Project-1: Sandbox, Firewall & Access Control
CS4371/CS5378 Spring 2025

FLR
Reed Kotrla, Robert Jones, Posan Gc

February 10, 2025

1 Introduction

This project focuses on building a sandbox environment for security experiments, building virtual machines, setting up firewalls, and implementing network security policies. The objectives include learning networking tools, analyzing security policies, and verifying secure configurations. The goals of this project was to design and implement a secure network infrastructure that has solid control rules, analyze network traffic before and after an implementation to get a full understanding of the security, by allowing only certain devices access and blocking external pings.

2 Task-II & III: Network Setup and Diagnosis

2.1 Virtual Machine Setup

Below is a screenshot showing the VM setup in the virtual machine manager:

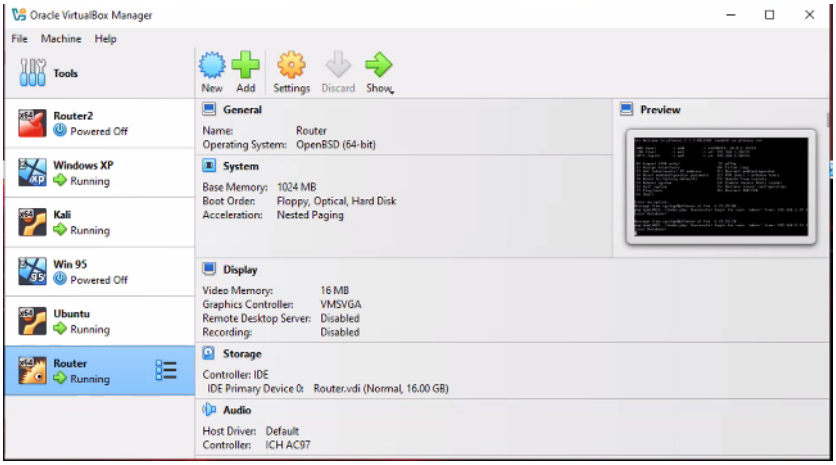


Figure 1: Virtual Machines in Virtual Machine Manager

2.2 NMap Scans

We used NMap to scan network A and B. The commands used are:

```
% Quick Scan of Network A:
nmap -T4 -F 192.168.A.0/24
% Regular Scan of Network A:
nmap 192.168.A.0/24
% Quick Scan of Network B:
nmap -T4 -F 192.168.B.0/24
% Regular Scan of Network B:
nmap 192.168.B.0/24
```

2.3 Wireshark Traffic Analysis

Screenshots of Wireshark captures before implementing security policies:

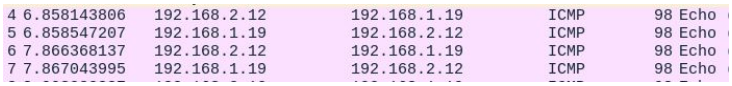


Figure 2: Ping from B.1 to A.1

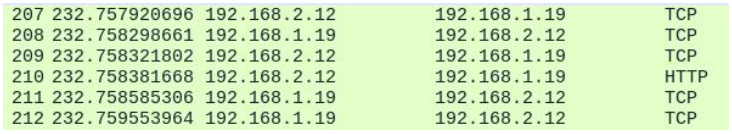


Figure 3: Curl from B.1 to A.1

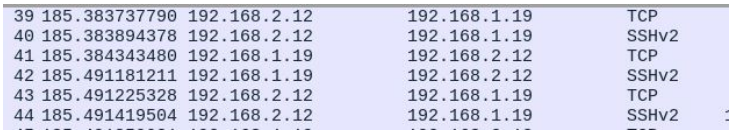


Figure 4: SSH from B.1 to A.1

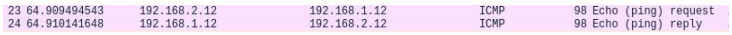


Figure 5: Ping from B.1 to A.2

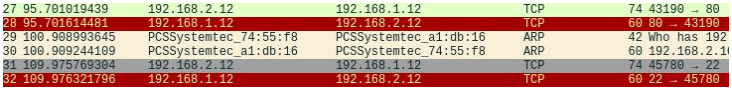


Figure 6: SSH and Curl from B.1 to A.2

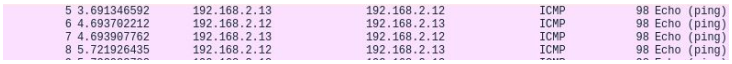


Figure 7: Ping from B.1 to B.2

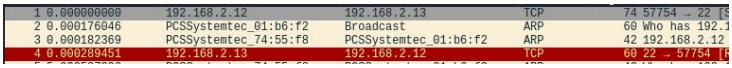


Figure 8: SSH from B.1 to B.2

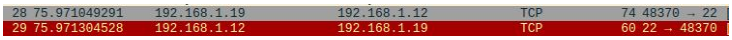


Figure 9: SSH from A.2 to A.1

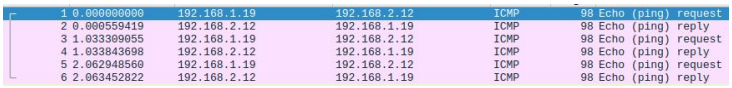


Figure 10: Ping from A.2 to A.1

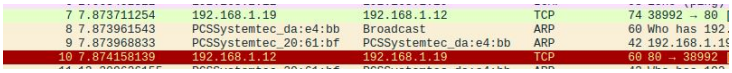


Figure 11: Curl from A.2 to A.1

2.4 Router Configuration

Here are the web services allowed between computers.

```
A) The Ubuntu Server (A.1) provides only web service to
   external computers (in Network B).
(B) The Ubuntu Server (A.1) provides only SSH and web service
    to the companys workstations (A.2).
(C) The Ubuntu Server (A.1) shall not access any services on
   external computers (no outbound to B), except it is
   allowed to ping external hosts (see item G).
(D) The Windows XP Workstations (A.2) shall not provide any
   services.
(E) Workstations (A.2) can access the Ubuntu servers (A.1)
    SSH and web services.
(F) Workstations (A.2) can access only the web service
   provided by external computers through Ubuntu server.
(G) Both the Ubuntu server (A.1) and the XP workstations (A
   .2) can ping any other computers.
(H) External computers cannot ping either the Ubuntu server (
   A.1) or the XP workstations (A.2).  show the web services
   allowed between computers
```

3 Task-IV and V: Access Control and Policy Enforcement

3.1 Access Control Matrix

The access control matrix is shown below, outlining the permitted and restricted interactions between different network entities.

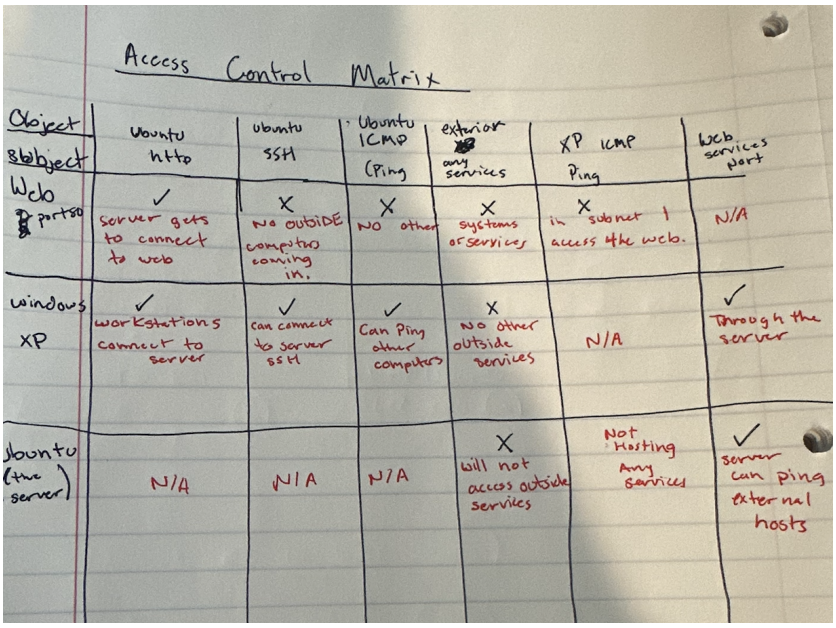


Figure 12: Access Control Matrix

3.2 Policy Limitations

The following policies **CANNOT** be completely enforced by the router rules on R:

- Restricting access based on user authentication levels.
- Preventing unauthorized data exfiltration through encrypted channels.
- Enforcing time-based access restrictions.

3.3 Router Rules on R

Below is a screenshot of the implemented router rules, along with an explanation of each rule's purpose.

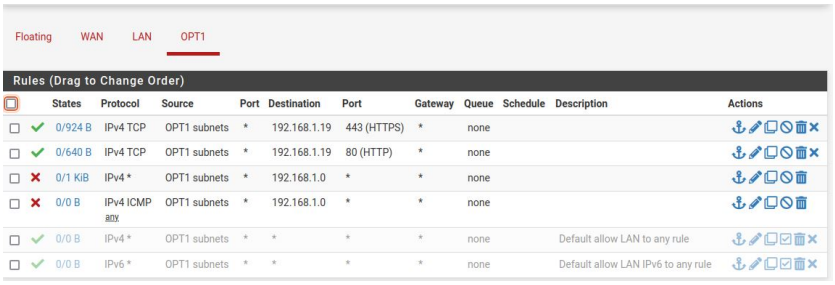


Figure 13: WAN Rules

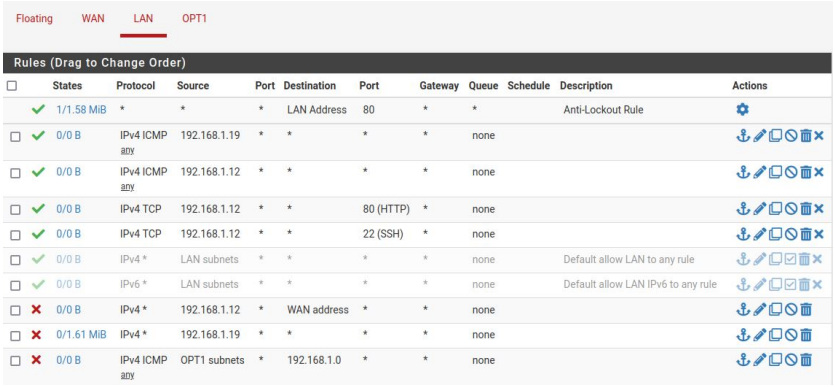


Figure 14: LAN Rules

Rule Purposes

```
LAN Firewall Rules
Allow HTTP traffic to LAN Address (Anti-Lockout Rule).
Allow ICMP (Ping) from 192.168.1.19 to any destination.
Allow ICMP (Ping) from 192.168.1.12 to any destination.
Allow HTTP traffic from 192.168.1.12.
Allow SSH traffic from 192.168.1.12.
Allow all traffic from LAN subnets (default rule).
Block traffic from 192.168.1.12 to WAN Address.
Block all other traffic from 192.168.1.19.
Block ICMP from OPT1 subnets to 192.168.1.0.

WAN Rules
Allow HTTPS traffic from OPT1 subnets to 192.168.1.19.
Allow HTTP traffic from OPT1 subnets to 192.168.1.19.
Block all other IPv4 traffic from OPT1 subnets to
   192.168.1.0.
Block ICMP from OPT1 subnets to 192.168.1.0.
Allow all other traffic from OPT1 subnets (default rule).
```

3.4 NMap Results of Exposed Computers and Ports in Network A

The following screenshots display the results of NMap scans performed on Network A to identify exposed computers and open ports.

3.5 Wireshark Results: Web Service Checks

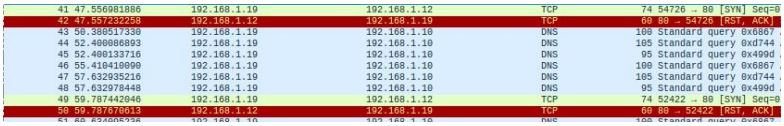
Screenshots of Wireshark captures for web service interactions between different nodes:

```
┌──(victim㉿vbox)-[~]
└─$ ping 192.168.1.19
PING 192.168.1.19 (192.168.1.19) 56(84) bytes of data.
^C
── 192.168.1.19 ping statistics ──
3 packets transmitted, 0 received, 100% packet loss, time 2057ms


┌──(victim㉿vbox)-[~]
└─$ nmap -T4 -F 192.168.1.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-09 23:17 CST
Nmap scan report for 192.168.1.19
Host is up (0.00045s latency).
Not shown: 98 filtered tcp ports (no-response)
PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https

Nmap done: 256 IP addresses (1 host up) scanned in 37.30 seconds

┌──(victim㉿vbox)-[~]
└─$ nmap 192.168.1.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-09 23:18 CST
Nmap scan report for 192.168.1.19
Host is up (0.00044s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT    STATE  SERVICE
80/tcp  open   http
443/tcp closed https

Nmap done: 256 IP addresses (1 host up) scanned in 132.16 seconds
```

Figure 15: NMap Results of Network A

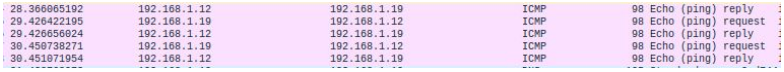
	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	00000000.08002701b6f2	00000000.ffffffffffff	IPX SAP	60	General Query
2	7.899340780	192.168.2.12	192.168.1.19	ICMP	98	Echo (ping) request
3	8.934223523	192.168.2.12	192.168.1.19	ICMP	98	Echo (ping) request
4	9.964265515	192.168.2.12	192.168.1.19	ICMP	98	Echo (ping) request
5	10.994055987	192.168.2.12	192.168.1.19	ICMP	98	Echo (ping) request
6	13.131442950	PCSSystemtec_74:55:f8	PCSSystemtec_a1:db:16	ARP	42	Who has 192.168.2.1
7	13.131616570	PCSSystemtec_a1:db:16	PCSSystemtec_74:55:f8	ARP	60	192.168.2.10 is at
8	26.019692364	192.168.2.12	192.168.1.19	TCP	74	37442 → 22 [SYN] Se
9	27.042447451	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
10	28.068343476	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
11	29.092479660	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
12	30.118112091	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
13	31.143168369	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
14	33.156125171	192.168.2.12	192.168.1.19	TCP	74	[TCP Retransmission
15	58.113581671	192.168.2.12	192.168.1.19	TCP	74	43024 → 80 [SYN] Se
16	58.114047648	192.168.1.19	192.168.2.12	TCP	74	80 → 43024 [SYN, AC
17	58.114068202	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se
18	58.114098294	192.168.2.12	192.168.1.19	HTTP	142	GET / HTTP/1.1
19	58.114352873	192.168.1.19	192.168.2.12	TCP	66	80 → 43024 [ACK] Se
20	58.114811106	192.168.1.19	192.168.2.12	TCP	4410	80 → 43024 [ACK] Se
21	58.114819952	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se
22	58.114855469	192.168.1.19	192.168.2.12	TCP	2962	80 → 43024 [PSH, AC
23	58.114855528	192.168.1.19	192.168.2.12	TCP	2962	80 → 43024 [ACK] Se
24	58.114860101	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se
25	58.114874912	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se
26	58.114897193	192.168.1.19	192.168.2.12	HTTP	856	HTTP/1.1 200 OK (t
27	58.114900479	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se
28	58.115007784	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [FIN, AC
29	58.115331370	192.168.1.19	192.168.2.12	TCP	66	80 → 43024 [FIN, AC
30	58.115337181	192.168.2.12	192.168.1.19	TCP	66	43024 → 80 [ACK] Se

Figure 16: Web service check: B.1 to A.1



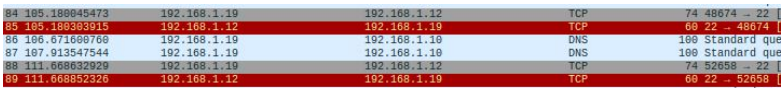
41	47.556981886	192.168.1.19	192.168.1.12	TCP	74 54726 → 80 [SYN] Seq=0
42	47.557232258	192.168.1.12	192.168.1.19	TCP	60 80 → 54726 [RST, ACK]
43	50.380517330	192.168.1.19	192.168.1.10	DNS	100 Standard query 0x6867
44	52.400086893	192.168.1.19	192.168.1.10	DNS	105 Standard query 0xd744
45	52.400133716	192.168.1.19	192.168.1.10	DNS	95 Standard query 0x499d
46	55.410410090	192.168.1.19	192.168.1.10	DNS	100 Standard query 0x6867
47	57.632935216	192.168.1.19	192.168.1.10	DNS	105 Standard query 0xd744
48	57.632978448	192.168.1.19	192.168.1.10	DNS	95 Standard query 0x499d
49	59.787442046	192.168.1.19	192.168.1.12	TCP	74 52422 → 80 [SYN] Seq=0
50	59.787670613	192.168.1.12	192.168.1.19	TCP	60 80 → 52422 [RST, ACK]

Figure 17: Web service check: B.1 to A.2



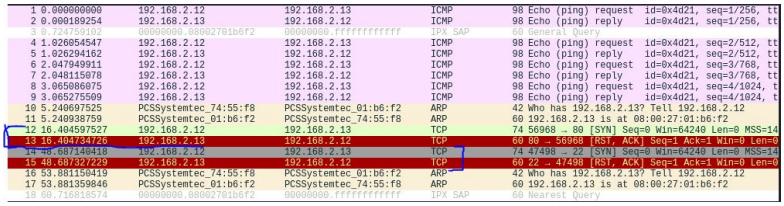
28.366065192	192.168.1.12	192.168.1.19	ICMP	98 Echo (ping) reply
29.426422195	192.168.1.19	192.168.1.12	ICMP	98 Echo (ping) request
29.426656024	192.168.1.12	192.168.1.19	ICMP	98 Echo (ping) reply
30.450738271	192.168.1.19	192.168.1.12	ICMP	98 Echo (ping) request
30.451071954	192.168.1.12	192.168.1.19	ICMP	98 Echo (ping) reply

Figure 18: Web service check: B.1 to B.2



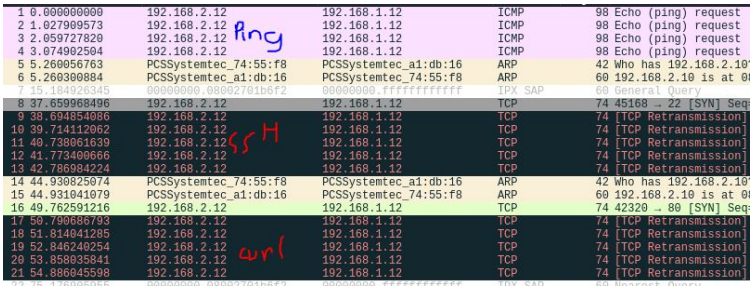
84	105.180045473	192.168.1.19	192.168.1.12	TCP	74 48674 → 22 [
85	105.180303915	192.168.1.12	192.168.1.19	TCP	60 22 → 48674 [
86	106.671600760	192.168.1.19	192.168.1.10	DNS	100 Standard que
87	107.913547544	192.168.1.19	192.168.1.10	DNS	100 Standard que
88	111.668632929	192.168.1.19	192.168.1.12	TCP	74 52658 → 22 [
89	111.668852326	192.168.1.12	192.168.1.19	TCP	60 22 → 52658 [

Figure 19: Web service check: B.1 to A.2



1	0.000000000	192.168.2.12	192.168.2.13	ICMP	98 Echo (ping) request id=0x4d21, seq=1/256, tt
2	0.000189254	192.168.2.13	192.168.2.12	ICMP	98 Echo (ping) reply id=0x4d21, seq=1/256, tt
3	0.724759102	00000000.08002701b6f2	00000000.ffffffffffff	IPX SAP	60 General Query
4	1.026054547	192.168.2.12	192.168.2.13	ICMP	98 Echo (ping) request id=0x4d21, seq=2/512, tt
5	1.026294162	192.168.2.13	192.168.2.12	ICMP	98 Echo (ping) reply id=0x4d21, seq=2/512, tt
6	2.047949911	192.168.2.12	192.168.2.13	ICMP	98 Echo (ping) request id=0x4d21, seq=3/768, tt
7	2.048115078	192.168.2.13	192.168.2.12	ICMP	98 Echo (ping) reply id=0x4d21, seq=3/768, tt
8	3.065086075	192.168.2.12	192.168.2.13	ICMP	98 Echo (ping) request id=0x4d21, seq=4/1024, t
9	3.065275509	192.168.2.13	192.168.2.12	ICMP	98 Echo (ping) reply id=0x4d21, seq=4/1024, t
10	5.240697525	PCSSystemtec_74:55:f8	PCSSystemtec_01:b6:f2	ARP	42 Who has 192.168.2.13? Tell 192.168.2.12
11	5.240938759	PCSSystemtec_01:b6:f2	PCSSystemtec_74:55:f8	ARP	60 192.168.2.13 is at 08:00:27:01:b6:f2
12	16.404597527	192.168.2.12	192.168.2.13	TCP	74 56968 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=14
13	16.404734726	192.168.2.13	192.168.2.12	TCP	60 80 → 56968 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
14	48.687404410	192.168.2.12	192.168.2.13	TCP	74 47498 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=14
15	48.687327229	192.168.2.13	192.168.2.12	TCP	60 22 → 47498 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16	53.881150419	PCSSystemtec_74:55:f8	PCSSystemtec_01:b6:f2	ARP	42 Who has 192.168.2.13? Tell 192.168.2.12
17	53.881359846	PCSSystemtec_01:b6:f2	PCSSystemtec_74:55:f8	ARP	60 192.168.2.13 is at 08:00:27:01:b6:f2
18	60.716816574	00000000.08002701b6f2	00000000.ffffffffffff	IPX SAP	60 Nearest Query

Figure 20: Web service check: B.1 to A.2



1	0.000000000	192.168.2.12	192.168.1.12	ICMP	98 Echo (ping) request
2	1.027909573	192.168.2.12	192.168.1.12	ICMP	98 Echo (ping) request
3	2.059727820	192.168.2.12	192.168.1.12	ICMP	98 Echo (ping) request
4	3.074902504	192.168.2.12	192.168.1.12	ICMP	98 Echo (ping) request
5	5.260056763	PCSSystemtec_74:55:f8	PCSSystemtec_a1:db:16	ARP	42 Who has 192.168.2.10?
6	5.260300884	PCSSystemtec_a1:db:16	PCSSystemtec_74:55:f8	ARP	60 192.168.2.10 is at 0
7	15.104926345	00000000.08002701b6f2	00000000.ffffffffffff	IPX SAP	60 General Query
8	37.659968496	192.168.2.12	192.168.1.12	TCP	74 45168 → 22 [SYN] Seq
9	38.694854086	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
10	39.714112062	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
11	40.738061639	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
12	41.773400666	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
13	42.786984224	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
14	44.930825074	PCSSystemtec_74:55:f8	PCSSystemtec_a1:db:16	ARP	42 Who has 192.168.2.10?
15	44.931041079	PCSSystemtec_a1:db:16	PCSSystemtec_74:55:f8	ARP	60 192.168.2.10 is at 0
16	49.762591216	192.168.2.12	192.168.1.12	TCP	74 42320 → 80 [SYN] Seq
17	50.790686793	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
18	51.814041285	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
19	52.846240254	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
20	53.858035841	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]
21	54.886045598	192.168.2.12	192.168.1.12	TCP	74 [TCP Retransmission]

Figure 21: Web service check: B.1 to A.2

3.6 Allowed Web Services

The web services permitted between different computers in the network are:

- HTTP (port 80) allowed from B.1 to A.1 and A.2
- SSH (port 22) allowed only within Network A
- ICMP requests blocked from B to A, A can still ping to B.

3.7 Differences Between Task-III and Task-V Scans

Key differences observed between the scans conducted in Task-III and Task-V:

- Reduction in exposed open ports due to firewall implementation.
- Blocked ICMP responses after policy enforcement.
- Limited access to certain services that were previously open.

4 Task-VI: Testing Local Router Security Policy

4.1 Local Router Configuration Rules

Below is a screenshot of the firewall rules applied to the local A.1 router and explanations of their functions.

4.2 Wireshark Results: Local Network Web Service Checks

Wireshark captures demonstrating web service interactions within the local network:

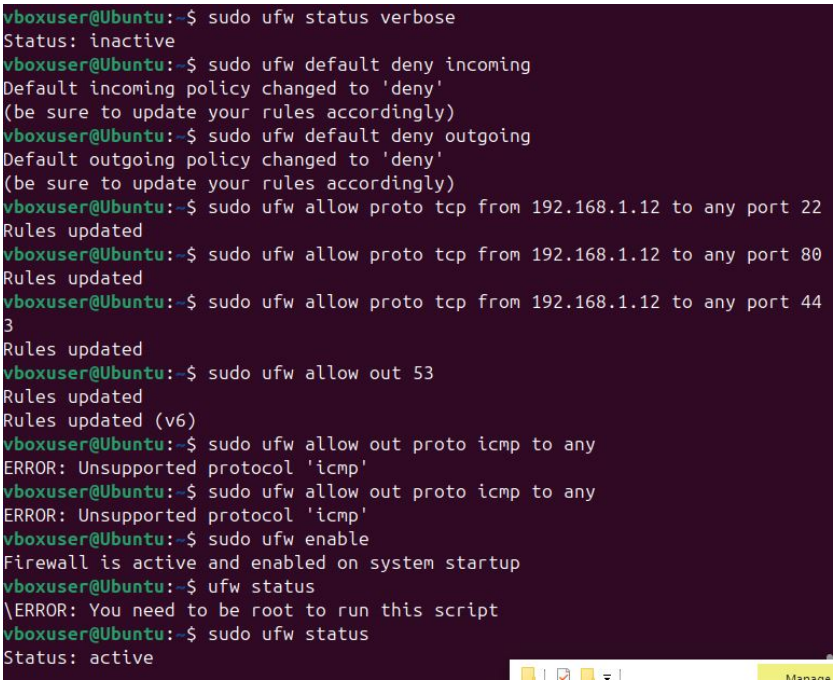


Figure 22: Local Router Rules on A.1

4.3 Allowed Web Services

Updated list of allowed web services between computers:

- HTTP (port 80) remains accessible within permitted nodes.
- SSH (port 22) still limited to specific network segments.
- Additional restrictions on unauthorized access to classified data.

4.4 Differences Between Task-V and Task-VI Scans

Comparison of security policy enforcement between Task-V and Task-VI:

- Strengthened access controls in Task-VI.
- Stricter policy enforcement leading to fewer detected open ports.
- Enhanced logging and tracking mechanisms in Task-VI implementation.

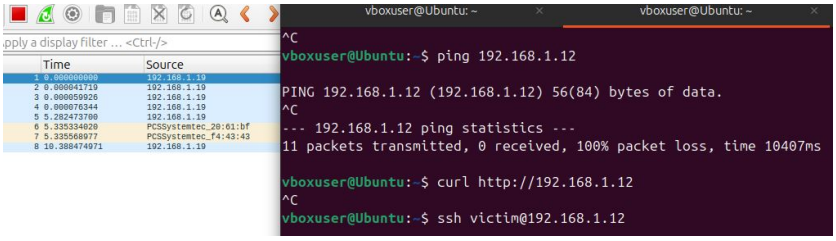


Figure 23: Web service check: B.1 to B.2

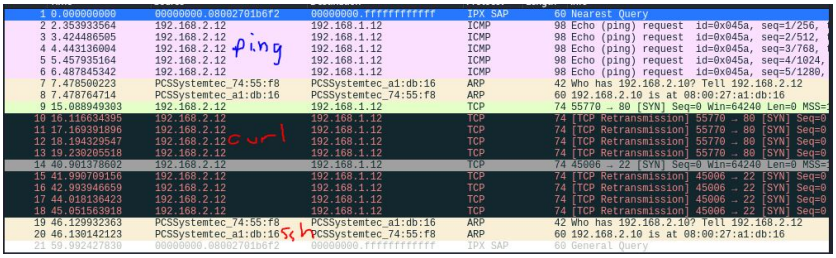


Figure 24: Web service check: B.1 to A.1 (Local Network)

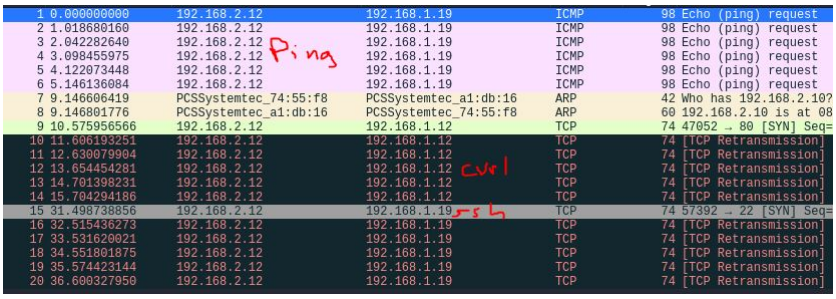


Figure 25: Web service check: B.1 to B.2

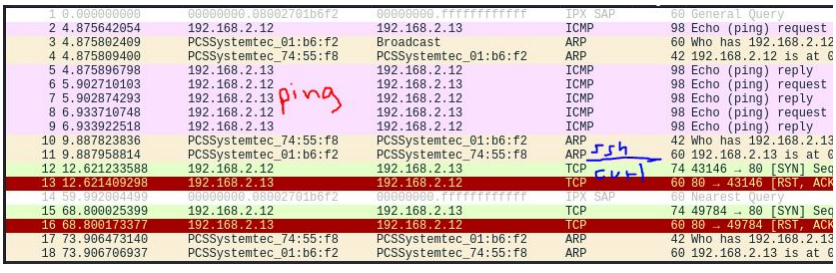


Figure 26: Web service check: B.1 to A.1 (Local Network)

4.5 Security Policy Effectiveness in Protecting Classified Data

The security policy aims to prevent classified data leaks from Computer A.1. However, potential vulnerabilities include:

- Insider threats—authorized users transferring data manually.
- Lack of monitoring on encrypted traffic.
- Missing policies for data loss prevention (DLP) strategies.

Recommendations for stronger enforcement:

- Implement stricter logging and access controls.
- Use encryption for stored and transmitted data.
- Deploy intrusion detection systems (IDS) to monitor abnormal access patterns.

5 Conclusion

This project provided hands-on experience in setting up a virtual sandbox, configuring network security, and analyzing traffic. Challenges faced included network configuration issues and firewall misconfigurations, which were resolved through debugging and testing. The project successfully enforced a security policy to control network traffic between internal and external systems. One of the key takeaways from this project was the importance of layered security. Layered security is incredibly important because simply blocking or allowing traffic through a firewall isn't enough. A well structured security policy must take into consideration for outbound and inbound rules, service restrictions, and any potential attacks. By limiting external access to only essential services (such as HTTP to the web server) and restricting the internal workstations from exposing services, we successfully minimized the attack surface of the network. Seeing how the pre to post security scans were also provided a lot of insight to us as in how a firewall could be manipulated and exposed by attackers. The Access Control Matrix and new router rules helped in creating strong boundaries so there wouldn't be any unauthorized access nor data leaks. Overall this project really helped in strengthening our skills in network security, access control and monitoring traffic. Doing a project like this which can be easily applied in a real life setting taught us the importance of having a secure network. For the future it would be interesting to test attack scenarios and improving our firewall policies