



Lab 5 - Middleboxes

Group D :

BIENFAIT Alexandre	513930
BOISTEL Julien	440915
HUMBLET Raphaël	514085
MUTKOWSKI Philippe	494470

Teacher :

DRICOT Jean-Michel

Teaching Assistant :

CHASSAGNE Aurélien

December 2023

Academic year 2023-2024

Contents

1	Mission 1 - NAT	2
1.1	Mission 1.1 - Configuration	2
1.1.1	Current knowledge of NAT	2
1.1.2	Configuration of the routers	3
1.2	Mission 1.2 - Analysis	3
1.2.1	Ping a	3
1.2.2	Ping b	4
1.2.3	Ping c	4
1.2.4	Ping d	5
1.2.5	Ping e	6
1.2.6	Ping f	6
1.2.7	Ping g	7
1.3	NAT - Improved knowledge	7
2	Mission 2 - Firewall	9
2.1	Knowledge before labs	9
2.2	Blacklist VS Whitelist	9
2.3	Accept ping (ICMP) from PC1 but not PC2	10
2.4	Refuse UDP communication unless intended for port 80 of PC4	10
2.5	PC3 accepts all UDP communication from subnet with PC1 and PC2	11
2.6	Mission 2.3 - Protection against data exfiltration	12
2.7	Firewall conclusion	13
3	Conclusion	13

The objective of this lab is to understand Middleboxes (here NAT and firewall), and how NAT and firewalls work by configuring them on the following topology. Those will be implemented on the routers names **Firewall** and **NAT**.

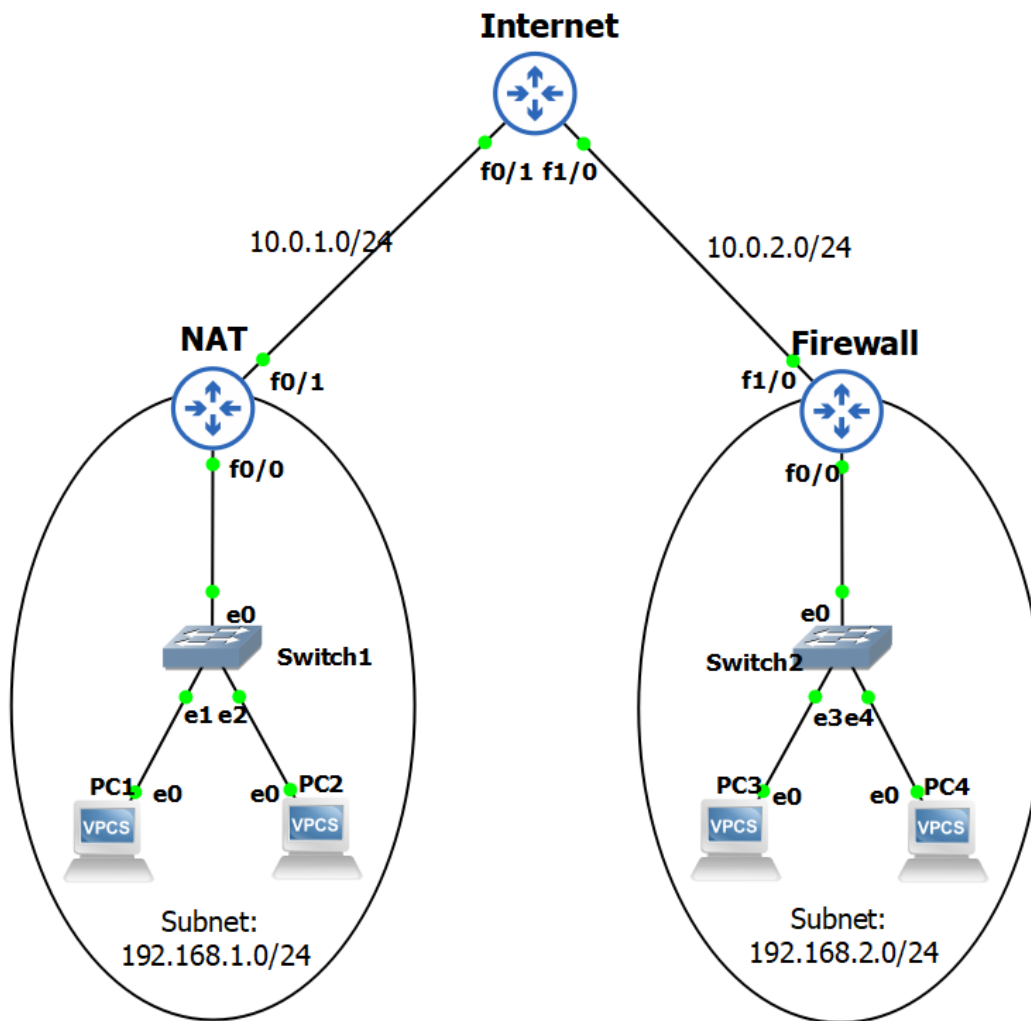


Figure 1: Initial topology

1 Mission 1 - NAT

1.1 Mission 1.1 - Configuration

1.1.1 Current knowledge of NAT

Before doing the mission, nobody in the group knows what is NAT and how it works. But after a quick research, NAT (Network Address Translation) is a technique used to map private IP addresses of devices to share a single public IP for accessing the internet.

NAT works by modifying the source or destination IP addresses in network traffic, allowing thus private addresses to be concealed behind a public address.

It looks like it plays an important role in enhancing the network security while conserving public IP addresses.

1.1.2 Configuration of the routers

First, the inside of the NAT needs to be configured. In the case of this lab, the inside is linked to the FastEthernet 0/0 (subnet 192.168.1.0) and the outside is linked to the interface FastEthernet 1/0.

```
Pro Inside global    Inside local    Outside local    Outside global
udp 10.0.1.1:1111    192.168.1.2:80  ---            ---
udp 10.0.1.1:2222    192.168.1.3:80  ---            ---
NAT#show ip nat statistics
Total active translations: 2 (2 static, 0 dynamic; 2 extended)
Outside interfaces:
  FastEthernet0/1
Inside interfaces:
  FastEthernet0/0
Hits: 0 Misses: 0
CEF Translated packets: 0, CEF Punted packets: 0
Expired translations: 0
Dynamic mappings:
Appl doors: 0
Normal doors: 0
Queued Packets: 0
```

Figure 2: NAT Translation table and statistics

It can be seen that the 2 PCs of the NAT subnet (192.168.1.0/24) have an inside local IP address that differs but they are seen from outside the NAT as a same IP address, only on different ports (1111 for PC1 and 2222 for PC2).

1.2 Mission 1.2 - Analysis

1.2.1 Ping a

The objective here is to try to ping PC4 from PC1 at the source port 8080 and destination port 80. The protocol used is UDP (-P 17).

```
PC1>>> ping 192.168.2.3 -P 17 -p 80 -s 8080
```

As it can be seen on figure 3, the source and destination of the packets inside and outside are not changed by the NAT router. This is because the source port (specified by -s 8080) is on 8080 which is not defined in the NAT translation table. Therefore, the packets will not be affected by the NAT to go from the inside to the outside of the NAT.

Inside						Outside					
12	60.489443	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56	17	52.501621	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56
13	60.563634	192.168.1.2	192.168.1.2	UDP	98 80 → 8080 Len=56	18	52.501756	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply
14	61.566032	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56	19	52.553842	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56
15	61.611155	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56	20	53.571281	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56
16	62.592216	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	21	53.601869	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56
17	62.612879	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56	22	54.592858	c4:03:0b:80:00:01	c4:03:0b:80:00:01	LOOP	60 Reply
18	62.666645	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56	23	54.624260	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56
19	63.669129	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56	24	54.656755	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56
20	63.707691	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56	25	55.677284	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56
21	64.709547	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56	26	55.706805	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56
22	64.771954	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56	27	56.720869	192.168.1.2	192.168.2.3	UDP	98 8080 → 80 Len=56
23	72.818541	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	28	56.762060	192.168.2.3	192.168.1.2	UDP	98 80 → 8080 Len=56
24	72.818541	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	29	62.359885	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply
25	83.170198	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	30	64.819457	c4:03:0b:80:00:01	c4:03:0b:80:00:01	LOOP	60 Reply
Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0						Frame 20: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0					
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: c4:03:0b:80:00:00						Ethernet II, Src: c4:03:0b:80:00:01 (c4:03:0b:80:00:01), Dst: c4:05:0b:b8:00:00					
Configuration Test Protocol (loopback)						Internet Protocol Version 4, Src: 192.168.1.2, Dst: 192.168.2.3					
Data (40 bytes)						User Datagram Protocol, Src Port: 8080, Dst Port: 80					
						Data (56 bytes)					

Figure 3: Ping from PC1 to PC4 on port 8080 - inside the NAT (left) - outside the NAT (right)

1.2.2 Ping b

Here, a ping from PC1 to PC4 (from inside the NAT to the outside of the NAT).

```
>>> PC1: ping 192.168.2.3 -P 17 -p 80 -s 80
```

As it can be seen, the NAT worked. Inside the NAT (from PC1 to NAT router), the source and destination are the private IP addresses with the right port numbers. Outside the NAT, the IP address of PC1 has become the public address of NAT with the port number 1111 that is assigned to PC1 in the translation table.

This means that outside of the NAT, the destination does not know the private IP address of PC1 (source), which is the objective of NAT.

Inside						Outside					
110	638.746643	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	199	638.756664	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
111	638.799113	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	200	638.790146	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
112	639.801188	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	201	631.810000	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
113	639.851478	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	202	631.841253	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
114	640.852875	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	203	631.852218	c4:05:0b:b8:00:01	COP/VTP/JDTP/PagP/UDL	COP	356 Device ID: Intern
115	640.905564	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	204	632.864502	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
116	641.907293	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	205	632.904850	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
117	641.941816	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	206	633.911797	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
118	642.945129	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	207	633.937969	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
119	643.001864	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	208	634.949743	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
120	647.035806	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	209	634.992012	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
Frame 199: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						Frame 199: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0					
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: c4:03:0b:80:00:00						Ethernet II, Src: c4:03:0b:80:00:01 (c4:03:0b:80:00:01), Dst: c4:05:0b:b8:00:00					
Configuration Test Protocol (loopback)						Internet Protocol Version 4, Src: 10.0.1.1, Dst: 192.168.2.3					
Data (40 bytes)						User Datagram Protocol, Src Port: 1111, Dst Port: 80					
						Data (56 bytes)					

Figure 4: Ping from PC1 to PC4 on port 80 - inside the NAT (left) - outside the NAT (right)

1.2.3 Ping c

The Fig 5 is the Wireshark capture of the

```
>>> PC2: ping 192.168.2.3 -P 17 -p 80 -s 80
```

The packet appears as malformed on Wireshark because the port defined for the PC2 on the NAT is port 2222, which is a port that was already predefined for CIP I/O (Ethernet packet exchange). Wireshark can then identify that the packet does

not have the expected form. The problem did not appear on Fig 4 because the port 1111 is not predefined.

Inside						Outside					
192.1048.027776	192.168.1.3	192.168.2.3	UDP	98 80 → 80	Len=56	339.1040.037009	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80	Len=56[Mal]
193.1048.067554	192.168.2.3	192.168.1.3	UDP	98 80 → 80	Len=56	340.1040.060158	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222	Len=56[Mal]
194.1049.069370	192.168.1.3	192.168.2.3	UDP	98 80 → 80	Len=56	341.1041.071844	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80	Len=56[Mal]
195.1049.125470	192.168.2.3	192.168.1.3	UDP	98 80 → 80	Len=56	342.1041.115862	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222	Len=56[Mal]
196.1050.127544	192.168.1.3	192.168.2.3	UDP	98 80 → 80	Len=56	343.1042.136177	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80	Len=56[Mal]
197.1050.187466	192.168.2.3	192.168.1.3	UDP	98 80 → 80	Len=56	344.1042.178406	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222	Len=56[Mal]
198.1051.189819	192.168.1.3	192.168.2.3	UDP	98 80 → 80	Len=56	345.1043.191650	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80	Len=56[Mal]
199.1051.232056	192.168.2.3	192.168.1.3	UDP	98 80 → 80	Len=56	346.1043.222084	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222	Len=56[Mal]
200.1052.237895	192.168.1.3	192.168.2.3	UDP	98 80 → 80	Len=56	347.1043.860354	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply	
201.1052.299997	192.168.2.3	192.168.1.3	UDP	98 80 → 80	Len=56	348.1044.250157	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80	Len=56[Mal]
202.1055.431805	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		349.1044.290901	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222	Len=56[Mal]
203.1061.912444	192.168.1.1	224.0.0.9	RIPv2	86 Response		350.1047.432606	c4:03:0b:80:00:01	c4:03:0b:80:00:01	LOOP	60 Reply	
304.1067.437003	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		351.1051.523392	10.0.1.1	224.0.0.9	RIPv2	66 Response	
305.1067.437003	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		352.1054.333197	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply	
Frame 192: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						Frame 339: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0					
Ethernet II, Src: Private_66:68:01 (00:50:79:66:68:01), Dst: c4:03:0b:80:00:00						Ethernet II, Src: c4:03:0b:80:00:01 (c4:03:0b:80:00:01), Dst: c4:05:0b:b8:00:00					
Internet Protocol Version 4, Src: 192.168.1.3, Dst: 192.168.2.3						Internet Protocol Version 4, Src: 10.0.1.1, Dst: 192.168.2.3					
User Datagram Protocol, Src Port: 80, Dst Port: 80						User Datagram Protocol, Src Port: 2222, Dst Port: 80					
Data (56 bytes)						Ethernet/IP (Industrial Protocol)					
						[Malformed Packet: CIP I/O]					

Figure 5: Ping from PC2 to PC4 on port 80 - inside the NAT (left) - outside the NAT (right)

1.2.4 Ping d

Here, PC4 sends a ping to PC1 at source and destination ports 80 by using UDP protocols:

```
>>> PC4: ping 192.168.1.2 -P 17 -p 80 -s 80
```

Ping shows timeout (figure 7) even though the packets arrived at the destination. This is because the ping arrives to the destination (private address of PC1), but the replied ping comes from another source (public address of PC1). As it can be seen on figure 6, on the inside of NAT, the IP are the same as asked (from PC4 to PC1). But outside the NAT, the destination is 192.168.1.2 for the request but for the reply, it becomes 10.0.1.1 which is the public address of the NAT.

Inside						Outside					
343.1924.783610	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56	617.1916.774471	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56
344.1924.783692	192.168.1.2	192.168.2.3	UDP	98 80 → 80	Len=56	618.1916.795175	10.0.1.1	192.168.2.3	UDP	98 1111 → 80	Len=56
345.1926.784032	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56	619.1918.776904	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56
346.1926.784133	192.168.1.2	192.168.2.3	UDP	98 80 → 80	Len=56	620.1918.795253	10.0.1.1	192.168.2.3	UDP	98 1111 → 80	Len=56
347.1928.791279	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56	621.1919.218071	10.0.1.1	224.0.0.9	RIPv2	66 Response	
348.1928.791532	192.168.1.2	192.168.2.3	UDP	98 80 → 80	Len=56	622.1920.549543	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply	
349.1929.005174	192.168.1.1	224.0.0.9	RIPv2	86 Response		623.1920.782059	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56
350.1930.789198	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56	624.1920.803547	10.0.1.1	192.168.2.3	UDP	98 1111 → 80	Len=56
351.1930.789274	192.168.1.2	192.168.2.3	UDP	98 80 → 80	Len=56	625.1922.780022	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56
352.1932.493003	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		626.1922.800957	10.0.1.1	192.168.2.3	UDP	98 1111 → 80	Len=56
353.1932.792248	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56	627.1924.504299	c4:03:0b:80:00:01	c4:03:0b:80:00:01	LOOP	60 Reply	
354.1932.792646	192.168.1.2	192.168.2.3	UDP	98 80 → 80	Len=56	628.1924.782611	192.168.2.3	192.168.1.2	UDP	98 80 → 80	Len=56
355.1943.030327	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		629.1924.803354	10.0.1.1	192.168.2.3	UDP	98 1111 → 80	Len=56
356.1943.030327	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply		630.1930.812536	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply	
Frame 343: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						Frame 617: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0					
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: Private_66:68:01						Ethernet II, Src: c4:05:0b:b8:00:01 (c4:05:0b:b8:00:01), Dst: c4:03:0b:80:00:00					
Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2						Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2					
User Datagram Protocol, Src Port: 80, Dst Port: 80						User Datagram Protocol, Src Port: 80, Dst Port: 80					
Data (56 bytes)						Data (56 bytes)					

Figure 6: Ping from PC4 to PC1 on port 80 - inside the NAT (left) - outside the NAT (right)

```
PC4> ping 192.168.1.2 -P 17 -p 80 -s 80

192.168.1.2 udp_seq=1 timeout
192.168.1.2 udp_seq=2 timeout
192.168.1.2 udp_seq=3 timeout
192.168.1.2 udp_seq=4 timeout
192.168.1.2 udp_seq=5 timeout
```

Figure 7: Ping from PC4 to PC1 timeout

But in real life, the packets would probably not arrive at the destination because someone that is outside the NAT does not have the address of a PC inside the NAT.

Another case that could happen is that PCs inside a NAT can have the same IP address as another PC from the internet. So in real life, it will not work most of the time due to either one of those two reasons.

1.2.5 Ping e

This is a ping from PC4 to the private address of PC1 on the destination port number 1111.

```
>>> PC4: ping 192.168.1.2 -P 17 -p 1111 -s 80
```

Inside						Outside					
426	2407.132223	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56	769	2399.110406	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56
427	2408.192854	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56	770	2399.143026	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56
428	2408.192944	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56	771	2400.183745	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56
429	2409.249302	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56	772	2400.204447	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56
430	2409.249648	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56	773	2401.239957	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56
431	2410.292343	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56	774	2401.260778	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56
432	2410.292498	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56	775	2402.057531	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply
433	2411.337823	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56	776	2402.286474	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56
434	2411.337892	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56	777	2402.303908	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56
435	2413.541265	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	778	2403.337123	192.168.2.3	192.168.1.2	UDP	98 80 → 1111 Len=56
436	2417.920687	c4:03:0b:80:00:00	CDP/VTP/DTP/PagP/UD...	CDP	351 Device ID: NAT Port ID	779	2403.349067	192.168.1.2	192.168.2.3	UDP	98 1111 → 80 Len=56
Frame 343: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0						Frame 769: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0					
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: Private_66:66:66						Ethernet II, Src: c4:05:0b:b8:00:01 (c4:05:0b:b8:00:01), Dst: c4:03:0b:80:00:00					
Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2						Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2					
User Datagram Protocol, Src Port: 80, Dst Port: 80						User Datagram Protocol, Src Port: 80, Dst Port: 1111					
Data (56 bytes)						Data (56 bytes)					

Figure 8: Ping from PC4 to PC1 on port 1111 - inside the NAT (left) - outside the NAT (right)

The ping was successful, but it's not good practice. PC4 should not have access to the private IP address of PC1. Also, the address of PC1 could be used elsewhere on the internet (as explained in section 1.2.4).

1.2.6 Ping f

This is a ping from PC4 to the public address of PC1 (NAT public address) on the destination port number 1111.

```
>>> PC4: ping 10.0.1.1 -P 17 -p 1111 -s 80
```

Inside					Outside				
483 2644.781740	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	859 2636.772464	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
484 2644.781842	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	860 2636.793053	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
485 2645.845063	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	861 2637.835688	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
486 2645.845221	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	862 2637.856100	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
487 2646.907894	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	863 2638.898457	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
488 2646.907996	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	864 2638.919999	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
489 2648.000383	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	865 2639.987276	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
490 2648.000600	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	866 2640.012224	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56
491 2649.073630	192.168.2.3	192.168.1.2	UDP	98 80 → 80 Len=56	867 2641.056663	192.168.2.3	10.0.1.1	UDP	98 80 → 1111 Len=56
492 2649.073787	192.168.1.2	192.168.2.3	UDP	98 80 → 80 Len=56	868 2641.084642	10.0.1.1	192.168.2.3	UDP	98 1111 → 80 Len=56

Frame 343: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0	0000 00 50 79 66	Frame 769: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0	0000 c4 03 0b 8e
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: Private_66:68:	0010 00 54 cf 2c	Ethernet II, Src: c4:05:0b:b8:00:01 (c4:05:0b:b8:00:01), Dst: c4:03:0b:80:00:	0010 00 54 d1 0e
Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2	0020 01 02 00 50	Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.2	0020 01 02 00 56
User Datagram Protocol, Src Port: 80, Dst Port: 80	0030 0e 0f 10 11	User Datagram Protocol, Src Port: 80, Dst Port: 1111	0030 0e 0f 10 11
Data (56 bytes)	0040 1e 1f 20 21	Data (56 bytes)	0040 1e 1f 20 21
	0050 2e 2f 30 31		

Figure 9: Ping from PC4 to NAT on port 1111 - inside the NAT (left) - outside the NAT (right)

Here, the ping is done correctly. PC4 reaches the NAT first, then the NAT translates the public address to the private address thanks to the port number. It allows then to let the ping go to the correct PC. As it can be seen on the inside, the destination address is PC1 with port 80 and on the outside, the destination address is NAT with port 1111 (linked to port 80!).

1.2.7 Ping g

This is a ping from PC4 to NAT public address on port 2222 (which is linked to PC2 private address).

```
>>> PC4: ping 10.0.1.1 -P 17 -p 2222 -s 80
```

Inside					Outside				
597 3123.832386	192.168.2.3	192.168.1.3	UDP	98 80 → 80 Len=56	1020 3115.822832	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222 Len=56[Malform
598 3123.832697	192.168.1.3	192.168.2.3	UDP	98 80 → 80 Len=56	1021 3115.843455	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80 Len=56[Malform
599 3124.906179	192.168.2.3	192.168.1.3	UDP	98 80 → 80 Len=56	1022 3116.281932	c4:05:0b:b8:00:01	c4:05:0b:b8:00:01	LOOP	60 Reply
600 3124.906509	192.168.1.3	192.168.2.3	UDP	98 80 → 80 Len=56	1023 3116.896293	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222 Len=56[Malform
601 3125.968858	192.168.2.3	192.168.1.3	UDP	98 80 → 80 Len=56	1024 3116.917331	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80 Len=56[Malform
602 3125.968992	192.168.1.3	192.168.2.3	UDP	98 80 → 80 Len=56	1025 3117.956448	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222 Len=56[Malform
603 3126.720538	c4:03:0b:80:00:00	c4:03:0b:80:00:00	LOOP	60 Reply	1026 3117.981485	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80 Len=56[Malform
604 3127.019739	192.168.2.3	192.168.1.3	UDP	98 80 → 80 Len=56	1027 3118.736066	c4:03:0b:80:00:01	c4:03:0b:80:00:01	LOOP	60 Reply
605 3127.019874	192.168.1.3	192.168.2.3	UDP	98 80 → 80 Len=56	1028 3119.019567	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222 Len=56[Malform
606 3128.103482	192.168.2.3	192.168.1.3	UDP	98 80 → 80 Len=56	1029 3119.035848	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80 Len=56[Malform
607 3128.103874	192.168.1.3	192.168.2.3	UDP	98 80 → 80 Len=56	1030 3120.002374	192.168.2.3	10.0.1.1	CIP I/O	98 80 → 2222 Len=56[Malform
					1031 3120.123611	10.0.1.1	192.168.2.3	CIP I/O	98 2222 → 80 Len=56[Malform

Frame 597: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0	0000 00 50 7	Frame 1001: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0	0000 c4 03 0b 80 00 01
Ethernet II, Src: c4:03:0b:80:00:00 (c4:03:0b:80:00:00), Dst: Private_66:68:	0010 00 54 d	Ethernet II, Src: c4:03:0b:80:00:01 (c4:03:0b:80:00:01), Dst: c4:03:0b:80:00:	0010 01 00 00 00 00
Internet Protocol Version 4, Src: 192.168.2.3, Dst: 192.168.1.3	0020 01 03 0	Configuration Test Protocol (loopback)	0020 00 00 00 00 00
User Datagram Protocol, Src Port: 80, Dst Port: 80	0030 0e 0f 1		0030 00 00 00 00 00
Data (56 bytes)	0040 1e 1f 2		
	0050 2e 2f 3		

Figure 10: Ping from PC4 to NAT on port 2222 - inside the NAT (left) - outside the NAT (right)

Port 2222 is already used. So the protocol used is not the same inside (UDP) and outside (CIP I/O) the NAT system. The message will arrive, but maybe not in the right form due to the change of protocol.

1.3 NAT - Improved knowledge

A NAT configured on a router allows the concealing of precise IP addresses of a subnet's devices. The packets can still be transmitted to these devices by creating a correspondence between a NAT's port and a device IP address. Those addresses can then stay private and the devices can only share a single public address outside the subnet.

NAT maintains the translation table that records the mapping between private IP addresses and ports to a public IP address and ports.

Thanks to NAT, the external hosts can not communicate directly a host of the network "protected" by NAT by using the private IP address of the destination host. But someone in the NAT "covered" network can send to anyone its messages without them knowing the private address of the source.

NAT is useful because it allows:

- **Address conservation:** it enables the conservation of public addresses by allowing multiple devices within a private network to share a single public IP address.
- **Security enhancement:** provides a level of security by hiding the internal network structure from external entities. The use of private IP addresses internally and a single public IP address externally adds a layer of obscurity, making it more challenging for external entities to directly access internal devices.
- **Network flexibility:** it allows organizations to renumber their internal networks without affecting external communication. Internal devices can use private IP addresses, and the NAT device handles the translation to the public IP address when communicating with external networks.

But it also has its disadvantages:

- **Application layer issues**
- **Complexity and maintenance:** it adds complexity to the network configurations and maintaining the translation tables, handling special cases and troubleshooting connectivity issues can require additional efforts.
- **End-to-End communication challenges :** it may be challenging for external devices to initiate connections to internal devices due to the dynamic nature of port assignments and address translations performed by the NAT device.

2 Mission 2 - Firewall

Note that here, a new topology has been used for both missions. But NAT and firewalls could easily co-exist to have a more secure network.

2.1 Knowledge before labs

A firewall is a wall of fire.



Figure 11: Supa dupa firewall

In case Figure 11 wasn't clear enough, a firewall is a security equipment that monitors the activity and blocks traffic based on rules specified by the admin. Usually, the firewall is placed between a trusted network and suspicious outside networks.

2.2 Blacklist VS Whitelist

There are two main strategies when creating a firewall. On one side, it is possible to block all packets except the ones coming from or going to desired addresses: these are stored in a list called **whitelist**. On the other side, it is also possible allow all communication to a subnet except some types arriving from suspicious addresses for example: those suspicious communications will be blocked when in the **blacklist**. Both strategies have their advantages and disadvantages.

Whitelist:

- **Pros:** High security, only trusted sources are allowed
- **Cons:** Requires constant updates as new entities needs to be manually added or removed

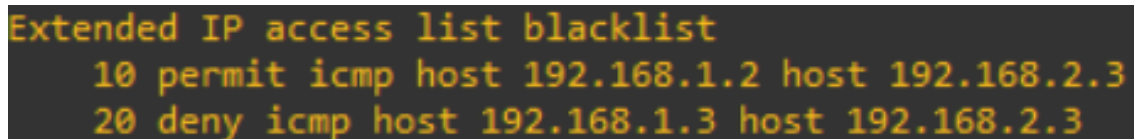
Blacklist:

- **Pros:** Simplicity, only known threats are blocked
- **Cons:** Incomplete protection, relying only on known threats condemn the firewall to let some malicious packets arrive before they are identified as threats

2.3 Accept ping (ICMP) from PC1 but not PC2

Here the objective is to accept pings from PC1 (192.168.1.2) and not from PC2 (192.168.1.3)

```
>>> permit icmp host 192.168.1.2 host 192.168.2.3
>>> deny icmp host 192.168.1.3 host 192.168.2.3
```



```
Extended IP access list blacklist
10 permit icmp host 192.168.1.2 host 192.168.2.3
20 deny icmp host 192.168.1.3 host 192.168.2.3
```

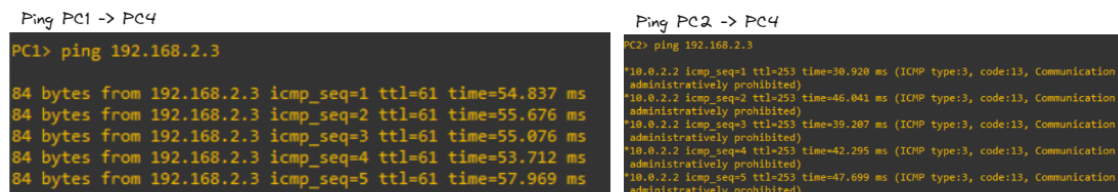
Figure 12: Configuration of the firewall for step 1

Note that here it is only configured for messages that go to PC4 instead of the entire subnet, it should not change the way it works because here only pings to PC4 are done for the testing.

If the objective is to protect the entire subnet, the command would have been:

```
>>> permit icmp host 192.168.1.2 192.168.2.0 0.0.0.255
>>> deny icmp icmp host 192.168.1.3 192.168.2.0 0.0.0.255
```

The figure 13 shows that pings from PC1 to PC4 succeeds, while pings from PC2 to PC4 are blocked by the firewall (communication administratively prohibited).



Ping PC1 -> PC4	Ping PC2 -> PC4
PC1> ping 192.168.2.3	PC2> ping 192.168.2.3
84 bytes from 192.168.2.3 icmp_seq=1 ttl=61 time=54.837 ms	*10.0.2.2 icmp_seq=1 ttl=253 time=30.920 ms (ICMP type:3, code:13, Communication administratively prohibited)
84 bytes from 192.168.2.3 icmp_seq=2 ttl=61 time=55.676 ms	*10.0.2.2 icmp_seq=2 ttl=253 time=46.041 ms (ICMP type:3, code:13, Communication administratively prohibited)
84 bytes from 192.168.2.3 icmp_seq=3 ttl=61 time=55.076 ms	*10.0.2.2 icmp_seq=3 ttl=253 time=39.207 ms (ICMP type:3, code:13, Communication administratively prohibited)
84 bytes from 192.168.2.3 icmp_seq=4 ttl=61 time=53.712 ms	*10.0.2.2 icmp_seq=4 ttl=253 time=42.295 ms (ICMP type:3, code:13, Communication administratively prohibited)
84 bytes from 192.168.2.3 icmp_seq=5 ttl=61 time=57.969 ms	*10.0.2.2 icmp_seq=5 ttl=253 time=47.699 ms (ICMP type:3, code:13, Communication administratively prohibited)

Figure 13: Ping from PC1 and PC2 to PC4

2.4 Refuse UDP communication unless intended for port 80 of PC4

The objective of this step is to create a whitelist by activating the firewall to block all UDP communication except those who goes to PC4 (192.168.2.3) on the port 80. This is done by entering the following commands in the ACL that will be linked to the interface FastEthernet1/0.

```
>>> permit udp any host 192.168.2.3 eq 80
>>> deny udp any any
```

Firstly, precise which destination can be reached for UDP communication. In a second time, deny all other communications for UDP. This procedure allows accepting only packets with a specific destination and to deny all the rest.

Note that the priority of the ACL's permissions are in the same order as the order of the instructions entered for the router configuration. This is explained by the fact that the first permission has the highest priority, and every new permission will by default have a lower priority.

```
Firewall#show access-list
Extended IP access list blacklist
 10 permit icmp host 192.168.1.2 host 192.168.2.3
 20 deny icmp host 192.168.1.3 host 192.168.2.3
 30 permit udp any host 192.168.2.3 eq 80
 40 deny udp any any (75 matches)
```

Figure 14: Configuration of ACL for UDP communication to port 80 of PC4 only

```
PC1> ping 192.168.2.3 -P 17
*10.0.2.2 udp_seq=1 ttl=253 time=42.230 ms (ICMP type:3, code:13, Communication administratively prohibite
d)
*10.0.2.2 udp_seq=2 ttl=253 time=32.565 ms (ICMP type:3, code:13, Communication administratively prohibite
d)
*10.0.2.2 udp_seq=3 ttl=253 time=46.130 ms (ICMP type:3, code:13, Communication administratively prohibite
d)
*10.0.2.2 udp_seq=4 ttl=253 time=27.692 ms (ICMP type:3, code:13, Communication administratively prohibite
d)
*10.0.2.2 udp_seq=5 ttl=253 time=24.102 ms (ICMP type:3, code:13, Communication administratively prohibite
d)
PC1> ping 192.168.2.3 -P 17 -p 80
84 bytes from 192.168.2.3 udp_seq=1 ttl=61 time=50.021 ms
84 bytes from 192.168.2.3 udp_seq=2 ttl=61 time=60.772 ms
84 bytes from 192.168.2.3 udp_seq=3 ttl=61 time=45.793 ms
84 bytes from 192.168.2.3 udp_seq=4 ttl=61 time=50.643 ms
84 bytes from 192.168.2.3 udp_seq=5 ttl=61 time=69.824 ms
```

Figure 15: Ping from PC1 to PC4 and port 80 of PC4

The figure 15 shows that pings from PC1 to port 80 of PC4 succeeds, while pings from PC1 to PC4 are blocked by the firewall (communication administratively prohibited). This shows that the step 2 implementation is correct.

2.5 PC3 accepts all UDP communication from subnet with PC1 and PC2

This step consists in configuring the router to accept all UDP communication that comes from the subnet 192.168.1.0.

```
>>> 35 permit udp 192.168.1.0 0.0.0.255 host 192.168.2.2
```

Since all the UDP communication have already been denied in last step, the only thing to do here, is to allow communication from subnet 192.168.1.0/24 to PC3

(192.168.2.2). This is done by inserting the permission before the permission that denies everything (40). The permission will thus enter the whitelist.

```
Firewall#show access-list
Extended IP access list blacklist
 10 permit icmp host 192.168.1.2 host 192.168.2.3
 20 deny icmp host 192.168.1.3 host 192.168.2.3
 30 permit udp any host 192.168.2.3 eq 80
 35 permit udp 192.168.1.0 0.0.0.255 host 192.168.2.2
 40 deny udp any any
Firewall#
```

Figure 16: Configuration of the permission for UDP communication between subnet of PC1 and PC2 to PC3

PC1 and PC2 can indeed communicate with PC3 by using UDP.

Ping PC1 -> PC3	Ping PC2 -> PC3
<pre>PC1> ping 192.168.2.2 -P 17 84 bytes from 192.168.2.2 udp_seq=1 ttl=61 time=51.798 ms 84 bytes from 192.168.2.2 udp_seq=2 ttl=61 time=45.910 ms 84 bytes from 192.168.2.2 udp_seq=3 ttl=61 time=35.971 ms 84 bytes from 192.168.2.2 udp_seq=4 ttl=61 time=50.495 ms 84 bytes from 192.168.2.2 udp_seq=5 ttl=61 time=34.048 ms</pre>	<pre>PC2> ping 192.168.2.2 -P 17 84 bytes from 192.168.2.2 udp_seq=1 ttl=61 time=41.106 ms 84 bytes from 192.168.2.2 udp_seq=2 ttl=61 time=57.943 ms 84 bytes from 192.168.2.2 udp_seq=3 ttl=61 time=52.331 ms 84 bytes from 192.168.2.2 udp_seq=4 ttl=61 time=58.455 ms 84 bytes from 192.168.2.2 udp_seq=5 ttl=61 time=51.919 ms</pre>

Figure 17: Ping from PC1 and PC2 to PC3

2.6 Mission 2.3 - Protection against data exfiltration

To protect any exfiltration of data from the subnet, a new ACL (whitelist) has to be configured for the interface FastEthernet0/0 of the Firewall router. The permission of whitelist is the following:

```
>>> deny 192.168.2.0 0.0.0.255 any
```

```
Firewall#show access-list
Extended IP access list blacklist
 10 permit icmp host 192.168.1.2 host 192.168.2.3
 20 deny icmp host 192.168.1.3 host 192.168.2.3
 30 permit udp any host 192.168.2.3 eq 80
 35 permit udp 192.168.1.0 0.0.0.255 host 192.168.2.2 (10 matches)
 40 deny udp any any (27 matches)
Extended IP access list whitelist
 10 deny ip 192.168.2.0 0.0.0.255 any
```

Figure 18: Configuration of the whitelist that has been added to interface F0/0

```

PC3> ping 192.168.1.1 -P 17
*192.168.2.1 udp_seq=1 ttl=255 time=10.998 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 udp_seq=2 ttl=255 time=7.736 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 udp_seq=3 ttl=255 time=14.860 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 udp_seq=4 ttl=255 time=7.802 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 udp_seq=5 ttl=255 time=3.788 ms (ICMP type:3, code:13, Communication administratively prohibited)
PC3> ping 192.168.1.1
*192.168.2.1 icmp_seq=1 ttl=255 time=8.397 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 icmp_seq=2 ttl=255 time=7.257 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 icmp_seq=3 ttl=255 time=10.724 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 icmp_seq=4 ttl=255 time=6.609 ms (ICMP type:3, code:13, Communication administratively prohibited)
*192.168.2.1 icmp_seq=5 ttl=255 time=8.235 ms (ICMP type:3, code:13, Communication administratively prohibited)

```

Figure 19: Ping from PC3 to PC1 using UDP and ICMP protocols

As seen on Figure 19, the communication is now denied from devices of the subnet to the outside for every protocol.

This method can be used for data protection, for example a bank won't authorise its workers to communicate private information outside the subnet of the bank.

2.7 Firewall conclusion

Now that the firewall has been set up, the understanding of the group improved subsequently. A firewall is a router that allows filtering packets communication depending on the protocol used (UDP, ICM, TCP, etc), the source or the destination.

Each packet passing through the firewall router is checked and if the ACL of the firewall interface says that the packet should be denied (because of a certain source, destination, protocol, ...), the packet will not continue his way.

It works as a barrier between an internal network and an untrusted external network, such as internet. It thus plays a crucial role in network security.

It is also possible to add multiple firewalls to an interface. This could be useful, but the order in which they are linked to the interface is important, as the priority of the permission defines how they should react.

3 Conclusion

In conclusion, this lab deepened our understanding of Network Address Translation (NAT) and firewalls. NAT emerged as a crucial tool for mapping private to public IP addresses, enhancing security, and conserving address resources. While it offers advantages, challenges like end-to-end communication were noted.

The firewall configurations demonstrated its important role in network security, controlling traffic based on source, destination, and protocol. Specific ACL setups showcased selective permissions, emphasizing the granularity of control firewalls provide. The addition of a whitelist illustrated its practical use in safeguarding against data exfiltration.

In summary, this lab underlined the importance of NAT and firewalls in securing networks and managing data flows effectively.