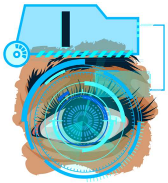
Jr Penetration Tester > Network Security > Nmap Live Host Discovery



Nmap Live Host Discovery

Learn how to use Nmap to discover live hosts using ARP scan, ICMP scan, and TCP/UDP ping scan.

Medium 120 min

Show Split View | Start AttackBox | Help | Save Room

3754 | Options

Task 1 ✓ Introduction

Task 2 ✓ Subnetworks

Task 3 ✓ Enumerating Targets

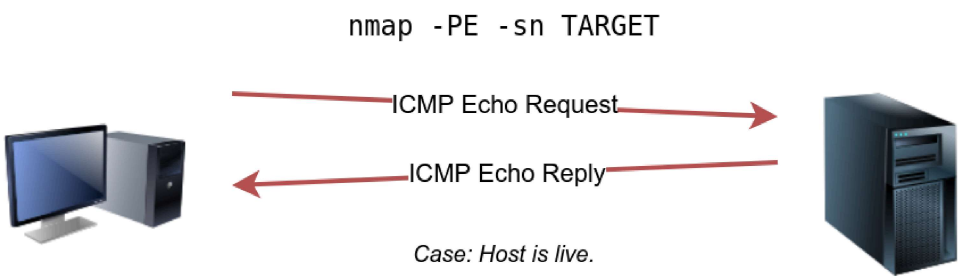
Task 4 ✓ Discovering Live Hosts

Task 5 ✓ Nmap Host Discovery Using ARP

Task 6 ✓ Nmap Host Discovery Using ICMP

We can ping every IP address on a target network and see who would respond to our `ping` (ICMP Type 8/Echo) requests with a ping reply (ICMP Type 0). Simple, isn't it? Although this would be the most straightforward approach, it is not always reliable. Many firewalls block ICMP echo; new versions of MS Windows are configured with a host firewall that blocks ICMP echo requests by default. Remember that an ARP query will precede the ICMP request if your target is on the same subnet.

To use ICMP echo request to discover live hosts, add the option `-PE`. (Remember to add `-sn` if you don't want to follow that with a port scan.) As shown in the following figure, an ICMP echo scan works by sending an ICMP echo request and expects the target to reply with an ICMP echo reply if it is online.



In the example below, we scanned the target's subnet using `nmap -PE -sn MACHINE_IP/24`. This scan will send ICMP echo packets to every IP address on the subnet. Again, we expect live hosts to reply; however, it is wise to remember that many firewalls block ICMP. The output below shows the result of scanning the virtual machine's class C subnet using `sudo nmap -PE -sn MACHINE_IP/24` from the AttackBox.

Pentester Terminal

```
pentester@TryHackMe$ sudo nmap -PE -sn 10.10.68.220/24

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-02 10:16 BST
Nmap scan report for ip-10-10-68-50.eu-west-1.compute.internal
(10.10.68.50)
Host is up (0.00017s latency).
MAC Address: 02:95:36:71:5B:87 (Unknown)
Nmap scan report for ip-10-10-68-52.eu-west-1.compute.internal
(10.10.68.52)
Host is up (0.00017s latency).
MAC Address: 02:48:E8:BF:78:E7 (Unknown)
Nmap scan report for ip-10-10-68-77.eu-west-1.compute.internal
(10.10.68.77)
Host is up (-0.100s latency).
Host is up (0.10s latency).
MAC Address: 02:6B:50:E9:C2:91 (Unknown)
Nmap scan report for ip-10-10-68-140.eu-west-1.compute.internal
(10.10.68.140)
Host is up (0.00021s latency).
```

Room completed (100%)

```
Nmap scan report for ip-10-10-68-142.eu-west-1.compute.internal
(10.10.68.142)
Host is up (0.00016s latency).
MAC Address: 02:C6:41:51:0A:0F (Unknown)
Nmap scan report for ip-10-10-68-220.eu-west-1.compute.internal
(10.10.68.220)
Host is up (0.00026s latency).
MAC Address: 02:25:3F:DB:EE:0B (Unknown)
Nmap scan report for ip-10-10-68-222.eu-west-1.compute.internal
(10.10.68.222)
Host is up (0.00025s latency).
MAC Address: 02:28:B1:2E:B0:1B (Unknown)
Nmap done: 256 IP addresses (8 hosts up) scanned in 2.11 seconds
```

The scan output shows that eight hosts are up; moreover, it shows their MAC addresses. Generally speaking, we don't expect to learn the MAC addresses of the targets unless they are on the same subnet as our system. The output above indicates that Nmap didn't need to send ICMP packets as it confirmed that these hosts are up based on the ARP responses it received.

We will repeat the scan above; however, this time, we will scan from a system that belongs to a different subnet. The results are similar but without the MAC addresses.

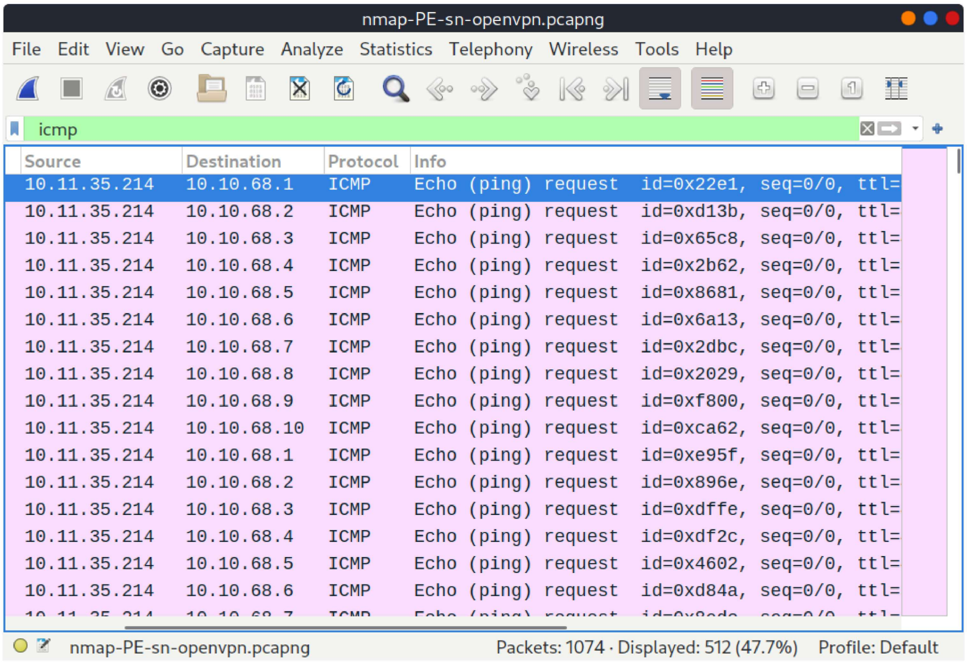
Pentester Terminal

```
pentester@TryHackMe$ sudo nmap -PE -sn 10.10.68.220/24

Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-02 12:16 EEST
Nmap scan report for 10.10.68.50
Host is up (0.12s latency).
Nmap scan report for 10.10.68.52
Host is up (0.12s latency).
Nmap scan report for 10.10.68.77
Host is up (0.11s latency).
```

```
Nmap scan report for 10.10.68.110
Host is up (0.11s latency).
Nmap scan report for 10.10.68.140
Host is up (0.11s latency).
Nmap scan report for 10.10.68.142
Host is up (0.11s latency).
Nmap scan report for 10.10.68.220
Host is up (0.11s latency).
Nmap scan report for 10.10.68.222
Host is up (0.11s latency).
Nmap done: 256 IP addresses (8 hosts up) scanned in 8.26 seconds
```

If you look at the network packets using a tool like Wireshark, you will see something similar to the image below. You can see that we have one source IP address on a different subnet than that of the destination subnet, sending ICMP echo requests to all the IP addresses in the target subnet to see which one will reply.

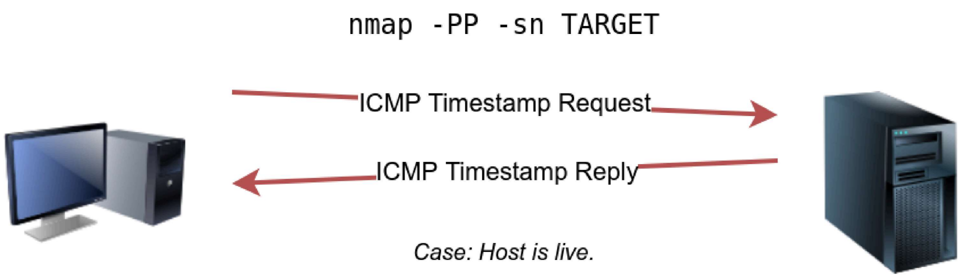

nmap-PE-sn-openvpn.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

icmp

Source	Destination	Protocol	Info
10.11.35.214	10.10.68.1	ICMP	Echo (ping) request id=0x22e1, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Echo (ping) request id=0xd13b, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Echo (ping) request id=0x65c8, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Echo (ping) request id=0x2b62, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Echo (ping) request id=0x8681, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Echo (ping) request id=0x6a13, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Echo (ping) request id=0x2dbc, seq=0/0, ttl=
10.11.35.214	10.10.68.8	ICMP	Echo (ping) request id=0x2029, seq=0/0, ttl=
10.11.35.214	10.10.68.9	ICMP	Echo (ping) request id=0xf800, seq=0/0, ttl=
10.11.35.214	10.10.68.10	ICMP	Echo (ping) request id=0xca62, seq=0/0, ttl=
10.11.35.214	10.10.68.1	ICMP	Echo (ping) request id=0xe95f, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Echo (ping) request id=0x896e, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Echo (ping) request id=0xdffe, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Echo (ping) request id=0xdf2c, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Echo (ping) request id=0x4602, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Echo (ping) request id=0xd84a, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Echo (ping) request id=0x8ede, seq=0/0, ttl=

nmap-PE-sn-openvpn.pcapng Packets: 1074 · Displayed: 512 (47.7%) Profile: Default

Because ICMP echo requests tend to be blocked, you might also consider ICMP Timestamp or ICMP Address Mask requests to tell if a system is online. Nmap uses timestamp request (ICMP Type 13) and checks whether it will get a Timestamp reply (ICMP Type 14). Adding the `-PP` option tells Nmap to use ICMP timestamp requests. As shown in the figure below, you expect live hosts to reply.



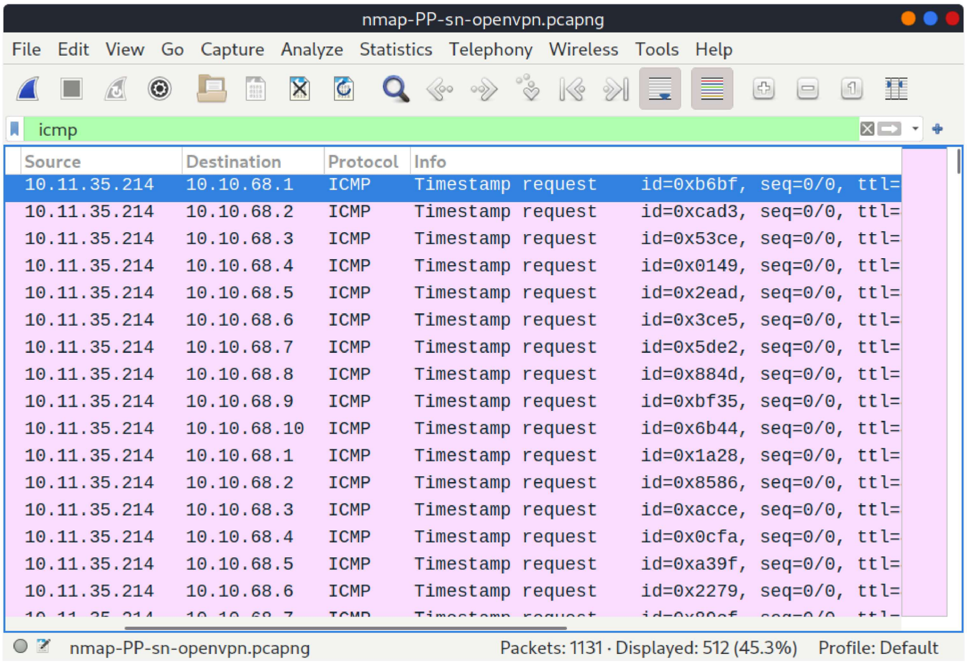
In the following example, we run `nmap -PP -sn MACHINE_IP/24` to discover the online computers on the target machine subnet.

Pentester Terminal

```
pentester@TryHackMe$ sudo nmap -PP -sn 10.10.68.220/24

Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-02 12:06 EEST
Nmap scan report for 10.10.68.50
Host is up (0.13s latency).
Nmap scan report for 10.10.68.52
Host is up (0.25s latency).
Nmap scan report for 10.10.68.77
Host is up (0.14s latency).
Nmap scan report for 10.10.68.110
Host is up (0.14s latency).
Nmap scan report for 10.10.68.140
Host is up (0.15s latency).
Nmap scan report for 10.10.68.209
Host is up (0.14s latency).
Nmap scan report for 10.10.68.220
Host is up (0.14s latency).
Nmap scan report for 10.10.68.222
Host is up (0.14s latency).
Nmap done: 256 IP addresses (8 hosts up) scanned in 10.93 seconds
```

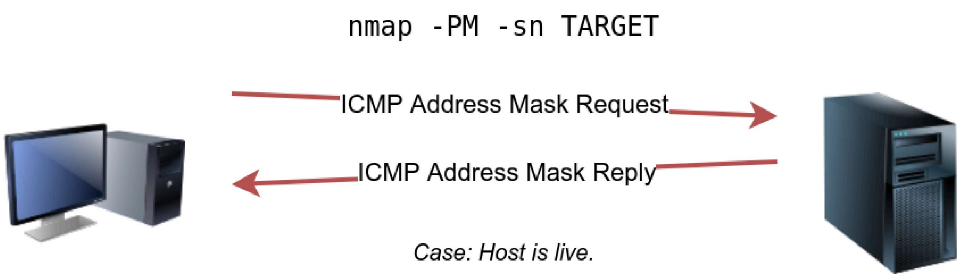
Similar to the previous ICMP scan, this scan will send many ICMP timestamp requests to every valid IP address in the target subnet. In the Wireshark screenshot below, you can see one source IP address sending ICMP packets to every possible IP address to discover online hosts.



Source	Destination	Protocol	Info
10.11.35.214	10.10.68.1	ICMP	Timestamp request id=0xb6bf, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Timestamp request id=0xcad3, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Timestamp request id=0x53ce, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Timestamp request id=0x0149, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Timestamp request id=0x2ead, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Timestamp request id=0x3ce5, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Timestamp request id=0x5de2, seq=0/0, ttl=
10.11.35.214	10.10.68.8	ICMP	Timestamp request id=0x884d, seq=0/0, ttl=
10.11.35.214	10.10.68.9	ICMP	Timestamp request id=0xbf35, seq=0/0, ttl=
10.11.35.214	10.10.68.10	ICMP	Timestamp request id=0x6b44, seq=0/0, ttl=
10.11.35.214	10.10.68.1	ICMP	Timestamp request id=0x1a28, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Timestamp request id=0x8586, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Timestamp request id=0xacce, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Timestamp request id=0x0cfa, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Timestamp request id=0xa39f, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Timestamp request id=0x2279, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Timestamp request

nmap-PP-sn-openvpn.pcapng Packets: 1131 · Displayed: 512 (45.3%) Profile: Default

Similarly, Nmap uses address mask queries (ICMP Type 17) and checks whether it gets an address mask reply (ICMP Type 18). This scan can be enabled with the option `-PM`. As shown in the figure below, live hosts are expected to reply to ICMP address mask requests.



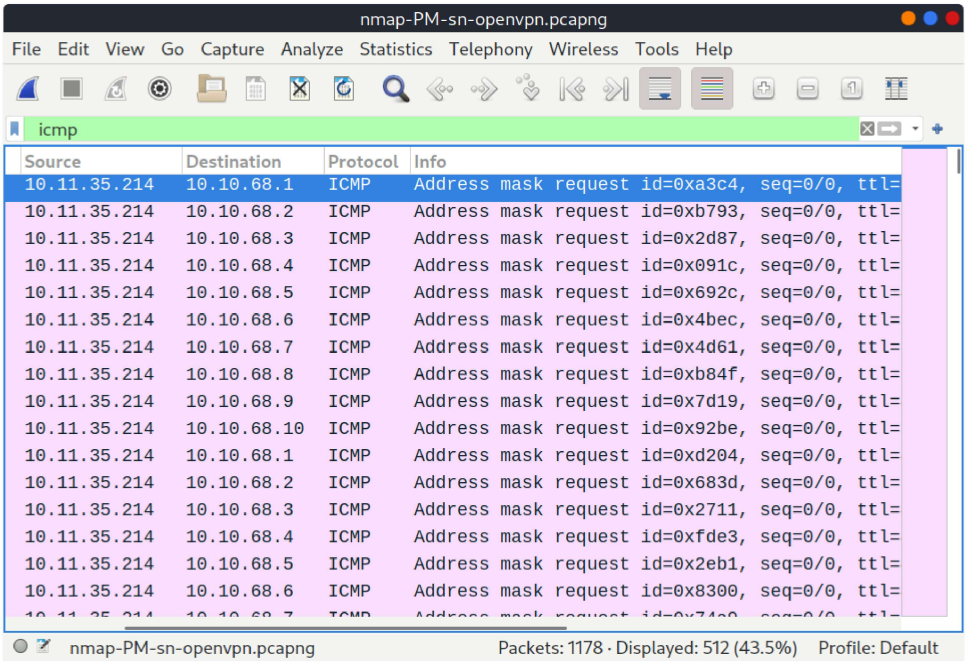
In an attempt to discover live hosts using ICMP address mask queries, we run the command `nmap -PM -sn MACHINE_IP/24`. Although, based on earlier scans, we know that at least eight hosts are up, this scan returned none. The reason is that the target system or a firewall on the route is blocking this type of ICMP packet. Therefore, it is essential to learn multiple approaches to achieve the same result. If one type of packet is being blocked, we can always choose another to discover the target network and services.

Pentester Terminal

```
pentester@TryHackMe$ sudo nmap -PM -sn 10.10.68.220/24

Starting Nmap 7.92 ( https://nmap.org ) at 2021-09-02 12:13 EEST
Nmap done: 256 IP addresses (0 hosts up) scanned in 52.17 seconds
```

Although we didn’t get any reply and could not figure out which hosts are online, it is essential to note that this scan sent ICMP address mask requests to every valid IP address and waited for a reply. Each ICMP request was sent twice, as we can see in the screenshot below.


nmap-PM-sn-openvpn.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

icmp

Source	Destination	Protocol	Info
10.11.35.214	10.10.68.1	ICMP	Address mask request id=0xa3c4, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Address mask request id=0xb793, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Address mask request id=0x2d87, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Address mask request id=0x091c, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Address mask request id=0x692c, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Address mask request id=0x4bec, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Address mask request id=0x4d61, seq=0/0, ttl=
10.11.35.214	10.10.68.8	ICMP	Address mask request id=0xb84f, seq=0/0, ttl=
10.11.35.214	10.10.68.9	ICMP	Address mask request id=0x7d19, seq=0/0, ttl=
10.11.35.214	10.10.68.10	ICMP	Address mask request id=0x92be, seq=0/0, ttl=
10.11.35.214	10.10.68.1	ICMP	Address mask request id=0xd204, seq=0/0, ttl=
10.11.35.214	10.10.68.2	ICMP	Address mask request id=0x683d, seq=0/0, ttl=
10.11.35.214	10.10.68.3	ICMP	Address mask request id=0x2711, seq=0/0, ttl=
10.11.35.214	10.10.68.4	ICMP	Address mask request id=0xfde3, seq=0/0, ttl=
10.11.35.214	10.10.68.5	ICMP	Address mask request id=0x2eb1, seq=0/0, ttl=
10.11.35.214	10.10.68.6	ICMP	Address mask request id=0x8300, seq=0/0, ttl=
10.11.35.214	10.10.68.7	ICMP	Address mask request id=0x74a0, seq=0/0, ttl=

nmap-PM-sn-openvpn.pcapng Packets: 1178 · Displayed: 512 (43.5%) Profile: Default

Answer the questions below

What is the option required to tell Nmap to use ICMP Timestamp to discover live hosts?

-PP

✓ Correct Answer

What is the option required to tell Nmap to use ICMP Address Mask to discover live hosts?

-PM

✓ Correct Answer

What is the option required to tell Nmap to use ICMP Echo to discover live hosts?

-PE

✓ Correct Answer

Task 7 ✓ Nmap Host Discovery Using TCP and UDP

Task 8 ✓ Using Reverse-DNS Lookup

Task 9 ✓ Summary

Created by

tryhackme strategos

Room Type	Users in Room	Created
Free Room. Anyone can deploy virtual machines in the room (without being subscribed)!	168,140	1064 days ago