**Project**

**Network Administration** Student: Ismail Yasin

**Date 14.01.2024**

**Lighthouse Labs**

**Table of Contents...**

**Introduction Section…………………………………………………………………………………………………………………… 1.1  
 EVE Lab Topology……………………………………………………………………………………………………………...………1.2**

**Network Device information………………………………………………………………………………………………………..2.0** Host Win 1………………………………………………………………………………………………………………...  
 Host Win 2………………………………………………………………………………………………………………...  
 Host Window server…………………………………………………………………………………………………..   
 Linux………………………………………………………………………………………………………………………….  
 Kali Linux………………………………………………………………………………………………………….………..

**Information Collection Methodology…………………………………………………………………………………………….3.0**

Nmap tool……………………………………………………………………………………………………………….3.1  
 Wireshark………………………………………………………………………………………………………………..3.2  
**Recommendations of detecting Nmap and Wireshark packet sniffing**……………………………3.3

**References…………………………………………………………………………………………………………………………………….4.0**

**Introduction…2.0**

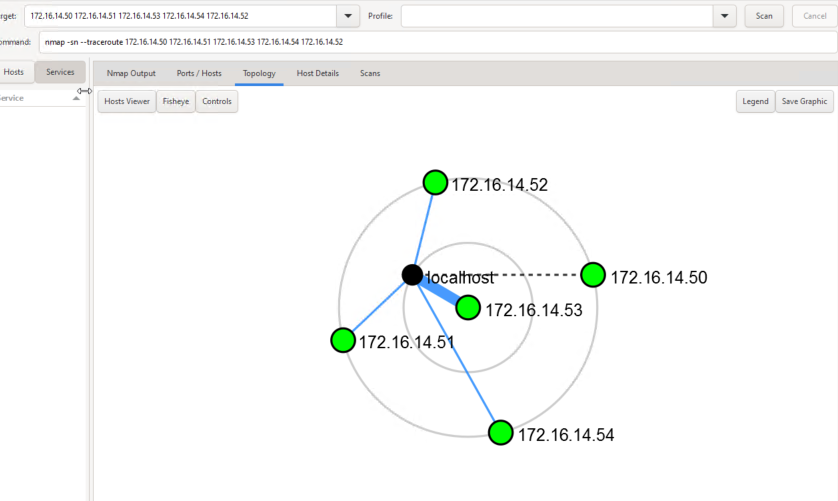
My project I will embark on a comprehensive exploration of the devices in EVE lab environment. In this endeavor, our primary objective is to generate a detailed report encapsulating valuable information about each device, shedding light on their information.

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 |



**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 |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 |

|  |  |  |  |
| --- | --- | --- | --- |
| **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 |

|  |  |  |  |
| --- | --- | --- | --- |
| #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 |

**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**

**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.

**Information collection methods….**

**Wireshark.**

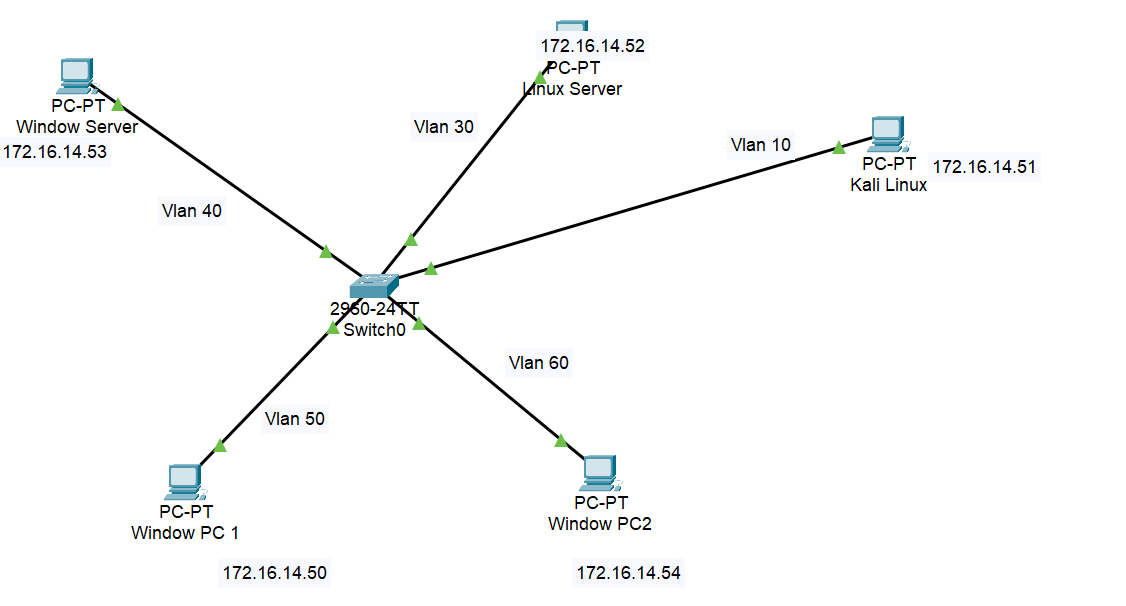
Wireshark is a network protocol analyzer that allows me to capture and inspect the data traveling back and forth on a network in real-time (Hanna, K. T. (2024). In my case, after running Nmap for host discovery, I use Wireshark to capture all the TCP SYN packets exchanged between the machine where Nmap 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](https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/)). 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](https://www.esecurityplanet.com/networks/what-is-a-vlan/)). 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/](https://www.esecurityplanet.com/networks/what-is-a-vlan/%20)
2. Hanna, K. T. (2024). *What is wireshark?: Definition from TechTarget*. WhatIs. [https://www.techtarget.com/whatis/definition/Wireshark](https://www.techtarget.com/whatis/definition/Wireshark%20)
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/](https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/%20)