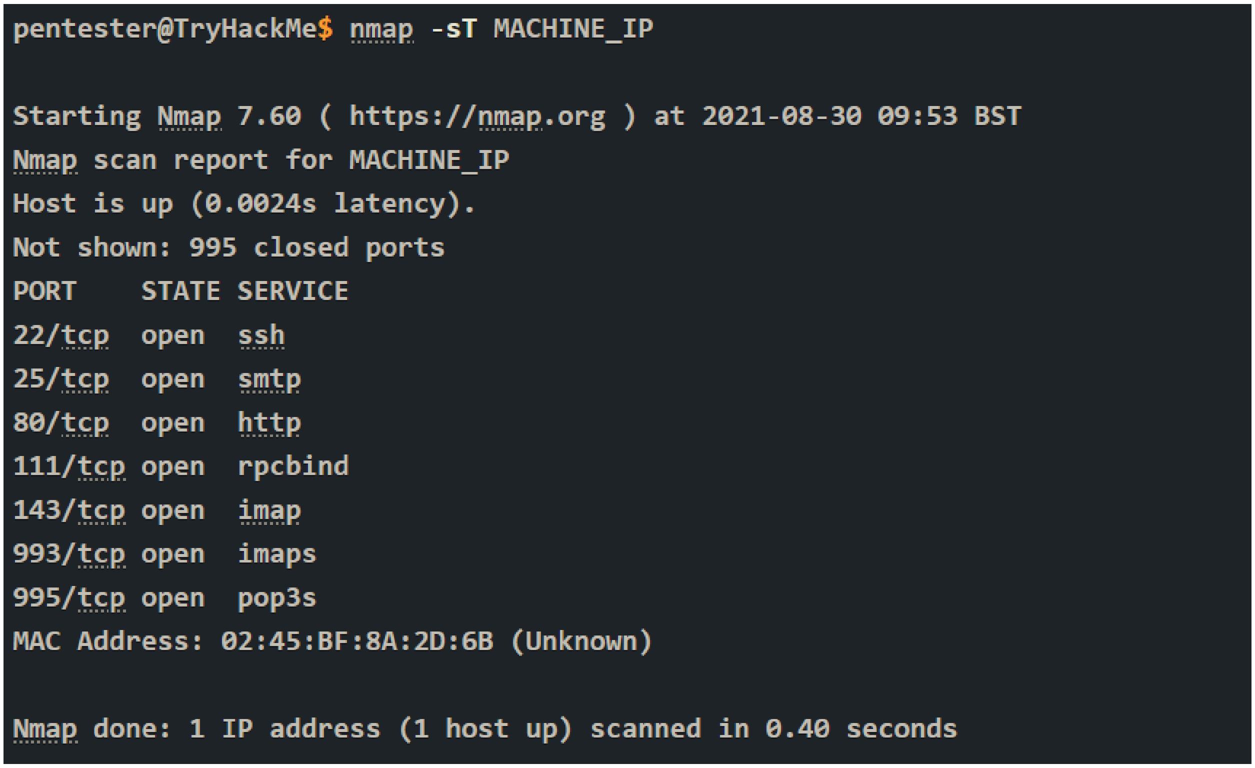
We notice that port 143 is open, so it replied with a SYN/ACK, and Nmap completed the 3-way handshake by sending an ACK. The figure below shows all the packets exchanged between our Nmap host and the target system's port 143. The first three packets are the TCP 3-way handshake being completed. Then, the fourth packet tears it down with an RST/ACK packet.


nmap-sT-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==10.10.252.27 && tcp.port==143

```
Source          Destination     Protocol  Info
10.10.113.174   10.10.252.27    TCP       34466 → 143 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 S
10.10.252.27    10.10.113.174   TCP       143 → 34466 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=(
10.10.113.174   10.10.252.27    TCP       34466 → 143 [ACK] Seq=1 Ack=1 Win=62848 Len=0 TSva
10.10.113.174   10.10.252.27    TCP       34466 → 143 [RST, ACK] Seq=1 Ack=1 Win=62848 Len=(
```

To illustrate the -sT (TCP connect scan), the following command example returned a detailed list of the open ports.



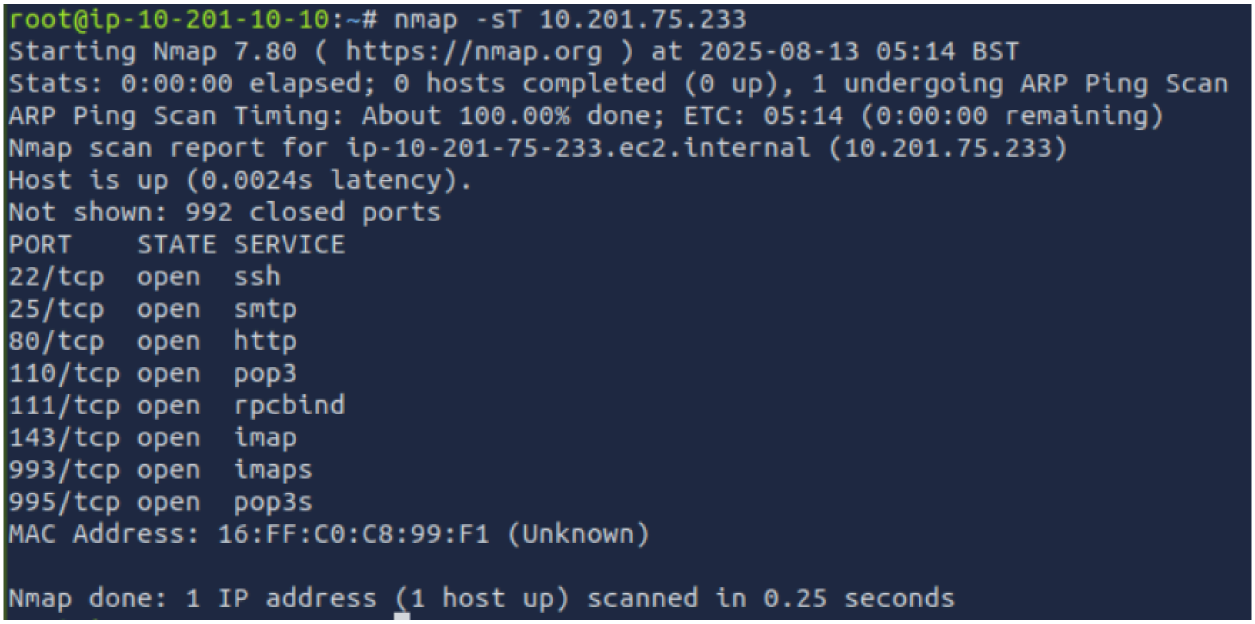
```
pentester@TryHackMe$ nmap -sT MACHINE_IP

Starting Nmap 7.60 ( https://nmap.org ) at 2021-08-30 09:53 BST
Nmap scan report for MACHINE_IP
Host is up (0.0024s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
111/tcp open  rpcbind
143/tcp open  imap
993/tcp open  imaps
995/tcp open  pop3s
MAC Address: 02:45:BF:8A:2D:6B (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds
```

Note that we can use -F to enable fast mode and decrease the number of scanned ports from 1000 to 100 most common ports.

It is worth mentioning that the -r option can also be added to scan the ports in consecutive order instead of random order. This option is useful when testing whether ports open in a consistent manner, for instance, when a target boots up.
****

Answer the questions below:

Launch the VM. Open the AttackBox and execute nmap -sT MACHINE_IP via the terminal. A new service has been installed on this VM since our last scan, as shown in the terminal window above. Which port number was closed in the scan above but is now open on this target VM?



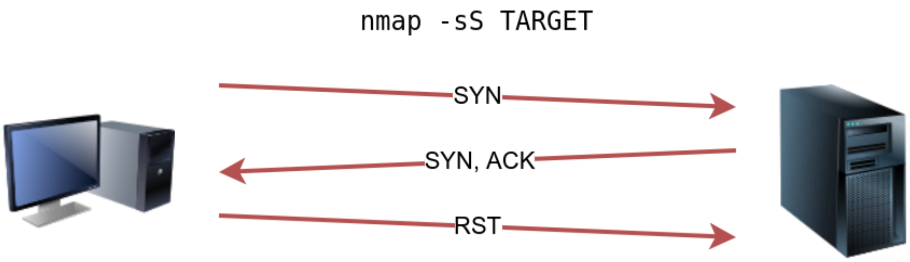
```
root@ip-10-201-10-10:~# nmap -sT 10.201.75.233
Starting Nmap 7.80 ( https://nmap.org ) at 2025-08-13 05:14 BST
Stats: 0:00:00 elapsed; 0 hosts completed (0 up), 1 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 100.00% done; ETC: 05:14 (0:00:00 remaining)
Nmap scan report for ip-10-201-75-233.ec2.internal (10.201.75.233)
Host is up (0.0024s latency).
Not shown: 992 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
110/tcp  open  pop3
111/tcp  open  rpcbind
143/tcp  open  imap
993/tcp  open  imaps
995/tcp  open  pop3s
MAC Address: 16:FF:C0:C8:99:F1 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
```

Answer: 110

What is Nmap's guess about the newly installed service?
Answer: pop3
****

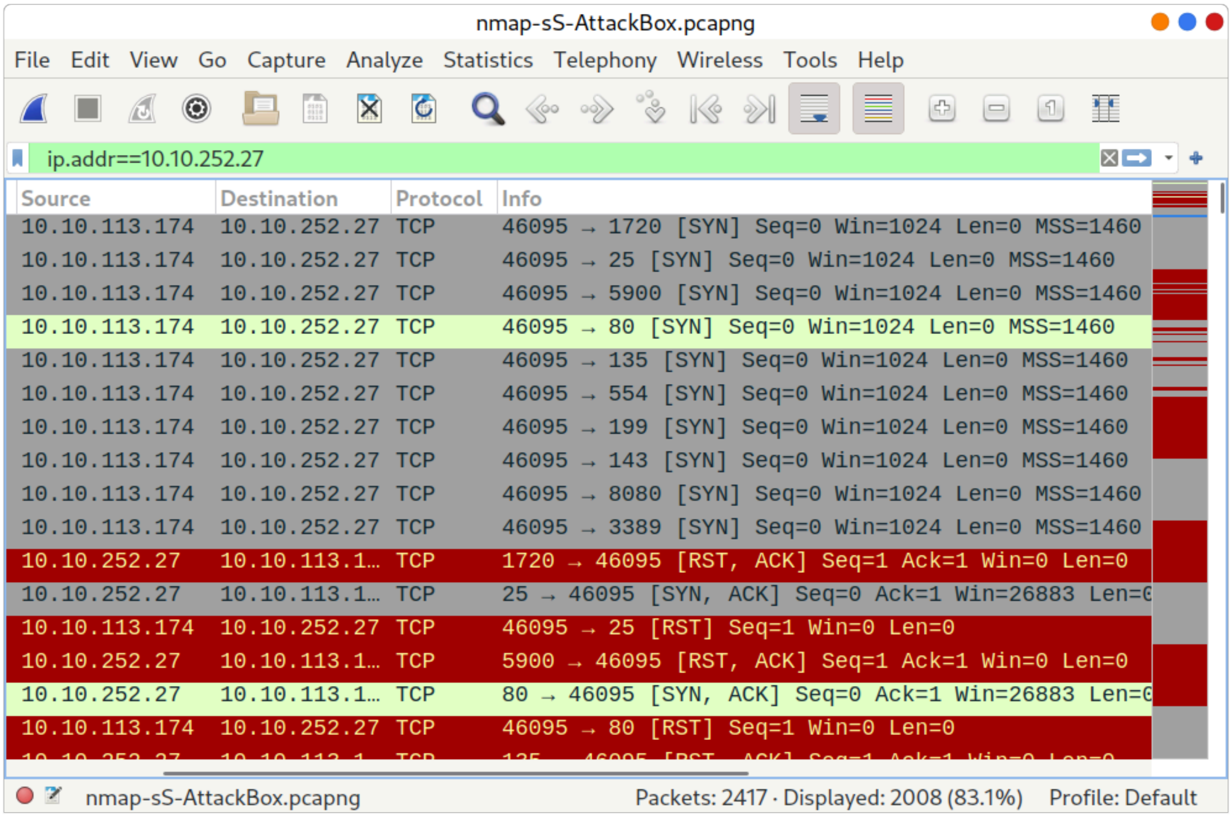
TCP SYN Scan

Unprivileged users are limited to connect scan. However, the default scan mode is SYN scan, and it requires a privileged (root or sudoer) user to run it. SYN scan does not need to complete the TCP 3-way handshake; instead, it tears down the connection once it receives a response from the server. Because we didn't establish a TCP connection, this decreases the chances of the scan being logged. We can select this scan type by using the -sS option. The figure below shows how the TCP SYN scan works without completing the TCP 3-way handshake.



Case: TCP port is open.

The following screenshot from Wireshark shows a TCP SYN scan. The behaviour in the case of closed TCP ports is similar to that of the TCP connect scan.

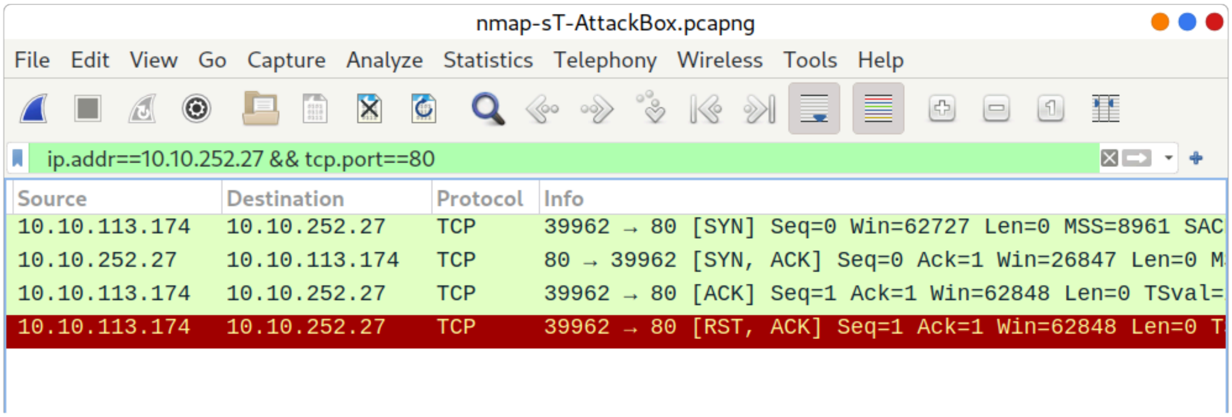

nmap-sS-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==10.10.252.27

Source	Destination	Protocol	Info
10.10.113.174	10.10.252.27	TCP	46095 → 1720 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 25 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 5900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 554 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 199 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 143 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 8080 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.113.174	10.10.252.27	TCP	46095 → 3389 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.252.27	10.10.113.1…	TCP	1720 → 46095 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
10.10.252.27	10.10.113.1…	TCP	25 → 46095 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0
10.10.113.174	10.10.252.27	TCP	46095 → 25 [RST] Seq=1 Win=0 Len=0
10.10.252.27	10.10.113.1…	TCP	5900 → 46095 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
10.10.252.27	10.10.113.1…	TCP	80 → 46095 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0
10.10.113.174	10.10.252.27	TCP	46095 → 80 [RST] Seq=1 Win=0 Len=0
10.10.252.27	10.10.113.1…	TCP	135 → 46095 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

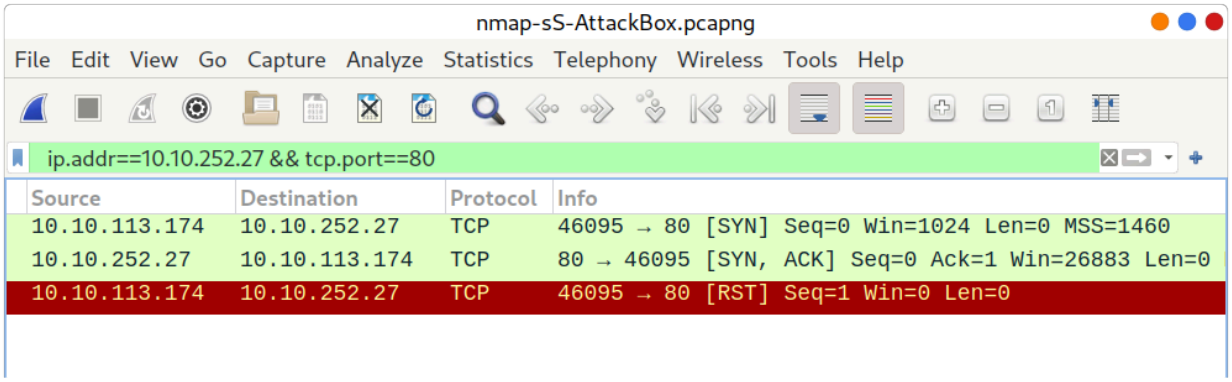
nmap-sS-AttackBox.pcapng Packets: 2417 · Displayed: 2008 (83.1%) Profile: Default

To better see the difference between the two scans, consider the following screenshot. In the upper half of the following figure, we can see a TCP connect scan -sT traffic. Any open TCP port will require Nmap to complete the TCP 3-way handshake before closing the connection. In the lower half of the following figure, we see how a SYN scan -sS does not need to complete the TCP 3-way handshake; instead, Nmap sends an RST packet once a SYN/ACK packet is received.


nmap-sT-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr==10.10.252.27 && tcp.port==80
```

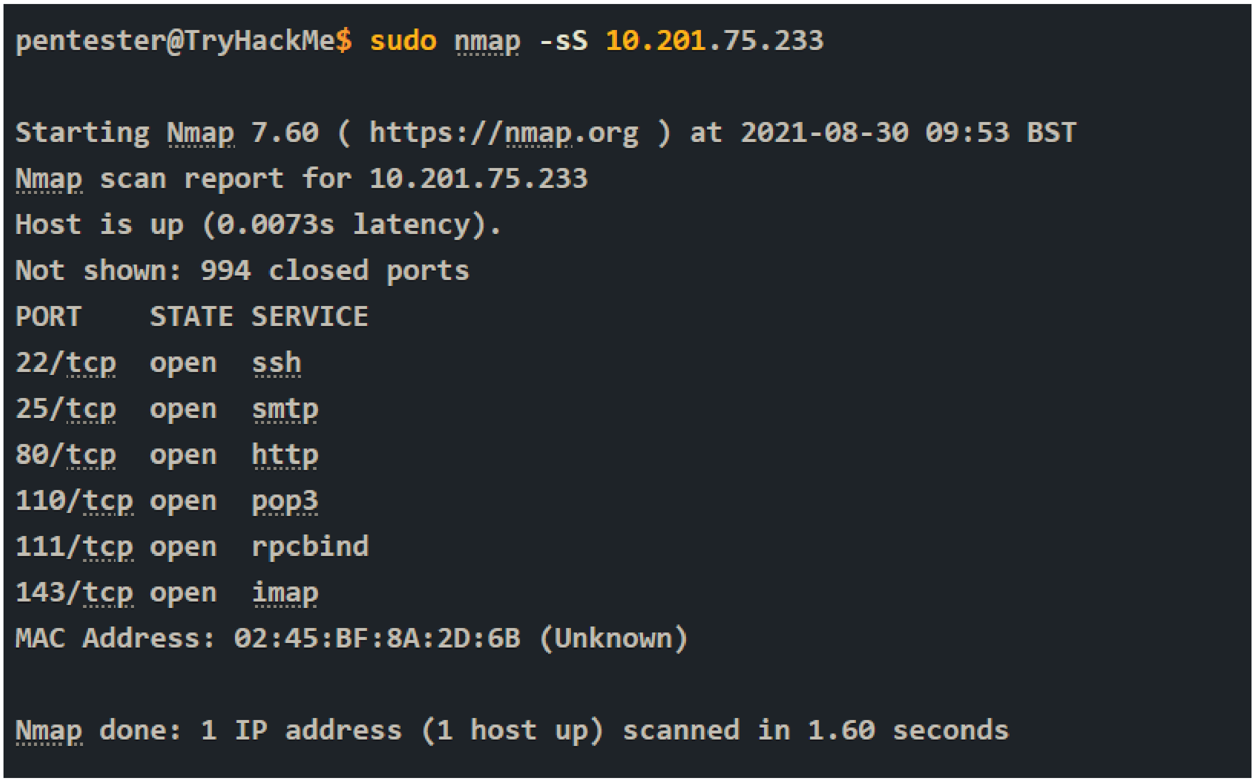
```
Source          Destination     Protocol  Info
10.10.113.174   10.10.252.27    TCP       39962 → 80 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SAC
10.10.252.27    10.10.113.174   TCP       80 → 39962 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 M
10.10.113.174   10.10.252.27    TCP       39962 → 80 [ACK] Seq=1 Ack=1 Win=62848 Len=0 TSval=
10.10.113.174   10.10.252.27    TCP       39962 → 80 [RST, ACK] Seq=1 Ack=1 Win=62848 Len=0 T
```


nmap-sS-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr==10.10.252.27 && tcp.port==80
```

```
Source          Destination     Protocol  Info
10.10.113.174   10.10.252.27    TCP       46095 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
10.10.252.27    10.10.113.174   TCP       80 → 46095 [SYN, ACK] Seq=0 Ack=1 Win=26883 Len=0
10.10.113.174   10.10.252.27    TCP       46095 → 80 [RST] Seq=1 Win=0 Len=0
```

TCP SYN scan is the default scan mode when running Nmap as a privileged user, running as root or using sudo, and it is a very reliable choice. It has successfully discovered the open ports you found earlier with the TCP connect scan, yet no TCP connection was fully established with the target.

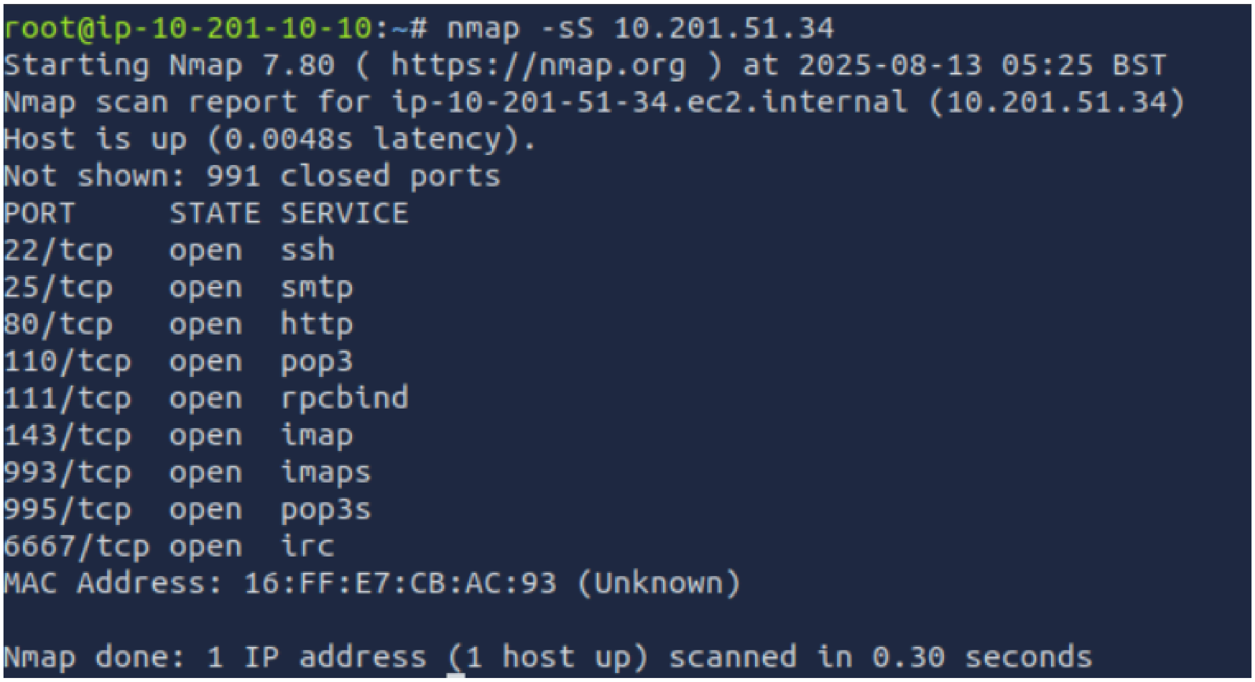


```
pentester@TryHackMe$ sudo nmap -sS 10.201.75.233

Starting Nmap 7.60 ( https://nmap.org ) at 2021-08-30 09:53 BST
Nmap scan report for 10.201.75.233
Host is up (0.0073s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
110/tcp open  pop3
111/tcp open  rpcbind
143/tcp open  imap
MAC Address: 02:45:BF:8A:2D:6B (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.60 seconds
```

****

Answer the questions below:

Launch the VM. Some new server software has been installed since the last time we scanned it. On the AttackBox, use the terminal to execute nmap -sS MACHINE_IP. What is the new open port?



```
root@ip-10-201-10-10:~# nmap -sS 10.201.51.34
Starting Nmap 7.80 ( https://nmap.org ) at 2025-08-13 05:25 BST
Nmap scan report for ip-10-201-51-34.ec2.internal (10.201.51.34)
Host is up (0.0048s latency).
Not shown: 991 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
25/tcp   open  smtp
80/tcp   open  http
110/tcp  open  pop3
111/tcp  open  rpcbind
143/tcp  open  imap
993/tcp  open  imaps
995/tcp  open  pop3s
6667/tcp open  irc
MAC Address: 16:FF:E7:CB:AC:93 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.30 seconds
```

Answer:6667

What is Nmap’s guess of the service name?

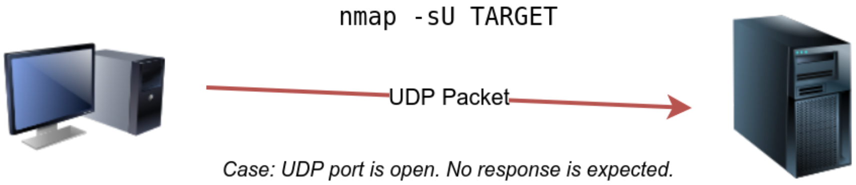
Answer: IRC

****

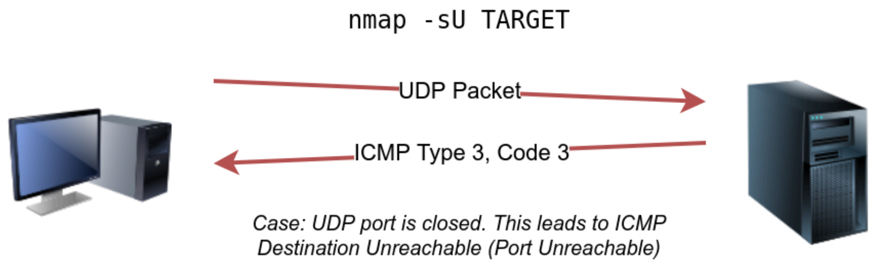
UDP Scan

UDP is a connectionless protocol, and hence it does not require any handshake for connection establishment. We cannot guarantee that a service listening on a UDP port would respond to our packets. However, if a UDP packet is sent to a closed port, an ICMP port unreachable error (type 3, code 3) is returned. You can select UDP scan using the -sU option; moreover, you can combine it with another TCP scan.

The following figure shows that if we send a UDP packet to an open UDP port, we cannot expect any reply in return. Therefore, sending a UDP packet to an open port won’t tell us anything.



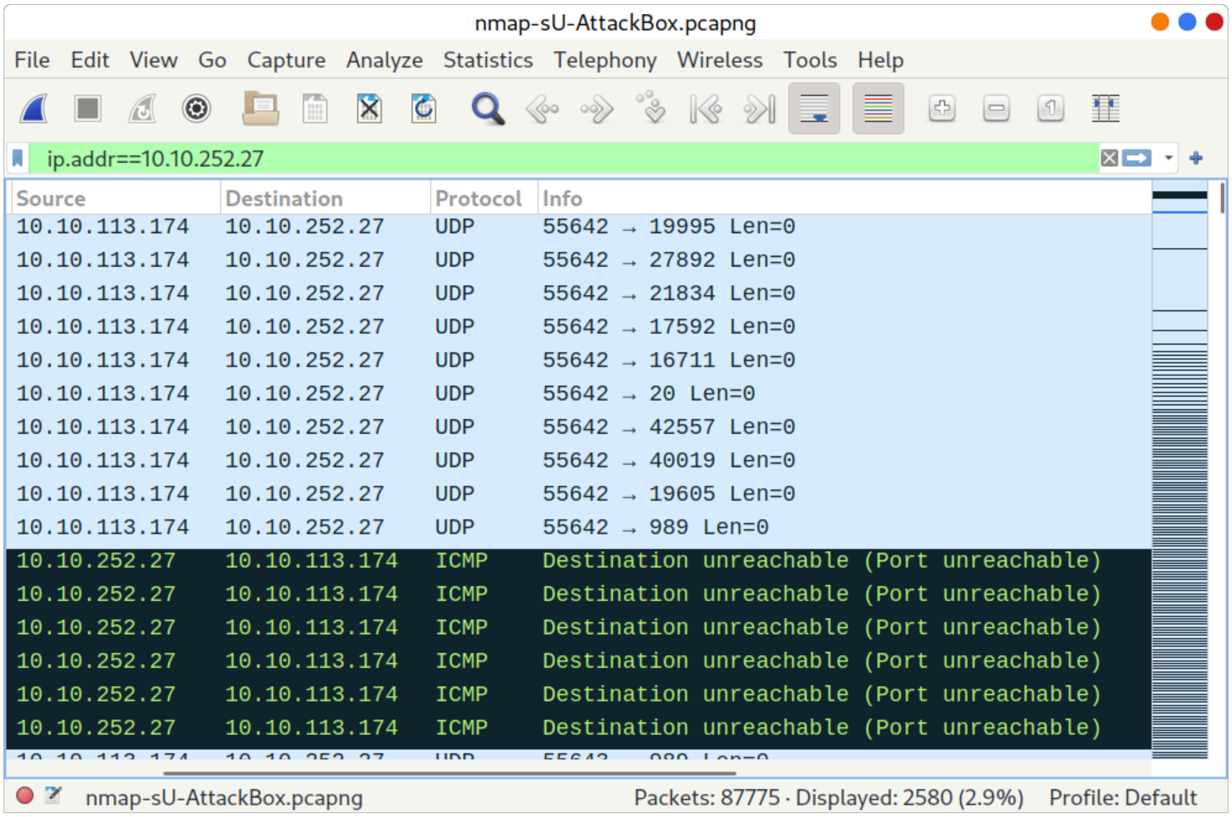
However, as shown in the figure below, we expect to get an ICMP packet of type 3, destination unreachable, and code 3, port unreachable. In other words, the UDP ports that don’t generate any response are the ones that Nmap will state as open.



```
nmap -sU TARGET
```

Case: UDP port is closed. This leads to ICMP Destination Unreachable (Port Unreachable)

In the Wireshark capture below, we can see that every closed port will generate an ICMP packet destination unreachable (port unreachable).



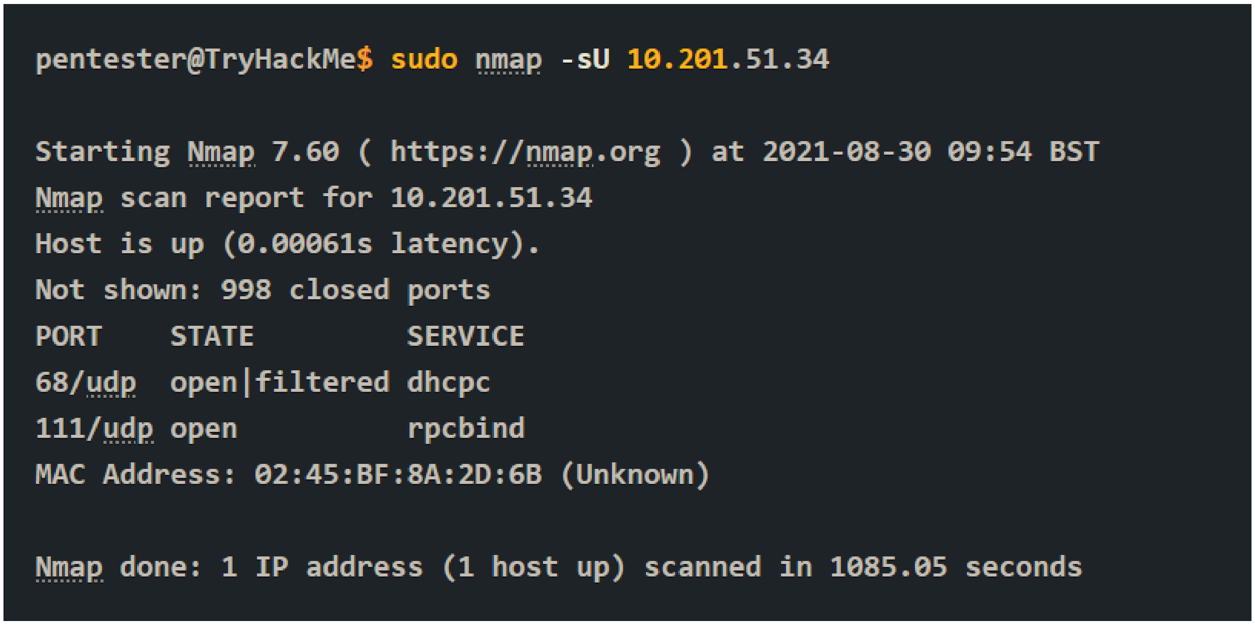
nmap-sU-AttackBox.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==10.10.252.27

```
Source          Destination     Protocol  Info
10.10.113.174   10.10.252.27    UDP       55642 → 19995 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 27892 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 21834 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 17592 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 16711 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 20 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 42557 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 40019 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 19605 Len=0
10.10.113.174   10.10.252.27    UDP       55642 → 989 Len=0
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.252.27    10.10.113.174   ICMP      Destination unreachable (Port unreachable)
10.10.113.174   10.10.252.27    UDP       55642 → 989 Len=0
```

nmap-sU-AttackBox.pcapng   Packets: 87775 · Displayed: 2580 (2.9%)   Profile: Default

Launching a UDP scan against this Linux server proved valuable, and indeed, we learned that port 111 is open. On the other hand, Nmap cannot determine whether UDP port 68 is open or filtered.

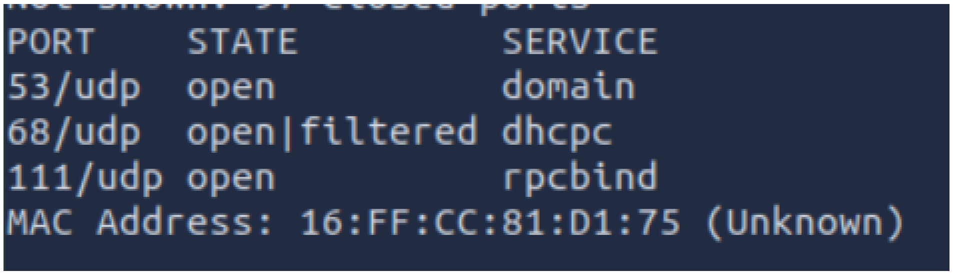


```
pentester@TryHackMe$ sudo nmap -sU 10.201.51.34

Starting Nmap 7.60 ( https://nmap.org ) at 2021-08-30 09:54 BST
Nmap scan report for 10.201.51.34
Host is up (0.00061s latency).
Not shown: 998 closed ports
PORT    STATE         SERVICE
68/udp  open|filtered dhcpc
111/udp open          rpcbind
MAC Address: 02:45:BF:8A:2D:6B (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1085.05 seconds
```

****

Answer the questions below:

Launch the VM. On the AttackBox, use the terminal to execute nmap -sU -F -v MACHINE_IP. A new service has been installed since the last scan. What is the UDP port that is now open?



```
Not shown: 97 closed ports
PORT    STATE         SERVICE
53/udp  open          domain
68/udp  open|filtered dhcpc
111/udp open          rpcbind
MAC Address: 16:FF:CC:81:D1:75 (Unknown)
```

Answer: 53

What is the service name according to Nmap?
Answer: domain

****

Fine-Tuning Scope and Performance

You can specify the ports you want to scan instead of the default 1000 ports. Specifying the ports is intuitive by now. Let's see some examples:

- port list: -p22,80,443 will scan ports 22, 80 and 443.

- port range: -p1-1023 will scan all ports between 1 and 1023 inclusive, while -p20-25 will scan ports between 20 and 25 inclusive.

You can request the scan of all ports by using -p-, which will scan all 65535 ports. If you want to scan the most common 100 ports, add -F. Using --top-ports 10 will check the ten most common ports.

You can control the scan timing using -T<0-5>. -T0 is the slowest (paranoid), while -T5 is the fastest. According to Nmap manual page, there are six templates:

- paranoid (0)
- sneaky (1)
- polite (2)
- normal (3)
- aggressive (4)
- insane (5)

To avoid IDS alerts, you might consider -T0 or -T1. For instance, -T0 scans one port at a time and waits 5 minutes between sending each probe, so you can guess how long scanning one target would take to finish. If you don't specify any timing, Nmap uses normal -T3. Note that -T5 is the most aggressive in terms of speed; however, this can affect the accuracy of the scan results due to the increased likelihood of packet loss. Note that -T4 is often used during CTFs and when learning to scan on practice targets, whereas -T1 is often used during real engagements where stealth is more important.

Alternatively, you can choose to control the packet rate using --min-rate <number> and --max-rate <number>. For example, --max-rate 10 or --max-rate=10 ensures that your scanner is not sending more than ten packets per second.

Moreover, you can control probing parallelization using --min-parallelism <numprobes> and --max-parallelism <numprobes>. Nmap probes the targets to discover which hosts are live and which ports are open; probing parallelization specifies the number of such probes that can be run in parallel. For instance, --min-parallelism=512 pushes Nmap to maintain at least 512 probes in parallel; these 512 probes are related to host discovery and open ports.
****

Answer the questions below:

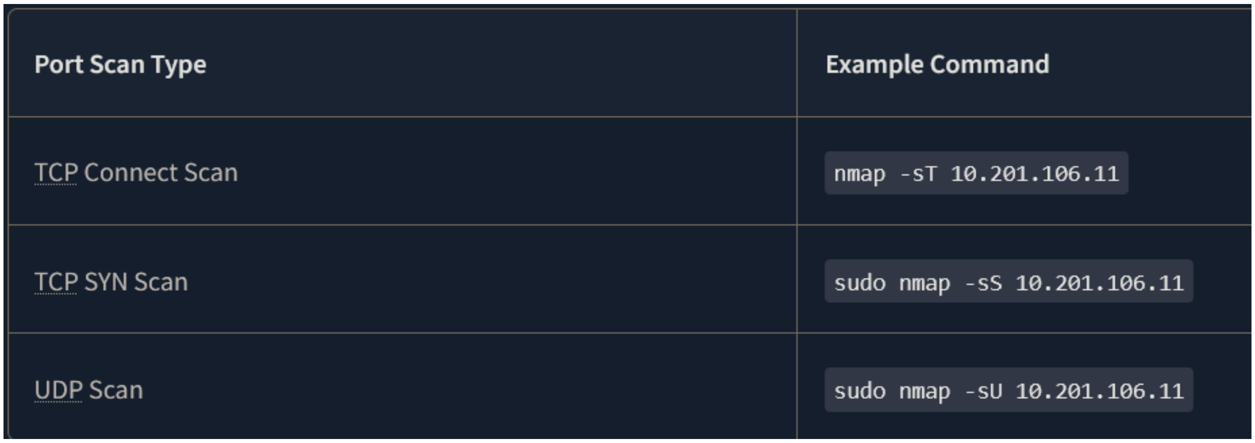
What is the option to scan all the TCP ports between 5000 and 5500?
Answer: -p5000-5500

How can you ensure that Nmap will run at least 64 probes in parallel?
Answer: --min-parallelism=64

What option would you add to make Nmap very slow and paranoid?
Answer: -T0
****

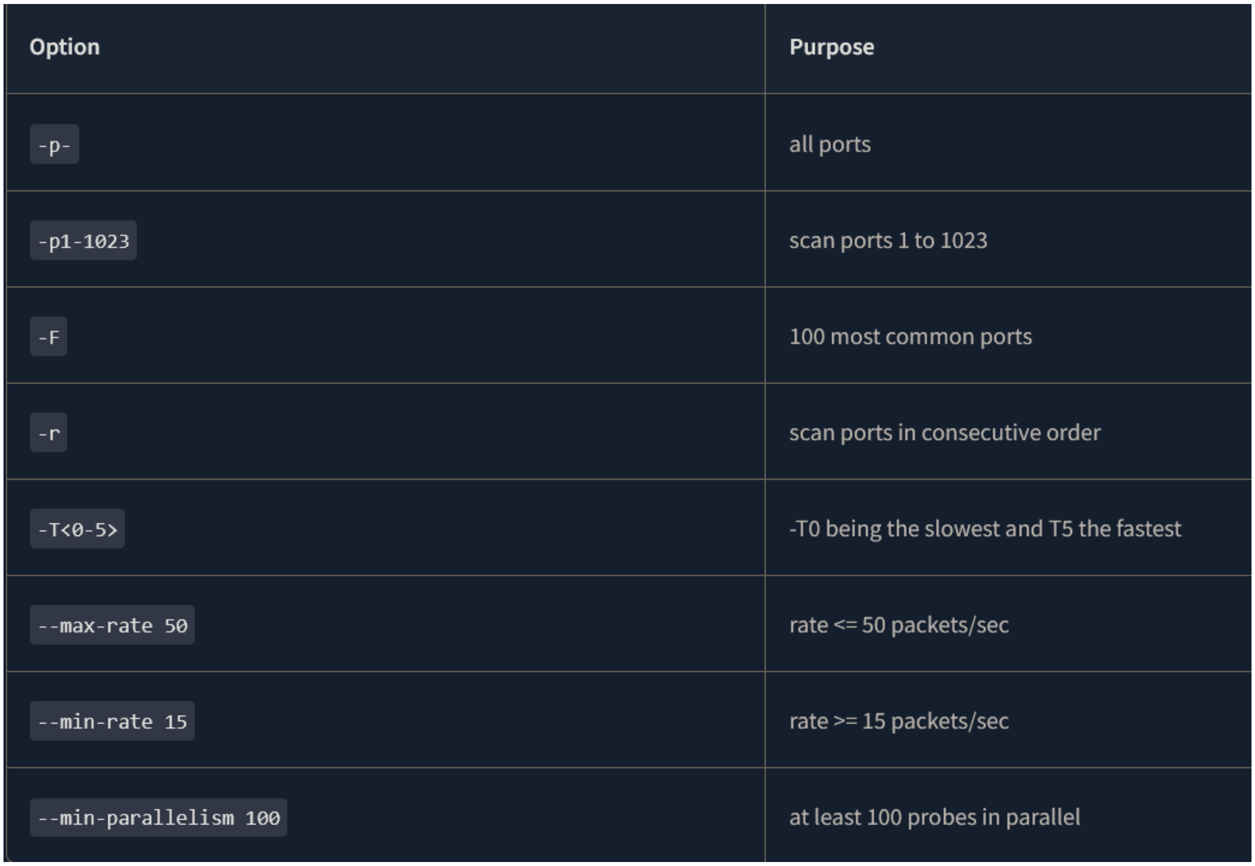
Summary

This room covered three types of scans.



Port Scan Type	Example Command
TCP Connect Scan	`nmap -sT 10.201.106.11`
TCP SYN Scan	`sudo nmap -sS 10.201.106.11`
UDP Scan	`sudo nmap -sU 10.201.106.11`

These scan types should get you started discovering running TCP and UDP services on a target host.



Option	Purpose
`-p-`	all ports
`-p1-1023`	scan ports 1 to 1023
`-F`	100 most common ports
`-r`	scan ports in consecutive order
`-T<0-5>`	-T0 being the slowest and T5 the fastest
`--max-rate 50`	rate <= 50 packets/sec
`--min-rate 15`	rate >= 15 packets/sec
`--min-parallelism 100`	at least 100 probes in parallel