

OLD DOMINION UNIVERSITY

CYSE 301 CYBERSECURITY TECHNIQUES AND OPERATIONS

Assignment #3 Sword vs. Shield

Jo Brown

01196622

Below is the snippet of a sample lab report.

TASK A

1. Use Nmap to profile the basic information about the subnet topology (including open ports information, operation systems, etc.) You need to get the service and backend software information associated with each opening port in each VM

```
(root@kali)-[~]
# nmap 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-09 20:55 EDT
Nmap scan report for 192.168.10.2
Host is up (0.0025s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
443/tcp   open  https

Nmap scan report for 192.168.10.18
Host is up (0.0038s latency).
Not shown: 968 filtered tcp ports (no-response), 30 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh

Nmap scan report for 192.168.10.19
Host is up (0.0040s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds

Nmap done: 256 IP addresses (3 hosts up) scanned in 25.33 seconds
```

This is the screenshot showing the nmap results for the open ports, which also shows the services they are running

2. Run Wireshark in Internal Kali VM while External Kali is scanning the network. Discuss the traffic pattern you observed. What do you find? Please write a 200-word essay to discuss your findings

No.	Time	Source	Destination	Protocol	Length	Info
790	11.100067000	192.168.217.3	192.168.10.10	TCP	50	48350 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
890	17.100847800	192.168.10.13	192.168.217.3	TCP	54	89 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
891	17.107454100	192.168.217.3	192.168.10.13	TCP	50	48350 → 23 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
892	17.107461700	192.168.217.3	192.168.10.10	TCP	50	48350 → 22 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
893	17.107602000	192.168.10.13	192.168.217.3	TCP	54	92 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
894	17.122876400	192.168.217.3	192.168.10.13	TCP	50	48350 → 587 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
895	17.122916000	192.168.10.13	192.168.217.3	TCP	54	587 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
896	17.125928200	192.168.217.3	192.168.10.13	TCP	50	48350 → 2120 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
897	17.125935500	192.168.217.3	192.168.10.13	TCP	50	48350 → 8080 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
898	17.125942900	192.168.217.3	192.168.10.13	TCP	50	48350 → 1025 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
899	17.125950300	192.168.217.3	192.168.10.13	TCP	50	48350 → 113 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
900	17.125959500	192.168.217.3	192.168.10.13	TCP	50	48350 → 1123 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
911	17.126629400	192.168.10.13	192.168.217.3	TCP	54	1720 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
912	17.126643100	192.168.10.13	192.168.217.3	TCP	54	8880 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
913	17.126656000	192.168.10.13	192.168.217.3	TCP	54	1025 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
914	17.126669100	192.168.10.13	192.168.217.3	TCP	54	135 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
915	17.126682000	192.168.10.13	192.168.217.3	TCP	54	1723 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
916	17.131819200	192.168.217.3	192.168.10.13	TCP	50	48350 → 9900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
917	17.131846900	192.168.217.3	192.168.10.13	TCP	50	48350 → 3306 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
918	17.131817700	192.168.217.3	192.168.10.13	TCP	50	48350 → 110 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
919	17.131849600	192.168.217.3	192.168.10.13	TCP	50	48350 → 199 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
920	17.135858800	192.168.217.3	192.168.10.13	TCP	50	48350 → 150 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
914	17.126669100	192.168.10.13	192.168.217.3	TCP	54	135 → 48350 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2796	16.222905500	192.168.217.3	192.168.10.13	TCP	50	48350 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2812	16.346960900	192.168.217.3	192.168.10.10	TCP	50	48350 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460

The first screenshot is a selection of the traffic generated by the external attacker as it was running its Nmap scan, the second is of the traffic generated specifically by a port being sniffed.

Regarding my findings in this matter. Utilizing wireshark on the Internal Kali machine, which worked in the background while I was scanning the network of the pfsense firewall, I found several things. Firstly, the traffic generated by the External Kali attacker was very noticeable, highlighted in red

TASK B

In order to receive full credits, you need to fill the table (add more rows if needed), implement the firewall rule(s), show me the screenshot of your firewall table, and verify the results.

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

Rule #	Interface	Action	Source IP	Destination IP	Protocol (port # if applicable)
2	WAN	Block	192.168.217.3	192.168.10.18	ICMP

[Add the screenshot here]

Rules (Drag to Change Order)												Actions
☐	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description		
☒	0/1.09 MiB	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule		
☐	☒ 0/47 KiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	 	
☐	☒ 0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	 	
☐	☒ 0/0 B	IPv4 ICMP	192.168.217.3	*	192.168.10.18	*	*	none		any	 	

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 (port # if applicable)
2	LAN	Block	192.168.217.3	Any LAN	ICMP

[Add the screenshot here]

Rules (Drag to Change Order)											
☐	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☒	0/1.09 MiB	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule	
☐	0/0 B	IPv4 ICMP any	192.168.217.3	*	LAN address	*	*	none			

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 towards Ubuntu.

Rule #	Interface	Action	Source IP	Destination IP	Protocol (port # if applicable)
1	LAN	Block	192.168.217.3	Any LAN	Any
2	LAN	Pass	192.168.217.3	192.168.10.18	TCP

[Add the screenshot here]

Rules (Drag to Change Order)											
☐	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☒	1/1.10 MiB	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule	
☐	0/0 B	IPv4 *	192.168.217.3	*	*	*	*	none			
☐	0/0 B	IPv4 TCP	192.168.217.3	*	192.168.10.18	*	*	none			

4. Keep the firewall policies you created in Task B.3 and repeat Task A.1. What's the difference?

```
└─# nmap 192.168.10.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-10 02:03 EDT
Nmap scan report for 192.168.10.2
Host is up (0.0023s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
443/tcp   open  https

Nmap scan report for 192.168.10.13
Host is up (0.0035s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE
8000/tcp  open  http-alt
8089/tcp  open  unknown

Nmap scan report for 192.168.10.18
Host is up (0.0084s latency).
Not shown: 968 filtered tcp ports (no-response), 30 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh

Nmap scan report for 192.168.10.19
Host is up (0.0045s latency).
Not shown: 997 filtered tcp ports (no-response)
```

I am not sure what I've done wrong, but nothing has seemed to change.