**Assignment 04**

**Port Discovery**

**CS4061**

**Ethical Hacking Concepts and Practices**

**Submitted by:** Alina Asim

**Roll number:** 20i-0628

**Date:** 24-March-2024

**Table of Contents**

[ **Introduction** 2](#_Toc162170213)

[ **Steps** 3](#_Toc162170214)

[**ICMP ping scan:** 3](#_Toc162170215)

[**UDP Ping Scan** 4](#_Toc162170216)

[**Port Scan** 4](#_Toc162170217)

[**TCP Scan** 5](#_Toc162170218)

[**SYN Scan (Open Full scan)** 5](#_Toc162170219)

[Stealth Scan (Half Open Scan) 6](#_Toc162170220)

[**Inverse TCP Flag Scan** 7](#_Toc162170221)

[**ACK Flag Scan** 11](#_Toc162170222)

[**TTL Based Scan** 11](#_Toc162170223)

[**Window Scan** 12](#_Toc162170224)

[ **Summary** 13](#_Toc162170225)

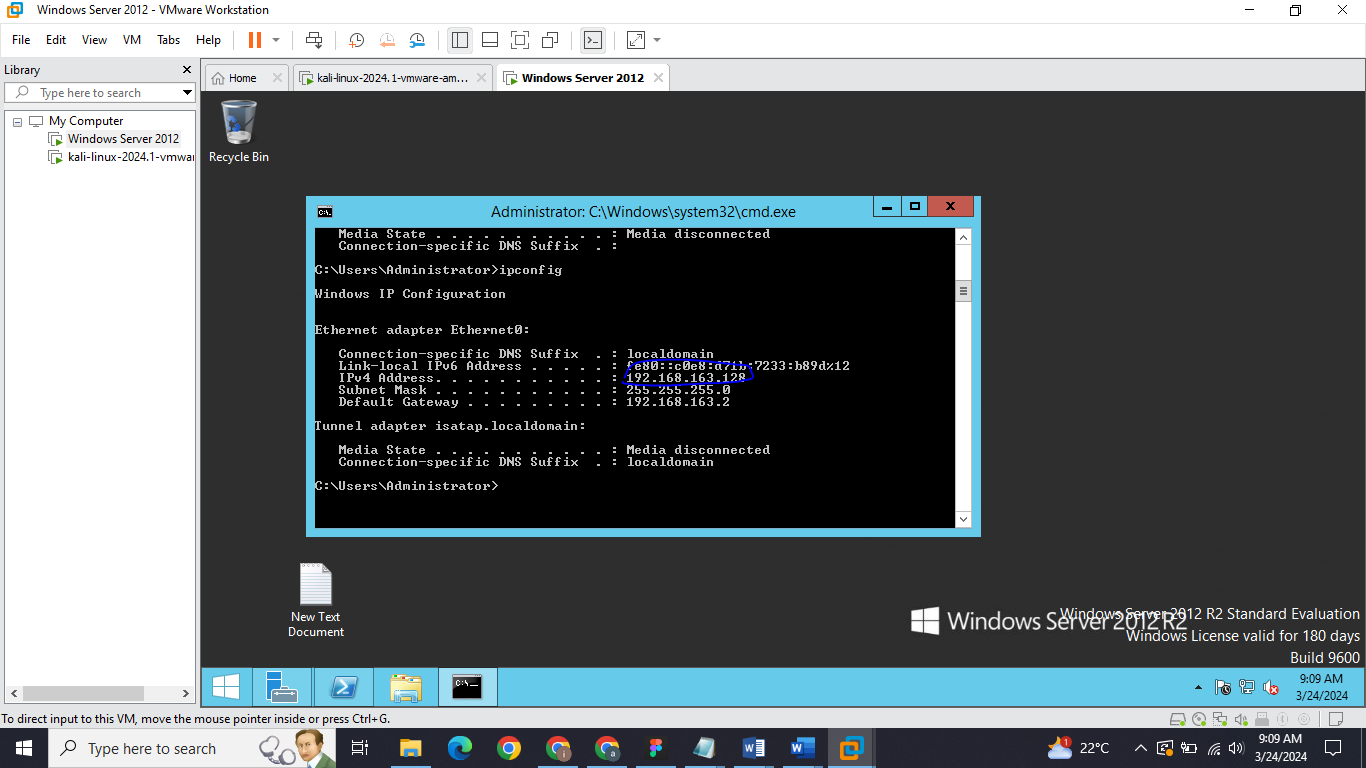
[ **References** 14](#_Toc162170226)

# **Introduction**

Introduction: In this assignment, I'll be exploring various port discovery techniques using tools like Nmap, Hping, or Scapy. Each technique serves a specific purpose in assessing network security. Screenshots of scan results, along with detailed command descriptions, will be provided. Additionally, if needed, assistance from generative AI tools like ChatGPT or Gemini will be sought, with relevant screenshots included. Both Wi-Fi and Ethernet interfaces will be utilized, with the attacker machine running Parrot or Kali OS, and victim machines having varied operating systems.

# **Steps**

To identify the IP range of our victim machine, we first use the **ipconfig** command on the victim machine, which in this case is a Windows server. This command provides us with both the IP address and subnet information. Once we have this information, we assign the IP range based on the subnet to execute the host scanning commands from our attacker machine. This ensures that our scanning commands target the correct range of IP addresses where potential victim machines might reside.



## **ICMP ping scan:**

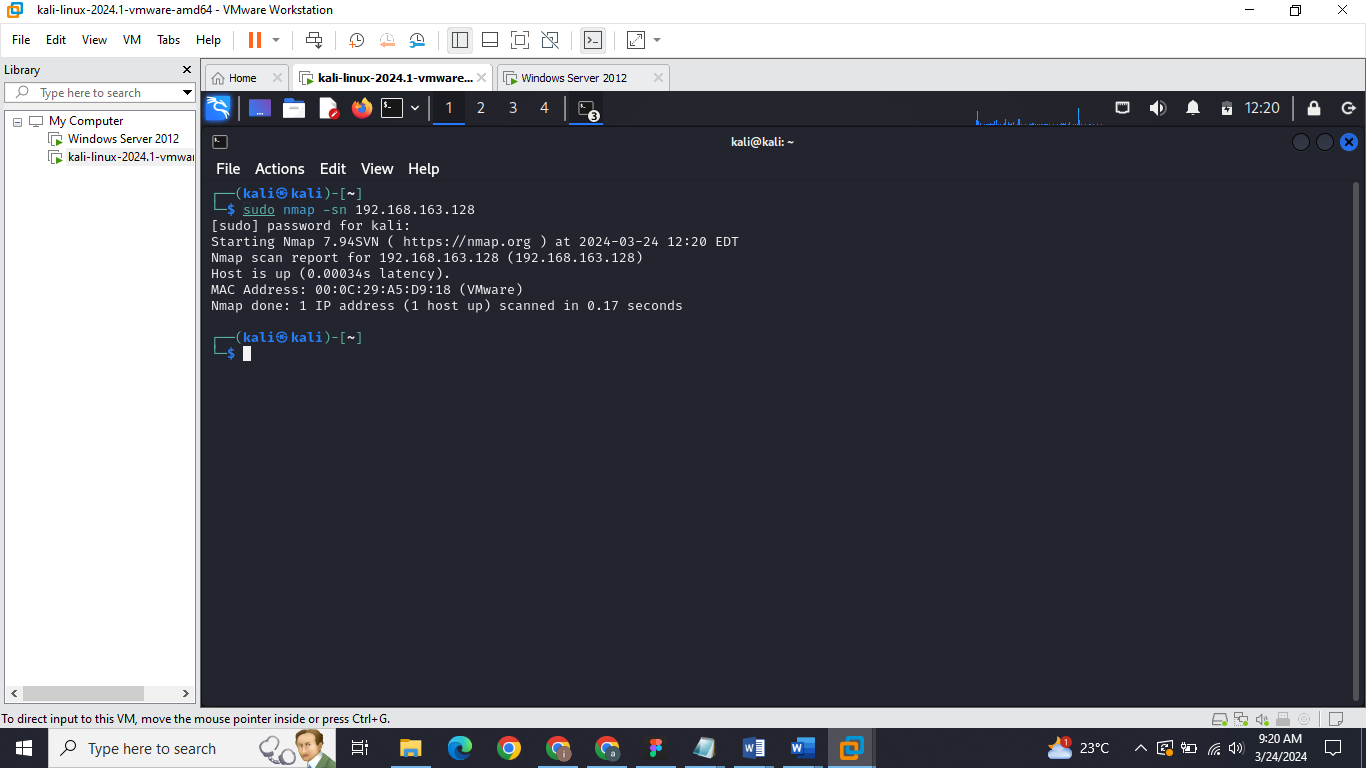
**Command:**

*sudo**nmap –sn <Target >*

*sudo**nmap -sn 192.168.163.128*

The command **nmap -sn 192.168.163.128** initiates an ICMP ping scan using Nmap, specifically tailored for host discovery. With the **-sn** option indicating "No port scan," the tool bypasses port scanning and solely focuses on identifying active hosts within the designated target network. ICMP Echo Request packets are dispatched to the target hosts, and their responses, typically ICMP Echo Replies, determine host availability. By analyzing these responses, Nmap swiftly compiles a list of online hosts, along with their respective IP addresses and status (up or down).

**Output:**



The output indicates that the Nmap scan successfully determined the status of the target IP address (192.168.163.128) as "up," implying that the host is active and reachable on the network. It also provides details such as the MAC address (00:0C:29:A5:D9:18), which reveals the manufacturer (VMware) of the network interface card. The latency, indicating the response time from the host, was extremely low (0.00034 seconds), suggesting fast network responsiveness.

## **UDP Ping Scan**

### **Port Scan**

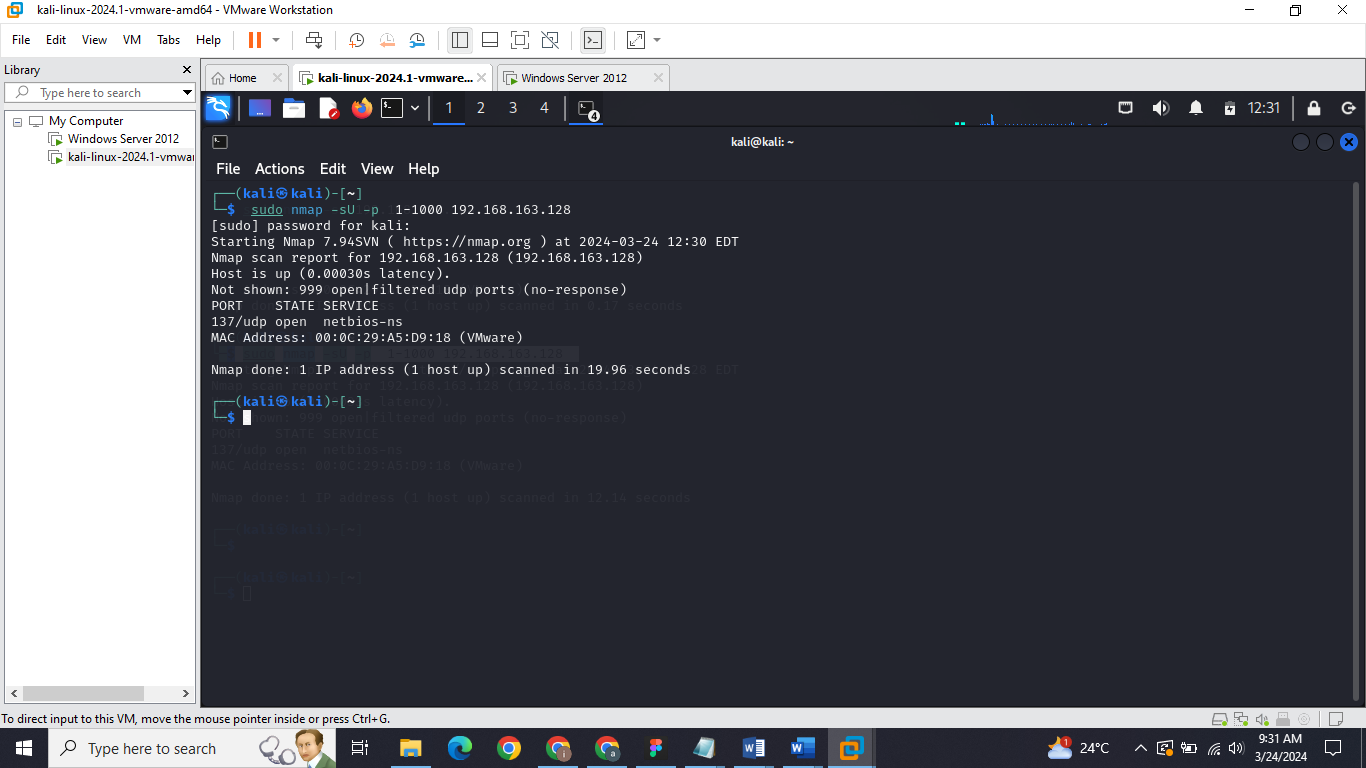
**Command:**

*sudo nmap -sU -p <port range> <target IP range>*

*sudo nmap -sU -p 1-1000 192.168.163.128*

The command nmap -sU -p 0-65535 192.168.163.128 initiates a UDP port scan using Nmap. It scans all 65,536 UDP ports on the target host or network specified by <target>. UDP (User Datagram Protocol) is a connectionless protocol commonly used for various network services. With -sU, Nmap specifically instructs to perform a UDP scan. For each port within the specified range (0 to 65535), Nmap sends UDP packets. If a response is received, it indicates an open port, suggesting the presence of a UDP service.

**Output:**



The output indicates that the UDP port scan identified the target host (192.168.163.128) as active and responsive, with a low latency. Out of the scanned UDP ports (1-1000), only port 137/udp was found to be open, revealing the presence of the "netbios-ns" service. However, 999 ports appeared to be either open or filtered but did not respond to the scan. The MAC address (00:0C:29:A5:D9:18) of the target host's network interface card suggests it belongs to a VMware virtual machine.

## **TCP Scan**

### **SYN Scan (Open Full scan)**

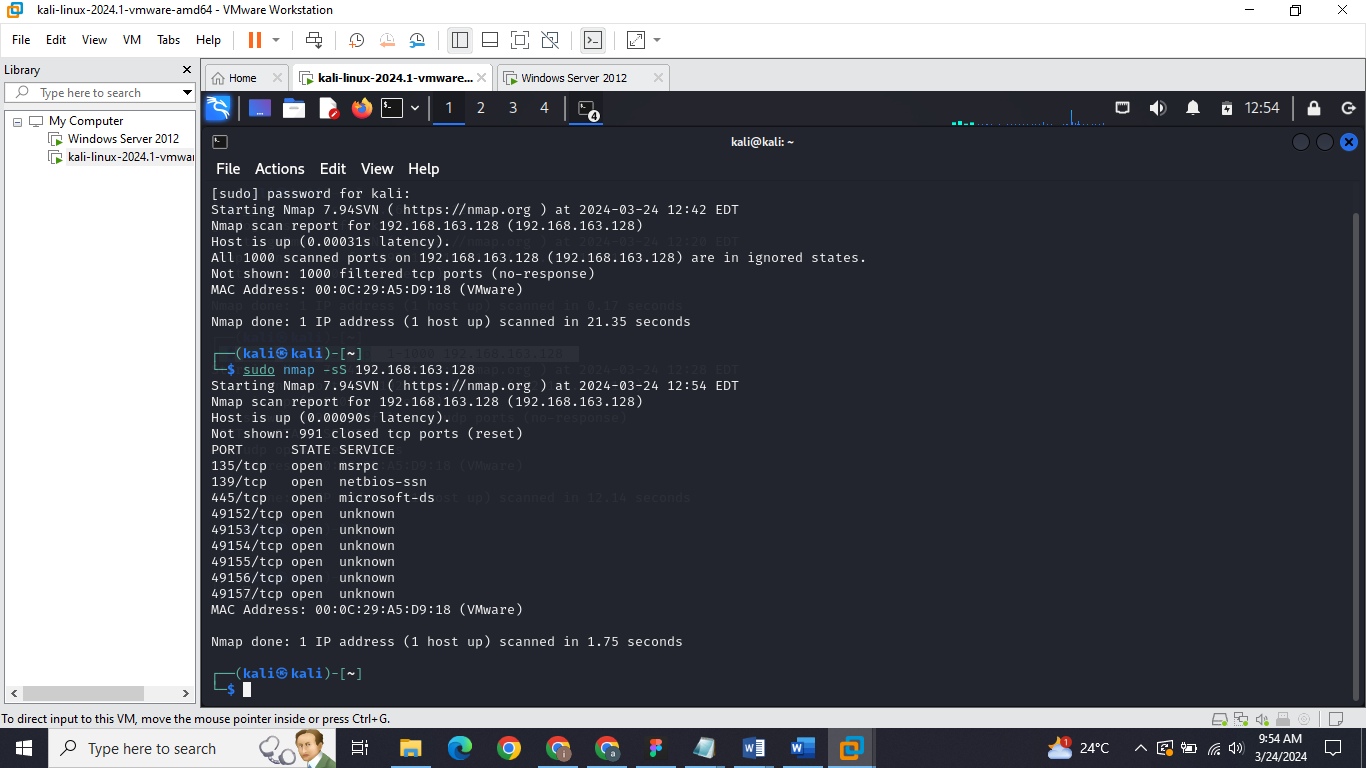
**Command:**

*sudo nmap -sS <target IP >*

*sudo nmap -sS 192.168.163.128*

The command **nmap -sS <target>** initiates a TCP SYN scan using Nmap. It sends SYN packets to target ports on the specified host(s). If a port responds with SYN/ACK, it's considered open; if it responds with RST, it's closed. This stealthy scan provides insights into available services without completing the full TCP handshake, making it less detectable by intrusion detection systems.

**Output:**



In this Nmap output, the scan reveals that the target host (192.168.163.128) is up and responsive with a quick response time. It identifies several open TCP ports: 135/tcp (MSRPC), 139/tcp (NetBIOS-SSN), 445/tcp (Microsoft-DS), and ports in the range 49152-49157/tcp with unknown services. The MAC address (00:0C:29:A5:D9:18) indicates the host is a VMware virtual machine.

### **Stealth Scan (Half Open Scan)**

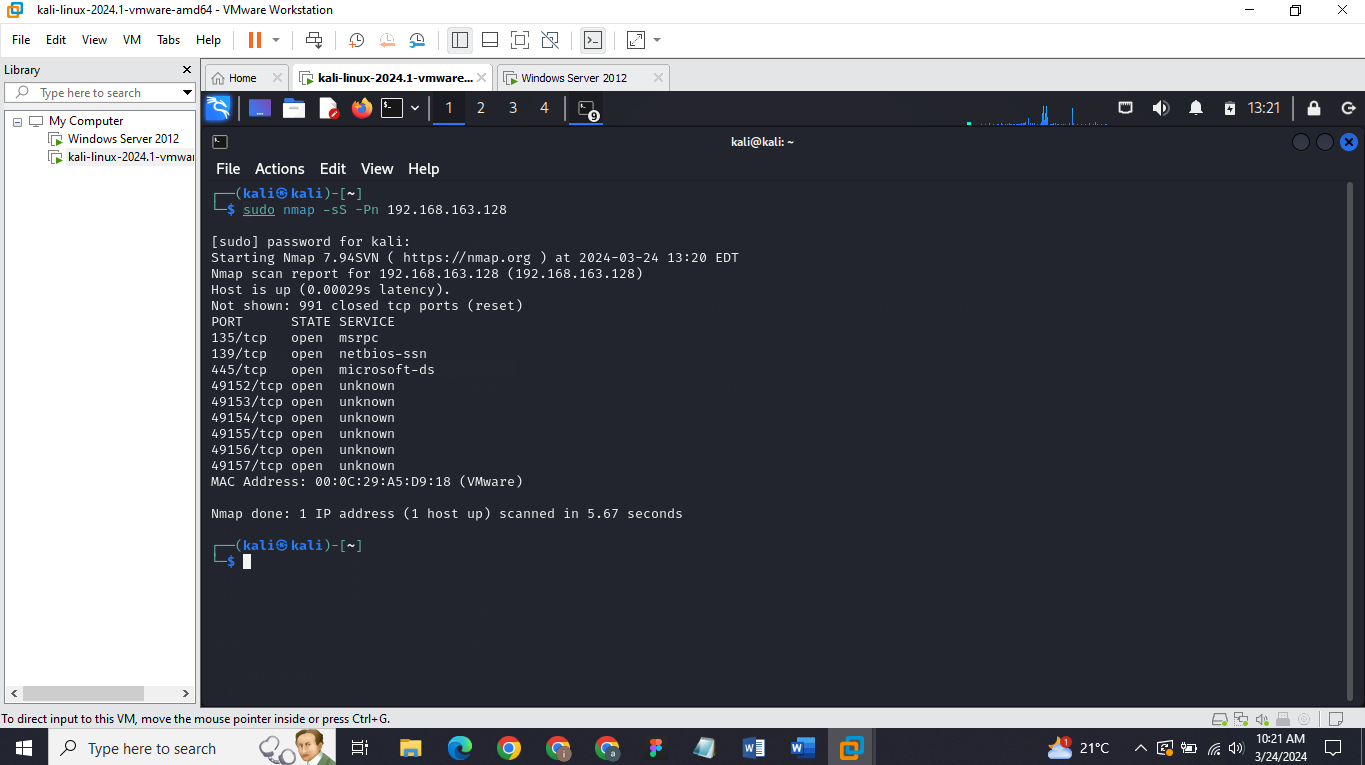
**Command:**

*Sudo nmap –sS -Pn <target>*

*sudo nmap –sS -Pn 192.168.163.128*

The command sudo nmap -sS -Pn 192.168.163.128 enables Nmap to conduct a Stealth Scan on the target IP address (192.168.163.128) without engaging in host discovery. This Stealth Scan, executed with the -sS option, involves sending SYN packets to target ports while withholding the completion of the TCP handshake, enhancing stealthiness. By utilizing the -Pn option, Nmap bypasses host discovery, treating all hosts as online, thereby accelerating the scanning process.

**Output:**



The output from the command sudo nmap -sS -Pn 192.168.163.128 confirms a successful Stealth Scan (Half Open Scan) on the target host (192.168.163.128). It reveals open TCP ports, including 135, 139, 445, and others in the range 49152-49157, running various services like MSRPC and NetBIOS-SSN. This scan is stealthier than a default SYN Scan because it doesn't complete the full TCP handshake, reducing the chances of detection by intrusion detection systems.

### **Inverse TCP Flag Scan**

#### **FIN Scan**

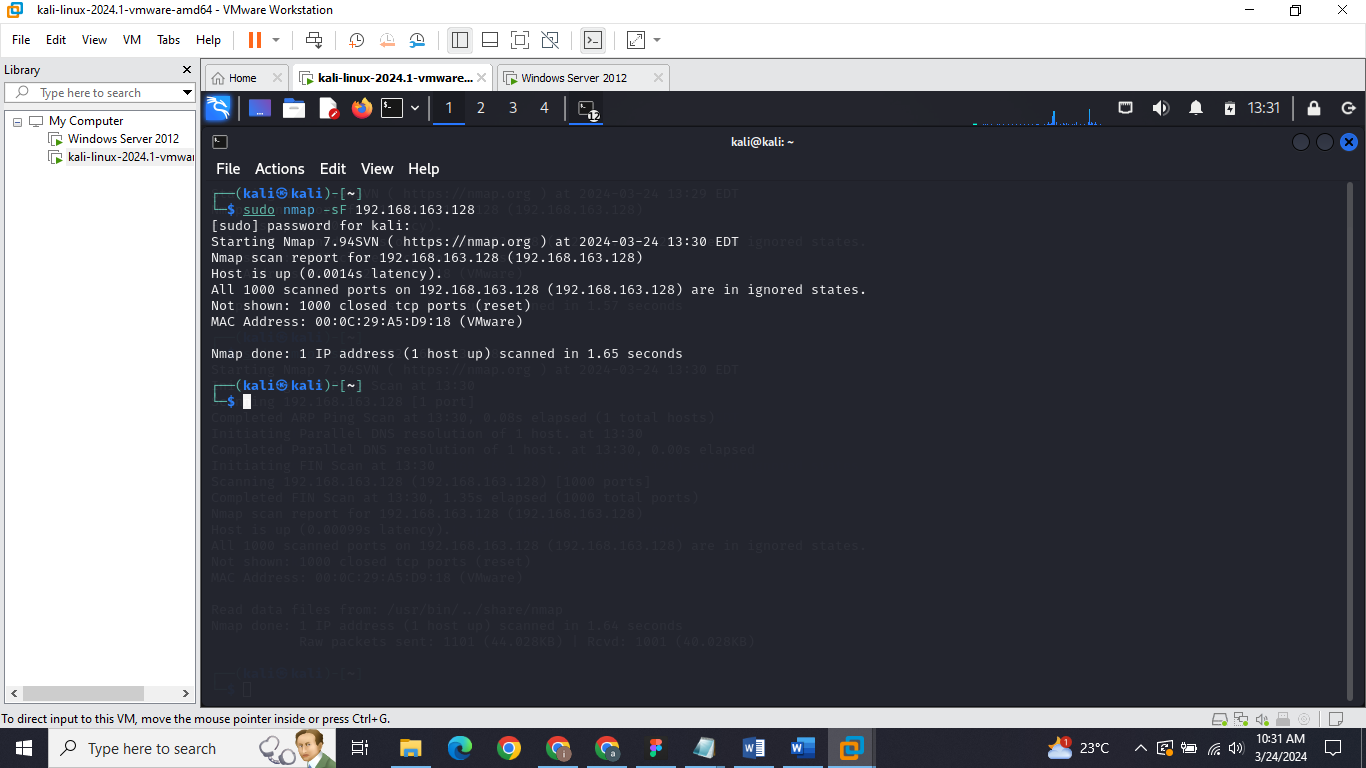
**Command:**

*sudo nmap -sF <target IP >*

*sudo nmap –sF 192.168.163.128*

The command sudo nmap -sF <target IP> initiates a FIN Scan, a stealthy port scanning technique employed by Nmap. In this scan, Nmap sends TCP packets with the FIN (Finish) flag set to the target ports. Unlike traditional scans that utilize SYN or ACK packets, the FIN Scan exploits the behavior of some systems, which may respond differently to unexpected FIN packets. If a port is open, it typically ignores the FIN packet, resulting in no response. Conversely, if the port is closed, it may respond with an RST (Reset) packet. This scan is stealthy because it doesn't establish a full TCP connection, making it less likely to be logged by intrusion detection systems.

**Output:**

In this output, the command sudo nmap -sF 192.168.163.128 executed a FIN Scan on the target IP address 192.168.163.128. However, the scan results indicate that all 1000 scanned ports on the target are in "ignored" states. This suggests that the FIN packets sent by Nmap received no response from any of the scanned ports. Typically, this lack of response occurs because most systems nowadays are configured to drop unexpected FIN packets, thereby making the FIN Scan less effective. Consequently, the scan detected no open ports, and all scanned ports are reported as closed. The MAC address provided (00:0C:29:A5:D9:18) corresponds to a VMware virtual machine.

#### **XMAS Scan**

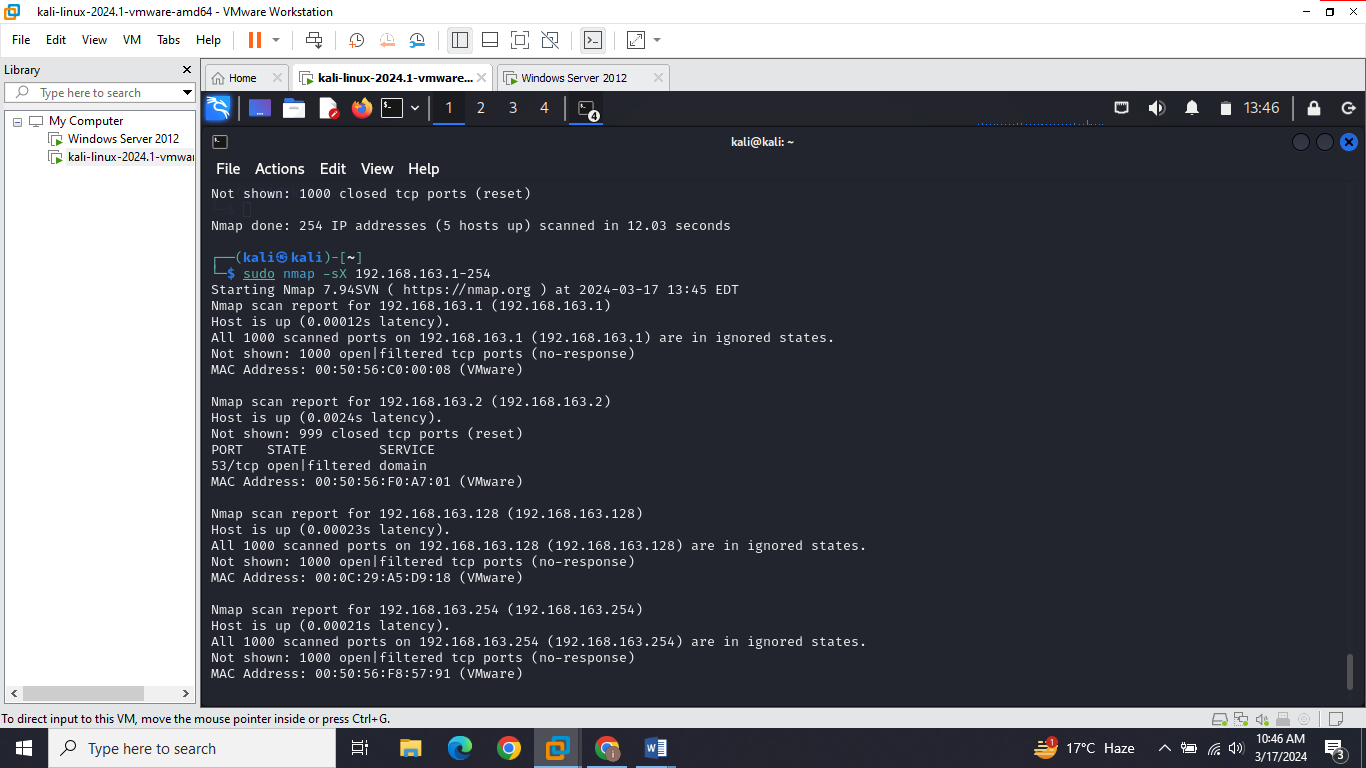
**Command:**

*sudo nmap -sX <target IP range>*

*sudo nmap -sX 192.168.163.1-254*

Similar to the NULL scan, this scan sends packets with the FIN, URG, and PSH flags set. If the port is closed, it should respond with a RST packet. If no response is received, it's marked as state filtered.

**Output:**



#### **Null Scan**

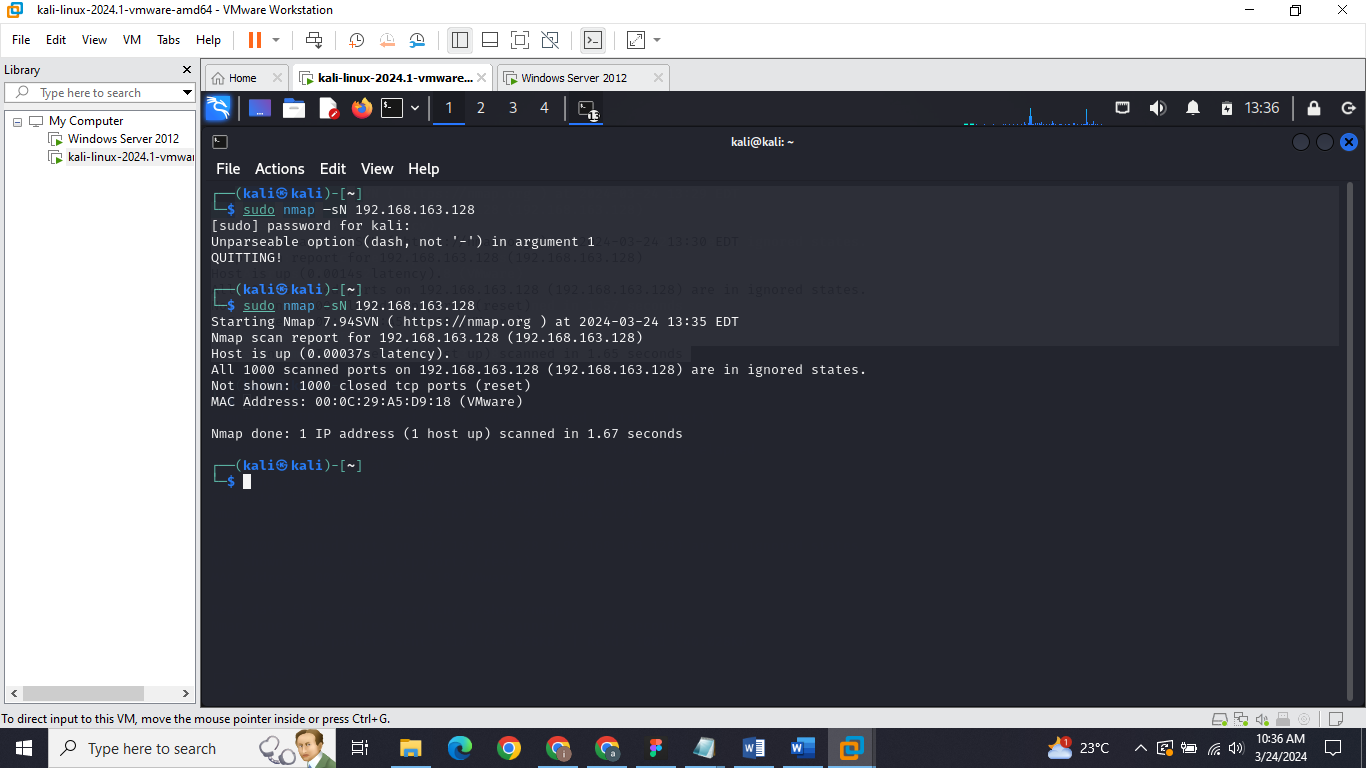
**Command:**

*sudo nmap -sN <target IP range>*

*sudo nmap –sN 192.168.163.128*

The command sudo nmap -sN <target IP range> initiates a TCP Null Scan using Nmap. In this scan, Nmap sends TCP packets with no TCP flags set to the target's ports. It analyzes responses to determine if ports are open, closed, or filtered. This technique leverages varying responses from systems to identify potential vulnerabilities while minimizing detection.

**Output:**



In this output, the command sudo nmap -sN 192.168.163.128 performed a TCP Null Scan on the target IP address 192.168.163.128. However, the results indicate that all 1000 scanned ports on the target are in "ignored" states, meaning there were no responses to the Null packets sent by Nmap. This suggests that the Null Scan did not elicit any discernible responses from the target's ports. As a result, no open ports were identified, and all scanned ports are reported as closed. The MAC address provided (00:0C:29:A5:D9:18) corresponds to a VMware virtual machine

#### **Maimon Scan**

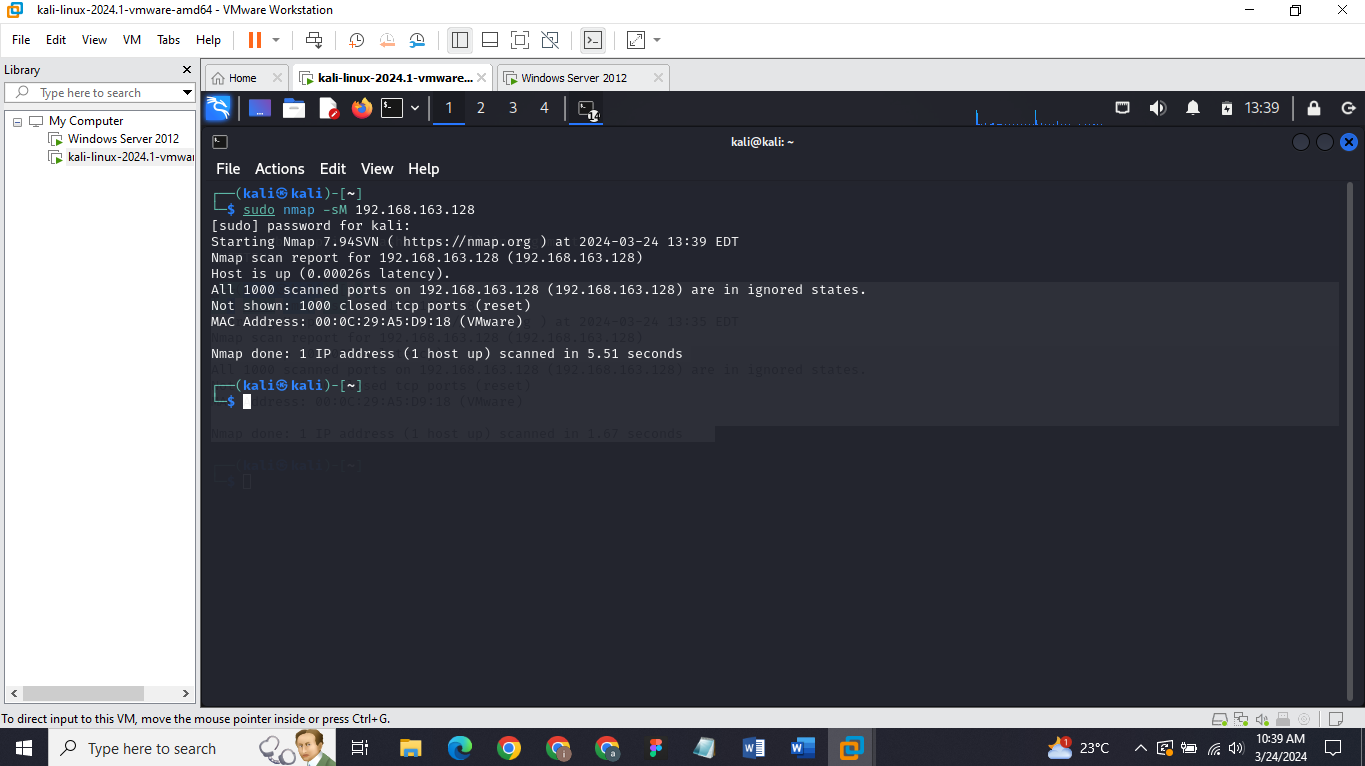
**Command:**

*sudo nmap -sM <target IP >*

*sudo nmap –sM 192.168.163.128*

The command sudo nmap -sM <target IP> initiates a TCP Maimon Scan using Nmap. In this scan, Nmap sends TCP packets with the "Maimon" flag set to the target's ports. This flag is not part of the TCP protocol, thus making it a non-standard scan technique. The purpose of the Maimon Scan is to determine how the target system reacts to these unusual packets.

**Output:**



In this output, the command **sudo nmap -sM 192.168.163.**128 executed a TCP Maimon Scan on the target IP address 192.168.163.128. However, the scan results indicate that all 1000 scanned ports on the target are in "ignored" states. This suggests that the Maimon Scan did not elicit any discernible responses from the target's ports.

## **ACK Flag Scan**

### **TTL Based Scan**

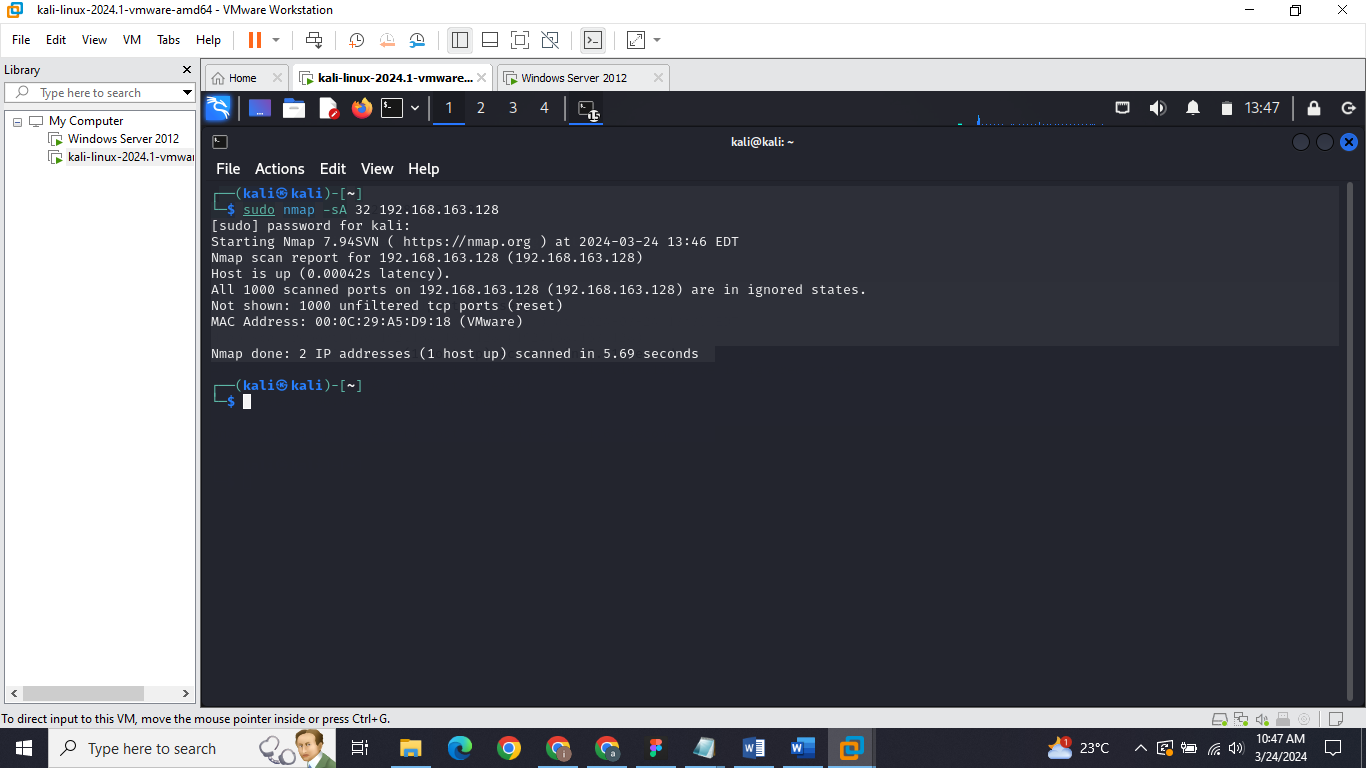
**Command:**

*sudo nmap -sA --ttl <value> <target>*

*sudo nmap -sA 32 192.168.163.128*

The command **sudo nmap -sA --ttl <value> <target>** launches an ACK Scan with a custom TTL (Time-to-Live) value using Nmap. This scan sends TCP ACK packets to the target's ports, observing their responses. By specifying a TTL value, you control the maximum number of router hops for the packets, optimizing scan performance across varied network configurations. The results unveil how the target's ports react to the ACK packets, aiding in the identification of firewall rules and network filtering mechanisms

**Output:**



In this output, the command sudo nmap -sA 32 192.168.163.128 executed an ACK Scan on the target IP address 192.168.163.128 with a custom TTL (Time-to-Live) value of 32. However, the scan results indicate that all 1000 scanned ports on the target are in "ignored" states. This suggests that the ACK Scan did not elicit any discernible responses from the target's ports. As a result, no open or closed ports were identified, and all scanned ports are reported as unfiltered.

### **Window Scan**

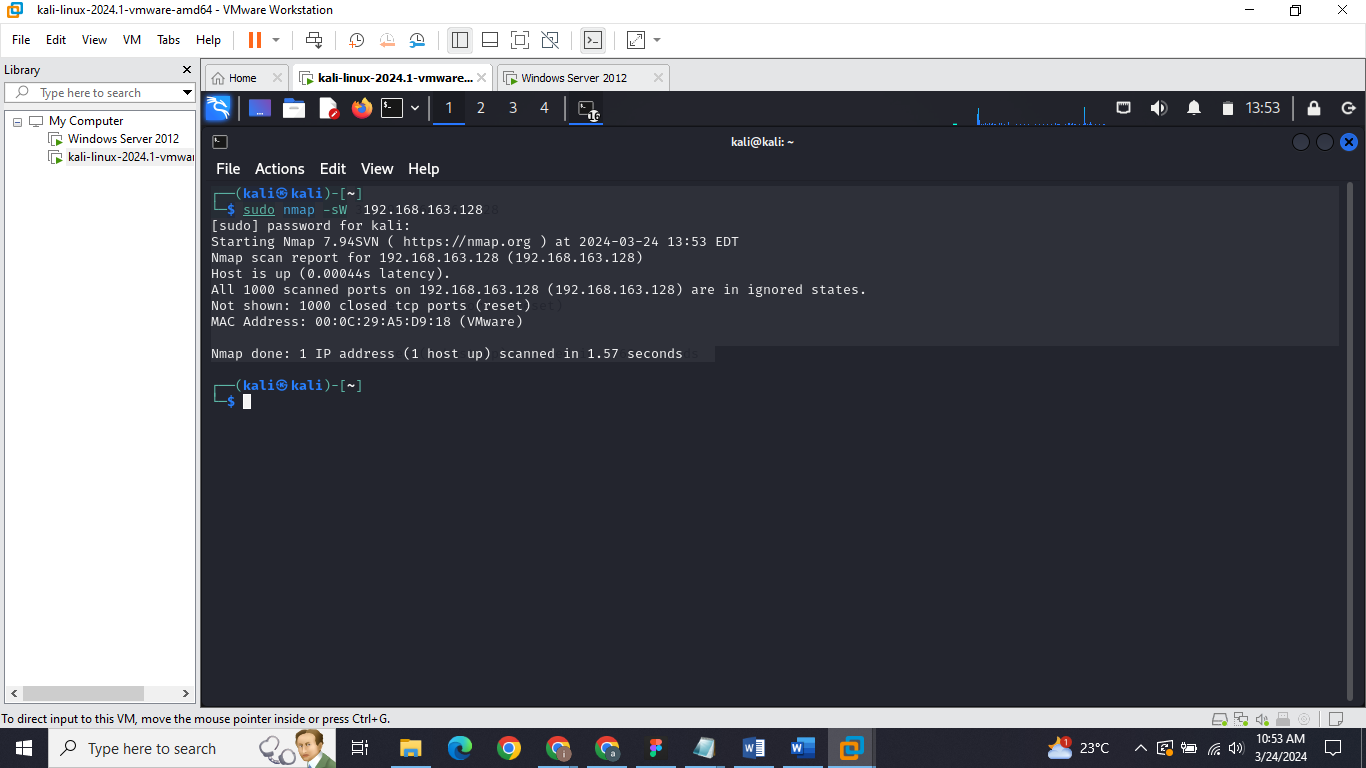
**Command:**

*sudo nmap -sW <target>*

*sudo nmap -sW 192.168.163.128*

The command sudo nmap -sW <target> launches a Window Scan with Nmap. This scan sends TCP packets with the SYN flag set and varying window sizes to the target's ports. By analyzing how ports respond to these packets, the scan identifies open and closed ports on the target system.

**Output:**

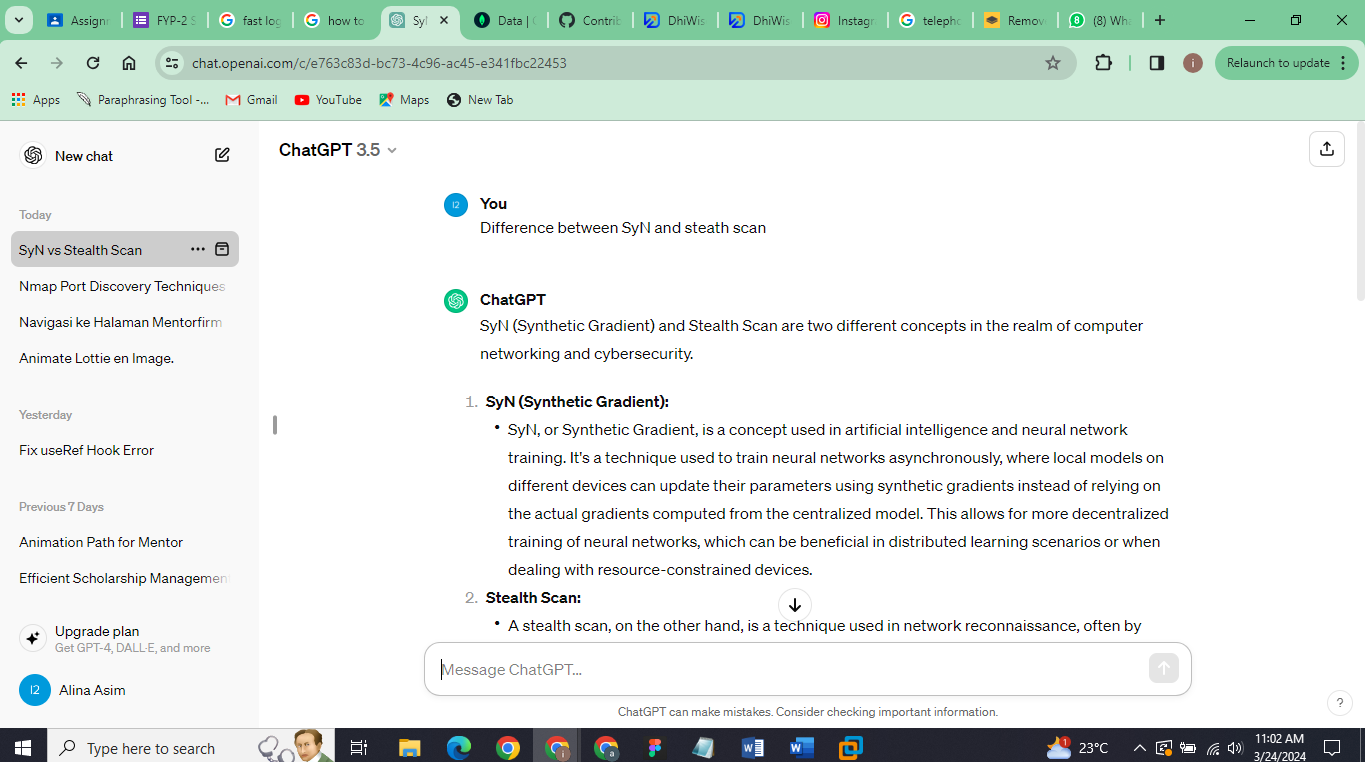


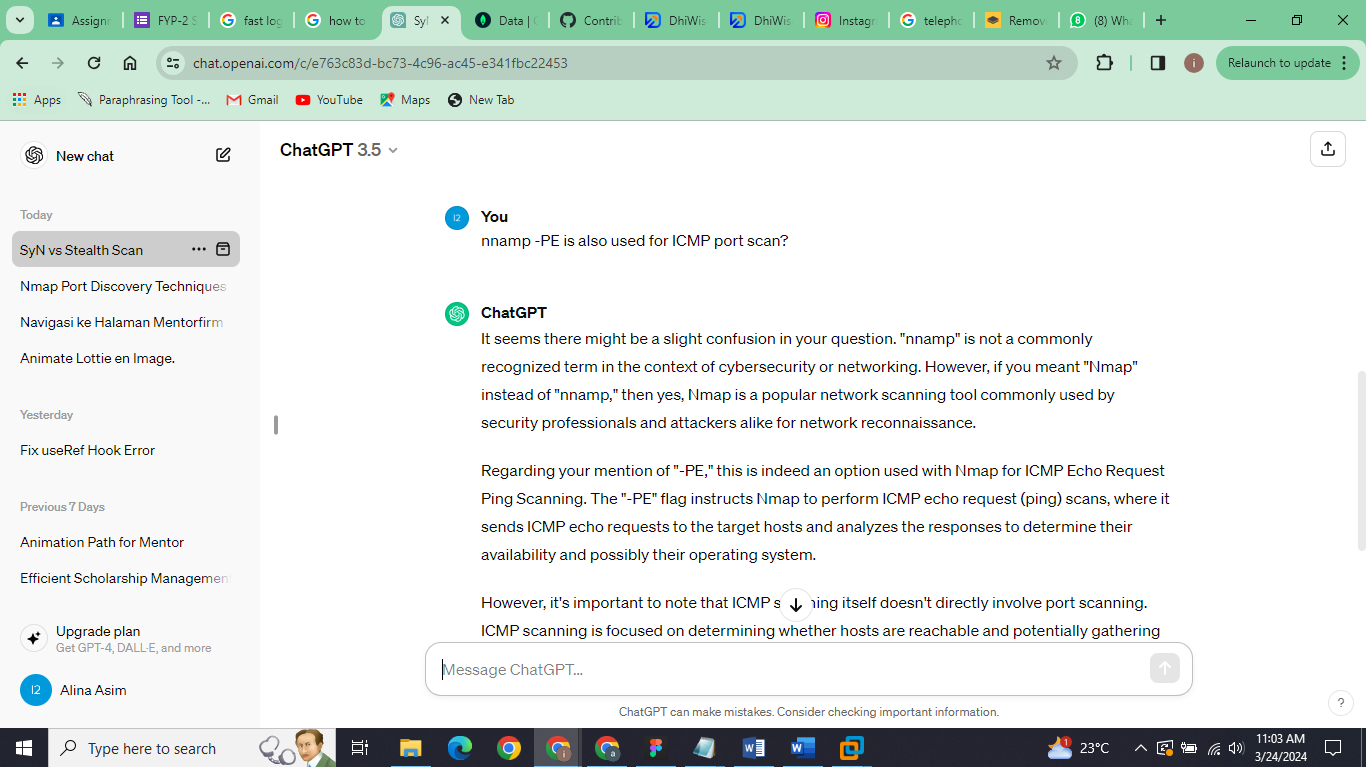
The command sudo nmap -sW 192.168.163.128 executed a Window Scan on the target IP address 192.168.163.128. However, all 1000 scanned ports on the target are reported as "ignored," indicating that the scan did not yield any actionable information about the target's ports.

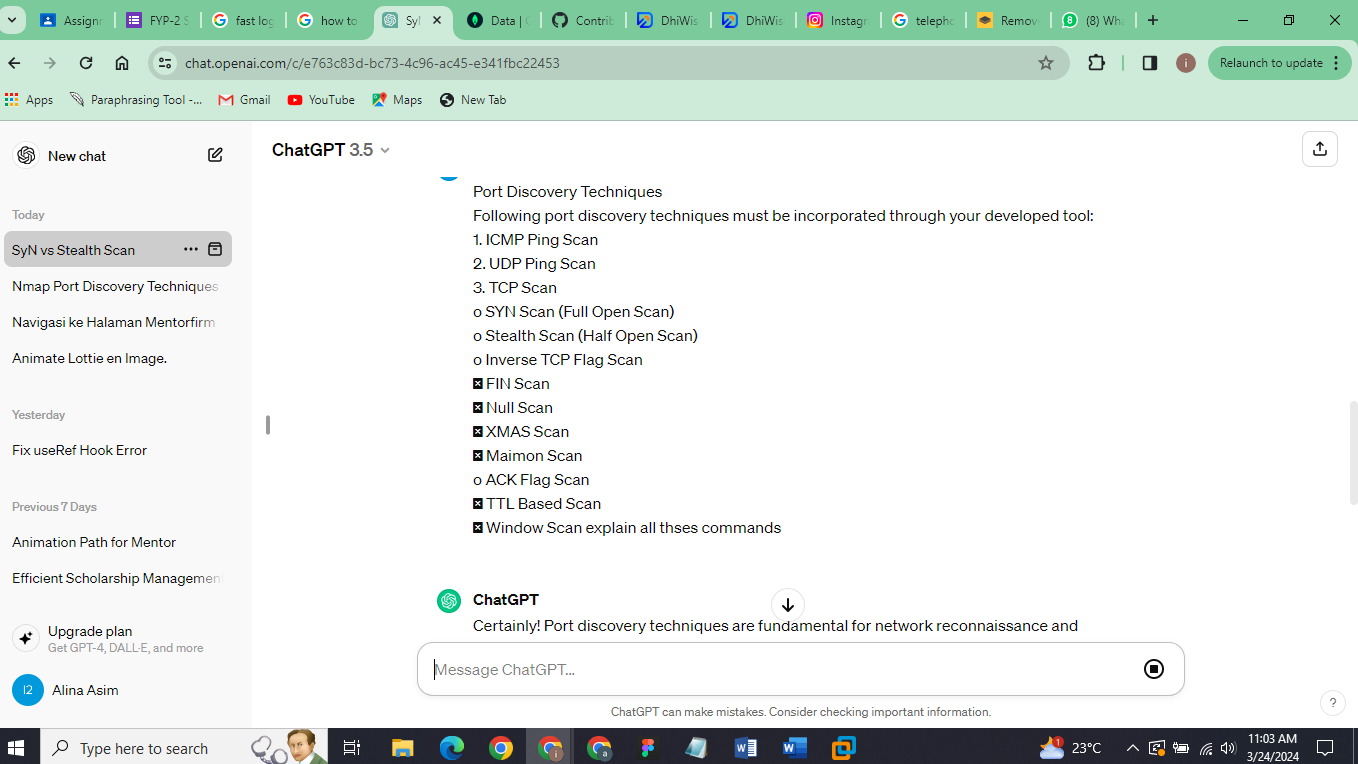
# **Summary**

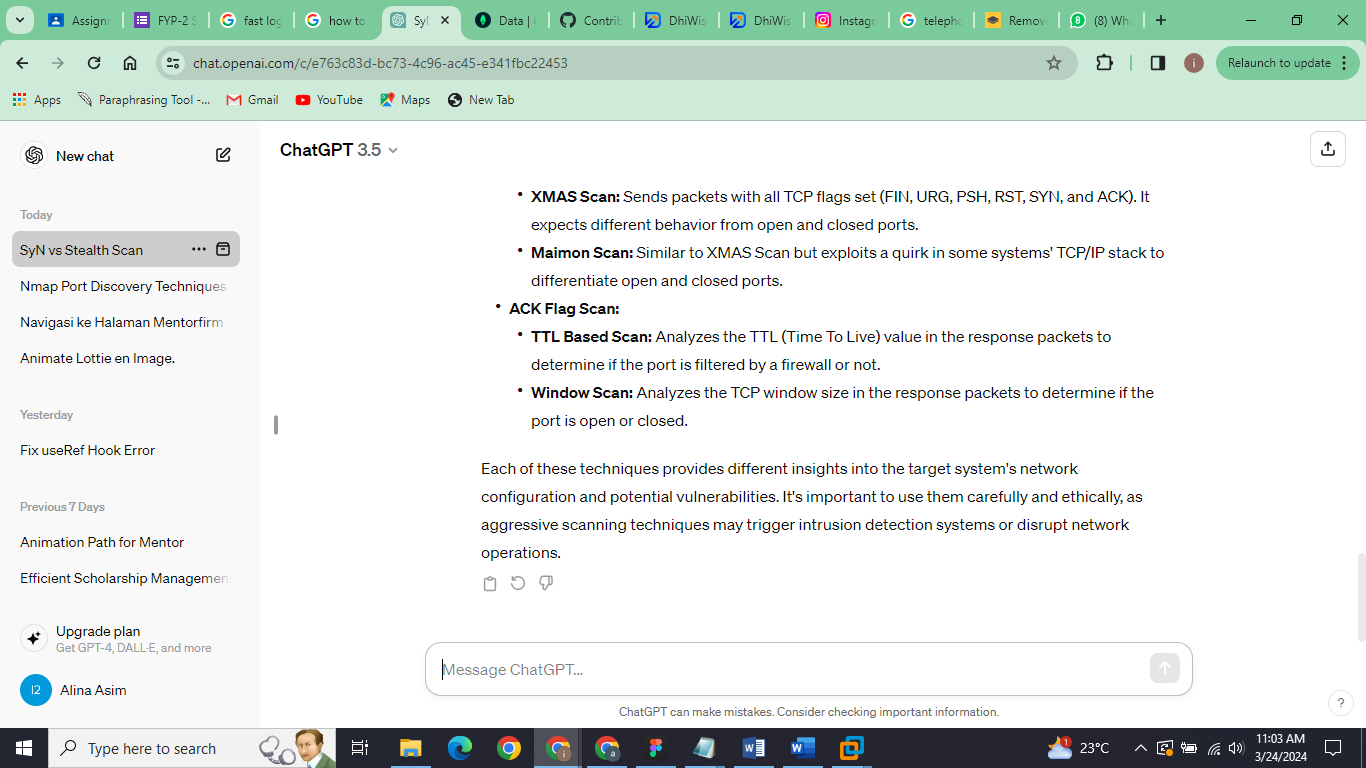
In conclusion, this assignment offered practical exposure to utilizing tools like Nmap for exploring diverse network scanning techniques aimed at port discovery. Kali Linux served as the attacker machine, executing the scans, while a Windows Server 2012 virtual machine acted as the victim machine, subject to the scans. Through this hands-on experience, a comprehensive understanding of network reconnaissance methods and security assessment practices was achieved.

# **Resources**









# **References**

<https://nmap.org/book/synscan.html>

<https://etutorials.org/Networking/network+security+assessment/Chapter+4.+IP+Network+Scanning/4.2+TCP+Port+Scanning/>

<https://chat.openai.com/share/65c574c4-68d2-4fee-a2d2-2440adde8278>