Old Dominion University

CYSE 301: Cybersecurity Technique and Operations
Assignment 3: Sword vs. Shield

Jessiah Davis

#01235044

Task A: Sword - Network Scanning (5 + 15+ 20 = 40 points)

Power on the listed VMs,

• External Kali

• pfSense

• Ubuntu

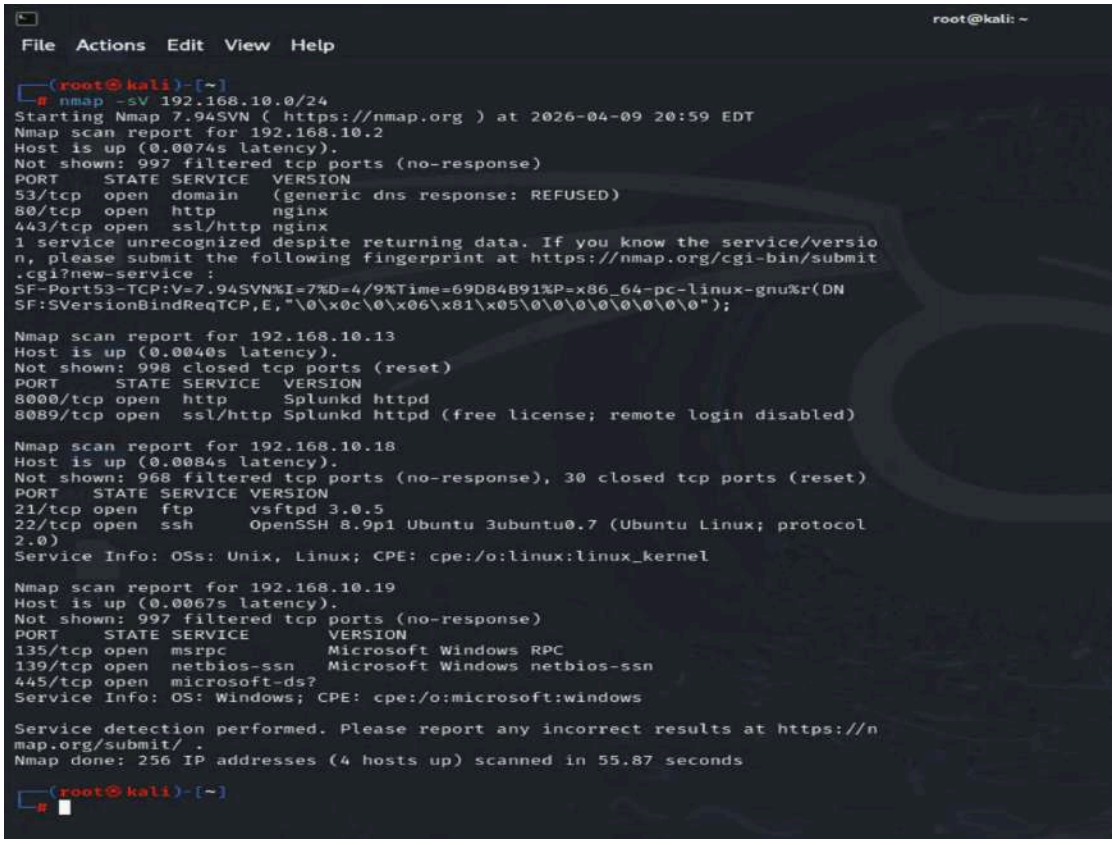
• Windows Server 2022

Make sure not to add/delete any firewall rule/policies before continuing.

1. Run Wireshark in the Internal Kali VM while External Kali is scanning the network.

2. Use Nmap in External Kali to profile the basic information about the subnet topology

(including open ports information, operating systems, etc.)



After running `nmap -sV 192.168.10.0/24`, four active IP addresses were identified: 192.168.10.2, 192.168.10.13, 192.168.10.18, and 192.168.10.19. On 192.168.10.2, three ports are open: 53/tcp (DNS), 80/tcp (HTTP running nginx), and 443/tcp (SSL/HTTP also running nginx). The OS is not explicitly identified. For 192.168.10.13, ports 8000/tcp and 8089/tcp are open, both running Splunk httpd services, with one indicating a free license and remote login disabled. No OS details are provided. On 192.168.10.18, port 21/tcp (FTP running vsftpd 3.0.5) and 22/tcp (SSH running OpenSSH 8.9p1 on Ubuntu) are open. The OS is identified as Linux/Unix. Finally, 192.168.10.19 shows 135/tcp (MSRPC), 139/tcp (NetBIOS-SSN), and 445/tcp (Microsoft-DS) open, indicating a Windows system. Overall, the scan reveals a mix of Linux and Windows hosts with services including web servers, file transfer, remote access, and network services, which could present multiple potential attack surfaces depending on configuration and patch levels.

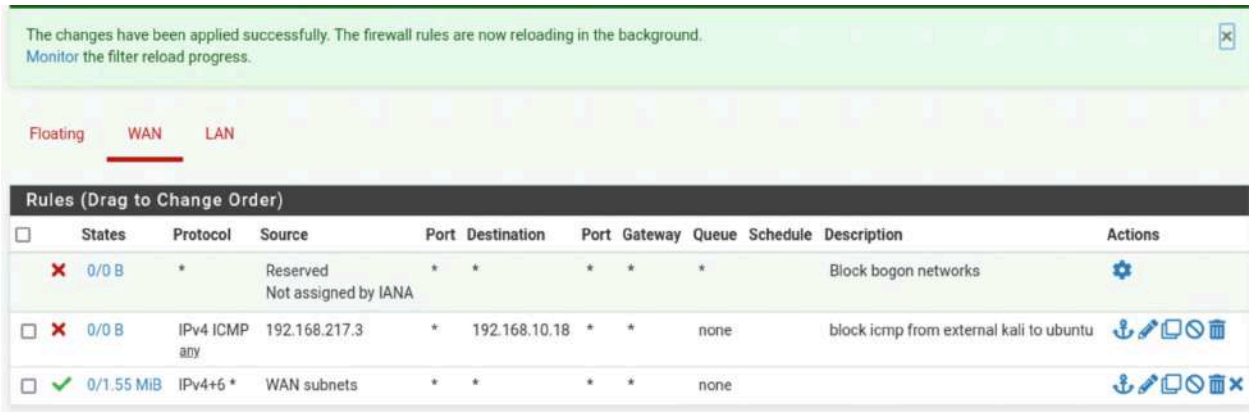
IP Adresses	Open Ports	Service Versions	Operating system detection	Backend software information
192.168.10.2	53, 80, 443	DNS, HTTP, nginx, HTTPS nginx	Not detected	Nginx web server
192.168.10.13	8000, 8089	splunk httpd	Not detected	Splunk web interface
192.168.10.18	21, 22	Vsftpd 3.0.5, OpenSHH 8.9p1	linux/Unix (ubuntu)	Ubuntu linux, OpenSSH
192.168.10.19	135, 139, 445	MSRPC, NetBIOS-SSN, Microsoft-DS	Windows	Microsoft windows

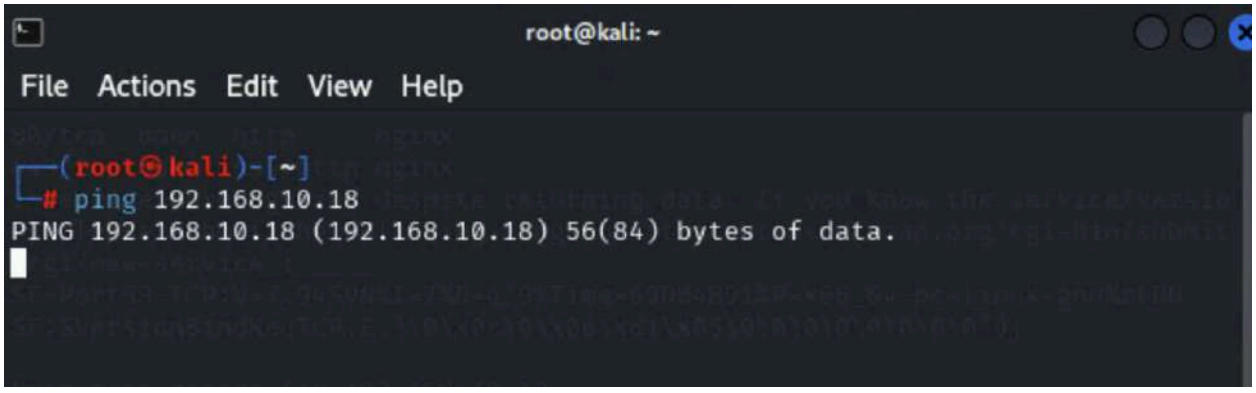
Task B: Shield – Protect your network with a firewall (10 + 15+ 15 + 20 = 60 points)

In order to receive full credits, you need to fill the table (add more rows if needed), implement the firewall rule(s), attach the screenshot of your rule configured in the firewall table, and attach the screenshot of the rule and the Ping test verification of the rules.

Important: Open Pfsense using a browser on the Internal Kali.

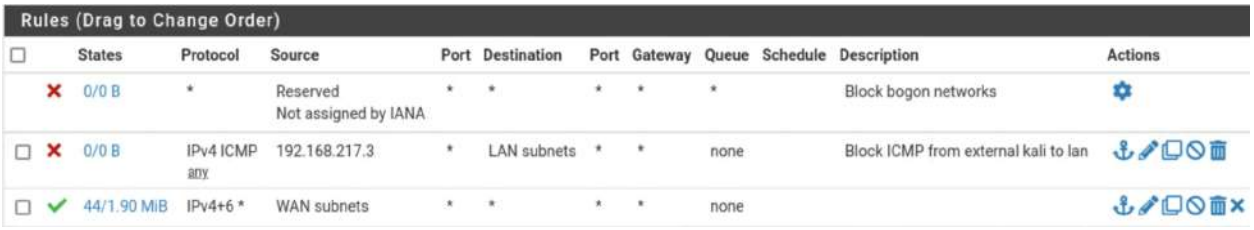
1. Configure the pfSense firewall rule to block the ICMP traffic from External Kali to the Ubuntu VM.

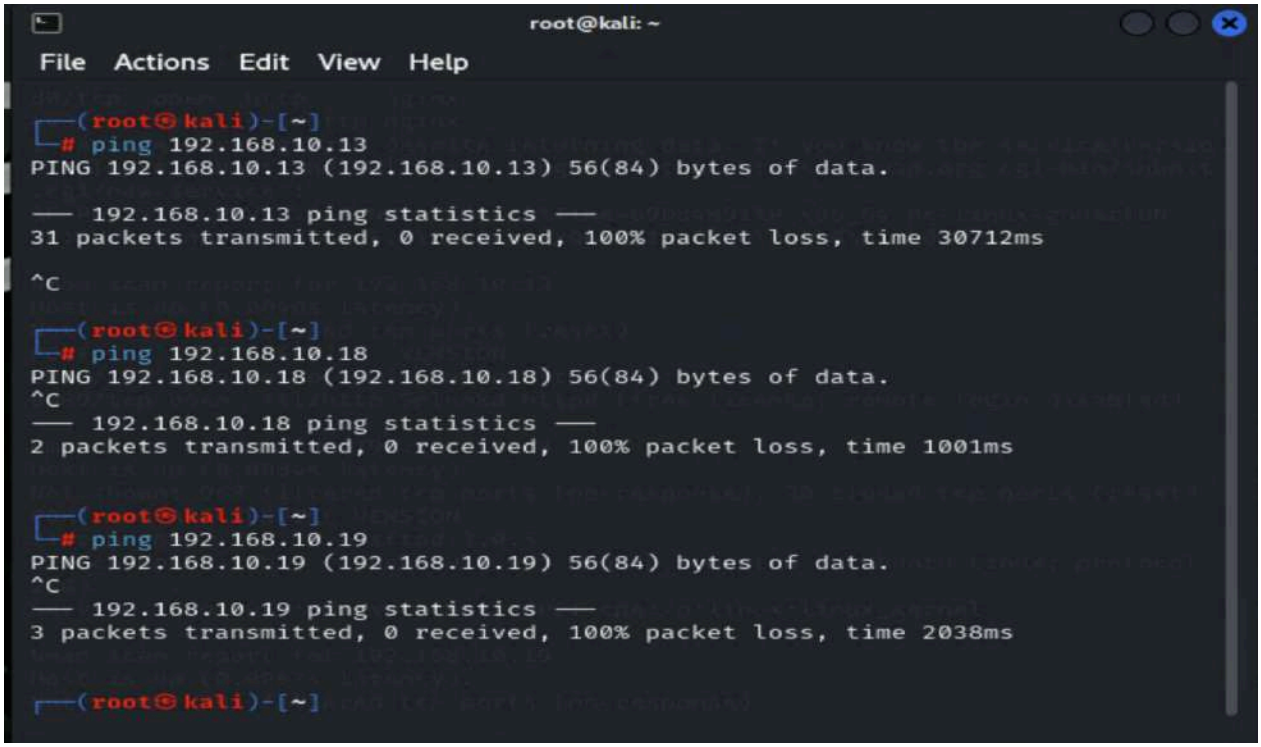




Rule	interface	action	Source IP	Destination IP	Protocol
1	WAN	block	192. 168.217.3	192.168.10.1 8	ICMP

2. Clear the previous firewall policies and configure the pfSense firewall to block all ICMP traffic from External Kali to the LAN side

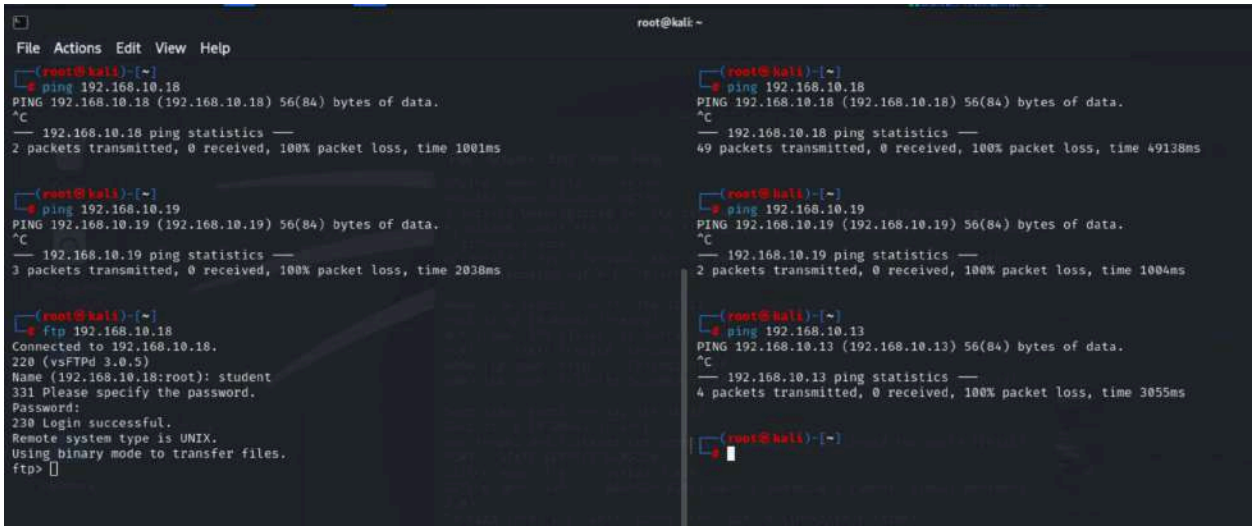


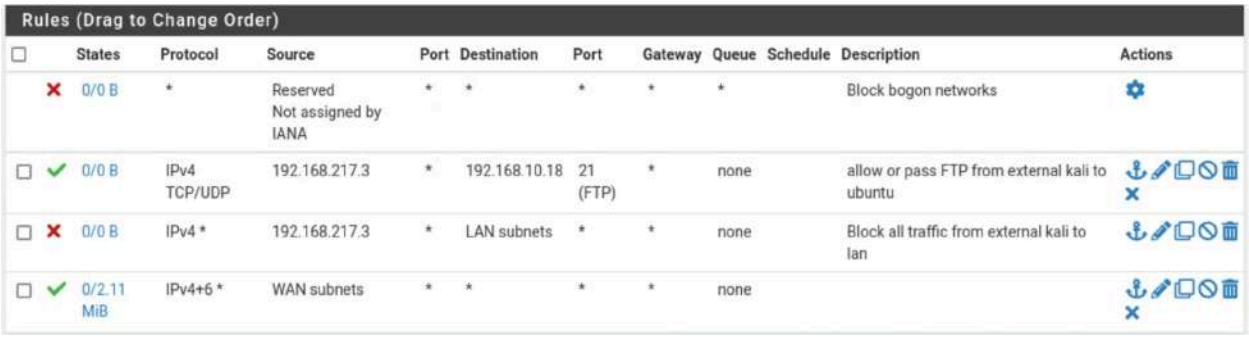


Rule	Interface	action	Source IP	Destination Ip	Protocol
1	WAN	Block	192.168.217.3	Any	ICMP

Blocking ICMP provides several security advantages. It reduces network visibility to outsiders by preventing common reconnaissance techniques such as ping sweeps, which attackers use to identify active hosts. This can slow down or discourage initial discovery phases of an attack. Additionally, blocking ICMP can help mitigate certain denial-of-service attacks, like ICMP flood (ping flood), which attempt to overwhelm a system with traffic. However, there are operational disadvantages. ICMP is essential for normal network diagnostics and troubleshooting. Tools like ping and traceroute rely on it to test connectivity and identify routing issues. Blocking ICMP can make it harder for administrators to detect network failures, latency problems, or misconfigurations. So while blocking ICMP can improve security, it can also reduce network visibility and complicate maintenance.

3. Clear the previous firewall policies and configure the pfSense firewall to block ALL traffic from External Kali to the LAN side, except for the FTP protocol directed towards Ubuntu. You can add more rows in the table here, if necessary

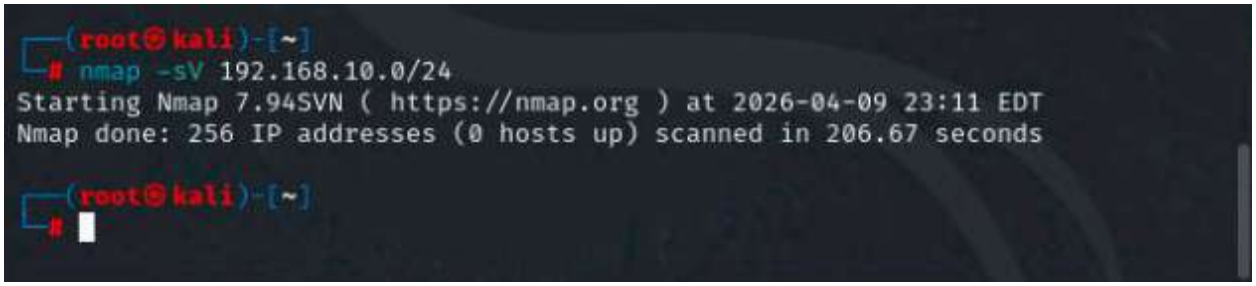




Rule	interface	action	Source IP	Destination IP	Protocol
1	WAN	Pass	192.168.217.3	192.168.10.11	FTP (port21)
2	WAN	Block	192.168.217.3	Any	Any

4. Keep the firewall policies you created in Task B.3 and repeat Task A.1 to rescan the subnet using

nmap in External Kali. Now, compare the results/ findings, and complete the following table and add your reflection as asked here:



So Nmap shows “filtered” instead of “closed” when a firewall blocks the probe packets without sending a response. A “closed” port means the host is reachable but the port is not in use and actively rejects the connection. In contrast, “filtered” indicates that a firewall is dropping the packets entirely, so Nmap cannot determine whether the port is open or closed. Firewall behavior significantly impacts scan results by controlling how packets are handled. If a firewall drops packets silently, Nmap cannot gather information, resulting in no hosts being discovered. If it rejects packets, Nmap may label ports as “closed.” Strong firewall rules reduce visibility and limit reconnaissance. Almost all of the critical information is hidden from attackers such as the IP addresses, active host, and operating system details. Overall this prevents the attackers from seeing information that could expose vulnerabilities and make it hard in the future to attack.