Group no: - 8

Group Member:- **Shrut Shah – 19BCP125**

Shubham Kathiriya – 19BCP127

Vedant Patel – 19BCP138

Subject: - Cyber Security Lab

Division:-2

Lab 7:- Network Scanner

Aim:-

Study and implementing the Programme Network Scanner using Python.

Introduction:-

What is Network Scanner?

- A network scanner is a software tool that scans the network for connected devices.
- Also network scanner is one major tool for analysing the hosts that are available on the network. A network scanner is an IP scanner that is used for scanning the networks that are connected to several computers.
- It is also used for diagnostic and investigative purposes to find and categorize what devices are running on a network.
- This tool takes an IP address or a range of IP addresses as input and then scans each IP Addresses sequentially and determines whether a device is present on that particular IP address or not.

- It scans the network and returns an IP address and it's corresponding MAC address if the device is present.
- A popular tool that's commonly used Cyber Security professionals is nmap.
- Network scanning allows companies to:
    - Keep a tab on the available UDP and TCP network services
    - Access the operating systems in use by monitoring the IP responses
    - Identify the filtering systems between nodes.

- Network scanning involves network port scanning and vulnerability scanning.
- In port scanning, the scanner sends data packets to a specified service port number over the network. This helps to identify the available network services on a particular system for troubleshooting.
- Vulnerability scanning allows the scanner to detect known vulnerabilities of computing systems available on a network. This process helps the scanner to identify specific weak spots in application software or the operating system.
- Both network port and vulnerability scanning gather relevant information from the network. This information, when used by unauthorized personnel, poses a serious threat to the company.
- Network scanning is also closely related to packet sniffing or passive scanning.

❖ Types of Scaning:-

- Scanning has three types:

1. Port scanning - used to list open ports and services
2. Network scanning - used to list IP addresses
3. Vulnerability scanning - used to discover the presence of known vulnerabilities

❖ Scaning Techiniques:-

Port scanning techniques are extremely useful when it comes to identifying open ports. Scanning techniques represent different categories which are used based on protocol types. They are categorized into three categories:

1. Scanning ICMP network services
2. Scanning TCP network services
3. Scanning UDP network services

❖ How it Work?

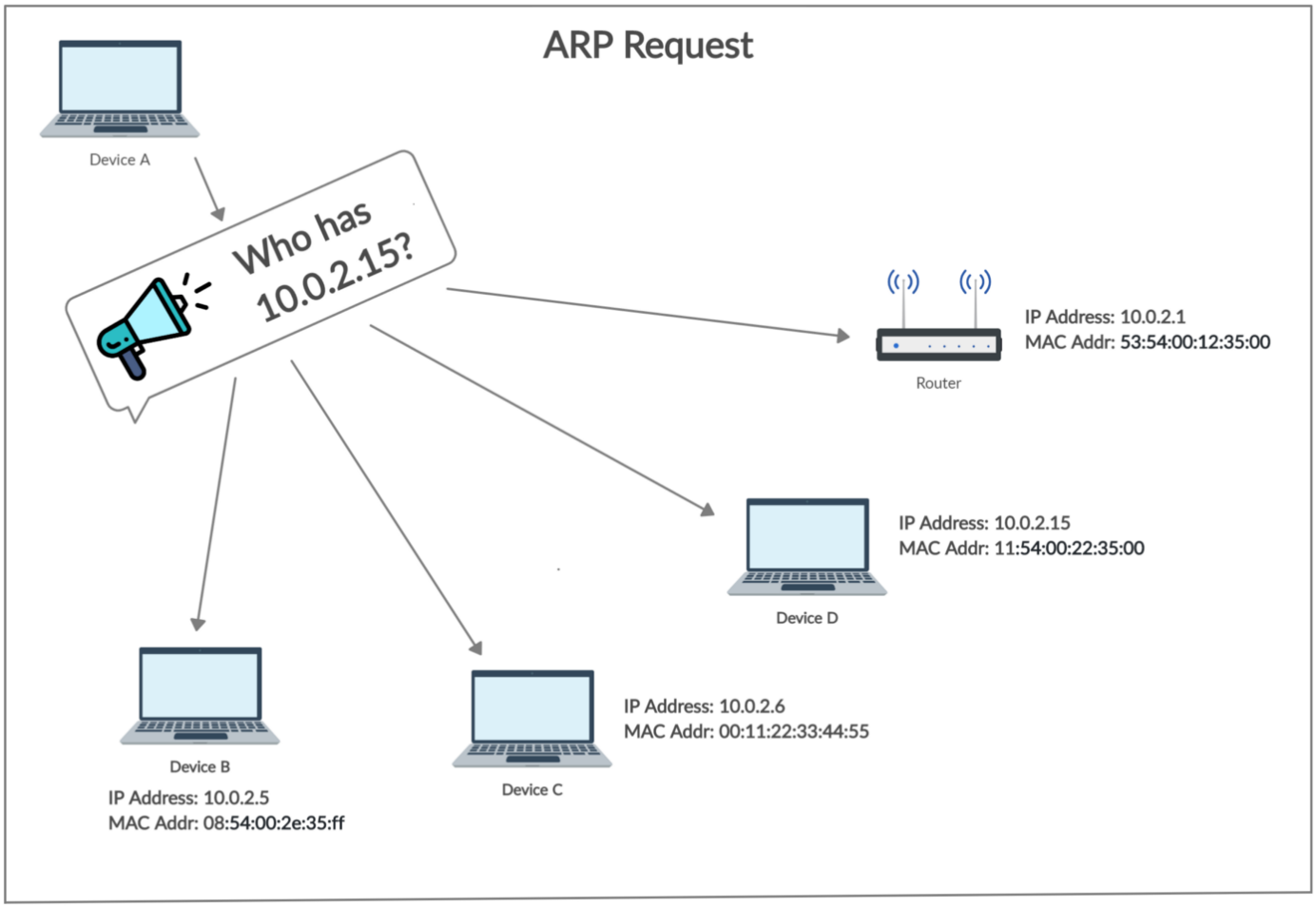
- To understand how the Network Scanner scans the entire network we need to first understand what is ARP (Address Resolution Protocol).
- In a network, most of the computers use the IP Address to communicate with other devices, however, in reality, the communication happens over the MAC Address.
- ARP is used to find out the MAC Address of a particular device whose IP address is known.
- For instance, a device wants to communicate with the other device on the network, then the sending device uses ARP to find the MAC Address of the device that it wants to communicate with.

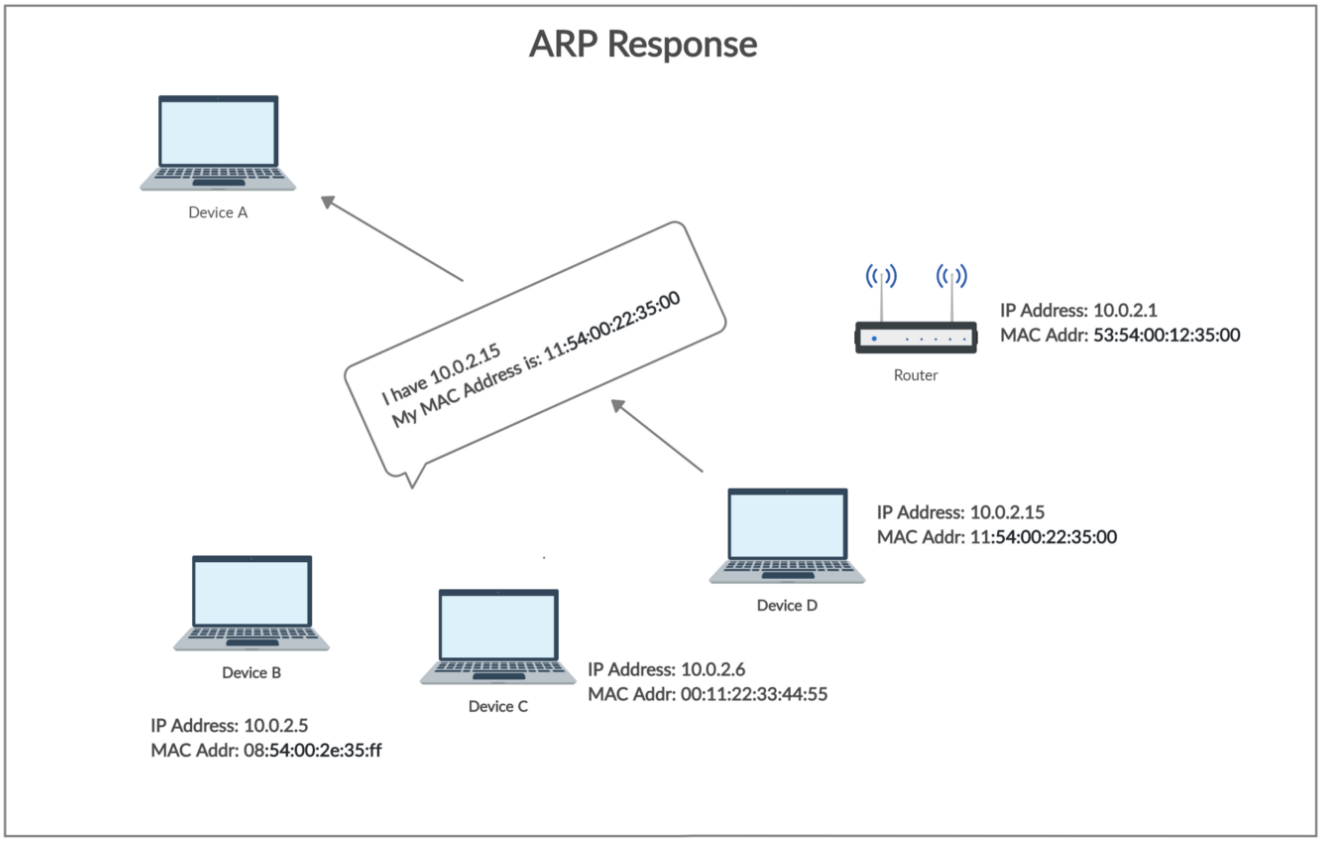
- ARP involves two steps to find the MAC address:

1. The sending device sends an ARP Request containing the IP Address of the device it wants to communicate with. This request is broadcasted meaning every device in the network will receive this but only the device with the intended IP address will respond.

2. After receiving the broadcast message, the device with the IP address equal to the IP address in the message will send an ARP Response containing its MAC Adress to the sender.

- Network Scanner uses ARP Request and Response to scan the entire network to find active devices on the network and also to find their MAC Addresses.
- If it is still not clear what ARP is and how it works then refer to the images of ARP Request and Response below.


ARP Request
Device A
Who has 10.0.2.15?
Router
IP Address: 10.0.2.1
MAC Addr: 53:54:00:12:35:00
Device D
IP Address: 10.0.2.15
MAC Addr: 11:54:00:22:35:00
Device B
IP Address: 10.0.2.5
MAC Addr: 08:54:00:2e:35:ff
Device C
IP Address: 10.0.2.6
MAC Addr: 00:11:22:33:44:55


ARP Response
Device A
I have 10.0.2.15
My MAC Address is: 11:54:00:22:35:00
Router
IP Address: 10.0.2.1
MAC Addr: 53:54:00:12:35:00
Device D
IP Address: 10.0.2.15
MAC Addr: 11:54:00:22:35:00
Device B
IP Address: 10.0.2.5
MAC Addr: 08:54:00:2e:35:ff
Device C
IP Address: 10.0.2.6
MAC Addr: 00:11:22:33:44:55

Implemetation:-

❖ Python Module which are needed:-

1. **argparse**:
2. **Scapy**: Enables the user to send, sniff and dissect and forge network packets. This capability allows the development of tools that can probe, scan, or attack networks.

❖ Python Code:-

```
#to print only value of dictionary not key so modification in client print
import scapy.all as scapy

def scan(ip):
    arp_request = scapy.ARP(pdst=ip)
    broadcast = scapy.Ether(dst="ff:ff:ff:ff:ff:ff")
    arp_request_broadcast = broadcast/arp_request
    answered_list = scapy.srp(arp_request_broadcast, timeout=1, verbose=False)[0]
    #it will only return me answered list by writing index [0]

    clients_list = []

    for element in answered_list:
        clients_dict = {"ip": element[1].psrc, "Mac": element[1].hwsrc}
        clients_list.append(clients_dict)
    return clients_list

def print_result(results_list):
    print("IP \t\t\t\t MAC Address\n-----------------------------------------------")
    for client in results_list:
        print(client["ip"] + "\t\t" + client["Mac"])


#scan("192.168.254.2")
scan_result = scan("192.168.254.2/24")
print_result(scan_result)
```

Ouput of the python code:-

```
C:\Users\Subham\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/
Subham/PycharmProjects/pythonProject/main.py

IP                    MAC Address
-----------------------------------------------
192.168.152.1       52:54:00:12:35:00
192.168.152.2       00:50:56:c0:42:00
192.168.152.3       08:00:27:22:d8:10
```

Benefits:-

1. ***Increased Network Performance***

- Network scanning plays a key role in increasing network performance and maximizing the speed of network operations.
- In a complex organizational network, multiple subnets of various IP addresses are assigned to several devices to improve their performance on the system. Scanning these devices helps to remove clogs and creates a free flow for optimal performance.

2. Protection Against Cyberattacks

- Network scanning is so useful that cybercriminals also use it to discover vulnerabilities in a network.
- When you fail to scan your network for threats and vulnerabilities, you're indirectly inviting attackers for a visit.
- Carrying out regular network scanning is an effective way to keep your system free from cyberattacks.
- It's similar to implementing intrusion detection systems to spot emerging threats.

3. Save Time and Money

- Scanning your network manually is tedious and time-consuming.
- The scanning process could linger for long.
- Your work is on hold, making you lose money in the long run.