

Step-1:
Sandbox, Firewall & Access Control

Red Team

Jair Ramirez, David Garcia, and Zach Lay

CS 4371 Computer System Security
Texas State University

September 24th, 2024

Section I - Introduction

1. Background

This project provided our team with hands-on experience creating and securing virtual network environments using a virtual machine manager, virtual machines, and a router/firewall. The primary goal of Project-1 was to simulate a real-world company network divided into two networks: Network A and Network B, each containing isolated machines connected through Router R, which enforces firewall rules between the networks. Network A consists of two internal machines: an Ubuntu server and a Windows XP workstation, while Network B has two external machines running Kali Linux and Windows 95. The scope includes setting up a virtual environment, installing operating systems and programs like Wireshark, configuring firewall rules, and analyzing network traffic using tools like ping and curl. This project allowed the team to develop valuable technical skills by working with real-world examples and industry security software.

Section II - Network Setup and Diagnostics (Tasks II & III)

1. Screenshot of Virtual Machine Manager

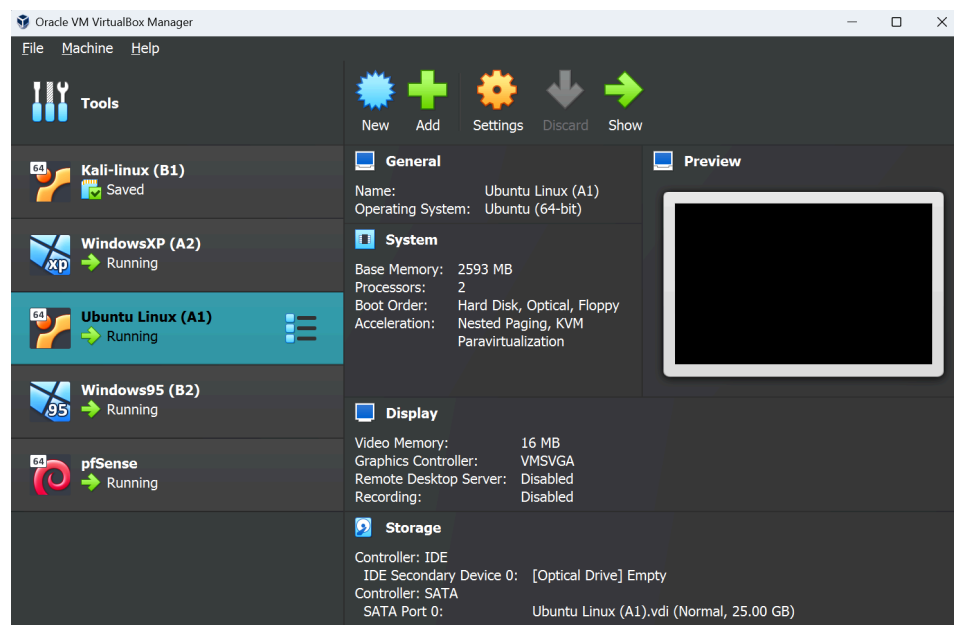


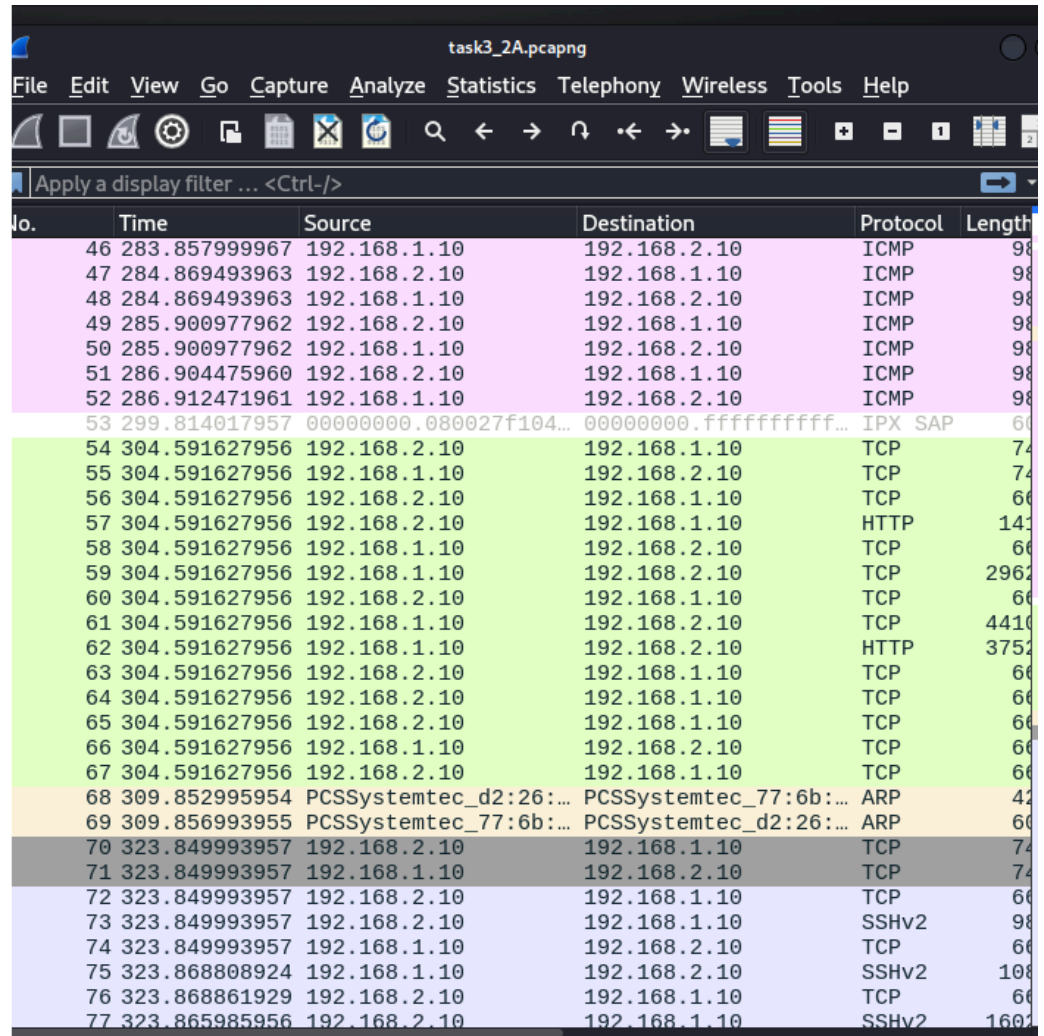
Fig 1: VM Manager with 5 virtual machines. A.1(Ubuntu), A.2(Windows XP), B.1(Kali), B.2(Windows 95) + Router (pfsense)

2. NMap Commands for Scanning Computers and Service Ports

Scanning Network A: To perform a regular NMap scan on Network A, we used the command ‘sudo nmap 192.168.1.0/24’, where 192.168.1.0/24 refers to the Network A subnet. When noting the service ports after the first scan, we identified the following were open from the Ubuntu Server A.1: Port 22 (SSH), Port 80 (HTTP), and Port 443 (HTTPS). Windows XP Workstation: Port 135 (MS

RPC), Port 139 (NetBIOS), and Port 445 (Microsoft-DS SMB). When scanning Network B, we used the following NMap command for a normal scan: ‘sudo nmap 192.168.2.0/24’. After this scan, we noted the open service port 22 (SSH) on the Kali machine, with no significant ports open on the Windows 95 machine.

3. Wireshark Result Screenshots



No.	Time	Source	Destination	Protocol	Length
46	283.857999967	192.168.1.10	192.168.2.10	ICMP	98
47	284.869493963	192.168.2.10	192.168.1.10	ICMP	98
48	284.869493963	192.168.1.10	192.168.2.10	ICMP	98
49	285.900977962	192.168.2.10	192.168.1.10	ICMP	98
50	285.900977962	192.168.1.10	192.168.2.10	ICMP	98
51	286.904475960	192.168.2.10	192.168.1.10	ICMP	98
52	286.912471961	192.168.1.10	192.168.2.10	ICMP	98
53	299.814017957	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60
54	304.591627956	192.168.2.10	192.168.1.10	TCP	74
55	304.591627956	192.168.1.10	192.168.2.10	TCP	74
56	304.591627956	192.168.2.10	192.168.1.10	TCP	60
57	304.591627956	192.168.2.10	192.168.1.10	HTTP	141
58	304.591627956	192.168.1.10	192.168.2.10	TCP	60
59	304.591627956	192.168.1.10	192.168.2.10	TCP	2962
60	304.591627956	192.168.2.10	192.168.1.10	TCP	60
61	304.591627956	192.168.1.10	192.168.2.10	TCP	4410
62	304.591627956	192.168.1.10	192.168.2.10	HTTP	3752
63	304.591627956	192.168.2.10	192.168.1.10	TCP	60
64	304.591627956	192.168.2.10	192.168.1.10	TCP	60
65	304.591627956	192.168.2.10	192.168.1.10	TCP	60
66	304.591627956	192.168.1.10	192.168.2.10	TCP	60
67	304.591627956	192.168.2.10	192.168.1.10	TCP	60
68	309.852995954	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42
69	309.856993955	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60
70	323.849993957	192.168.2.10	192.168.1.10	TCP	74
71	323.849993957	192.168.1.10	192.168.2.10	TCP	74
72	323.849993957	192.168.2.10	192.168.1.10	TCP	60
73	323.849993957	192.168.2.10	192.168.1.10	SSHv2	98
74	323.849993957	192.168.1.10	192.168.2.10	TCP	60
75	323.868808924	192.168.1.10	192.168.2.10	SSHv2	108
76	323.868861929	192.168.2.10	192.168.1.10	TCP	60
77	323.865985956	192.168.2.10	192.168.1.10	SSHv2	1601

Fig 2: Wireshark Screenshot from B.1 machine monitoring requests to from B.1 to A.1. A.1's ip is 192.168.1.10, B.1's ip is 192.168.2.10

No.	Time	Source	Destination	Protocol	Length	Info
40	19.892494	192.168.1.10	192.168.2.10	ICMP	98	Echo (ping) request
41	20.887012	192.168.2.10	192.168.1.10	ICMP	98	Echo (ping) reply
42	20.887379	192.168.1.10	192.168.2.10	ICMP	98	Echo (ping) request
43	21.886697	192.168.2.10	192.168.1.10	ICMP	98	Echo (ping) reply
44	21.887155	192.168.1.10	192.168.2.10	ICMP	98	Echo (ping) request
45	39.595474	192.168.2.10	192.168.1.10	TCP	74	36356 → 80
46	39.595934	192.168.1.10	192.168.2.10	TCP	74	80 → 36356
47	39.596506	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
48	39.596566	192.168.2.10	192.168.1.10	HTTP	141	GET / HTTP/1.1
49	39.596852	192.168.1.10	192.168.2.10	TCP	66	80 → 36356
50	39.599500	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
51	39.599530	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
52	39.599537	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
53	39.599565	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
54	39.599571	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
55	39.599585	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
56	39.599591	192.168.1.10	192.168.2.10	TCP	1514	80 → 36356
57	39.599597	192.168.1.10	192.168.2.10	HTTP	856	HTTP/1.1
58	39.599955	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
59	39.601013	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
60	39.601061	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
61	39.601500	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
62	39.601965	192.168.1.10	192.168.2.10	TCP	66	80 → 36356
63	39.602397	192.168.2.10	192.168.1.10	TCP	66	36356 → 80
64	58.892107	192.168.2.10	192.168.1.10	TCP	74	40266 → 22
65	58.892461	192.168.1.10	192.168.2.10	TCP	74	22 → 40266
66	58.897977	192.168.2.10	192.168.1.10	TCP	66	40266 → 22
67	58.899104	192.168.2.10	192.168.1.10	SSHv2	98	Client: P
68	58.899441	192.168.1.10	192.168.2.10	TCP	66	22 → 40266
69	58.934505	192.168.1.10	192.168.2.10	SSHv2	108	Server: P
70	58.935220	192.168.2.10	192.168.1.10	TCP	66	40266 → 22

Fig 3: Wireshark Screenshot from A.1 machine monitoring requests to from B.1 to A.1. A.1's ip is 192.168.1.10, B.1's ip is 192.168.2.10

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	General Query
2	59.905829923	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	Nearest Query
3	119.920009999	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	General Query
4	179.882014001	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	Nearest Query
5	199.688106006	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
6	199.688106006	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
7	200.715592001	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
8	200.715592001	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
9	201.751074002	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
10	201.751074002	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
11	202.794552004	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
12	202.794552004	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
13	205.005446002	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42	Who has 192.168.2.1? Tell 192.168.2.10
14	205.005446002	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.1 is at 08:00:27:77:6b:9b
15	222.892498006	192.168.2.10	192.168.1.11	TCP	74	53948 → 80 [SYN] Seq=0 Win=32120 Len=0
16	222.892498006	192.168.1.11	192.168.2.10	TCP	60	80 → 53948 [RST, ACK] Seq=1 Ack=1 Win=0
17	229.053416004	192.168.2.10	192.168.1.11	TCP	74	47112 → 22 [SYN] Seq=0 Win=32120 Len=0
18	229.053416004	192.168.1.11	192.168.2.10	TCP	60	22 → 47112 [RST, ACK] Seq=1 Ack=1 Win=0
19	239.852014010	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	General Query
20	299.755090814	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	Nearest Query

Fig 4: Wireshark Screenshot from B.1 machine monitoring requests to from B.1 to A.2. A.2's ip is 192.168.1.11, B.1's ip is 192.168.2.10

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
2	0.000559	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
3	1.029248	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
4	1.029905	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
5	2.100041	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
6	2.100612	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
7	3.102478	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) request
8	3.103149	192.168.1.11	192.168.2.10	ICMP	98	Echo (ping) reply
9	23.216610	192.168.2.10	192.168.1.11	TCP	74	53948 → 80 [SYN]
10	23.217242	192.168.1.11	192.168.2.10	TCP	60	80 → 53948 [RST]
11	29.383130	192.168.2.10	192.168.1.11	TCP	74	47112 → 22 [SYN]
12	29.383957	192.168.1.11	192.168.2.10	TCP	60	22 → 47112 [RST]

Fig 5: Wireshark Screenshot from A.1 machine monitoring requests to from B.1 to A.2.

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	00000000.080027f104...	00000000.ffffffffff...	IPX SAP	60	Nearest Query
2	1.079460006	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) request id=0x748b, seq=1/250
3	1.079460006	PCSSystemtec_f1:04:...	Broadcast	ARP	60	Who has 192.168.2.10? Tell 192.168.2.11
4	1.079460006	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	192.168.2.10 is at 08:00:27:d2:26:79
5	1.079460006	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) reply id=0x748b, seq=1/250
6	2.122938004	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) request id=0x748b, seq=2/510
7	2.122938004	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) reply id=0x748b, seq=2/510
8	3.150424002	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) request id=0x748b, seq=3/760
9	3.154422002	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) reply id=0x748b, seq=3/760
10	6.148924003	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	Who has 192.168.2.11? Tell 192.168.2.10
11	6.148924003	PCSSystemtec_f1:04:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.11 is at 08:00:27:f1:04:ea
12	8.211892003	192.168.2.10	192.168.2.11	TCP	74	34822 → 80 [SYN] Seq=0 Win=32120 Len=0
13	8.211892003	192.168.2.11	192.168.2.10	TCP	60	80 → 34822 [RST, ACK] Seq=1 Ack=1 Win=0
14	19.890049996	192.168.2.10	192.168.2.11	TCP	74	53100 → 22 [SYN] Seq=0 Win=32120 Len=0
15	19.890049996	192.168.2.11	192.168.2.10	TCP	60	22 → 53100 [RST, ACK] Seq=1 Ack=1 Win=0
16	30.361448768	192.168.2.10	192.168.2.11	TCP	74	59994 → 22 [SYN] Seq=0 Win=32120 Len=0
17	30.348817998	192.168.2.11	192.168.2.10	TCP	60	22 → 59994 [RST, ACK] Seq=1 Ack=1 Win=0

Fig 6: Wireshark Screenshot from B.1 machine monitoring requests to from B.1 to B.1. B.1's ip is 192.168.2.10. B.2's ip is 192.168.2.11

4. What Web Services are Allowed Between Computers Before Task IV?

Before we implemented the security rules in Task IV, Ubuntu Server A.1 provided HTTP through Port 80 and HTTPS through Port 443, allowing web services to all machines in Networks A and B. This means that A.2, B.1, and B.2 can access web services provided by Ubuntu Server A.1. No other VMs in the sandbox are configured to host web services, which is why A.1 is the only server.

Section III - Security Policy Implementation (Tasks IV & V)

1. Access Control Matrix

Machine	Server-Provided Web Service	Server-Provided SSH Service	External-Provided Web Service	Workstation-Provided Web Service	Ping to Company Machines	Ping to External Machines
A.1 - Company Server	N/A	N/A	Block	N/A	Allow	Block
A.2 - Company Workstation	Allow	Allow	Allow	N/A	Allow	Block
B.1 - External Machine	Allow	Block	N/A	N/A	Block	Allow
B.2 - External Machine	Allow	Block	N/A	N/A	Block	Allow

2. Which Policies Cannot be Completely Enforced by the Router Rules of R

- Due to the simplicity and limitations of the pfSense firewalls, some policies and capabilities can't be fully enforced. For example, Router R can allow and block traffic based on criteria like protocol and IP address, but it can't filter traffic based on the type of traffic content, such as specific types of websites or applications. This would require application-level filtering, which is too high on the OSI model for a firewall like pfSense to be able to affect it. Another policy that can not be completely enforced is workstations providing web services. pfSense can block common ports like HTTP and HTTPS, but it can't prevent a workstation from internally hosting a locally accessible web service through internal routing.

3. R Router Rules Screenshot and Explanation

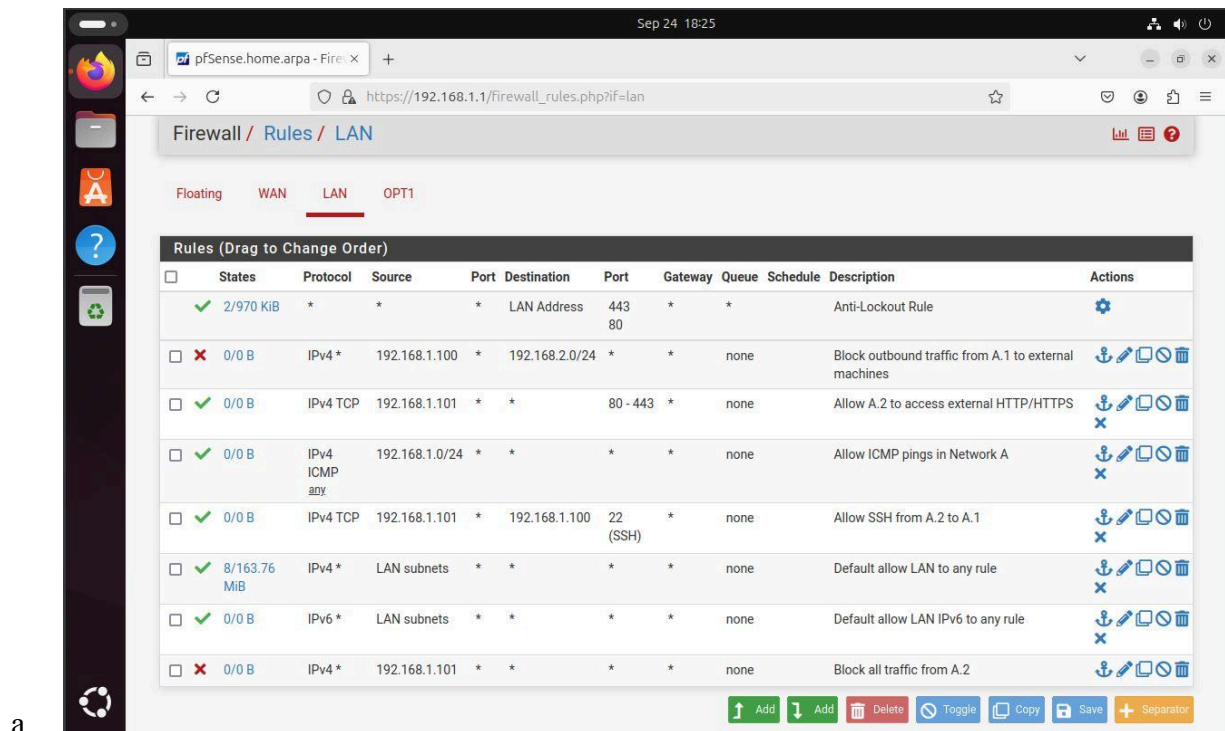


Fig 7:LAN Firewall Rules

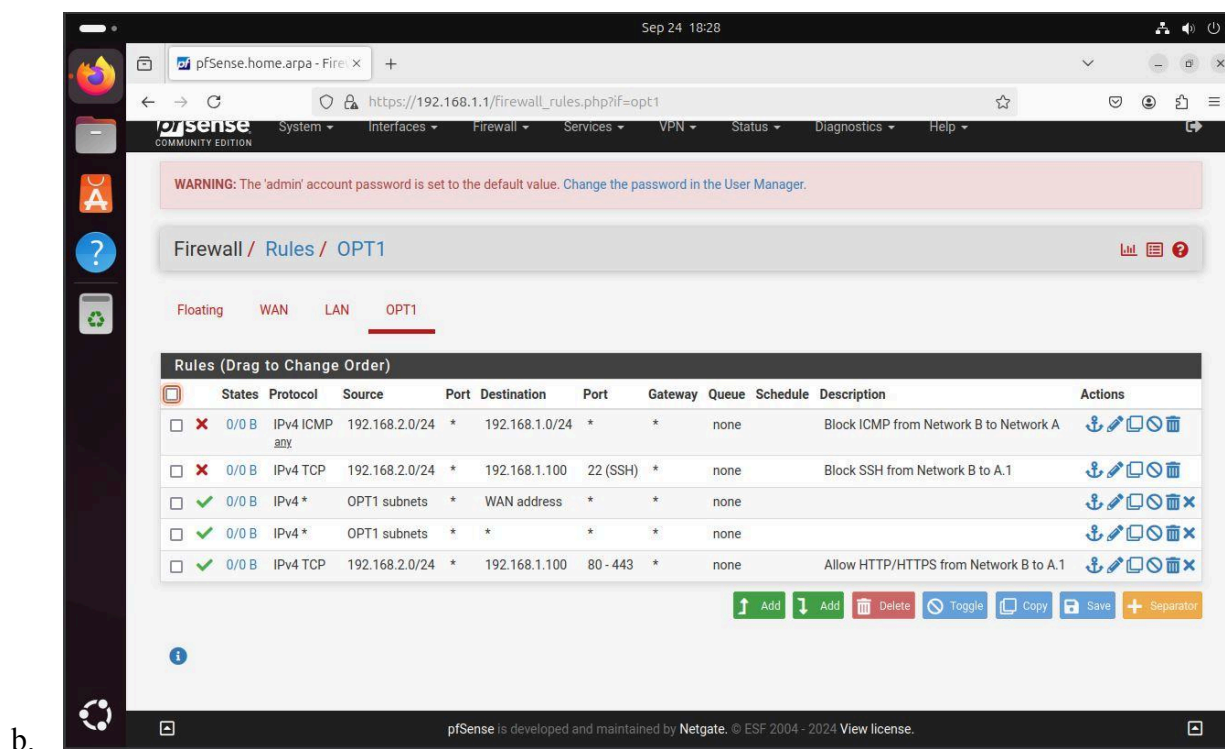


Fig 8:LAN 2 Firewall Rules

c. LAN/Network A Firewall Rules.

- i. Anti-lockout rule: Automatically generated
 - ii. Block Outbound Traffic from A.1 to External Machines: This blocks any outbound traffic from the Ubuntu Server to the external machines on Network B, following the security policy that prohibits A.1 from accessing external machines.
 - iii. Allow A.2 to Access External HTTP/HTTPS: This allows the Windows XP Workstation to access HTTP/S web services on the internet by allowing traffic from 192.168.1.101 (A.2's IP address) to external destinations over ports 80 (HTTP) and 443 (HTTPS).
 - iv. Allow ICMP Pings in Network A: This option allows any host in the 192.168.1.0/24 Network A subnet to send pings, ensuring that everything is working correctly.
 - v. Allow SSH from A.2 to A.1: This rule lets A.2 establish SSH connections to A.1 by opening up TCP traffic on port 22 (SSH) from 192.168.1.101 (A.2) to 192.168.1.100 (A.1).
 - vi. Default Allow LAN to Any Rule: Default pass rule.
 - vii. Block All Traffic from A.2: This rule blocks all traffic coming from 192.168.1.101 (A.2), making sure no other hosts/machines receive traffic from the Windows XP Machine.
- d. OPT1/Network B Firewall Rules.
- i. Block ICMP from Network B to Network A: This rule blocks pings, preventing Network B external machines from pinging the Network A company machines.
 - ii. Block SSH from Network B to A.1: Blocks SSH traffic from Network B to A.1, which prevents the Network B external machines from accessing A.1 via port 22.
 - iii. Allow OPT1 Subnets to WAN: Default rule.
 - iv. Allow HTTP/HTTPS from Network B to A.1: This rule allows external machines in Network B to access A.1's HTTP(S) web services over ports 80 and 443.

4. *NMap Network-A Computers and Ports Screenshots*


```

kali@kali: ~
File Actions Edit View Help
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-23 13:00 EDT
Nmap scan report for 192.168.1.10
Host is up (0.00036s latency).
Not shown: 83 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
81/tcp    closed hosts2-ns
88/tcp    closed kerberos-sec
106/tcp   closed pop3pw
110/tcp   closed pop3
111/tcp   closed rpcbind
113/tcp   closed ident
119/tcp   closed nntp
135/tcp   closed msrpc
139/tcp   closed netbios-ssn
143/tcp   closed imap
144/tcp   closed news
179/tcp   closed bgp
199/tcp   closed smux
389/tcp   closed ldap
427/tcp   closed svrloc
443/tcp   closed https

Nmap done: 256 IP addresses (1 host up) scanned in 22.07 seconds

```

Fig 9: NMAP showing exposed Network A ports after rules

5. Wireshark Routines Screenshots

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
2	1.024000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
3	2.044000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
4	3.060000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
5	4.084000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
6	5.072000000	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42	Who has 192.1
7	5.076000000	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.1 i
8	5.104000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
9	6.132000000	192.168.2.10	192.168.1.0	ICMP	98	Echo (ping) r
10	7.020000000	00000000.080027f104...	00000000.fffffffffff...	IPX SAP	60	General Query
11	20.473923758	192.168.2.10	192.168.1.10	TCP	74	42050 → 80 [S
12	20.472000000	192.168.1.10	192.168.2.10	TCP	74	80 → 42050 [S
13	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
14	20.474943067	192.168.2.10	192.168.1.10	HTTP	141	GET / HTTP/1.
15	20.472000000	192.168.1.10	192.168.2.10	TCP	66	80 → 42050 [A
16	20.472000000	192.168.1.10	192.168.2.10	TCP	2962	80 → 42050 [A
17	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
18	20.472000000	192.168.1.10	192.168.2.10	TCP	4410	80 → 42050 [P
19	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
20	20.472000000	192.168.1.10	192.168.2.10	TCP	1514	80 → 42050 [A
21	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
22	20.472000000	192.168.1.10	192.168.2.10	HTTP	2304	HTTP/1.1 200
23	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
24	20.477035689	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [F
25	20.472000000	192.168.1.10	192.168.2.10	TCP	66	80 → 42050 [F
26	20.472000000	192.168.2.10	192.168.1.10	TCP	66	42050 → 80 [A
27	32.914223031	192.168.2.10	192.168.1.10	TCP	74	42720 → 22 [S
28	33.936415594	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm
29	34.959578206	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm
30	35.984490351	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm
31	37.007650192	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm
32	67.012000000	00000000.080027f104...	00000000.fffffffffff...	IPX SAP	60	Nearest Query

a.

Fig 10: Wireshark Screenshots from B.1 ping/curl/ssh B.1 to A.1 post firewall rules.

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	00000000.080027f104...	00000000.fffffffffff...	IPX SAP	60	General Query
2	4.232000000	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) r
3	5.256000000	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) r
4	9.476000000	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42	Who has 192.1
5	9.476000000	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.1 i
6	9.952000000	192.168.2.10	192.168.1.11	TCP	74	41216 → 80 [S
7	10.984000000	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm
8	12.004000000	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm
9	20.108000000	192.168.2.10	192.168.1.11	TCP	74	53036 → 22 [S
10	21.128000000	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm
11	22.148000000	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm
12	23.176000000	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm

b.

Fig 11: Screenshot from B.1 machine monitoring B.1 ping/curl/ssh to A.1

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) r
2	0.000000000	PCSSystemtec_f1:04:...	Broadcast	ARP	60	Who has 192.1
3	0.000000000	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	192.168.2.10
4	0.000000000	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) r
5	1.016000000	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) r
6	1.020000000	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) r
7	5.232000000	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	Who has 192.1
8	5.232000000	PCSSystemtec_f1:04:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.11
9	6.256000000	192.168.2.10	192.168.2.11	TCP	74	46244 → 80 [S
10	6.264000000	192.168.2.11	192.168.2.10	TCP	60	80 → 46244 [R
11	9.944000000	00000000.080027f104...	00000000.ffffffffffff...	IPX SAP	60	Nearest Query
12	17.864000000	192.168.2.10	192.168.2.11	TCP	74	35506 → 22 [S
13	17.864000000	192.168.2.11	192.168.2.10	TCP	60	22 → 35506 [R

Fig 12: Screenshot from B.1 ping/curl/ssh B.1 to B.2 post firewall

No.	Time	Source	Destination	Protocol	Length	Info
8	6.336222414	192.168.1.10	192.168.1.11	ICMP	74	Echo (ping) reply id=0x
9	7.336204355	192.168.1.11	192.168.1.10	ICMP	74	Echo (ping) request id=0x
10	7.336204355	192.168.1.10	192.168.1.11	ICMP	74	Echo (ping) reply id=0x
11	8.348186080	192.168.1.11	192.168.1.10	ICMP	74	Echo (ping) request id=0x
12	8.348186080	192.168.1.10	192.168.1.11	ICMP	74	Echo (ping) reply id=0x
13	11.510128983	PCSSystemtec_9b:a2:...	PCSSystemtec_32:b0:...	ARP	42	Who has 192.168.1.11? Tell
14	11.511128965	PCSSystemtec_32:b0:...	PCSSystemtec_9b:a2:...	ARP	60	192.168.1.11 is at 08:00:2
15	30.318789480	fe80::a00:27ff:feb0...	ff02::1	ICMPv6	190	Router Advertisement from
16	35.371698312	PCSSystemtec_32:b0:...	Broadcast	ARP	60	Who has 192.168.1.1? Tell
17	38.863635317	192.168.1.11	192.168.1.10	TCP	62	1037 → 80 [SYN] Seq=0 Win=
18	38.863635317	192.168.1.10	192.168.1.11	TCP	62	80 → 1037 [SYN, ACK] Seq=6
19	38.864635299	192.168.1.11	192.168.1.10	TCP	60	1037 → 80 [ACK] Seq=1 Ack=
20	38.864635299	192.168.1.11	192.168.1.10	HTTP	402	GET / HTTP/1.1
21	38.864635299	192.168.1.10	192.168.1.11	TCP	54	80 → 1037 [ACK] Seq=1 Ack=
22	38.864635299	192.168.1.10	192.168.1.11	HTTP	304	HTTP/1.1 304 Not Modified
23	38.869635208	192.168.1.11	192.168.1.10	HTTP	392	GET /icons/ubuntu-logo.png

Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface
 Ethernet II, Src: PCSSystemtec_9b:a2:85 (08:00:27:9b:a2:85), Dst: 08:00:27:9b:a2:85
 Internet Protocol Version 4, Src: 192.168.1.10, Dst: 192.168.1.11
 User Datagram Protocol, Src Port: 48859, Dst Port: 80

- c. Fig 13: Ping/Curl/SSH A.1 monitoring A.1 to A.2
6. *What Web Services are Allowed Between Computers?*
 - a. After implementing the security policies in Task IV to the configuration of the Access Control Matrix, many web services were restricted. The Ubuntu Server now can not access any external web services and only provides HTTP(S) services to the machines in Network B. The XP Workstation can access web services internally and externally, but the Network B machines can not provide access or host their services, same with the XP Workstation.. A.1 continues to to serve web services through port 80 and 443.
7. *Share the Differences Between the Scans from Task III and V*
 - a. The main difference between Task III and V scans is the blocking of pings. Before the security policies, the Kali machine could ping on A.1 and A.2. After the security policies were enforced, all pings from the Kali machine to A.1 and

A.2 are now blocked, with the firewall restricting ICMP traffic between the two networks.

Section IV - Additional Security (Task VI)

1. Local A.1 Router Configuration Rules Screenshots and Explanations

a.

```
zeid@zeid-VirtualBox:~$ sudo ufw status
Status: active

To Action From
--
22/tcp ALLOW Anywhere
80/tcp ALLOW Anywhere
443/tcp REJECT Anywhere
Anywhere ALLOW 192.168.1.10
Anywhere ALLOW 192.168.1.11
Anywhere ALLOW 192.168.2.10
Anywhere ALLOW 192.168.2.11
22/tcp (v6) ALLOW Anywhere (v6)
80/tcp (v6) ALLOW Anywhere (v6)
443/tcp (v6) REJECT Anywhere (v6)
```

Fig 14: Internal Firewall Config

- b. Allow TCP Port 22: allows SSH connections on port 22, giving remote access to the server from any machine.
 - c. Allow TCP Port 80: allows HTTP traffic web services on port 80, permitting web access from any external machine.
 - d. Reject TCP Port 443: blocks port 443 (HTTPS traffic), making web services inaccessible from any external source.
 - e. Allow Access from Specific IPs (192.168.x.x). These rules allow access to the server from specific IP addresses of machines in Networks A and B, ensuring specific machines are accessing their own certain services or resources on the server.
 - f. Allow/Reject IPv6 Rules: Allows SSH and HTTP over IPv6 while HTTPS is blocked.
2. Wireshark Routines Screenshots

it

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.2.10	192.168.1.10	ICMP	98	Echo (ping) r
2	1.009086886	192.168.2.10	192.168.1.10	ICMP	98	Echo (ping) r
3	5.040415810	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42	Who has 192.1
4	5.036589478	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.1 i
5	7.034525616	192.168.2.10	192.168.1.10	TCP	74	41928 → 80 [S
6	7.032589478	192.168.1.10	192.168.2.10	TCP	74	80 → 41928 [S
7	7.032589478	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [A
8	7.035372492	192.168.2.10	192.168.1.10	HTTP	141	GET / HTTP/1.
9	7.032589478	192.168.1.10	192.168.2.10	TCP	66	80 → 41928 [A
10	7.032589478	192.168.1.10	192.168.2.10	TCP	5858	80 → 41928 [A
11	7.032589478	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [A
12	7.032589478	192.168.1.10	192.168.2.10	TCP	1514	80 → 41928 [P
13	7.032589478	192.168.1.10	192.168.2.10	HTTP	3752	HTTP/1.1 200
14	7.032589478	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [A
15	7.032589478	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [A
16	7.037017893	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [F
17	7.032589478	192.168.1.10	192.168.2.10	TCP	66	80 → 41928 [F
18	7.032589478	192.168.2.10	192.168.1.10	TCP	66	41928 → 80 [A
19	11.632589478	192.168.2.10	192.168.1.10	TCP	74	38982 → 22 [S
20	12.664589478	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm
21	13.684589478	192.168.2.10	192.168.1.10	TCP	74	[TCP Retransm

p

A

h

r

p

t

v

[

.

[

a.

Fig 15: B.1 Monitoring B.1 ping/curl/ssh to A.1 post internal firewall update

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) r
2	1.019604308	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) r
3	2.043646262	192.168.2.10	192.168.1.11	ICMP	98	Echo (ping) r
4	4.723615222	192.168.2.10	192.168.1.11	TCP	74	46778 → 80 [S
5	5.180282456	PCSSystemtec_d2:26:...	PCSSystemtec_77:6b:...	ARP	42	Who has 192.1
6	5.179589827	PCSSystemtec_77:6b:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.1 i
7	5.755242231	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm
8	13.263589827	192.168.2.10	192.168.1.11	TCP	74	48392 → 22 [S
9	14.267589827	192.168.2.10	192.168.1.11	TCP	74	[TCP Retransm

b.

Fig 16: B.1 monitoring Ping/Curl/Ssh from B.1 to A.2

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) r
2	0.000000000	PCSSystemtec_f1:04:...	Broadcast	ARP	60	Who has 192.1
3	0.000000000	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	192.168.2.10
4	0.000000000	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) r
5	1.032000000	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) r
6	1.032000000	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) r
7	2.060000000	192.168.2.10	192.168.2.11	ICMP	98	Echo (ping) r
8	2.060000000	192.168.2.11	192.168.2.10	ICMP	98	Echo (ping) r
9	5.032000000	PCSSystemtec_d2:26:...	PCSSystemtec_f1:04:...	ARP	42	Who has 192.1
10	5.032000000	PCSSystemtec_f1:04:...	PCSSystemtec_d2:26:...	ARP	60	192.168.2.11
11	7.288000000	192.168.2.10	192.168.2.11	TCP	74	41266 → 80 [S
12	7.288000000	192.168.2.11	192.168.2.10	TCP	60	80 → 41266 [R
13	11.244000000	192.168.2.10	192.168.2.11	TCP	74	47284 → 22 [S
14	11.244000000	192.168.2.11	192.168.2.10	TCP	60	22 → 47284 [R

Fig 17: B.1 monitoring ping/curl/ssh from B.1 to B.2

3. What Web Services are Allowed Between Computers?

- a. After implementing all of the security policies in the project, including Task VI, the web services between the machines continue to follow the Access Control Matrix, with Ubuntu Server A.1 being the only machine providing web services through ports 80 and 443 for HTTP and HTTPS, respectively. Those web services are only accessible to machines in Network B as per the security guidelines. The XP workstation can still access internal and external web services while still being restricted from hosting its own, the same as the Network B machines.

4. *Share the Differences Between the Scans from Task V and VI*
 - a. We did not find any significant differences between the scans from Task V and Task VI
5. *A.1 Security Policy Discussion*
 - a. Assuming that the company only stores classified business data in A.1, our team agreed that it is reasonably secure but can still be improved. There is a potential vulnerability with the XP Workstation A.2 machine that could become a possible weak link in the system. Since A.2 has access to internal and external web services, it can act as a bridge between the sensitive information on machine A.1 in Network A and the open internet with the external machines in Network B. A malicious actor could exfiltrate data, taking sensitive information from the internal network and funneling it to the external networking, sharing secrets from a private machine to the World Wide Web. Staying with the idea of a malicious actor using A.2 as a “bridge”, an attacker who gains access to A.1 has the ability to explore it and compromise the other machine on the network, A.1, which can be used to bypass the firewall rules set up between Networks A and B.

Section V - Conclusion

1. What We Learned

The project went smoothly and taught us a great deal about working with different operating systems in VMware and setting up a virtual sandbox for practice. Learning to manage the specific requirements of various machines, such as not assigning too much memory or CPU cores to older Windows operating systems, was an important lesson that will help us configure machines more efficiently in the future. The pfSense router tool proved to be extremely useful, allowing us to create a network setup that mimics the behavior of real-world separate networks. This provided an ideal environment to test and apply different firewall rules, as well as simulate attacks to observe their effects. By experimenting with firewall rules, we were able to see how they impacted connectivity in real time. Our main goal was to complete the project efficiently and correctly, while gaining a full understanding of the benefits of creating a virtual environment like this. It was a great refresher on using different operating systems and a fun introduction to deeper concepts in computer security.

2. Obstacles We Overcame

Although there were only a few obstacles, we did encounter some challenges when setting up the machines and router. For the Windows machines, both required a specific amount of RAM and CPU cores to function correctly, which took a combination of internet research and trial-and-error to solve. We also had significant trouble getting Windows 95 to install correctly until we were able to properly install the patch. Once all the machines were up and running, the rest of the process went smoothly until after we applied the first round of firewall rules on pfSense. The external network, for some reason, was unable to access A.1's web service, despite A.1 being able to ping both machines on the external network. We eventually discovered that the issue was caused by the incorrect

order of the firewall rules. After rearranging the rules to ensure they applied in the correct sequence, the web service began functioning properly, everything worked as expected, and we were able to finish up with the rest of the tasks of Project-1.