

Part 1: Port Scanning to Map a Network

Sample Screenshots of Zenmap and Nmap:

The screenshots show the Zenmap interface running an "Intense scan" on target 10.0.2.15/24 using the command nmap -T4 -A -v 10.0.2.15/24.

Screenshot 1 (Top): Shows the initial scan progress. The output window displays the start of the scan, including the loading of scripts and the initiation of NSE (Nmap Script Engine) at 21:32 on April 25, 2023. It also shows the initiation of an ARP Ping Scan and the scanning of 255 hosts.

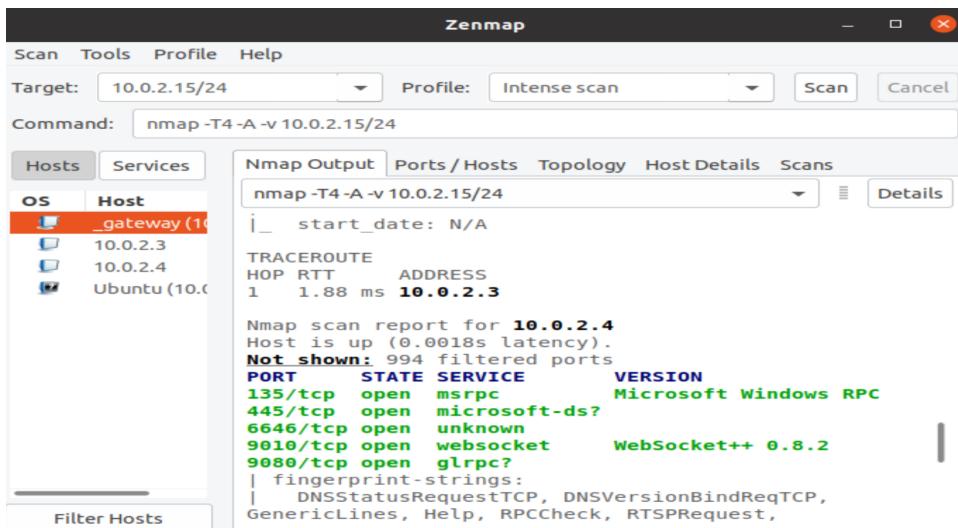
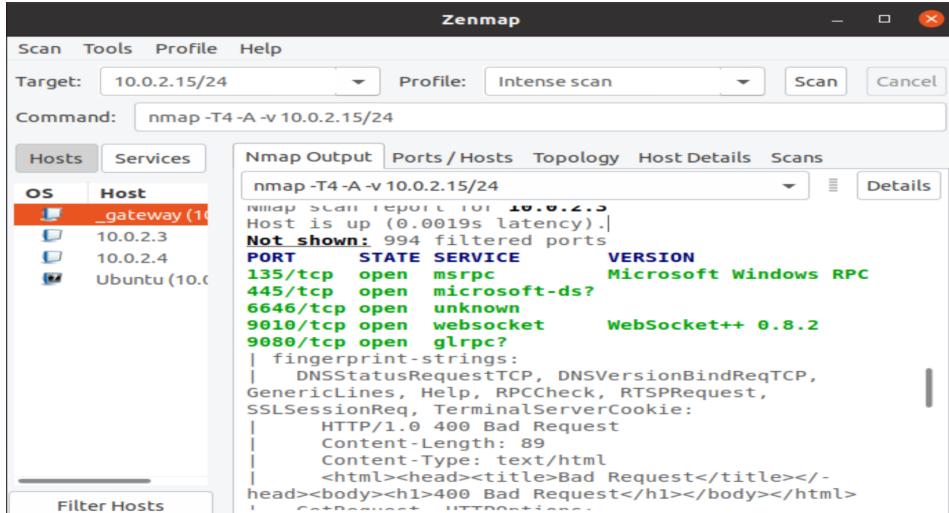
```
nmap -T4 -A -v 10.0.2.15/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-04-25 21:32 MDT
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 21:32
Completed NSE at 21:32, 0.00s elapsed
Initiating NSE at 21:32
Completed NSE at 21:32, 0.00s elapsed
Initiating NSE at 21:32
Completed NSE at 21:32, 0.00s elapsed
Initiating ARP Ping Scan at 21:32
Scanning 255 hosts [1 port/host]
Completed ARP Ping Scan at 21:32, 2.01s elapsed (255 total hosts)
Initiating Parallel DNS resolution of 255 hosts. at 21:32
Completed Parallel DNS resolution of 255 hosts. at 21:32, 0.05s elapsed
```

Screenshot 2 (Middle): Shows the completed scan results. The output window displays the completed NSE at 21:35, followed by the host report for the gateway (10.0.2.2). It lists 994 filtered ports and provides detailed information for several open ports, including service fingerprints and version numbers.

```
Completed NSE at 21:35, 1.06s elapsed
Initiating NSE at 21:35
Completed NSE at 21:35, 0.00s elapsed
Nmap scan report for _gateway (10.0.2.2)
Host is up (0.0018s latency).
Not shown: 994 filtered ports
PORT      STATE SERVICE          VERSION
135/tcp    open  msrpc           Microsoft Windows RPC
445/tcp    open  microsoft-ds?
6646/tcp   open  unknown
9010/tcp   open  websocket       WebSocket++ 0.8.2
9080/tcp   open  glrp?
| fingerprint-strings:
|   | DNSStatusRequestTCP, DNSVersionBindReqTCP,
|   | GenericLines, Help, RPCCheck, RTSPRequest,
|   | SSLSessionReq, TerminalServerCookie:
|   |   HTTP/1.0 400 Bad Request
|   |   Content-Length: 89
```

Screenshot 3 (Bottom): Shows the completed scan results with OS detection details. The output window displays the OS detection process, including the identification of the gateway as a QEMU user mode network gateway (98%), Oracle Virtualbox (96%), Samsung CLP-315W printer (88%), Dell 1815dn printer (88%), and VxWorks (88%). It also notes that there were no exact OS matches for the host due to non-ideal test conditions.

```
SF:x20Request</h1></body></html>";
MAC Address: 52:54:00:12:35:02 (QEMU virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose|bridge|printer
Running (JUST GUESSING): QEMU (98%), Oracle Virtualbox (96%), Samsung embedded (88%), Dell embedded (88%), Wind River VxWorks (88%)
OS_CPE: cpe:/a:qemu:qemu cpe:/o:oracle:virtualbox cpe:/h:samsung:clp-315w cpe:/h:dell:1815dn cpe:/o:windriver:vxworks
Aggressive OS guesses: QEMU user mode network gateway (98%), Oracle Virtualbox (96%), Samsung CLP-315W printer (88%), Dell 1815dn printer (88%), VxWorks (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
```



Utilized Several Nmap Scans:

- **Sudo nmap -sU <target IP address>**

```
s://nmap.org/submit/ .
Nmap done: 256 addresses (4 hosts up) scanned in 172.32 seconds
[sudo] password for sneakybot:
[sudo] password for sneakybot:
Starting Nmap 7.80 ( https://nmap.org ) at 2023-04-25 22:43 MDT
Nmap scan report for _gateway (10.0.2.2)
Host is up (0.00090s latency).
Not shown: 991 closed ports
PORT      STATE     SERVICE
67/udp    open|filtered dhcps
123/udp   open|filtered ntp
137/udp   open|filtered netbios-ns
500/udp   open|filtered isakmp
1900/udp  open|filtered upnp
5350/udp  open|filtered nat-t-like
5353/udp  open|filtered zeroconf
5355/udp  open|filtered llmnr
MAC Address: 52:54:00:12:35:02 (QEMU virtual NIC)

Nmap scan report for 10.0.2.3
Host is up (0.0011s latency).
Not shown: 993 filtered ports
PORT      STATE     SERVICE
67/udp    open|filtered dhcps
123/udp   open|filtered ntp
1900/udp  open|filtered upnp
```

- SU performs a UDP scan

- **sudo nmap -sS -O -sV -oN output.txt <target IP address or range>**

```
sneakybot@Ubuntu:~$ sudo nmap -sS -O -sV -oN output.txt 10.0.2.15/24
[sudo] password for sneakybot:
Starting Nmap 7.80 ( https://nmap.org ) at 2023-04-25 21:50 MDT
Nmap scan report for _gateway (10.0.2.2)
Host is up (0.00016s latency).
Not shown: 994 filtered ports
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
445/tcp   open  microsoft-ds?
6646/tcp  open  unknown
9010/tcp  open  websocket    WebSocket++ 0.8.2
9080/tcp  open  glrpc?
9100/tcp  open  jetdirect?
```

-sS: performs a SYN scan, which sends SYN packets to target ports to determine which are open.

-O: attempts to identify the OS on the target machine

-sV: performs version detection, finds the version of services running on open ports

-oN output.txt: saves the output of the scan in a file named "output.txt" in the current working directory.

- **List of all IP addresses of all machines in the network:**

Nmap scanned the IP address 10.0.2.2, 10.0.2.3, and 10.0.2.4 which had a latency of 0.0016 seconds.

- **OS and Version**

The host is likely running QEMU, Oracle Virtualbox, or Samsung embedded software.

- **Open Ports and services running on ports and versions**

PORT	STATE	SERVICE	VERSION
135/tcp	open	msrpc	Microsoft Windows RPC
445/tcp	open	microsoft-ds?	
6646/tcp	open	unknown	
9010/tcp	open	websocket	WebSocket++ 0.8.2
9080/tcp	open	glrpc?	
9100/tcp	open	jetdirect?	

- List of all IP addresses of all machines in the network:

Nmap also scanned Ubuntu (10.0.2.15) which had a 0.000059s latency

- OS and Version

The host has too many matching fingerprints, which makes it impossible to provide specific details about its operating system.

- Open Ports and services running on ports and versions

All 1000 scanned ports on Ubuntu (10.0.2.15) are closed

Part 2: Vulnerability Scanning

Nessus Screenshots:

Host	Vulnerabilities	Name	Family	Count
Windows	5.3	SMB Signing not required	Misc.	3
Linux	...	SSL (Multiple Issues)	General	4
Windows	3.3 *	DHCP Server Detection	Service detection	1
Windows	...	HTTP (Multiple Issues)	Web Servers	22
Windows	...	SMB (Multiple Issues)	Windows	18
Windows	...	Microsoft Windows (Multi...)	Windows	6
Windows	...	SSH (Multiple Issues)	General	5
Windows	...	TLS (Multiple Issues)	Service detection	2
Windows	...	DCE Services Enumeration	Windows	24
Windows	...	Nessus SYN scanner	Port scanners	23
Windows	...	Service Detection	Service detection	10
Windows	...	Common Platform Enumeratio...	General	4
Windows	...	Device Type	General	4
Windows	...	Ethernet MAC Address [High]	General	4
Windows	...	Nessus Scan Information	Settings	4

Host	Vulnerabilities
10.0.2.2	50
10.0.2.4	52
10.0.2.15	48
10.0.2.3	54

Top 5 Vulnerabilities/Info:

**** My Nessus only displayed three vulnerabilities, which were low, medium, and mixed.**

**** Professor has given me permission to list out info as well that Nessus provided.**

****Information is taken from Nessus Scan and Nist Website that lists all CVEs.**

Low:

DHCP Server Detection

Description:

This script contacts the remote DHCP server to retrieve information about the network layout, which could include sensitive details such as the NIS domain name or network web server list.

While this script does not exhibit any vulnerability, a local attacker may leverage DHCP to gain insights into the associated network. To prevent DHCP Server Detection, it is recommended to filter the network and eliminate any unused options. The primary vulnerability associated with this issue is CVE-2023-28488, which involves the client.c in gdhcp in ConnMan through version 1.41. Attackers operating a crafted DHCP server could exploit this vulnerability to trigger a stack-based buffer overflow and deny service, thereby terminating the connman process.

Medium:

SMB Signing not required

Description:

The remote SMB server has a weakness where it doesn't need message signing, allowing attackers to attack the server without authentication. To prevent these attacks, message signing should be turned on in the host's settings. On Windows, this can be done by enabling the 'Microsoft network server: Digitally sign communications (always)' policy setting.

On Samba, it can be done by enabling the 'server signing' setting. The problem is called CVE-2016-2115, and it affects Samba 3.x and 4.x. Attackers can modify the client-server data stream and trick SMB clients without requiring SMB signing in a DCERPC session over ncacn_np.

Medium:

SSL Certificate Cannot Be Trusted

Description:

The message means that the server's X.509 certificate might not be trustworthy for three reasons. First, the certificate chain's top, which the server sent, might not be recognized by a well-known public certificate authority. Second, the certificate chain could have an expired invalid certificate. Third, the signature on the certificate might not be checkable. Any problem with the chain makes it hard to confirm if the web server is real, making it easy for attackers to trick the remote host with man-in-the-middle attacks. This security flaw is linked to CVE-2010-4340, which involves libcloud before version 0.4.1. It fails to check SSL certificates for HTTPS connections, allowing attackers to bypass access controls and make fake certificates with man-in-the-middle attacks.

Info:

Nessus SYN scanner

Description:

This tool is a type of port scanner used by Nessus to quickly scan targets behind a firewall. It is not as strong as other scans that can break services, but it may still cause problems for weaker firewalls and may leave open connections on the target if the network is busy.

To keep your target safe, it is best to use an IP filter.

Info:

Enumerate IPv4 Interfaces via SSH

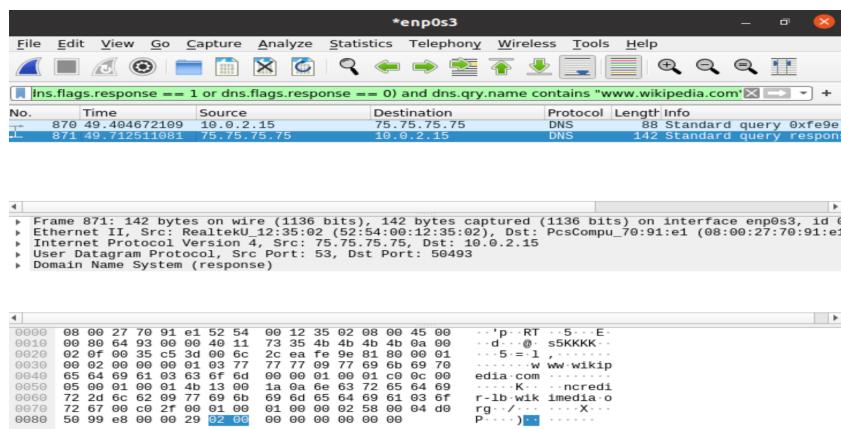
Description:

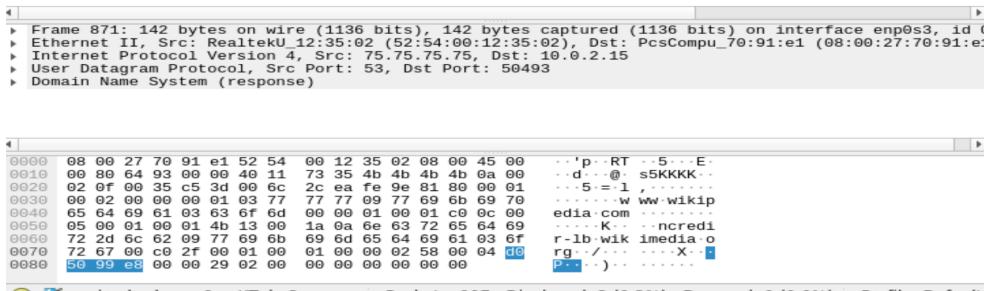
This message from Nessus says that it found out which network interfaces have IPv4 addresses on the remote computer by using SSH with the given login information. To make the computer safer, it's recommended to turn off any network interfaces that are not being used.

3.2

The transport layer protocol used in the DNS query packet is UDP, and the decimal number in the network layer header that identifies this protocol is 17.

3.3





3.4.

Raw Results:

1 0.000000000 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

2 1.005063308 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

3 2.010036189 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

6 3.014411317 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

7 4.015364477 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

8 5.015483046 PcsCompu_70:91:e1 Broadcast ARP 58 Who has 208.80.153.224? Tell 10.0.2.15

682 22.057669131 PcsCompu_70:91:e1 Broadcast ARP 58 Who has
208.80.153.224? Tell 10.0.2.15

870 49.404672109 10.0.2.15 75.75.75.75 DNS 88 Standard query 0xfe9e A
www.wikipedia.com OPT

871 49.712511081 75.75.75.75 10.0.2.15 DNS 142 Standard query response
0xfe9e A www.wikipedia.com CNAME ncredir-lb.wikimedia.org A 208.80.153.232 OPT

ARP Query/Response Pairs:

Query 1: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 1: Not captured

Query 2: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 2: Not captured

Query 3: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 3: Not captured

Query 4: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 4: Not captured

Query 5: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 5: Not captured

Query 6: "Who has 208.80.153.224? Tell 10.0.2.15"

Response 6: Not captured

DNS Query/Response Pairs:

Query 1: "Standard query 0x15a9 A doh2.gslb2.xfinity.com OPT"

Query source: IP address 10.0.2.15

Query destination: IP address 75.75.75.75

Response 1: Not captured

Query 2: "Standard query 0xfe9e A www.wikipedia.com OPT"

Query source: IP address 10.0.2.15

Query destination: IP address 75.75.75.75

Response 2: "Standard query response 0xfe9e A www.wikipedia.com CNAME
ncredir-lb.wikimedia.org A 208.80.153.232 OPT"

Response source: IP address 75.75.75.75

Response destination: IP address 10.0.2.15

DNS Query/Response Pairs:

Query 1: "Standard query 0x15a9 A doh2.gslb2.xfinity.com OPT"

Query source: IP address 10.0.2.15

Query destination: IP address 75.75.75.75

Response 1: Not captured

Query 2: "Standard query 0xfe9e A www.wikipedia.com OPT"

Query source: IP address 10.0.2.15

Query destination: IP address 75.75.75.75

Response 2: "Standard query response 0xfe9e A www.wikipedia.com CNAME
ncredir-lb.wikimedia.org A 208.80.153.232 OPT"

Response source: IP address 75.75.75.75

Response destination: IP address 10.0.2.15

Questions for ARP:

- **Info on query, MAC address, IP address, who answered, info in answer**

The ARP query was asking for the MAC address associated with the IP address 208.80.153.224.

The query was sent by the device with MAC address "PcsCompu_70:91:e1" and IP address 10.0.2.15, and was broadcasted to all devices on the network. There was no answer to the query in the captured packets.

The device with IP address 10.0.2.15 was trying to establish communication with the device associated with the IP address 208.80.153.224, but did not have the MAC address information necessary for the communication to occur. By sending an ARP query, it was waiting to obtain the MAC address of the device with the specified IP address.

Questions for DNS:

- **Info on query, MAC address, IP address, who answered, info in answer**

The query was asking for the IP address of "www.wikipedia.com" and was sent from IP address 10.0.2.15 with MAC address. The query was sent to the DNS server with IP address 75.75.75.75.

The response to the query was sent from the DNS server with IP address 75.75.75.75 and MAC address. It included the IP address of the CNAME "ncredir-lb.wikimedia.org" (208.80.153.232), which is the final destination for the query to resolve "www.wikipedia.com".

The query was made by a client device (with IP address 10.0.2.15) that was trying to access the website "www.wikipedia.com", and it needed to resolve the domain name to an IP address so it could establish a connection. The DNS server with IP address 75.75.75.75 was likely the primary DNS server configured on the client device.