Project

Network Administration

Student: Ismail Yasin

Date 14.01.2024

Lighthouse Labs

Table of Contents...

Introduction…2.0

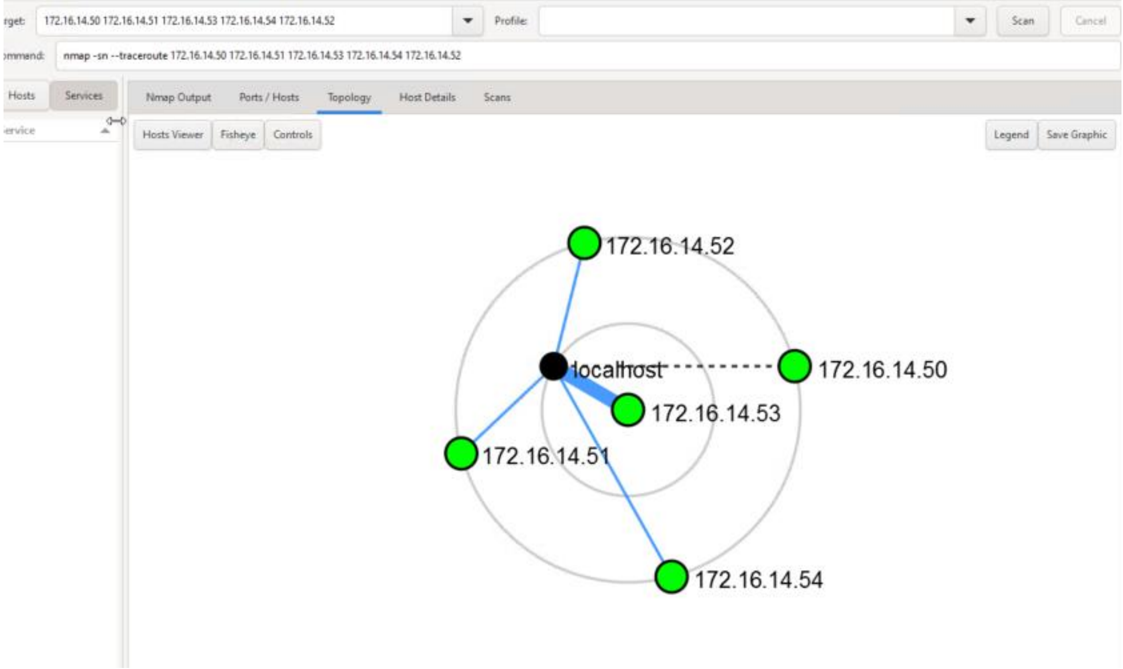
My project I will embark on a comprehensive exploration of the devices in EVE lab environment. In this endeavor, our primary objective is to generate a detailed report encapsulating valuable information about each device, shedding light on their information.

To achieve this, we will leverage two powerful and widely respected tools in the field of network scanning and analysis: Nmap and Wireshark. Nmap, a versatile network scanning tool, will be employed to conduct a systematic examination of the devices, unveiling critical details such as **open ports**, **services running**, and operating systems for hosts. Meanwhile, Wireshark, a packet analysis tool, will allow us to delve deeper into the network's communication.

EVE lab Topology….1.2

Before delving into the details of each device within our EVE Main Lab, it is prudent to familiarize ourselves with the overarching structure of our network lab. Understanding the topology of our EVE lab environment lays the foundation for a more meaningful interpretation of the subsequent device information. To disclose that information, we use this command to run in Nmap to virtualize the overall topology and see what it looks like in EVE environment.

```
Nmap --sn --traceroute 172.16.14.0/24
```



rget: 172.16.14.50 172.16.14.51 172.16.14.53 172.16.14.54 172.16.14.52

Profile:

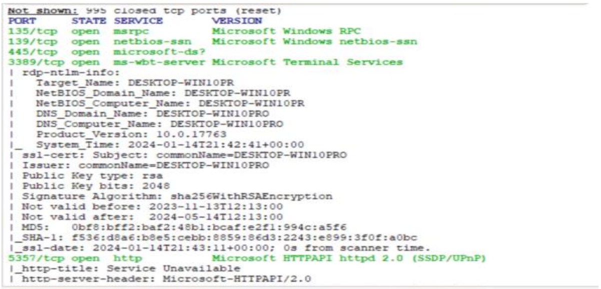
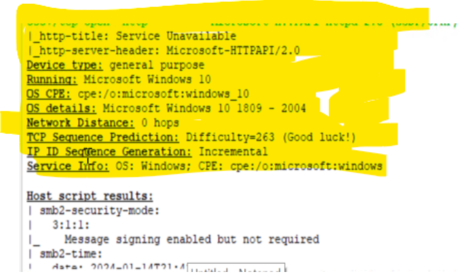
```
nmap -sn --traceroute 172.16.14.50 172.16.14.51 172.16.14.53 172.16.14.54 172.16.14.52
```

Network Devices information……1.3

To initiate the process of fetching detailed information and discovering the hosts within our EVE Lab environment, it is imperative to employ the powerful **Nmap network scanning tool, with** which we have already discovered the topology of the EVE Lab environment.

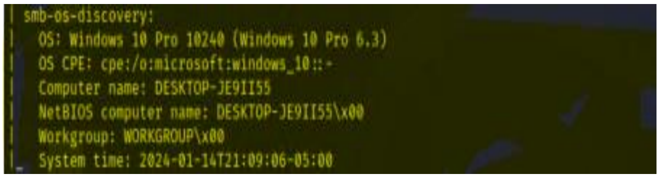
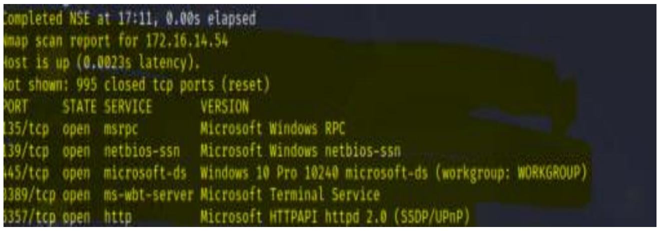
Given the current lack of complete insights into the device types, **available services**, **service versions**, and **open ports across our network**, Nmap will serve as the foundation for this exploration phase. Following the scanning procedure, a structured table will be created, providing a brief overview of each discovered host.

#nmap –T4 –A –v 172.16.14.50

Machine	**Host window 1**	**OSI layer**	 
Device Host Name	Desktop-WIN10PR		
Operating System & version	Microsoft Window 10 1809		
IP address	172.16.14.50/24	Layer 3 Network	
Open ports with associated services	TCP 135 open Microsoft windows RPC TCP 139 open Microsoft Windows Netbios-ssn TCP 445 open Microsoft-ds TCP 3389 open Microsoft Window Terminal Service TCP 5357 open Microsoft HTTPAPT httpd 2.0	Layer 3 Network	
MAC address	50:01:00:02:00:00	Layer 2 Data Link	
ARP Ping Scan elapsed time.	0.00s elapsed	Layer 2: Data Link	

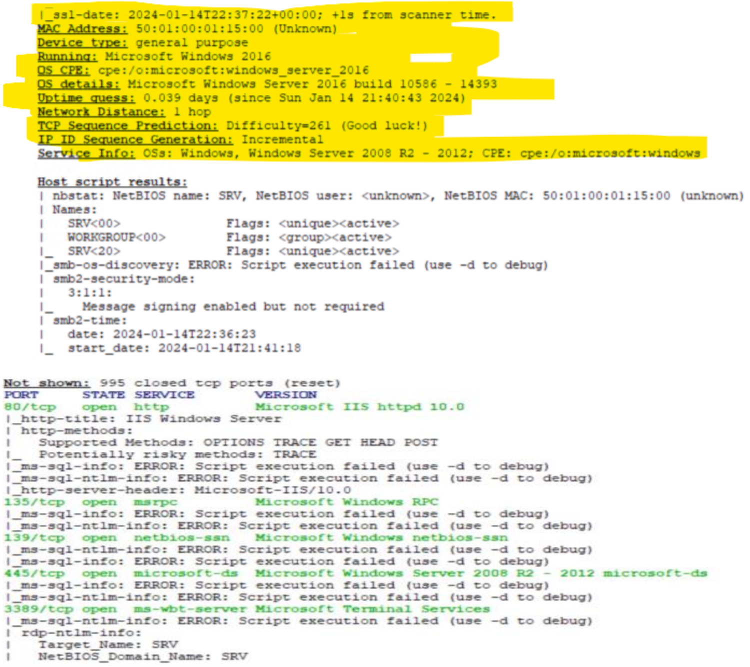
```
Not shown: 995 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: DESKTOP-WIN10PR
|   NetBIOS_Domain_Name: DESKTOP-WIN10PR
|   NetBIOS_Computer_Name: DESKTOP-WIN10PR
|   DNS_Domain_Name: DESKTOP-WIN10PRO
|   DNS_Computer_Name: DESKTOP-WIN10PRO
|   Product_Version: 10.0.17763
|_  System_Time: 2024-01-14T21:42:41+00:00
| ssl-cert: Subject: commonName=DESKTOP-WIN10PRO
| Issuer: commonName=DESKTOP-WIN10PRO
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2023-11-13T12:13:00
| Not valid after:  2024-05-14T12:13:00
| MD5:   0bf8:bff2:baf2:48b1:bcaf:e2f1:994c:a5f6
|_SHA-1: f536:d8a6:b8e5:cebb:8859:86d3:2243:e899:3f0f:a0bc
|_ssl-date: 2024-01-14T21:43:11+00:00; 0s from scanner time.
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Service Unavailable
| http-server-header: Microsoft-HTTPAPI/2.0
```

```
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Service Unavailable
|_http-server-header: Microsoft-HTTPAPI/2.0
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10
OS details: Microsoft Windows 10 1809 - 2004
Network Distance: 0 hops
TCP Sequence Prediction: Difficulty=263 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2024-01-14T21:4
```

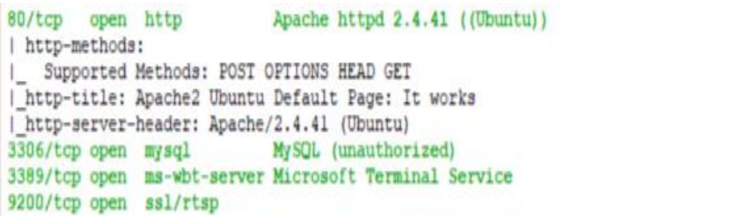
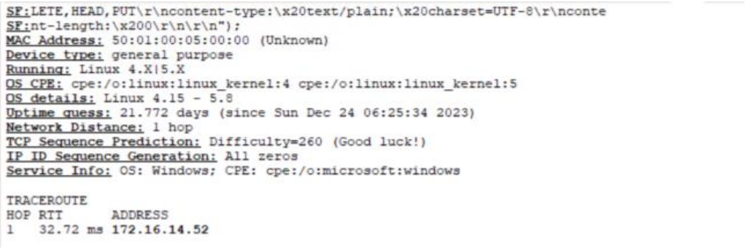
Machine	**Host window 2**	**OSI layer**	 
Device Host Name	Desktop-JE9ii5		
Operating System & version	Microsoft Window 10 Pro 6.3	.	
IP address	172.16.14.54/24	Layer 3 Network	
Open ports with associated services	TCP 135 open Microsoft windows RPC TCP 139 open Microsoft Windows Netbios-ssn TCP 445 open Microsoft-ds TCP 3389 open Microsoft Window Terminal Service TCP 5357 open Microsoft HTTPAPT httpd 2.0	Layer 3 Network	
MAC address	50:01:00:03:00:00	Layer 2 Data Link	
ARP Ping Scan elapsed time.	1.48s elapsed	Layer 2: Data Link	

```
| smb-os-discovery:
|   OS: Windows 10 Pro 10240 (Windows 10 Pro 6.3)
|   OS CPE: cpe:/o:microsoft:windows_10::-
|   Computer name: DESKTOP-JE9II55
|   NetBIOS computer name: DESKTOP-JE9II55\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2024-01-14T21:09:06-05:00
```

```
Completed NSE at 17:11, 0.00s elapsed
Nmap scan report for 172.16.14.54
Host is up (0.0023s latency).
Not shown: 995 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds  Windows 10 Pro 10240 microsoft-ds (workgroup: WORKGROUP)
3389/tcp open  ms-wbt-server Microsoft Terminal Service
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

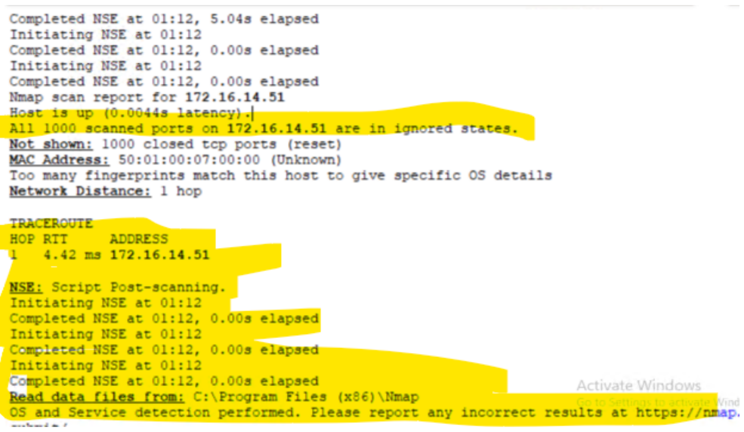
Machine	Host Window Server	OSI layer	
Device Host Name	Srv		
Operating System & version	Window Server 2016 build 10586		
IP address	172.16.14.53/24	Layer 3 Network	
Open ports with associated services	TCP port 80 open http Microsoft IIS 10.0 TCP port 135 open msrpc Microsoft Windows RPC TCP port 139 open netbios-ssn Microsoft Windows netbios-ssn TCP port 445 open Microsoft-ds Microsoft Window Server 2008 R2 TCP port 3389 open ms-wbt-server Microsoft Terminal Services.	Layer 3 Network	
MAC address	50:01:00:01:15:00	Layer 2 Data Link	
ARP Ping Scan elapsed time.	1.12s elapsed	Layer 2: Data Link	

```
|_ssl-date: 2024-01-14T22:37:22+00:00; +1s from scanner time.
MAC Address: 50:01:00:01:15:00 (Unknown)
Device type: general purpose
Running: Microsoft Windows 2016
OS CPE: cpe:/o:microsoft:windows_server_2016
OS details: Microsoft Windows Server 2016 build 10586 - 14393
Uptime guess: 0.039 days (since Sun Jan 14 21:40:43 2024)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Host script results:
| nbstat: NetBIOS name: SRV, NetBIOS user: <unknown>, NetBIOS MAC: 50:01:00:01:15:00 (unknown)
| Names:
|   SRV<00>              Flags: <unique><active>
|   WORKGROUP<00>        Flags: <group><active>
|_  SRV<20>              Flags: <unique><active>
|_smb-os-discovery: ERROR: Script execution failed (use -d to debug)
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2024-01-14T22:36:23
|_  start_date: 2024-01-14T21:41:18

Not shown: 995 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-title: IIS Windows Server
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_ms-sql-info: ERROR: Script execution failed (use -d to debug)
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
|_http-server-header: Microsoft-IIS/10.0
135/tcp  open  msrpc         Microsoft Windows RPC
|_ms-sql-info: ERROR: Script execution failed (use -d to debug)
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
|_ms-sql-info: ERROR: Script execution failed (use -d to debug)
445/tcp  open  microsoft-ds  Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
|_ms-sql-info: ERROR: Script execution failed (use -d to debug)
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ms-sql-ntlm-info: ERROR: Script execution failed (use -d to debug)
| rdp-ntlm-info:
|   Target_Name: SRV
|   NetBIOS_Domain_Name: SRV
```

Machine	Host Linux	OSI layer	 
Device Host Name	Srv		
Operating System & version	Linux 4.15 – 5.8		
IP address	172.16.14.52/24	Layer 3 Network	
Open ports with associated services	TCP port 80 open http Apache httpd 2.4.41 (Ubuntu) TCP port 3306 open Mysql TCP port 3389 open ms-wbt-server Microsoft Terminal Service TCP port 9200 open ssl/rtsp		
MAC address	50:01:00:05:00:00	Layer 2 Data Link	
ARP Ping Scan elapsed time.	0.00s elapsed	Layer 2: Data Link	

```
80/tcp   open  http          Apache httpd 2.4.41 ((Ubuntu))
| http-methods:
|_  Supported Methods: POST OPTIONS HEAD GET
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.41 (Ubuntu)
3306/tcp open  mysql         MySQL (unauthorized)
3389/tcp open  ms-wbt-server Microsoft Terminal Service
9200/tcp open  ssl/rtsp
```

```
SF:LETE,HEAD,PUT\r\ncontent-type:\x20text/plain;\x20charset=UTF-8\r\nconte
SF:nt-length:\x200\r\n\r\n");
MAC Address: 50:01:00:05:00:00 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Uptime guess: 21.772 days (since Sun Dec 24 06:25:34 2023)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=260 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE
HOP RTT      ADDRESS
1   32.72 ms 172.16.14.52
```

#nmap –T4 –A –v 172.16.14.51			
Machine	**Host Kali**	**OSI layer**	
Device Host Name	Undetected		
Operating System & version	Undetected		
IP address	172.16.14.51/24	Layer 3 Network	
Open ports with associated services	1,000 port scanned, no single port is open		
MAC address	50:01:00:07:00:00	Layer 2 Data Link	
ARP Ping Scan elapsed time.	0.00s elapsed	Layer 2: Data Link	

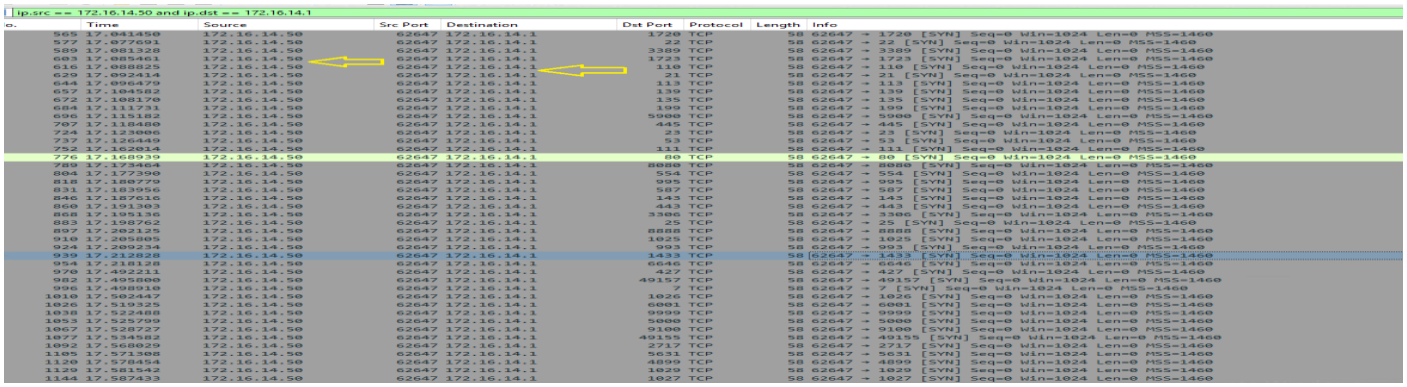
```
Completed NSE at 01:12, 5.04s elapsed
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Nmap scan report for 172.16.14.51
Host is up (0.0044s latency).
All 1000 scanned ports on 172.16.14.51 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 50:01:00:07:00:00 (Unknown)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   4.42 ms 172.16.14.51

NSE: Script Post-scanning.
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Initiating NSE at 01:12
Completed NSE at 01:12, 0.00s elapsed
Read data files from: C:\Program Files (x86)\Nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.
```

Wireshark packet capture..

TCP SYN packet capture in Wireshark. (Port scanning suspicious activity)

In this analysis, we delve into the data captured by Wireshark during the Nmap discovery process. Wireshark, a powerful network protocol analyzer, enables the real-time capture and inspection of data traversing a network. The provided snapshot illustrates Nmap's utilization of the ARP protocol for scanning purposes. Notably, the Nmap command initiates the exploration of the 172.16.14.1 network gateway. Within the captured data, we observe **TCP SYN communication between the scanning machine (host 172.16.14.50) and one of the target machines in EVE lab (host 172.16.14.1).** Nmap diligently sends ARP requests to over 1,000 ports in its quest to discover open Ports, although, in this instance, the target machine exhibits none. Subsequent examinations will spotlight how the Nmap tool successfully uncovers open ports on other hosts within the EVE lab environment.

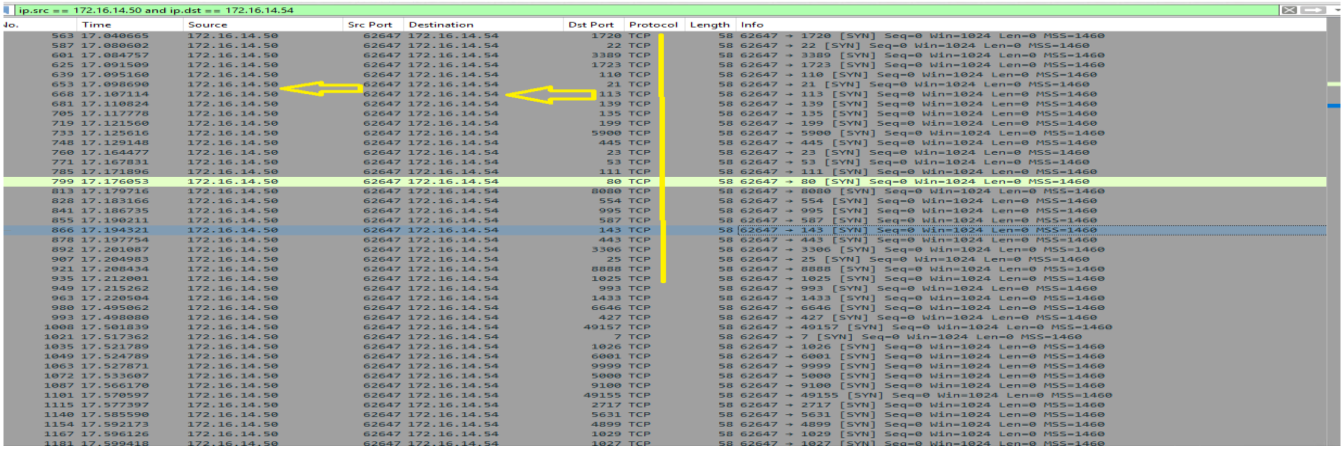
How Nmap trying to discover information see TCP SYN Source Add IP, Port and Destination Add and port
Fitering command: ip.src == 172.16.14.50 and ip.dst === 172.16.14.1



```
ip.src == 172.16.14.50 and ip.dst == 172.16.14.1
o.    Time       Source        Src Port Destination  Dst Port Protocol Length Info
 565 17.041450  172.16.14.50  62647    172.16.14.1   1720    TCP      58 62647 → 1720 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 577 17.077691  172.16.14.50  62647    172.16.14.1     22    TCP      58 62647 → 22 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 589 17.081328  172.16.14.50  62647    172.16.14.1   3389    TCP      58 62647 → 3389 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 603 17.085461  172.16.14.50  62647    172.16.14.1   1723    TCP      58 62647 → 1723 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 616 17.088825  172.16.14.50  62647    172.16.14.1    110    TCP      58 62647 → 110 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 629 17.092414  172.16.14.50  62647    172.16.14.1     21    TCP      58 62647 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 644 17.096479  172.16.14.50  62647    172.16.14.1    113    TCP      58 62647 → 113 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 657 17.104582  172.16.14.50  62647    172.16.14.1    139    TCP      58 62647 → 139 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 672 17.108170  172.16.14.50  62647    172.16.14.1    135    TCP      58 62647 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 684 17.111731  172.16.14.50  62647    172.16.14.1    199    TCP      58 62647 → 199 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 696 17.115182  172.16.14.50  62647    172.16.14.1   5900    TCP      58 62647 → 5900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 707 17.118480  172.16.14.50  62647    172.16.14.1    445    TCP      58 62647 → 445 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 724 17.123006  172.16.14.50  62647    172.16.14.1     23    TCP      58 62647 → 23 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 737 17.126449  172.16.14.50  62647    172.16.14.1     53    TCP      58 62647 → 53 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 752 17.162014  172.16.14.50  62647    172.16.14.1    111    TCP      58 62647 → 111 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 776 17.168939  172.16.14.50  62647    172.16.14.1     80    TCP      58 62647 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 789 17.173464  172.16.14.50  62647    172.16.14.1   8080    TCP      58 62647 → 8080 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 804 17.177390  172.16.14.50  62647    172.16.14.1    554    TCP      58 62647 → 554 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 818 17.180779  172.16.14.50  62647    172.16.14.1    995    TCP      58 62647 → 995 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 831 17.183956  172.16.14.50  62647    172.16.14.1    587    TCP      58 62647 → 587 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 846 17.187616  172.16.14.50  62647    172.16.14.1    143    TCP      58 62647 → 143 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 860 17.191303  172.16.14.50  62647    172.16.14.1    443    TCP      58 62647 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 868 17.195136  172.16.14.50  62647    172.16.14.1   3306    TCP      58 62647 → 3306 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 883 17.198762  172.16.14.50  62647    172.16.14.1     25    TCP      58 62647 → 25 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 897 17.202125  172.16.14.50  62647    172.16.14.1   8888    TCP      58 62647 → 8888 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 910 17.205805  172.16.14.50  62647    172.16.14.1   1025    TCP      58 62647 → 1025 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 924 17.209234  172.16.14.50  62647    172.16.14.1    993    TCP      58 62647 → 993 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 939 17.212828  172.16.14.50  62647    172.16.14.1   1433    TCP      58 62647 → 1433 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 954 17.218128  172.16.14.50  62647    172.16.14.1   6646    TCP      58 62647 → 6646 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 970 17.492211  172.16.14.50  62647    172.16.14.1    427    TCP      58 62647 → 427 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 982 17.495800  172.16.14.50  62647    172.16.14.1  49157    TCP      58 62647 → 49157 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 996 17.498910  172.16.14.50  62647    172.16.14.1      7    TCP      58 62647 → 7 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1010 17.502447  172.16.14.50  62647    172.16.14.1   1026    TCP      58 62647 → 1026 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1026 17.519325  172.16.14.50  62647    172.16.14.1   6001    TCP      58 62647 → 6001 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1038 17.522488  172.16.14.50  62647    172.16.14.1   9999    TCP      58 62647 → 9999 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1053 17.525799  172.16.14.50  62647    172.16.14.1   5000    TCP      58 62647 → 5000 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1067 17.528727  172.16.14.50  62647    172.16.14.1   9100    TCP      58 62647 → 9100 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1077 17.534582  172.16.14.50  62647    172.16.14.1  49155    TCP      58 62647 → 49155 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1092 17.568029  172.16.14.50  62647    172.16.14.1   2717    TCP      58 62647 → 2717 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1105 17.571308  172.16.14.50  62647    172.16.14.1   5631    TCP      58 62647 → 5631 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1120 17.578454  172.16.14.50  62647    172.16.14.1   4899    TCP      58 62647 → 4899 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1129 17.581542  172.16.14.50  62647    172.16.14.1   1029    TCP      58 62647 → 1029 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
1144 17.587433  172.16.14.50  62647    172.16.14.1   1027    TCP      58 62647 → 1027 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
```

TCP SYN packet capture in Wireshark. (Port port scan SYN data capture)

In this analysis, we focus on examining the data meticulously captured by Wireshark during the Nmap discovery process. The provided snapshot offers a indication Nmap use ARP protocol to discover its target. The Nmap command takes the initiative to explore the 172.16.14.1 network Window 2. Within the captured dataset, a notable observation emerges – the presence of TCP SYN communication between the scanning machine (host 172.16.14.50) and a specific target machine in the EVE lab environment (host 172.16.14.54). Nmap use ARP requests across more than 1,000 ports, aiming to disclose open ports on the target device (host Window 2 in EV lab). In this particular instance, the analysis not only reveals the absence of open ports on the target machine but also highlights Nmap's capability to extract detailed information, including detecting operating system version, and services running on those open ports for the specified target devices within the EVE lab. Here are one Example TCP SYN captured during Nmap scanning in EVE Lab environment.



```
ip.src == 172.16.14.50 and ip.dst == 172.16.14.54
No.   Time       Source          Src Port Destination   Dst Port Protocol Length Info
  563 17.040665  172.16.14.50    62647 172.16.14.54    1720 TCP        58 62647 → 1720 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  587 17.080602  172.16.14.50    62647 172.16.14.54      22 TCP        58 62647 → 22 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  601 17.084757  172.16.14.50    62647 172.16.14.54    3389 TCP        58 62647 → 3389 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  625 17.091509  172.16.14.50    62647 172.16.14.54    1723 TCP        58 62647 → 1723 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  639 17.095160  172.16.14.50    62647 172.16.14.54     110 TCP        58 62647 → 110 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  653 17.098690  172.16.14.50    62647 172.16.14.54      21 TCP        58 62647 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  668 17.107114  172.16.14.50    62647 172.16.14.54     113 TCP        58 62647 → 113 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  681 17.110824  172.16.14.50    62647 172.16.14.54     139 TCP        58 62647 → 139 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  705 17.117778  172.16.14.50    62647 172.16.14.54     135 TCP        58 62647 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  719 17.121560  172.16.14.50    62647 172.16.14.54     199 TCP        58 62647 → 199 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  733 17.125616  172.16.14.50    62647 172.16.14.54    5900 TCP        58 62647 → 5900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  748 17.129148  172.16.14.50    62647 172.16.14.54     445 TCP        58 62647 → 445 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  760 17.164477  172.16.14.50    62647 172.16.14.54      23 TCP        58 62647 → 23 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  771 17.167831  172.16.14.50    62647 172.16.14.54      53 TCP        58 62647 → 53 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  785 17.171896  172.16.14.50    62647 172.16.14.54     111 TCP        58 62647 → 111 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  799 17.176053  172.16.14.50    62647 172.16.14.54      80 TCP        58 62647 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  813 17.179716  172.16.14.50    62647 172.16.14.54    8080 TCP        58 62647 → 8080 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  828 17.183166  172.16.14.50    62647 172.16.14.54     554 TCP        58 62647 → 554 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  841 17.186735  172.16.14.50    62647 172.16.14.54     995 TCP        58 62647 → 995 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  855 17.190211  172.16.14.50    62647 172.16.14.54     587 TCP        58 62647 → 587 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  866 17.194321  172.16.14.50    62647 172.16.14.54     143 TCP        58 62647 → 143 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  878 17.197754  172.16.14.50    62647 172.16.14.54     443 TCP        58 62647 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  892 17.201087  172.16.14.50    62647 172.16.14.54    3306 TCP        58 62647 → 3306 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  907 17.204983  172.16.14.50    62647 172.16.14.54      25 TCP        58 62647 → 25 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  921 17.208434  172.16.14.50    62647 172.16.14.54    8888 TCP        58 62647 → 8888 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  935 17.212001  172.16.14.50    62647 172.16.14.54    1025 TCP        58 62647 → 1025 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  949 17.215262  172.16.14.50    62647 172.16.14.54     993 TCP        58 62647 → 993 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  963 17.220504  172.16.14.50    62647 172.16.14.54    1433 TCP        58 62647 → 1433 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  980 17.495062  172.16.14.50    62647 172.16.14.54    6646 TCP        58 62647 → 6646 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
  993 17.498080  172.16.14.50    62647 172.16.14.54     427 TCP        58 62647 → 427 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1008 17.501839  172.16.14.50    62647 172.16.14.54   49157 TCP        58 62647 → 49157 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1021 17.517362  172.16.14.50    62647 172.16.14.54       7 TCP        58 62647 → 7 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1035 17.521789  172.16.14.50    62647 172.16.14.54    1026 TCP        58 62647 → 1026 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1049 17.524789  172.16.14.50    62647 172.16.14.54    6001 TCP        58 62647 → 6001 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1063 17.527871  172.16.14.50    62647 172.16.14.54    9999 TCP        58 62647 → 9999 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1072 17.533607  172.16.14.50    62647 172.16.14.54    5000 TCP        58 62647 → 5000 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1087 17.566170  172.16.14.50    62647 172.16.14.54    9100 TCP        58 62647 → 9100 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1101 17.570597  172.16.14.50    62647 172.16.14.54   49155 TCP        58 62647 → 49155 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1115 17.577397  172.16.14.50    62647 172.16.14.54    2717 TCP        58 62647 → 2717 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1140 17.585590  172.16.14.50    62647 172.16.14.54    5631 TCP        58 62647 → 5631 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1154 17.592173  172.16.14.50    62647 172.16.14.54    4899 TCP        58 62647 → 4899 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1167 17.596126  172.16.14.50    62647 172.16.14.54    1029 TCP        58 62647 → 1029 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
 1181 17.599418  172.16.14.50    62647 172.16.14.54    1027 TCP        58 62647 → 1027 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
```

Information collection methods….

Wireshark.

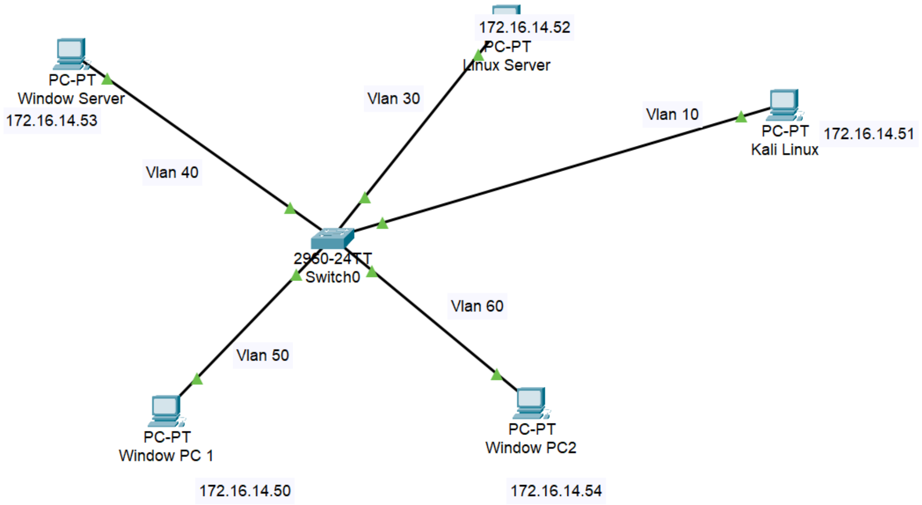
Wireshark is a network protocol analyzer that allows me to capture and inspect the data traveling back and forth on a network in real-time (Hanna, K. T. (2024). In my case, after running Nmap for host discovery, I use Wireshark to capture all the TCP SYN packets exchanged between the machine where Nmap is running (the switcher computer) and other machines in the lab. By capturing these packets, It allows me to analyze the communication between the devices in the network. This can provide valuable insights into the structure of the network and the types of services or applications running on the discovered hosts in EVE lab.

Nmap tool.

Nmap is short for Network Mapper. It is an open-source Linux command-line tool that is used to scan IP addresses and ports in a network and to detect installed applications (Shivanandhan, M. 2020). I used this tool in my discover for information about hosts include scanning open ports, operating system detection and service running for open ports.

Recommendation Securing EVE lab network.

I highly recommend implementing Vlan concept which separate network or and IP segmentation. These two techniques applying within the EVE lab environment to boost security. By adopting these can effectively stop attempts by tools like Nmap to discover hosts' information within the EVE lab network through the transmission of ICMP or ARP packets. Moreover, the integration of VLANs plays a pivotal role in enhancing security within the lab environment. VLANs enable logical partitioning within a single switch, allowing for the creation of multiple virtual local area networks. (Basan, M. 2023). This segmentation is particularly valuable when physical switch segmentation is impractical. These virtual partitions facilitate the division of a large network into smaller, more manageable broadcast domains, thereby enhancing overall network security.



Reference

1. Basan, M. (2023). *Vlans: Effective network segmentation for Security*. eSecurity Planet. https://www.esecurityplanet.com/networks/what-is-a-vlan/
2. Hanna, K. T. (2024). *What is wireshark?: Definition from TechTarget*. WhatIs. https://www.techtarget.com/whatis/definition/Wireshark
3. Shivanandhan, M. (2020). *What is nmap and how to use it – a tutorial for the greatest scanning tool of all time*. freeCodeCamp.org. https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/