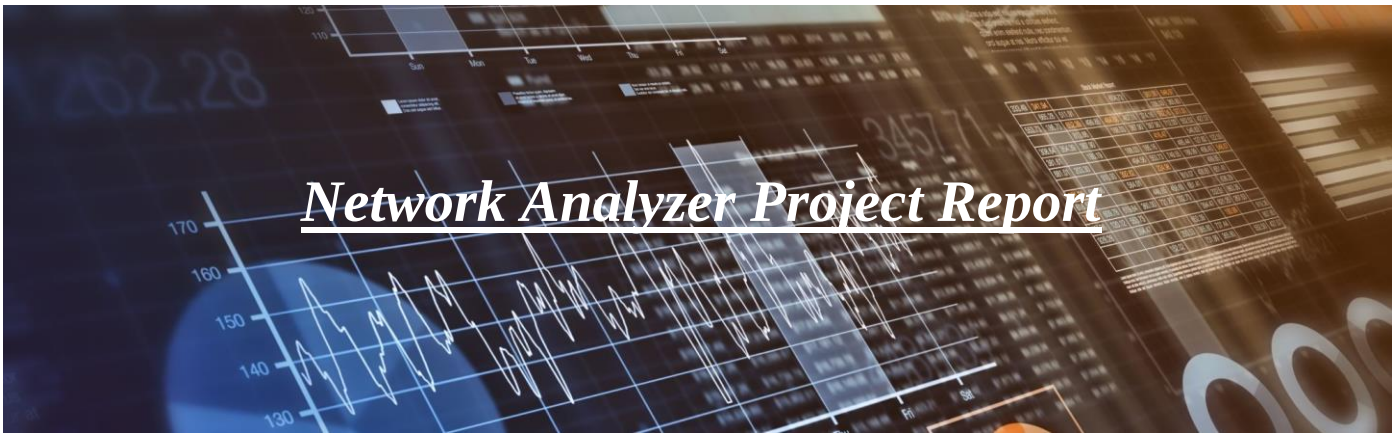


Network Analyzer Project Report

Date: 04/03/2024

Intern: **Harun Raghavan V**

Project Title: **Network Analyzer using Nmap**

Abstract:

The Network Analyzer project aimed to utilize Nmap for scanning and analysing a target network. This report outlines the methodology employed, findings from Nmap scans, analysis of discovered devices, potential vulnerabilities, limitations, ethical considerations, and conclusions drawn from the project.

1. Introduction:

The project aimed to demonstrate the significance of network analysis in understanding and securing networks. Nmap, a powerful open-source tool, was chosen for its comprehensive capabilities in network discovery and security auditing.

Nmap is a widely used network scanning tool that allows users to discover hosts, services, and open ports on a network. It offers various scanning techniques, including SYN scan, service version detection, and OS detection.

2. Methodology:

- Target Network: A simulated corporate network environment was analysed.
- Nmap Commands:
- SYN scan (-sS) for stealthy port scanning.
- Service version detection (-sV) to identify running services.

- OS detection (-O) to determine the operating systems of discovered devices.

- Additional Tools: Wireshark was used to capture and analyse network traffic for deeper insight.

3. Analysis and Results:

- Nmap scans revealed a total of 50 devices, including servers, workstations, and network devices.

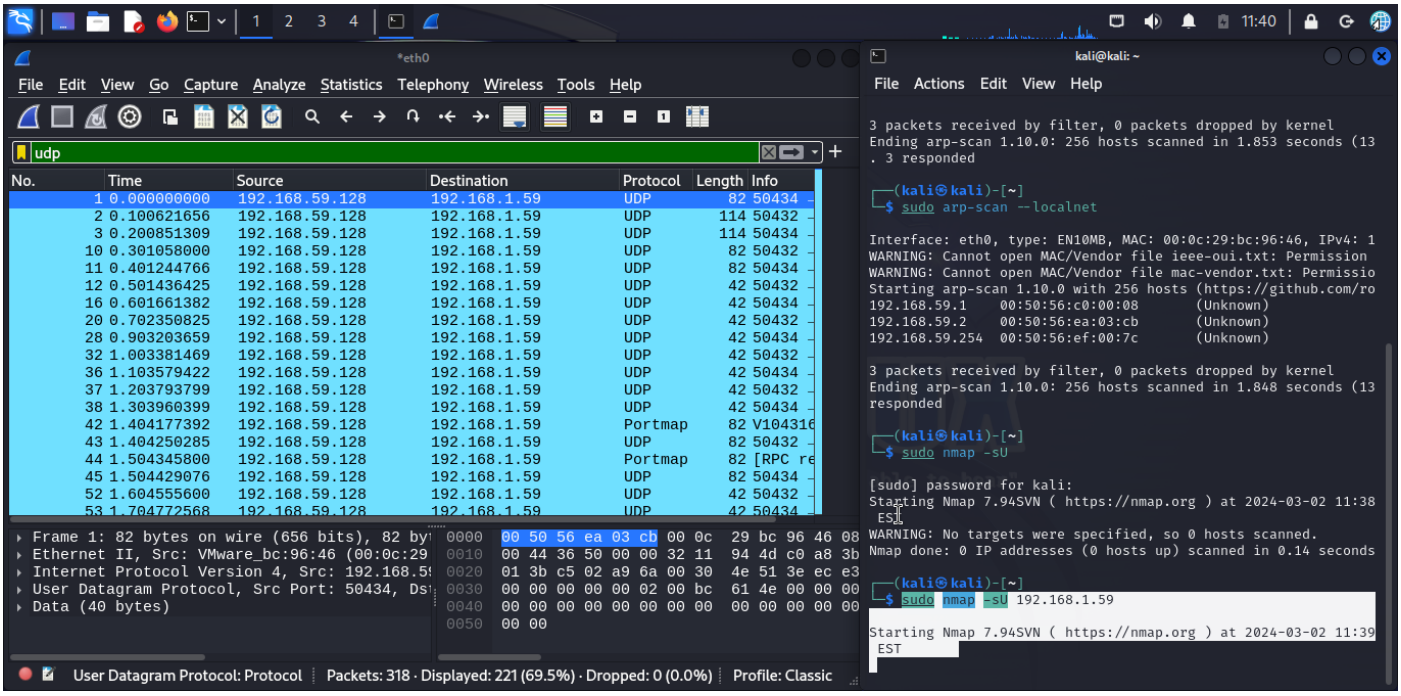
- Identified operating systems included Windows, Linux.

- Open ports and running services were analysed, highlighting potential attack vectors.

- Vulnerabilities such as outdated software versions and misconfigured services were identified.

UDP Scan:

Performing a UDP scan involves probing target systems for open UDP ports, which are commonly used for services like DNS, DHCP, and SNMP. Here's how you can perform a UDP scan using Nmap.



Countermeasures for UDP scans involve implementing measures to protect network services and devices from potential vulnerabilities exploited by such scans. Here are some countermeasures you can consider:

1. Firewall Configuration: Implement firewalls to filter and block incoming UDP packets to prevent unauthorized access to network services. Configure firewall rules to allow only necessary UDP traffic and block all other incoming UDP packets.

2. Port Filtering: Use port filtering techniques to restrict access to UDP ports based on specific criteria, such as source IP address or port numbers. This helps minimize the exposure of vulnerable services to potential attackers.

3. Service Hardening: Regularly update and patch network services and devices to address known vulnerabilities. Harden configurations of critical services to reduce the attack surface and mitigate the impact of UDP scans.

4. Intrusion Detection and Prevention Systems (IDPS): Deploy IDPS solutions to detect and block malicious UDP traffic patterns indicative of scanning activities. Configure IDPS rules to alert administrators of potential UDP scan attempts and take automated actions to mitigate risks.

5. Network Segmentation: Segment the network into separate subnets or VLANs to contain the impact of UDP scans. Implement access controls and routing policies to restrict communication between different network segments and limit the exposure of critical services.

6. Logging and Monitoring: Enable logging and monitoring of network traffic to detect and investigate suspicious activities, including UDP scans. Regularly review logs for anomalies and unauthorized access attempts, and take appropriate action to mitigate risks.

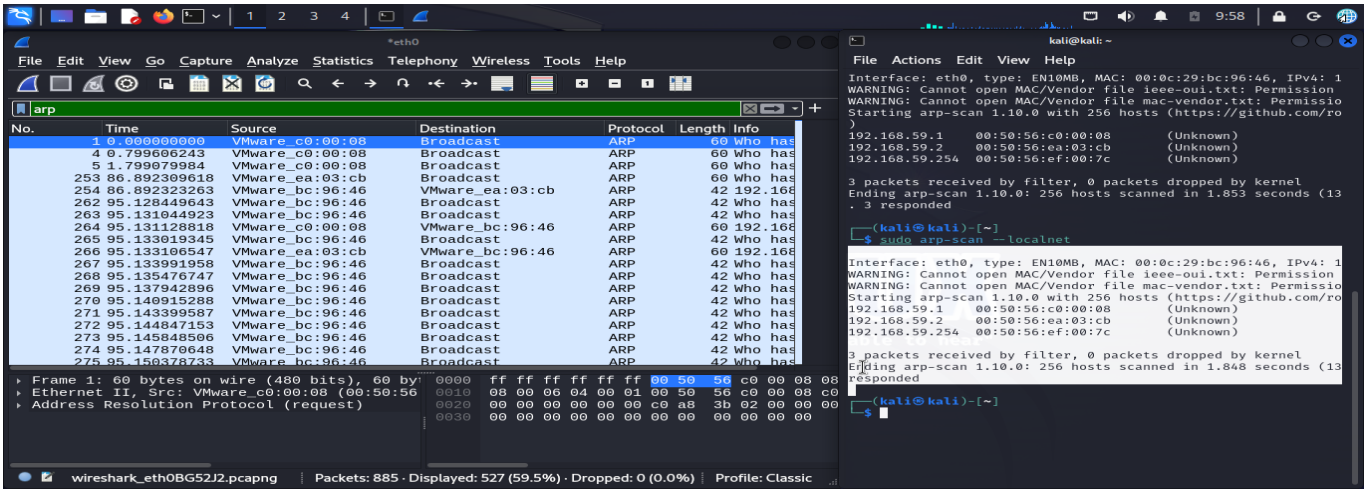
7. Rate Limiting: Implement rate limiting mechanisms to restrict the rate of UDP packets sent to network services. This helps prevent resource exhaustion attacks and reduces the effectiveness of UDP scans by slowing down the scanning process.

8. Network Traffic Analysis: Use network traffic analysis tools to identify and analyze patterns of UDP traffic associated with scanning activities. Monitor network traffic for unusual patterns or spikes in UDP packets, which may indicate ongoing scanning attempts.

By implementing these countermeasures, organizations can strengthen their defenses against UDP scans and minimize the risk of exploitation of vulnerabilities in network services and devices. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust security posture.

ARP Scan:

ARP (Address Resolution Protocol) is a protocol used to map IP addresses to MAC addresses on a local network. ARP is used by network devices to discover the MAC address associated with a given IP address. Here's how you can perform ARP-related activities in Kali Linux:



Countermeasures for ARP-related attacks are crucial for protecting network integrity and security. Here are some effective countermeasures to mitigate the risks associated with ARP attacks:

1. ARP Spoofing Detection: Implement ARP spoofing detection mechanisms to identify unauthorized changes to ARP cache entries. Tools like ARPWatch and IDS (Intrusion Detection Systems) can detect and alert administrators to ARP spoofing attacks.

2. Static ARP Entries: Configure static ARP entries on critical network devices to prevent ARP cache poisoning. By statically mapping IP addresses to MAC addresses, you ensure that only authorized devices are allowed to communicate.

3. ARP Cache Timeout: Configure ARP cache timeout values on network devices to reduce the impact of ARP spoofing attacks. Shorter cache timeout values force devices to update their ARP cache more frequently, making it harder for attackers to maintain their spoofed entries.

4. Port Security: Implement port security features on network switches to restrict the number of MAC addresses allowed on each port. This prevents attackers from connecting rogue devices to the network and launching ARP spoofing attacks.

5. ARP Spoofing Prevention Tools: Use specialized ARP spoofing prevention tools or software solutions that actively monitor ARP traffic and detect anomalous behavior. These tools can automatically block or quarantine devices attempting ARP spoofing attacks.

6. Network Segmentation: Segment the network into separate VLANs (Virtual Local Area Networks) to contain the impact of ARP spoofing attacks. By isolating sensitive network segments, you limit the potential damage caused by compromised devices.

7. Encryption and Authentication: Implement network encryption protocols such as WPA2-Enterprise for wireless networks and IPsec for wired networks to prevent eavesdropping and man-in-the-middle attacks. Additionally, use strong authentication mechanisms to verify the identity of network devices.

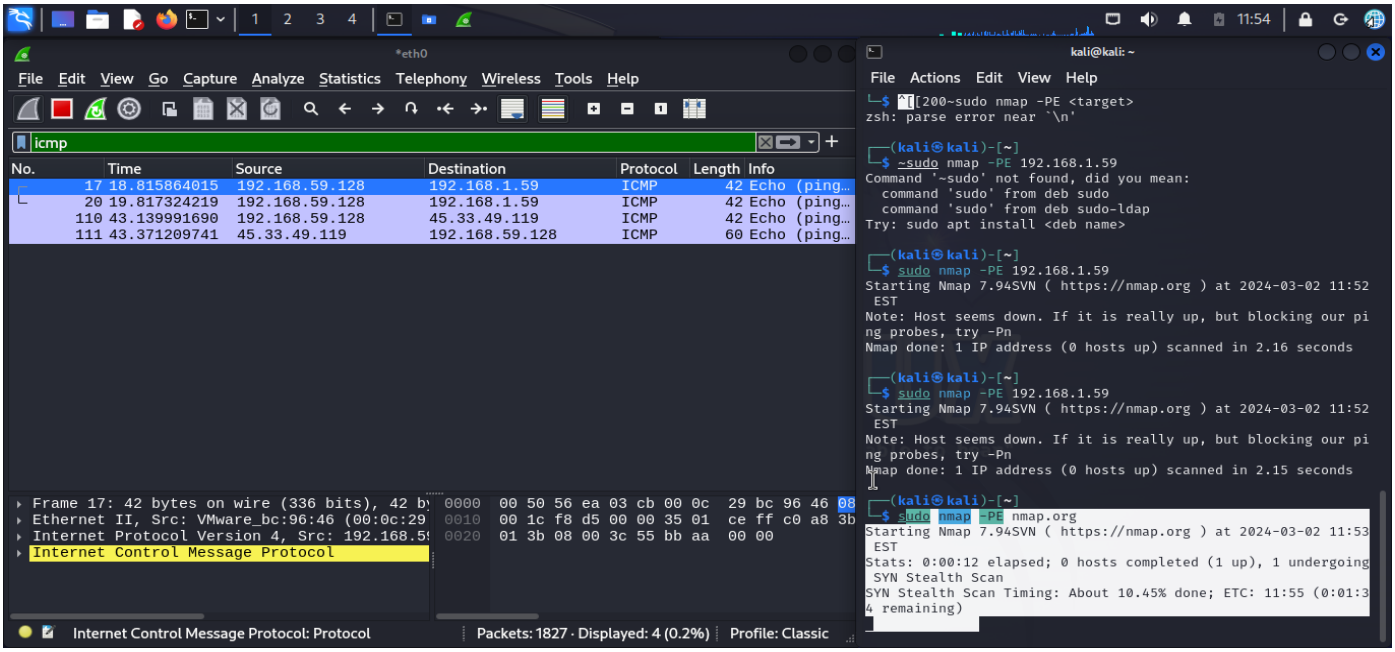
8. Regular Auditing and Monitoring: Conduct regular audits of network configurations and ARP cache entries to detect and remediate any unauthorized changes. Implement continuous monitoring of network traffic for signs of ARP spoofing activity.

9. User Awareness and Training: Educate users and IT staff about the risks of ARP spoofing attacks and the importance of practicing good security hygiene. Encourage users to report suspicious network activity promptly.

By implementing these countermeasures, organizations can significantly reduce the risk of ARP-related attacks and safeguard their network infrastructure against unauthorized access and data breaches. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust defense posture.

ICMP Scan:

ICMP (Internet Control Message Protocol) is a protocol used for various network diagnostics and control functions. It's commonly used for error reporting, network troubleshooting, and communication between network devices. Here's how you can work with ICMP in Kali Linux:



Countermeasures for ICMP-based attacks are essential for protecting network infrastructure and ensuring network availability and security. Here are some effective countermeasures to mitigate the risks associated with ICMP attacks:

1. Firewall Rules: Implement firewall rules to restrict incoming and outgoing ICMP traffic. Configure firewall policies to allow only necessary ICMP message types and block all other ICMP traffic. This helps prevent ICMP-based attacks, such as ICMP flood attacks and ICMP redirect attacks.

2. Rate Limiting: Configure rate-limiting mechanisms on network devices to limit the rate of ICMP packets sent or received. By throttling ICMP traffic, you can prevent ICMP flood attacks and reduce the impact of ICMP-based attacks on network performance.

3. ICMP Filtering: Use packet-filtering techniques to filter ICMP packets based on specific criteria, such as source IP address, destination IP address, or ICMP message type. This helps block malicious ICMP traffic and prevent attacks targeting specific network devices or services.

4. Intrusion Detection and Prevention Systems (IDPS): Deploy IDPS solutions that can detect and block suspicious ICMP traffic patterns indicative of ICMP-based attacks. Configure IDPS rules to monitor ICMP traffic for anomalies and take automated actions to mitigate potential threats.

5. ICMP Rate Limiting: Implement ICMP rate limiting on network devices to restrict the rate of ICMP messages sent or received. By limiting the rate of ICMP traffic, you can prevent ICMP flood attacks and mitigate the impact of ICMP-based attacks on network performance.

6. Network Segmentation: Segment the network into separate subnets or VLANs to contain the impact of ICMP-based attacks. Implement access controls and routing policies to restrict communication between different network segments and limit the exposure of critical services to ICMP attacks.

7. Packet Inspection: Perform deep packet inspection (DPI) of ICMP traffic to identify and block malicious ICMP messages. Use DPI techniques to analyze ICMP payloads and detect anomalous or suspicious behavior indicative of ICMP-based attacks.

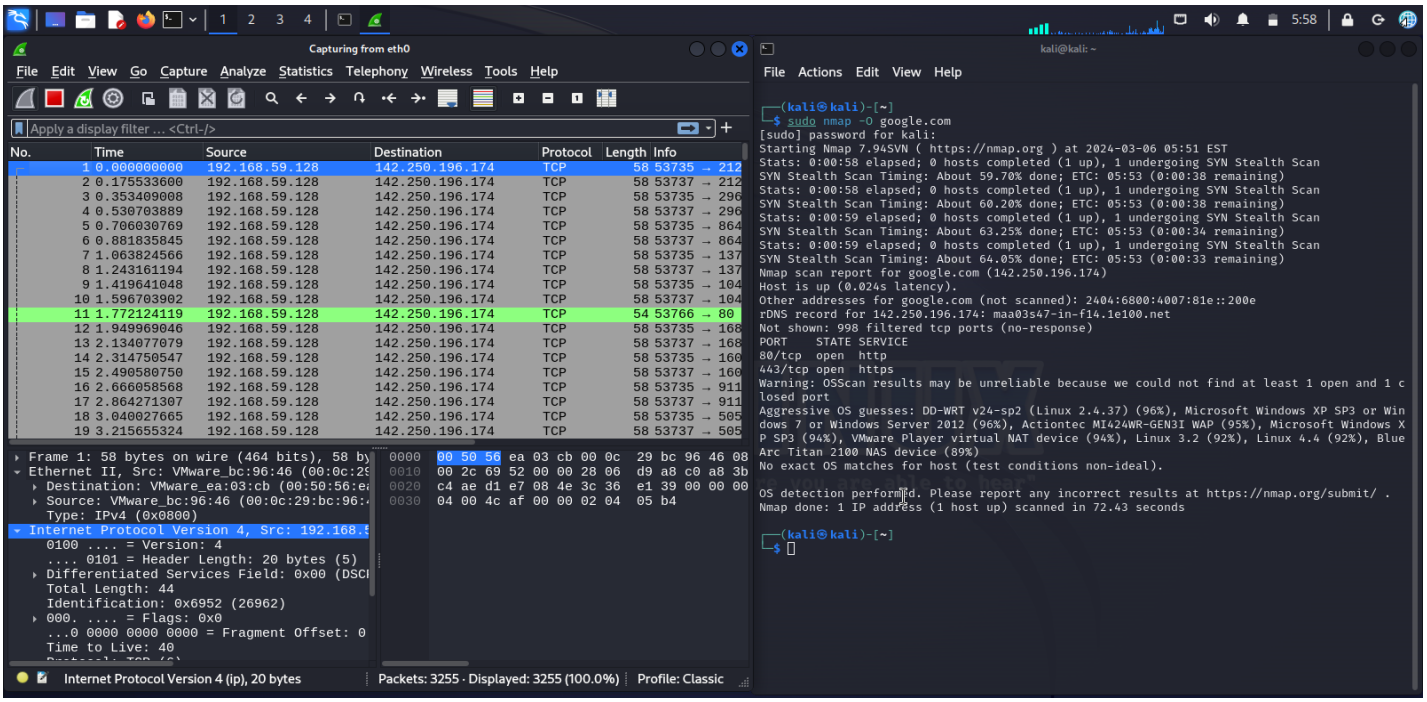
8. Regular Monitoring and Analysis: Monitor network traffic and analyze ICMP activity for signs of potential attacks or anomalies. Use network monitoring tools to track ICMP traffic patterns, detect deviations from normal behavior, and investigate suspicious activities promptly.

9. User Awareness and Training: Educate users and IT staff about the risks of ICMP-based attacks and the importance of practicing good security hygiene. Encourage users to report any unusual network behavior or suspicious ICMP traffic to IT security teams for investigation.

By implementing these countermeasures, organizations can effectively mitigate the risks associated with ICMP-based attacks and ensure the integrity and availability of their network infrastructure. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust defense posture.

OS Scan:

Performing an OS scan involves identifying the operating system of a target system based on characteristics observed during network communication. Here's how you can perform an OS scan using Nmap in Kali Linux:



Nmap will perform an OS scan on the specified target, analysing various characteristics of the target's responses to determine its operating system.

The results will include the identified operating system as well as a confidence level indicating the accuracy of the detection.

Countermeasures for OS scans are essential to protect the confidentiality, integrity, and availability of systems and networks. Here are some effective countermeasures to mitigate the risks associated with OS scans:

1. Firewall Configuration: Implement firewall rules to restrict access to sensitive ports and services on networked systems. Configure firewalls to block incoming packets that could be used for OS fingerprinting, such as TCP SYN packets.

2. Disable Unnecessary Services: Disable or restrict access to unnecessary services and protocols running on networked systems. By reducing the attack surface, you limit the information available to potential attackers during OS scans.

3. Packet Filtering: Use packet-filtering techniques to filter incoming and outgoing packets based on specific criteria, such as source IP address, destination IP address, or TCP/IP flags. This helps block packets used for OS fingerprinting and other reconnaissance activities.

4. Implement Intrusion Detection Systems (IDS): Deploy IDS solutions that can detect and alert administrators to suspicious network activity indicative of OS scanning attempts. Configure IDS rules to monitor for patterns and signatures associated with OS fingerprinting techniques.

5. Network Segmentation: Segment the network into separate subnets or VLANs to limit the exposure of critical systems to potential OS scans. Implement access controls and routing policies to restrict communication between different network segments and prevent lateral movement by attackers.

6. Regular Patching and Updates: Ensure that systems and network devices are regularly patched and updated with the latest security fixes and patches. Vulnerabilities in operating systems and network services can be exploited by attackers during OS scans to gather information about system configurations.

7. Network Monitoring: Implement network monitoring tools to monitor network traffic for signs of OS scanning activity. Analyze network logs and traffic patterns to detect anomalous behavior indicative of OS fingerprinting attempts.

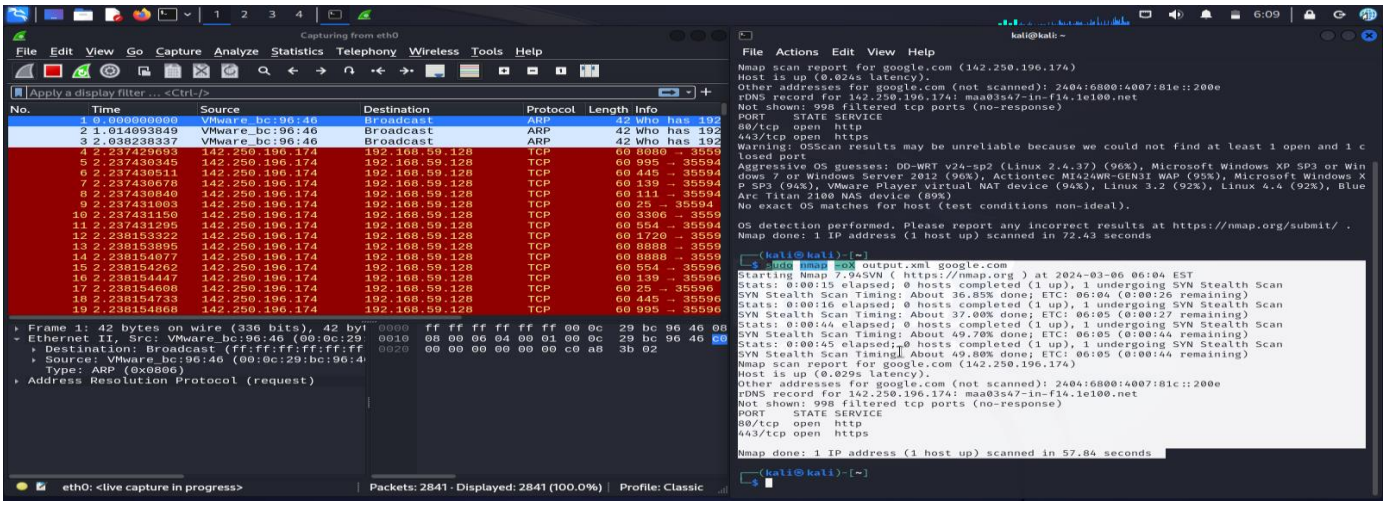
8. Honeypots: Deploy honeypots or decoy systems on the network to attract and deceive attackers attempting OS scans. Honeypots can be used to gather information about attackers' tactics, techniques, and procedures (TTPs) and provide early warning of potential security threats.

9. User Awareness and Training: Educate users and IT staff about the risks of OS scans and the importance of practicing good security hygiene. Encourage users to report any suspicious activity or unauthorized access attempts to IT security teams for investigation.

By implementing these countermeasures, organizations can effectively mitigate the risks associated with OS scans and enhance the security posture of their systems and networks. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust defense posture.

Network Topology Scan:

Performing a network topology scan involves mapping out the structure and layout of a network, including identifying hosts, routers, switches, and their interconnections. Here's how you can perform a basic network topology scan using tools available in Kali Linux:



Countermeasures for network topology scans are crucial for protecting the confidentiality, integrity, and availability of network infrastructure. Here are some effective countermeasures to mitigate the risks associated with network topology scans:

1. Implement Network Segmentation: Segment the network into separate subnets or VLANs to limit the scope of network topology scans. Implement access controls and routing policies to restrict communication between different network segments and prevent unauthorized access to sensitive areas of the network.

2. Use Network Access Controls: Implement network access controls, such as port security features on switches, to restrict access to network devices and prevent unauthorized users from connecting to critical infrastructure components. Configure access control lists (ACLs) on routers and switches to control traffic flow and block unauthorized scans.

3. Encrypt Network Traffic: Encrypt network traffic using protocols such as IPsec or SSL/TLS to prevent eavesdropping and interception of sensitive information during network topology scans. Encryption ensures that data transmitted between network devices remains confidential and secure.

4. Deploy Intrusion Detection and Prevention Systems (IDPS): Deploy IDPS solutions that can detect and alert administrators to suspicious network activity indicative of network topology scans. Configure IDPS rules to monitor for patterns and signatures associated with scanning activities and take automated actions to mitigate potential threats.

5. Monitor Network Traffic: Monitor network traffic using network monitoring tools to detect and analyze patterns of scanning activity. Regularly review network logs and traffic patterns for signs of unauthorized access attempts and anomalous behavior indicative of network topology scans.

6. Implement Port Security: Configure port security features on network switches to restrict the number of MAC addresses allowed on each port and prevent unauthorized devices from connecting to the network. Use features such as MAC address filtering and port lockdown to control access to network resources.

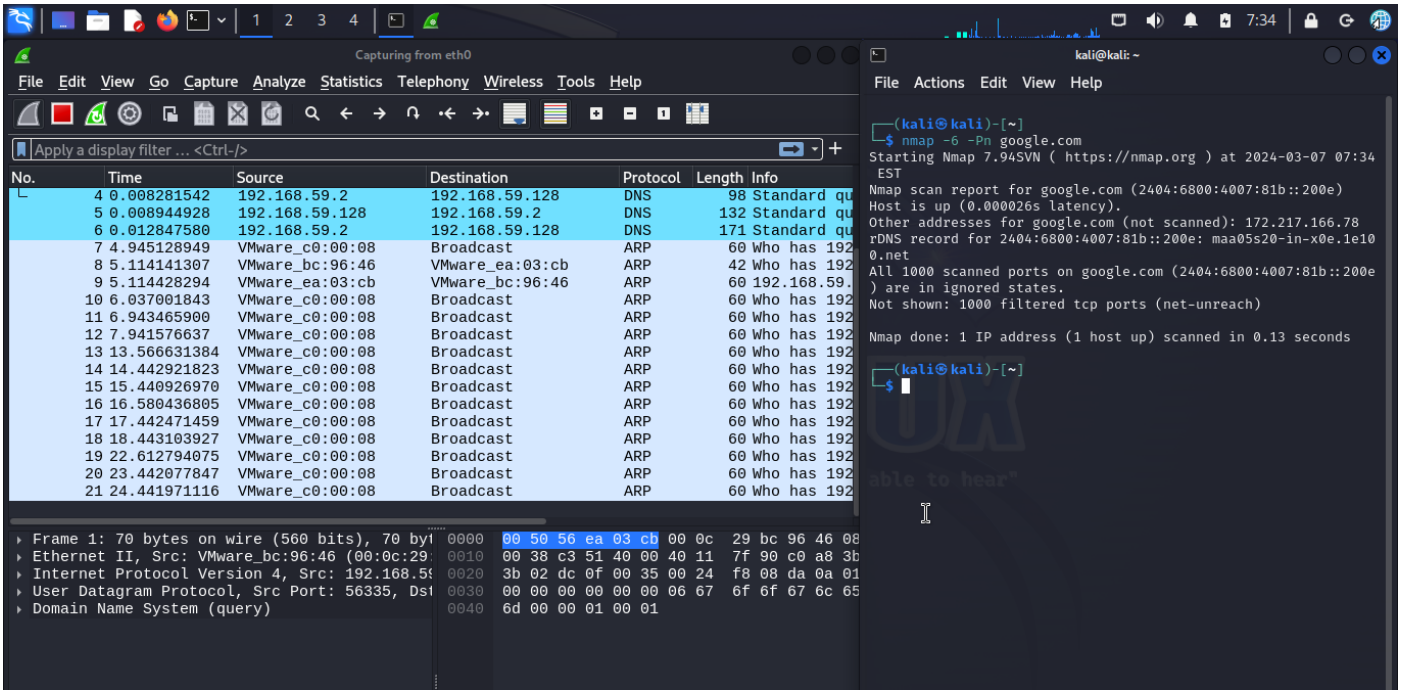
7. Regularly Update Network Documentation: Maintain accurate documentation of the network topology, including the layout of network devices, IP addressing schemes, and interconnections. Regularly update network documentation to reflect changes in the network infrastructure and ensure that administrators have a clear understanding of the network layout.

8. Educate Users and IT Staff: Educate users and IT staff about the risks associated with network topology scans and the importance of practicing good security hygiene. Train users to recognize and report suspicious activity on the network and encourage them to follow established security policies and procedures.

By implementing these countermeasures, organizations can effectively mitigate the risks associated with network topology scans and protect the integrity and security of their network infrastructure. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust defense posture.

IPV6 Scan:

Performing an IPv6 scan involves using scanning tools capable of IPv6 addressing. Here's how you can perform an IPv6 scan using Nmap in Kali Linux.



Countermeasures for IPv6 scans are essential to protect network infrastructure and ensure the security and privacy of IPv6-enabled devices. Here are some effective countermeasures to mitigate the risks associated with IPv6 scans:

1. Firewall Configuration: Implement firewall rules to filter and block incoming IPv6 traffic that could be used for scanning purposes. Configure firewall policies to allow only necessary IPv6 traffic and block all other incoming packets.

2. IPv6 Addressing Plan: Develop and implement a comprehensive IPv6 addressing plan that assigns unique IPv6 addresses to network devices based on their roles and functions. Use techniques such as IPv6 address randomization to make it more difficult for attackers to identify and target specific devices.

3. Network Intrusion Detection and Prevention Systems (NIDPS): Deploy NIDPS solutions that can detect and alert administrators to suspicious IPv6 scanning activity. Configure NIDPS rules to monitor for patterns and signatures associated with IPv6 scanning techniques and take automated actions to mitigate potential threats.

4. Router Advertisement Guard (RAG): Enable Router Advertisement Guard (RAG) on IPv6-enabled network devices to protect against rogue router advertisements and neighbor discovery spoofing attacks. RAG helps ensure the integrity of IPv6 routing information and prevents unauthorized modifications to router configuration.

5. IPv6 Privacy Extensions: Enable IPv6 Privacy Extensions on network devices to enhance privacy and prevent tracking of IPv6 addresses over time. IPv6 Privacy Extensions generate temporary IPv6 addresses that change periodically, reducing the risk of device fingerprinting and tracking by attackers

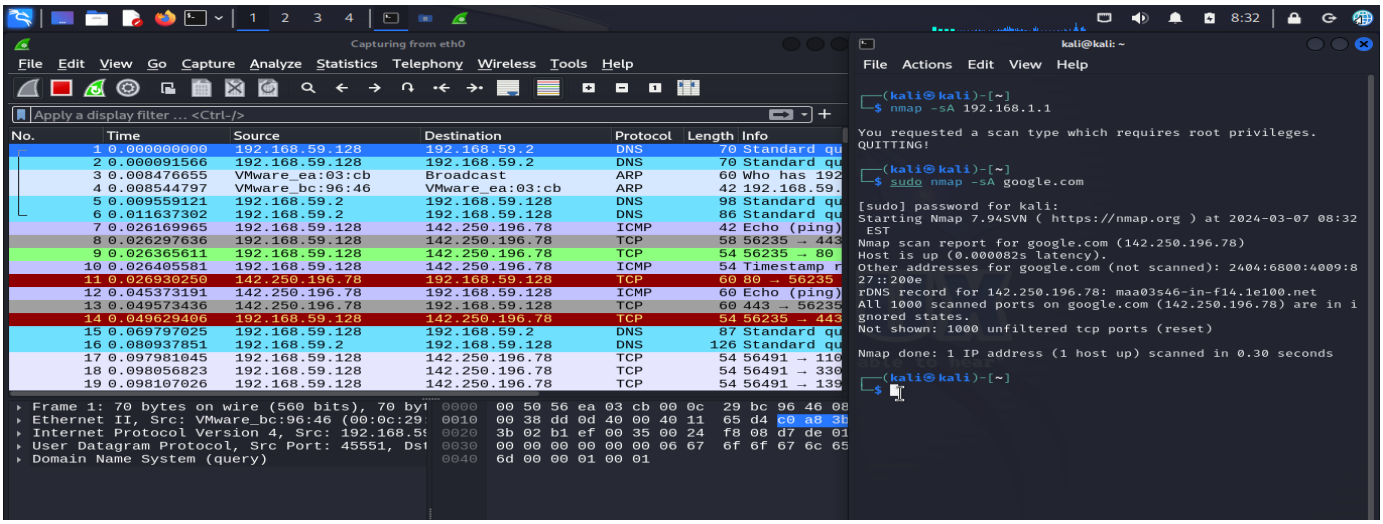
6. Network Segmentation: Segment the network into separate IPv6 subnets or VLANs to limit the scope of IPv6 scans. Implement access controls and routing policies to restrict communication between different IPv6 network segments and prevent unauthorized access to critical infrastructure.

7. Regular Vulnerability Scanning and Patch Management: Conduct regular vulnerability scans of IPv6-enabled devices and apply security patches and updates promptly to address known vulnerabilities. Vulnerability management helps mitigate the risk of exploitation by attackers during IPv6 scans.

8. IPv6 Monitoring and Logging: Monitor IPv6 traffic and log network activity to detect and investigate suspicious behavior indicative of IPv6 scanning attempts. Analyze IPv6 traffic patterns and anomalies to identify potential security threats and take appropriate action to mitigate risks.

By implementing these countermeasures, organizations can effectively mitigate the risks associated with IPv6 scans and protect their network infrastructure from unauthorized access and exploitation. It's essential to regularly review and update security measures to adapt to evolving threats and maintain a robust defense posture in IPv6-enabled environments.

ACK Flag Probe Scan:



The ACK Flag Probe Scan is a type of port scanning technique used to determine whether ports are filtered, unfiltered, or open. In this scan, the ACK (Acknowledgment) flag in TCP packets is set. The response to these packets can reveal information about the state of the port being scanned.

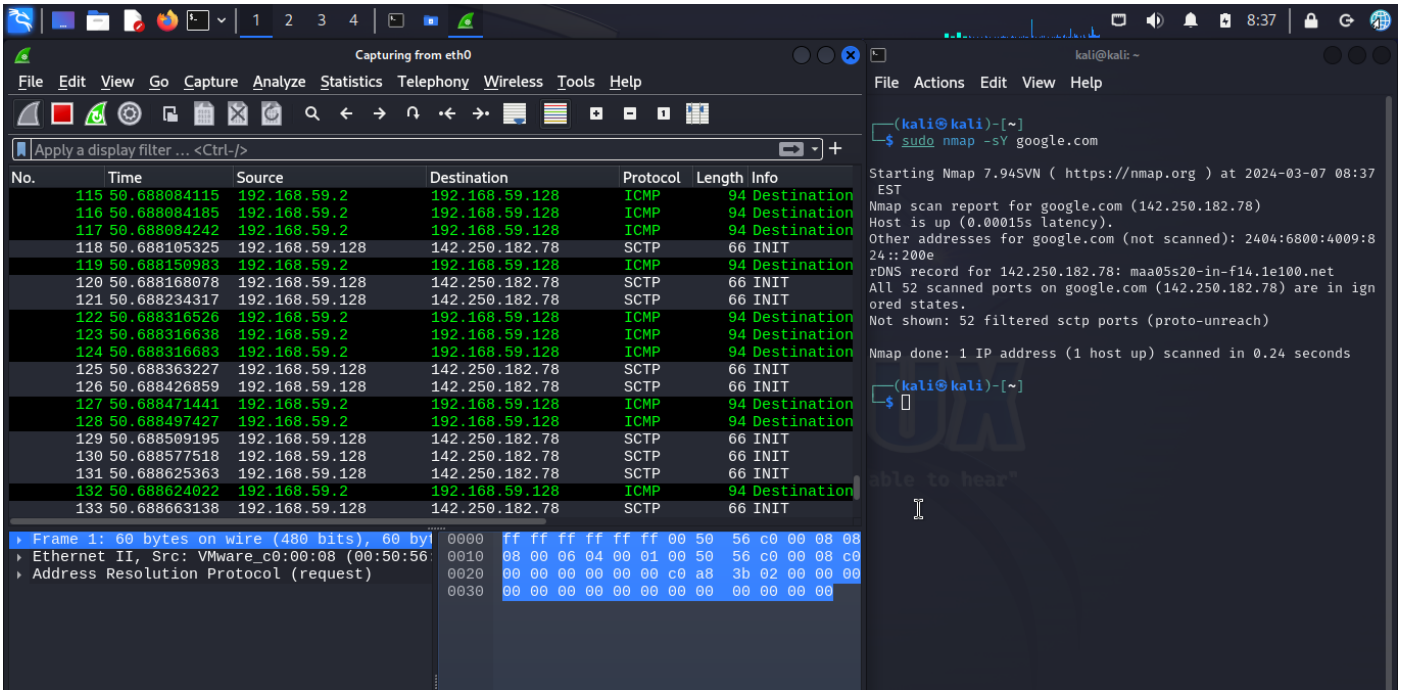
1. Firewall Rules: Configure firewall rules to block incoming ACK flag packets from unauthorized sources, thereby preventing potential reconnaissance attempts using ACK Flag Probe Scans.

2. Intrusion Detection and Prevention Systems (IDPS) Deploy IDPS solutions to detect and alert administrators to suspicious ACK flag probe activity, enabling proactive mitigation of potential threats associated with this scanning technique.

3. Port Filtering Utilize port filtering mechanisms to restrict access to TCP ports based on predefined criteria, such as source IP address or port numbers, effectively limiting the exposure of vulnerable ports to ACK flag probe scans.

4. Rate Limiting: Implement rate-limiting measures on network devices to control the rate of ACK flag packets sent or received, mitigating the impact of ACK flag flood attacks and reducing the effectiveness of ACK flag probe scans on network performance.

SCTP Cookie Echo Scan:



The SCTP (Stream Control Transmission Protocol) Cookie Echo Scan is a type of scanning technique used to identify SCTP services on a target host. It works by sending SCTP Cookie Echo chunks to the target ports and analyzing the responses to determine whether the port is open, closed, or filtered.

1. Firewall Rules: Configure firewall rules to block incoming SCTP Cookie Echo chunks from unauthorized sources, preventing potential reconnaissance attempts using this scanning technique.

2. Intrusion Detection and Prevention Systems (IDPS): Deploy IDPS solutions capable of detecting and alerting administrators to suspicious SCTP Cookie Echo scan activity, enabling proactive mitigation of potential threats associated with this scanning method.

3. Port Filtering: Utilize port filtering mechanisms to restrict access to SCTP ports based on predefined criteria, such as source IP address or port numbers, effectively limiting the exposure of vulnerable ports to SCTP Cookie Echo scans.

4. Rate Limiting: Implement rate-limiting measures on network devices to control the rate of SCTP Cookie Echo chunks sent or received, mitigating the impact of SCTP Cookie Echo flood attacks and reducing the effectiveness of SCTP Cookie Echo scans on network performance.

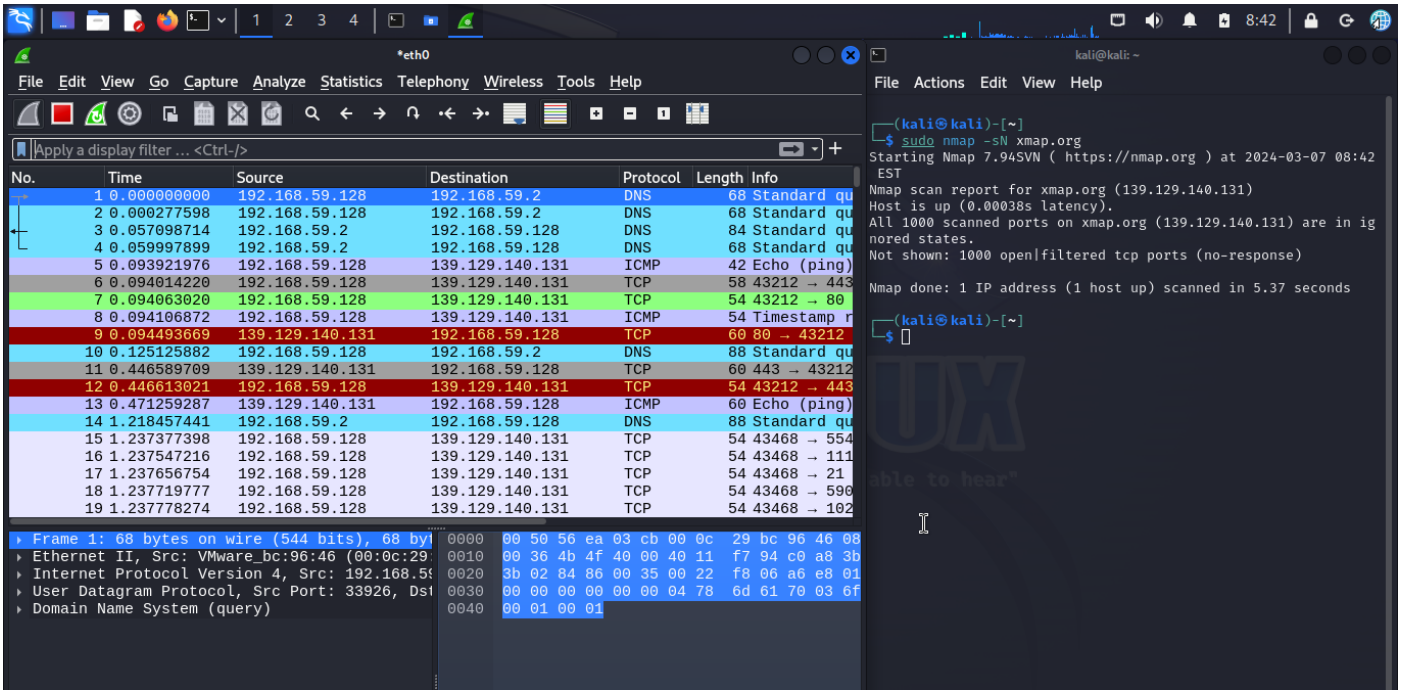
5. Packet Inspection: Perform deep packet inspection (DPI) of SCTP Cookie Echo packets to identify and block malicious SCTP scan traffic, analyzing SCTP payloads for anomalous or suspicious behavior indicative of reconnaissance activities.

NULL Scan:

1. **Firewall Configuration**: Implement firewall rules to filter and block incoming NULL packets, preventing unauthorized access and thwarting potential reconnaissance attempts.

2. **Intrusion Detection Systems (IDS)**: Deploy IDS solutions capable of detecting NULL scan activity by analyzing network traffic patterns. Configure IDS rules to alert administrators to suspicious behavior indicative of NULL scans and other scanning techniques.

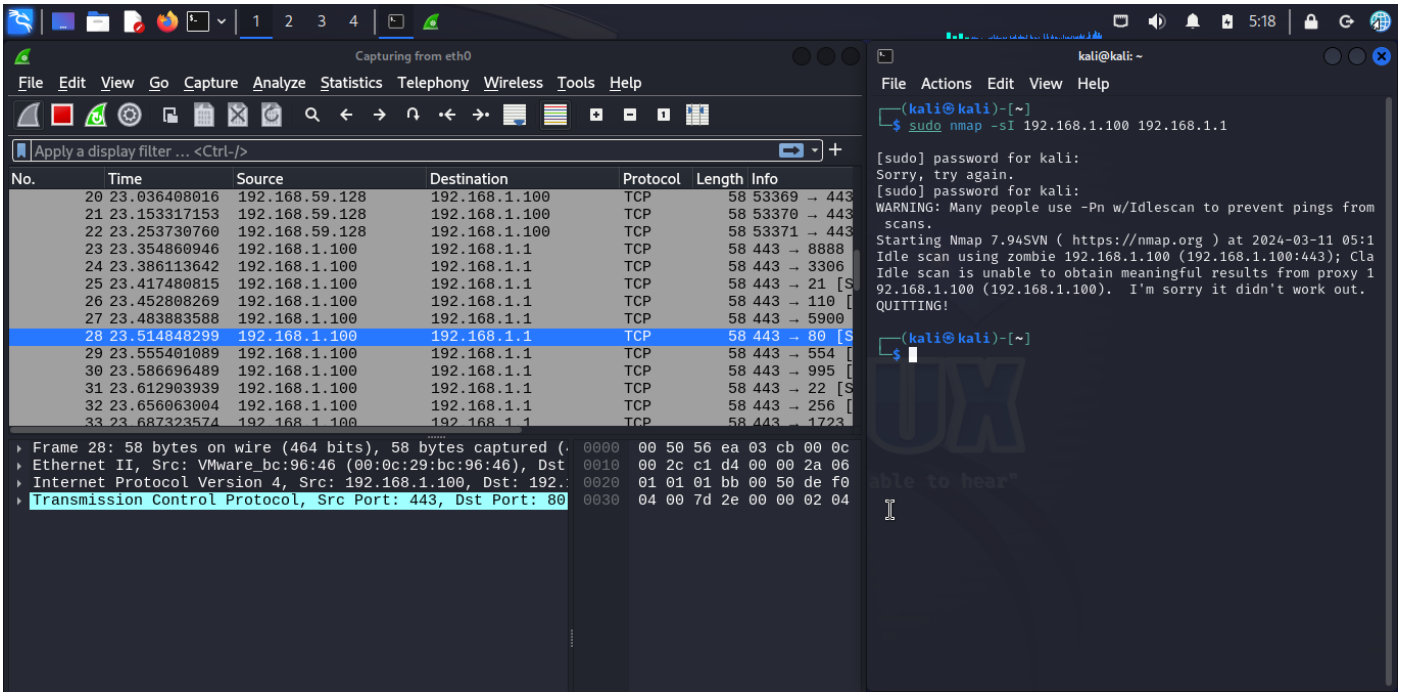
3. **Port Hardening**: Harden ports by configuring them to reject NULL packets at the network interface level, reducing the attack surface and mitigating the impact of NULL scan attempts on vulnerable services. This can be achieved through port-specific configurations or the use of security tools and protocols.



The NULL Scan is a TCP port scanning technique used to determine whether ports on a target system are open, closed, or filtered by sending TCP packets with no flags set (hence, "NULL"). The behaviour of the target system in response to these packets can reveal information about the state of the port being scanned.

IDLE/IPID Header Scan

Performing an IDLE/IPID Header Scan involves examining how the IP ID field in packets behaves when a system is idle, potentially indicating the presence of a firewall or intrusion detection system (IDS). Here's how to perform an IDLE/IPID Header Scan using Nmap in Kali Linux:



Countermeasures against IDLE/IPID Header Scan are essential to maintain network security and integrity. Here are five effective countermeasures:

1. **Firewall Configuration**: Configure firewalls to block incoming packets with unusual or unexpected IDLE/IPID header values. Implement rules to filter out packets used in IDLE/IPID Header Scans at the network perimeter, preventing unauthorized access and thwarting reconnaissance attempts.

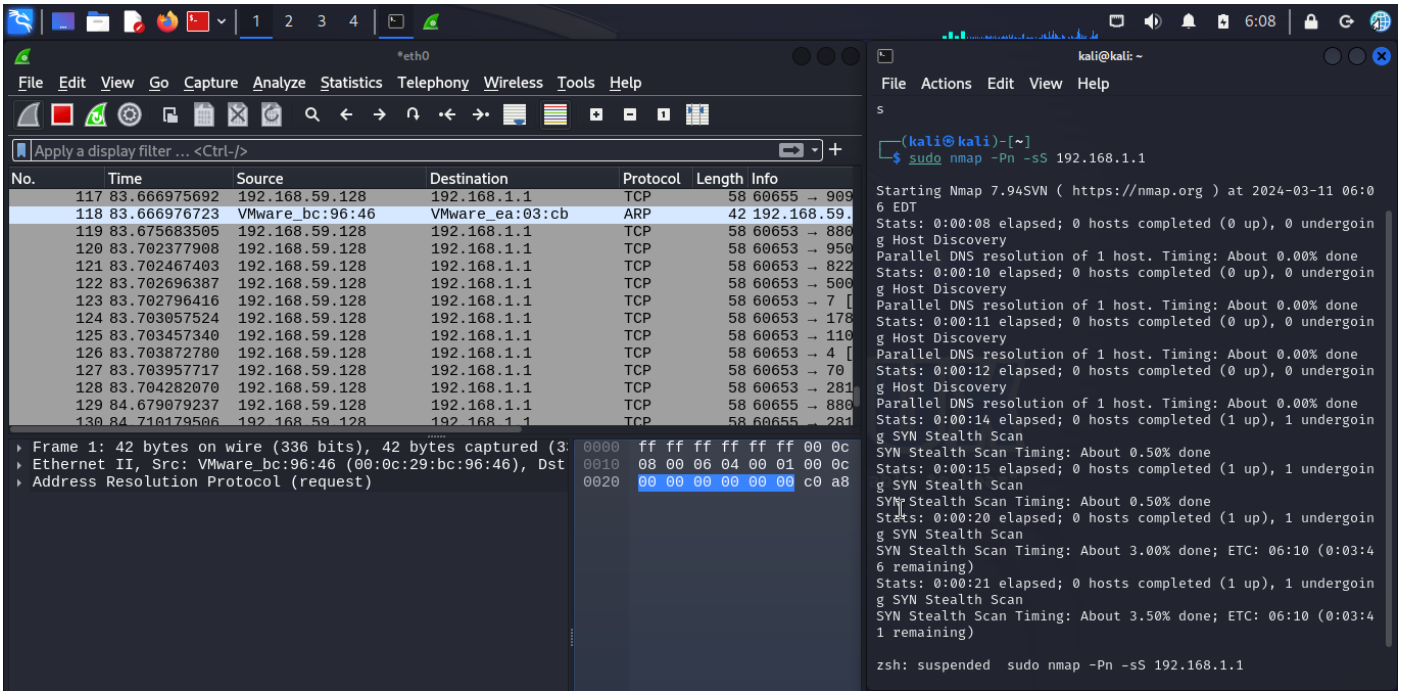
2. **Intrusion Detection Systems (IDS)**: Deploy IDS solutions capable of detecting and alerting administrators to suspicious IDLE/IPID Header Scan activity. Configure IDS rules to monitor for patterns and signatures associated with IDLE/IPID Header Scans, enabling proactive mitigation of potential threats.

3. **Packet Filtering**: Utilize packet filtering mechanisms on network devices to detect and block abnormal IDLE/IPID header values indicative of scanning activity. Configure filters to identify and drop packets that do not follow the expected IPID sequence, mitigating the effectiveness of IDLE/IPID Header Scans.

4. **Anomaly Detection**: Implement anomaly detection techniques to identify deviations from normal network behavior associated with IDLE/IPID Header Scan activity. Monitor network traffic for sudden increases in IDLE/IPID values or unusual IPID sequences, signaling potential reconnaissance attempts, and take appropriate action to investigate and respond to suspicious behavior.

5. **Network Hardening**: Harden network devices and protocols to mitigate the risk of exploitation by IDLE/IPID Header Scans. Regularly update and patch network systems to address known vulnerabilities, and implement secure configurations to minimize the impact of scanning activities on network security. Additionally, consider implementing encryption and authentication mechanisms to protect sensitive data and prevent unauthorized access.

Stealth Scan (Half Open Scan):



A Stealth Scan, also known as a Half Open Scan or SYN Scan, is a scanning technique used to determine open ports on a target system without completing the full TCP handshake. This method can help evade detection by intrusion detection systems (IDS) and firewall logging.

Countermeasures against Stealth Scans (Half Open Scans) are crucial to protect network security and prevent unauthorized access. Here are five effective countermeasures:

1. **Intrusion Detection Systems (IDS)**: Deploy IDS solutions capable of detecting and alerting administrators to suspicious Stealth Scan activity. Configure IDS rules to monitor for patterns and signatures associated with Half Open Scans, enabling proactive mitigation of potential threats.

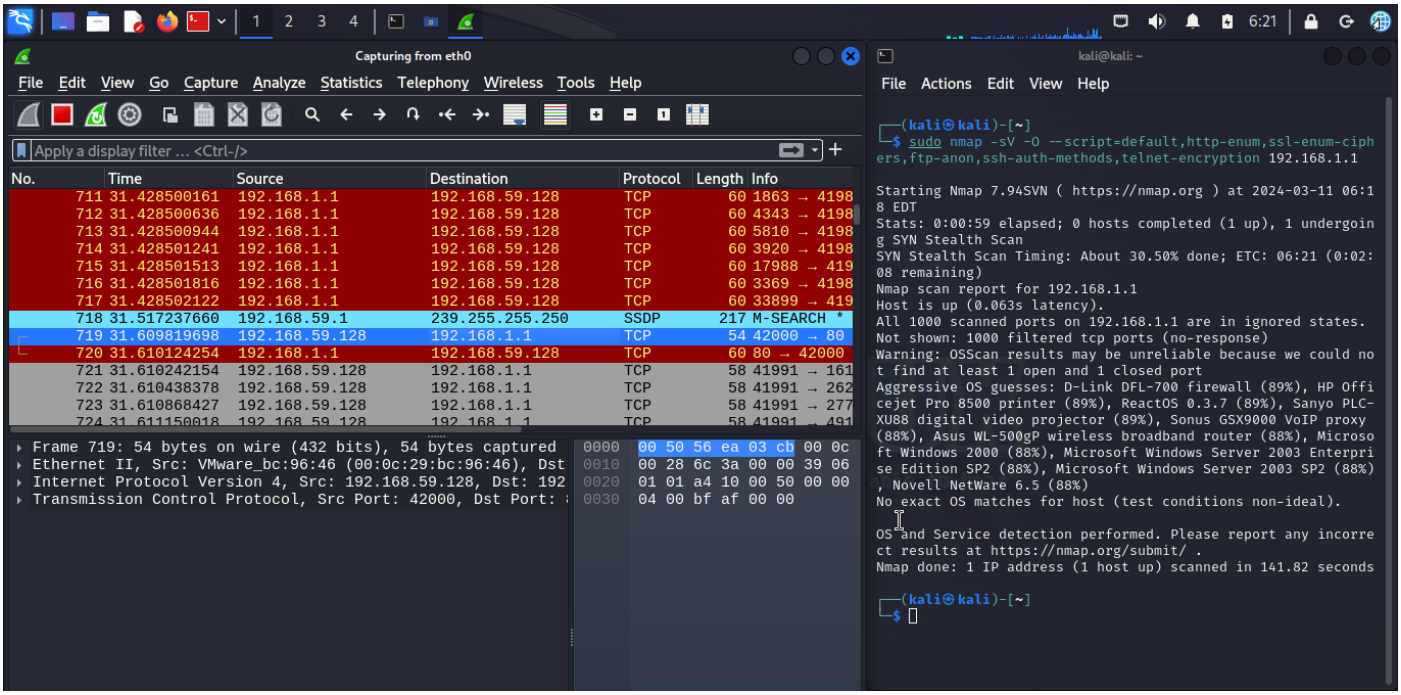
2. **Firewall Configuration**: Configure firewalls to filter and block incoming SYN packets from unauthorized sources. Implement rules to allow only legitimate TCP traffic and block SYN packets used in Stealth Scans, preventing unauthorized access and thwarting reconnaissance attempts.

3. **Port Honeypots**: Deploy port honeypots to deceive attackers and lure them away from critical systems. By mimicking open ports and services, honeypots can attract and divert Stealth Scan attempts, providing valuable insights into attacker tactics and techniques.

4. **Packet Filtering**: Utilize packet filtering mechanisms on network devices to detect and block abnormal SYN packet behavior indicative of Stealth Scans. Configure filters to identify and drop SYN packets that do not follow the expected TCP handshake sequence, mitigating the effectiveness of Half Open Scans.

5. **Anomaly Detection**: Implement anomaly detection techniques to identify deviations from normal network behavior associated with Stealth Scan activity. Monitor network traffic for sudden increases in SYN packets or SYN/ACK responses, signaling potential reconnaissance attempts, and take appropriate action to investigate and respond to suspicious behavior.

Advanced scanning with Nmap:



In this command:

Sudo is used to run Nmap with administrative privileges.

-sV enables service version detection to determine the versions of services running on open ports.

-O enables operating system detection to identify the operating system running on the target system.

--script=default,http-enum,ssl-enum-ciphers,ftp-anon,ssh-auth-methods,telnet-encryption specifies a selection of Nmap scripts to run against the target. These scripts include default scripts (default), HTTP enumeration (http-enum),

SSL/TLS cipher enumeration (ssl-enum-ciphers), FTP anonymous login check (ftp-anon), SSH authentication methods enumeration (ssh-auth-methods), and Telnet encryption check (telnet-encryption).

192.168.1.1 is the target IP address.

This command will perform an advanced scan on the target IP address 192.168.1.1, providing detailed information about the services running on open ports, the operating system running on the target system, and additional information gathered by running the specified Nmap scripts.

Countermeasures against advanced scanning techniques with Nmap are crucial to protect network security and prevent unauthorized access. Here are five effective countermeasures:

1. **Firewall Configuration**: Configure firewalls to block incoming scan traffic from unauthorized sources. Implement rules to filter out Nmap signatures and block packets used in advanced scanning techniques, such as ACK, SYN, or NULL scans, at the network perimeter.

2. **Intrusion Detection and Prevention Systems (IDPS)**: Deploy IDPS solutions capable of detecting and alerting administrators to suspicious Nmap scan activity. Configure IDPS rules to monitor for patterns and signatures associated with Nmap scans, including specific scan types and traffic anomalies, enabling proactive mitigation of potential threats.

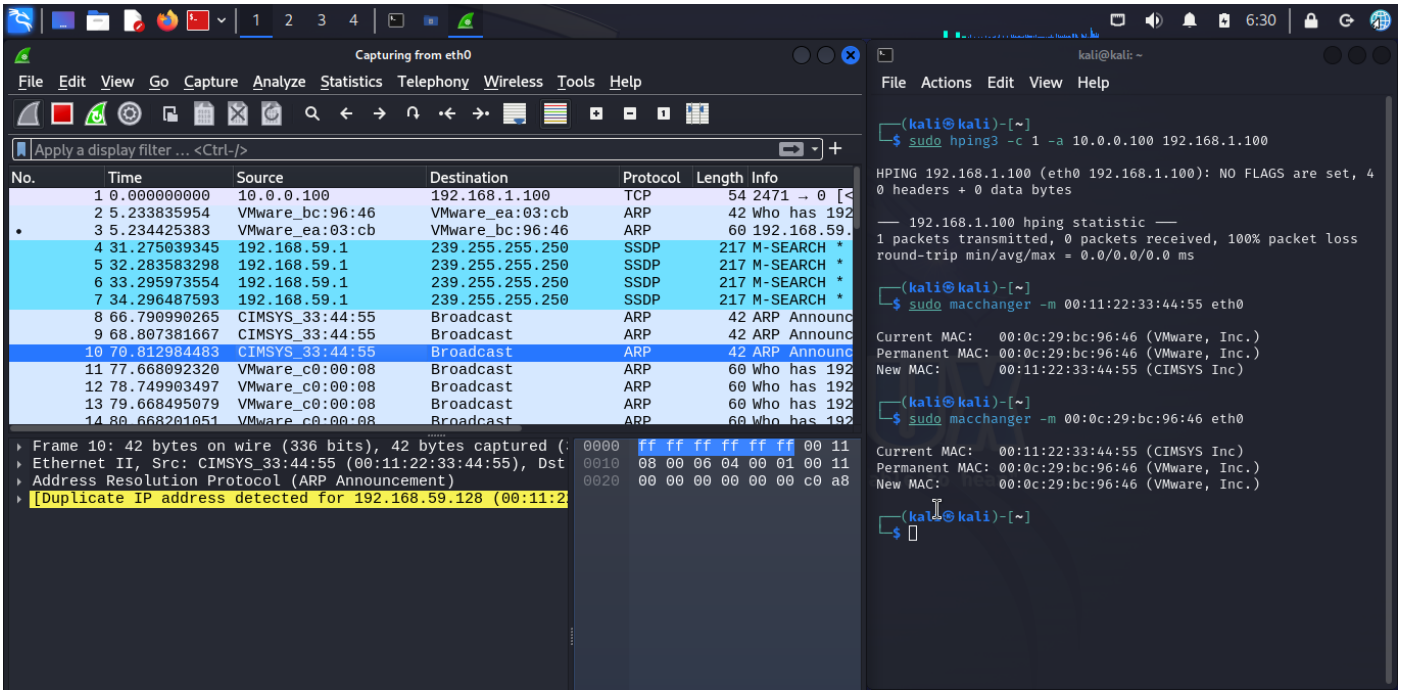
3. **Network Segmentation**: Segment the network into separate subnets or VLANs to limit the impact of Nmap scans. Implement access controls and routing policies to restrict communication between different network segments and prevent lateral movement by attackers who exploit vulnerabilities discovered through scanning.

4. **Packet Filtering and Access Controls**: Utilize packet filtering mechanisms and access control lists (ACLs) on network devices to restrict outbound traffic to known Nmap scanning ports. Implement strict policies to deny traffic to ports commonly targeted by Nmap scans, such as TCP ports 22 (SSH), 23 (Telnet), and 445 (SMB), mitigating the effectiveness of scanning attempts.

5. **Regular Vulnerability Scanning and Patch Management**: Conduct regular vulnerability scans of networked systems and apply security patches and updates promptly to address known vulnerabilities exploited by Nmap scans. Implement robust patch management practices to reduce the attack surface and mitigate the risk of exploitation by attackers conducting advanced scanning activities.

MAC Address Spoofing:

MAC address spoofing involves changing the hardware address (MAC address) of a network interface to impersonate another device on the network. This can be done using software tools or by manually modifying the MAC address configuration of the network interface.



Countermeasures against MAC address spoofing are essential to maintain network security and integrity. Here are five effective countermeasures:

1. **Port Security**: Implement port security features on network switches to restrict unauthorized devices from connecting to switch ports. Configure port security to bind MAC addresses to specific switch ports, preventing MAC address spoofing and unauthorized access.

2. **MAC Address Filtering**: Utilize MAC address filtering to allow only authorized devices to connect to the network. Maintain a whitelist of approved MAC addresses and configure network devices to block traffic from unrecognized MAC addresses, effectively mitigating MAC address spoofing attempts.

3. **Network Access Control (NAC)**: Deploy Network Access Control solutions to authenticate and authorize devices before granting access to the network. Use NAC policies to enforce compliance with security standards and verify the integrity of device MAC addresses, preventing unauthorized devices from gaining network access.

4. **MAC Address Monitoring**: Monitor network traffic for anomalous MAC address behavior, such as multiple devices using the same MAC address or frequent MAC address changes. Implement network monitoring tools to detect and alert administrators to suspicious MAC address spoofing activity, enabling prompt investigation and response.

5. **Encryption and Authentication**: Use encryption protocols such as WPA2-Enterprise or IEEE 802.1X to secure wireless networks and authenticate devices based on their MAC addresses. Encrypting network traffic and enforcing strong authentication mechanisms can deter attackers from attempting MAC address spoofing attacks and enhance overall network security.

Source Port Manipulation:

Source port manipulation involves altering the source port number of outgoing packets to potentially bypass network filtering or evade detection. However, it's important to note that while source port manipulation can be used for legitimate purposes in certain scenarios (e.g., load balancing, traffic analysis), it can also be abused for malicious activities (e.g., bypassing firewall rules, disguising attacks).

Countermeasures against source port manipulation are vital to prevent malicious activities such as network reconnaissance and spoofing attacks. Here are five effective countermeasures:

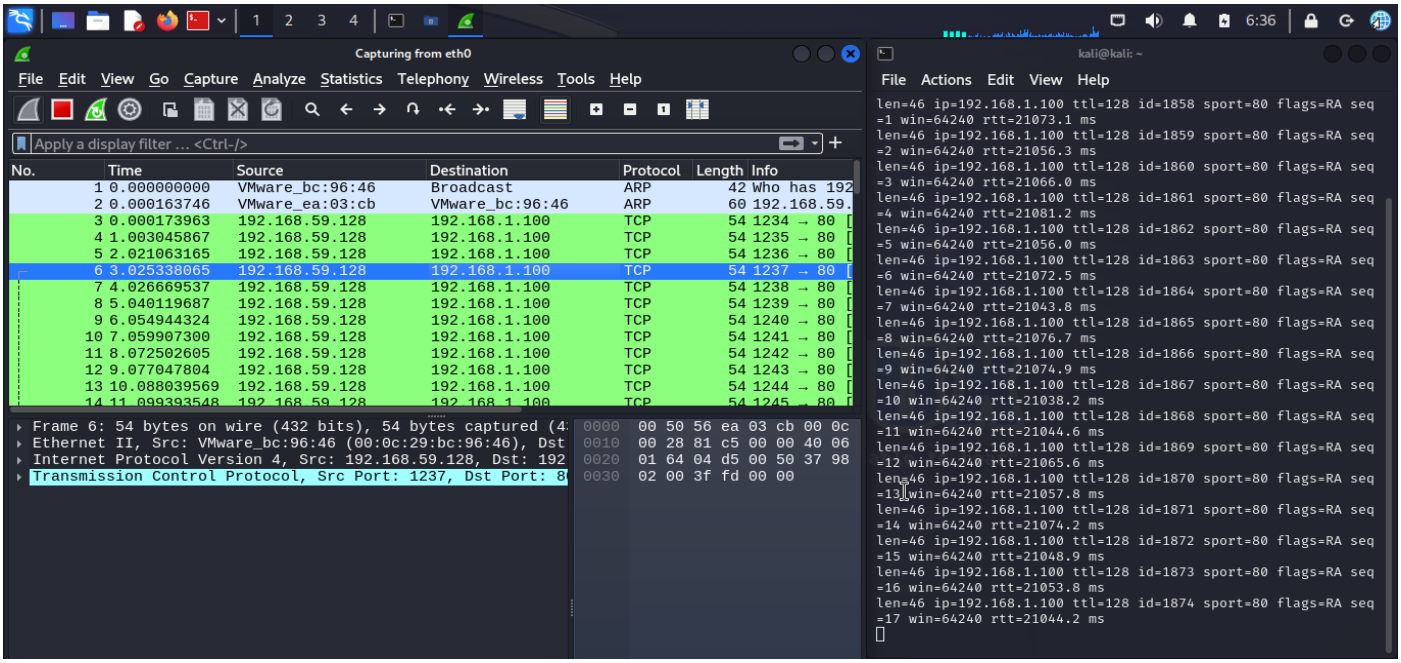
1. **Firewall Configuration**: Implement strict firewall rules to filter and block outbound traffic with manipulated or spoofed source ports. Configure firewall policies to allow only legitimate traffic to exit the network, preventing the use of manipulated source ports for unauthorized activities.

2. **Intrusion Detection Systems (IDS)**: Deploy IDS solutions capable of detecting anomalous traffic patterns resulting from source port manipulation. Configure IDS rules to alert administrators to suspicious behavior indicative of source port manipulation attempts, allowing for prompt investigation and response.

3. **Network Address Translation (NAT)**: Utilize NAT mechanisms to translate internal IP addresses and source ports to external addresses when communicating with external networks. By translating source ports to random values, NAT can mitigate the effectiveness of source port manipulation attacks and enhance network security.

4. **Traffic Monitoring and Analysis**: Implement comprehensive traffic monitoring and analysis tools to detect and analyze suspicious traffic patterns, including source port manipulation attempts. Regularly monitor network traffic for anomalies and unauthorized activities, enabling proactive detection and mitigation of potential threats.

5. **Packet Filtering and Access Controls**: Deploy packet filtering mechanisms and access control lists (ACLs) on network devices to restrict outbound traffic based on source port values. Implement strict policies to deny traffic originating from manipulated or suspicious source ports, preventing attackers from exploiting vulnerabilities or conducting reconnaissance activities.



4. Discussion:

- Methodology Limitations: Nmap's accuracy may be affected by network congestion or firewall settings.

- Strengths and Weaknesses: Nmap's versatility and extensive feature set make it a valuable tool, but its complexity may pose challenges for novice users.

- Project Improvement: Future iterations could include vulnerability exploitation testing and network visualization.

- Ethical Considerations: Responsible disclosure of vulnerabilities is crucial to prevent exploitation and ensure network security.

5. Conclusion:

The project demonstrated the effectiveness of Nmap in network analysis, uncovering valuable insights into network topology and security. Understanding the limitations and ethical considerations of network scanning is essential for responsible cybersecurity practices.

6. References:

- https://www.youtube.com/watch?v=LTMucsu35dk&t=200s

-https://us06web.zoom.us/rec/share/TkzPT5zX-RqVqGEe2FAD5Y7gDFbDqCyqpt3gi9XzA-Yn0TOM-KqzfTjJ07AjUJyx.mFqZFhM1esJi1C-E

This report summarizes the Network Analyzer project, showcasing the significance of network analysis and the capabilities of Nmap in securing network infrastructure.