Security: iptables firewall on linux

Introduction

Virtual infrastructure used for the task

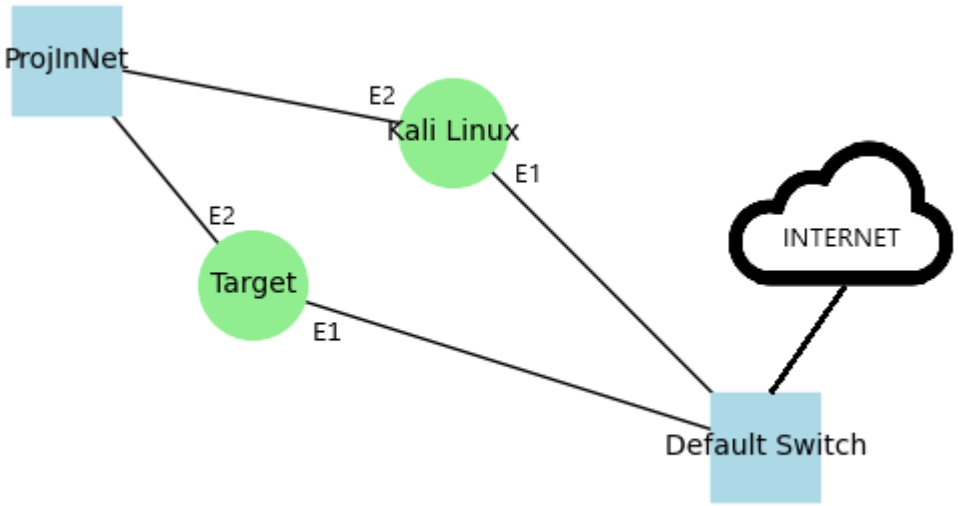


Diagram of the virtual environment

The following virtual environment was used to carry out this task:

- Hypervisor: Hyper-V 10.0.26100.1882
- Target machine: Ubuntu 24.04 Noble Numbat
- Kali Linux machine: Kali GNU/Linux 2024.4
- ProjInNet virtual switch: virtual switch added to hyper-v for the purpose of this item
- Virtual Switch Default Switch: this is a switch which acts as a simple NAT router and allows access to the internet from virtual machines.

IP addressing from the ProjInNet switch side:

- Kali Linux -. `10.10.10.1`
- Target - `10.10.10.254`

Pre-task

Content:

2.1 Personal Firewall
Check the iptables configuration on your computer by issuing the command
iptables L v n
Questions:
1. question 1: what policy is enabled by default on the machine ?
Question 2: Which chain should be modified to protect the machine against connections?
external connections ?
Include answers with explanations in the report.

Solution:

```
administrator@Target-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
administrator@Target-VM:~$
```

Answers to questions:

what policy is enabled by default on the machine ?

It is running the ACCEPT policy meaning all packets will be accepted by the machine as there is nothing blocked on the firewall.

Which chain should I modify to protect the machine from external connections?

INPUT because it is the chain responsible for incoming traffic to the machine.

Task 1

Content and solutions

Contents and solutions are intertwined as the task is extensive and consists of many annotations

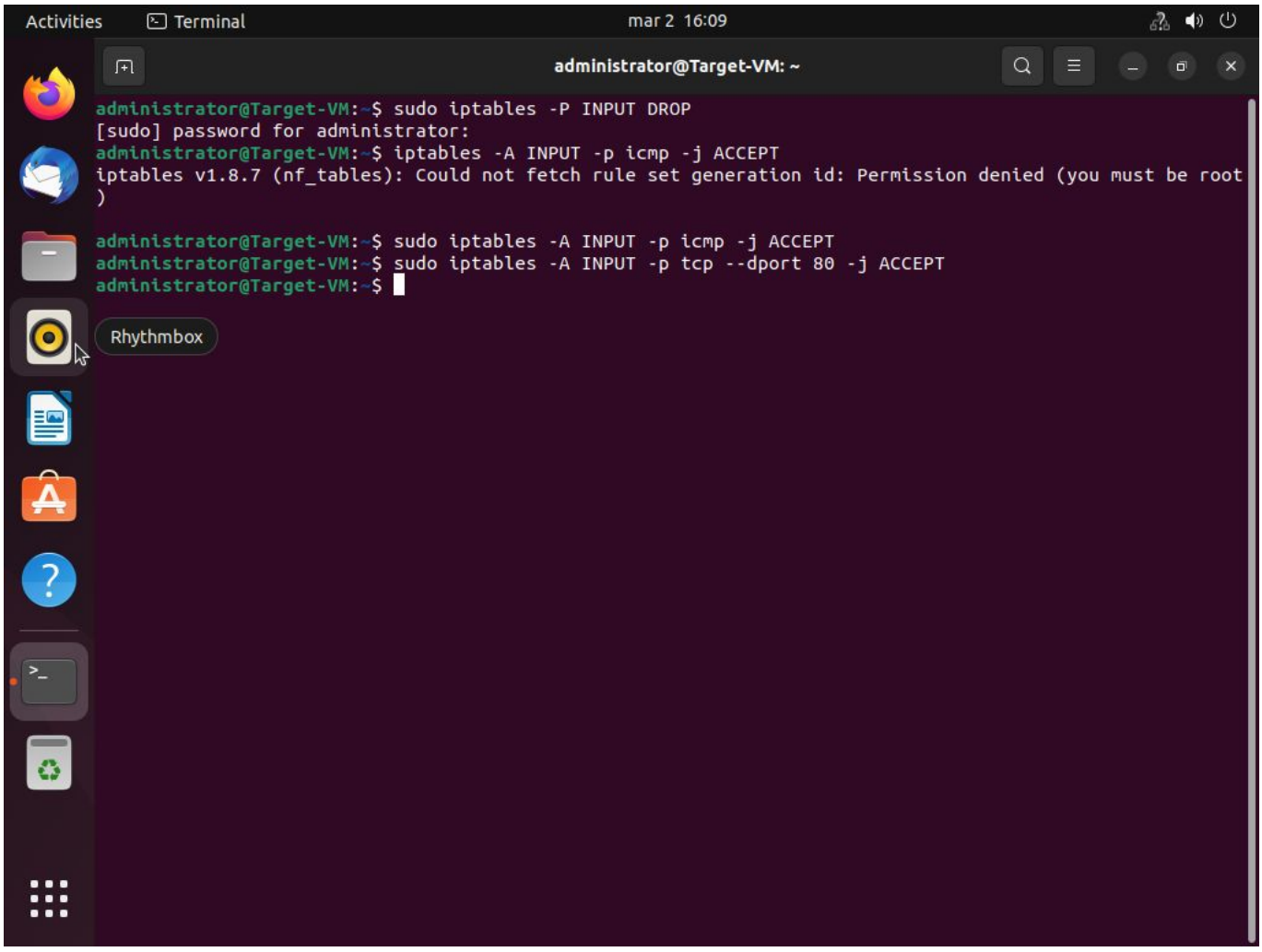
```
    Użytkownik maszyny A uruchamia usługi WWW i ssh .
    Użytkownik maszyny B sprawdza dostępność
    maszyny A oraz usług WWW I ssh za pomocą poleceń:
    ping adres_ IP_maszyny_A
    nmap sT Pn n p 80,22 v adres_ IP_maszyny_A
    Wynik umieść w sprawozdaniu.
```

```
kali@kali:~$ ping 10.10.10.254
PING 10.10.10.254 (10.10.10.254) 56(84) bytes of data.
64 bytes from 10.10.10.254: icmp_seq=1 ttl=64 time=0.329 ms
64 bytes from 10.10.10.254: icmp_seq=2 ttl=64 time=0.498 ms
64 bytes from 10.10.10.254: icmp_seq=3 ttl=64 time=0.562 ms
64 bytes from 10.10.10.254: icmp_seq=4 ttl=64 time=0.719 ms
^C
--- 10.10.10.254 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3059ms
rtt min/avg/max/mdev = 0.329/0.527/0.719/0.139 ms

kali@kali:~$ nmap -sT -Pn -n -p 80,22 -v 10.10.10.254
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-02 09:58 EST
Initiating Connect Scan at 09:58
Scanning 10.10.10.254 [2 ports]
Discovered open port 80/tcp on 10.10.10.254
Discovered open port 22/tcp on 10.10.10.254
Completed Connect Scan at 09:58, 0.00s elapsed (2 total ports)
Nmap scan report for 10.10.10.254
Host is up (0.00036s latency).

PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.02 seconds

kali@kali:~$
```

```
    Zad 1.1. Napisz wywołanie iptables modyfikujące domyślna politykę
maszyny A
    na taką, która będzie
    odrzucać jakikolwiek ruch przychodzący (podpowiedź: należy zmodyfikować
    domyślną politykę dla łańcucha
    wejściowego 'INPUT' z ACCEPT na DROP). Wywołanie ma składnię:
    iptables P chain target
    Z a d 1.2. Napisz wywołanie iptables dodające regułę do polityki na
maszyny A
```

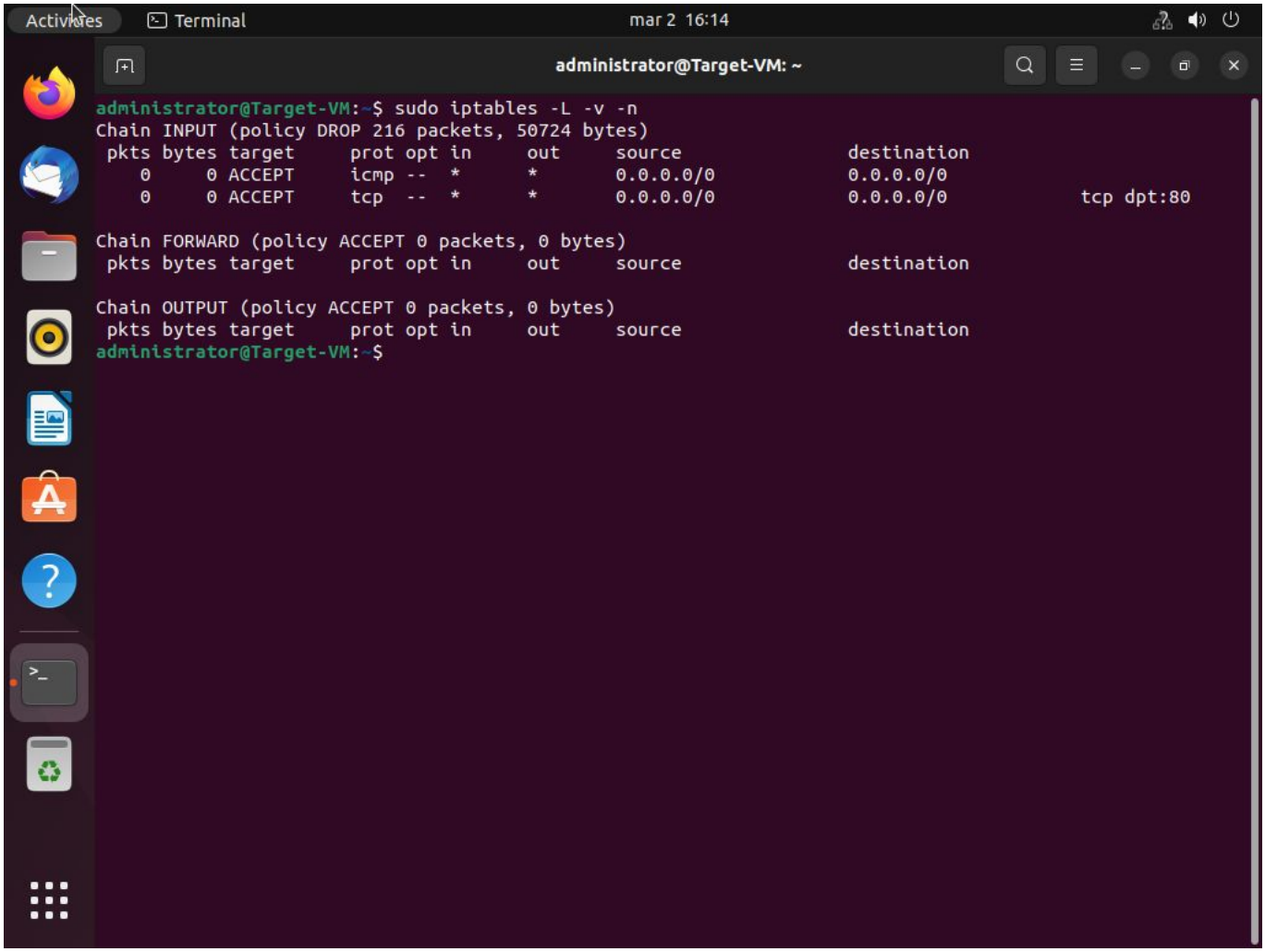
```
dla ruchu przychodzącego
powodujące akceptację całego ruchu ICMP (podpowiedź: prawie gotowe
polecenie znajduje się poniżej jedynie
w miejscu kropek należy wpisać odpowiednią wartość)
iptables A ... p icmp j ACCEPT
Uzupełnione polecenie umieść w sprawozdaniu.
Z ad 1.3. W analogiczny sposób, na maszynie A, wpisz komendę iptables,
 która pozwoli na akceptację
przychodzącego ruchu tcp na port nr 80 (podpowiedź: prawie gotowe
polecenie znajduje sie poniżej):
iptables A ... p tcp dport ... j ...
```

Here a picture of the desktop was used instead of a listing because the SSH connection was blocked so I couldn't copy command results from the Target machine



Command execution on the Target machine

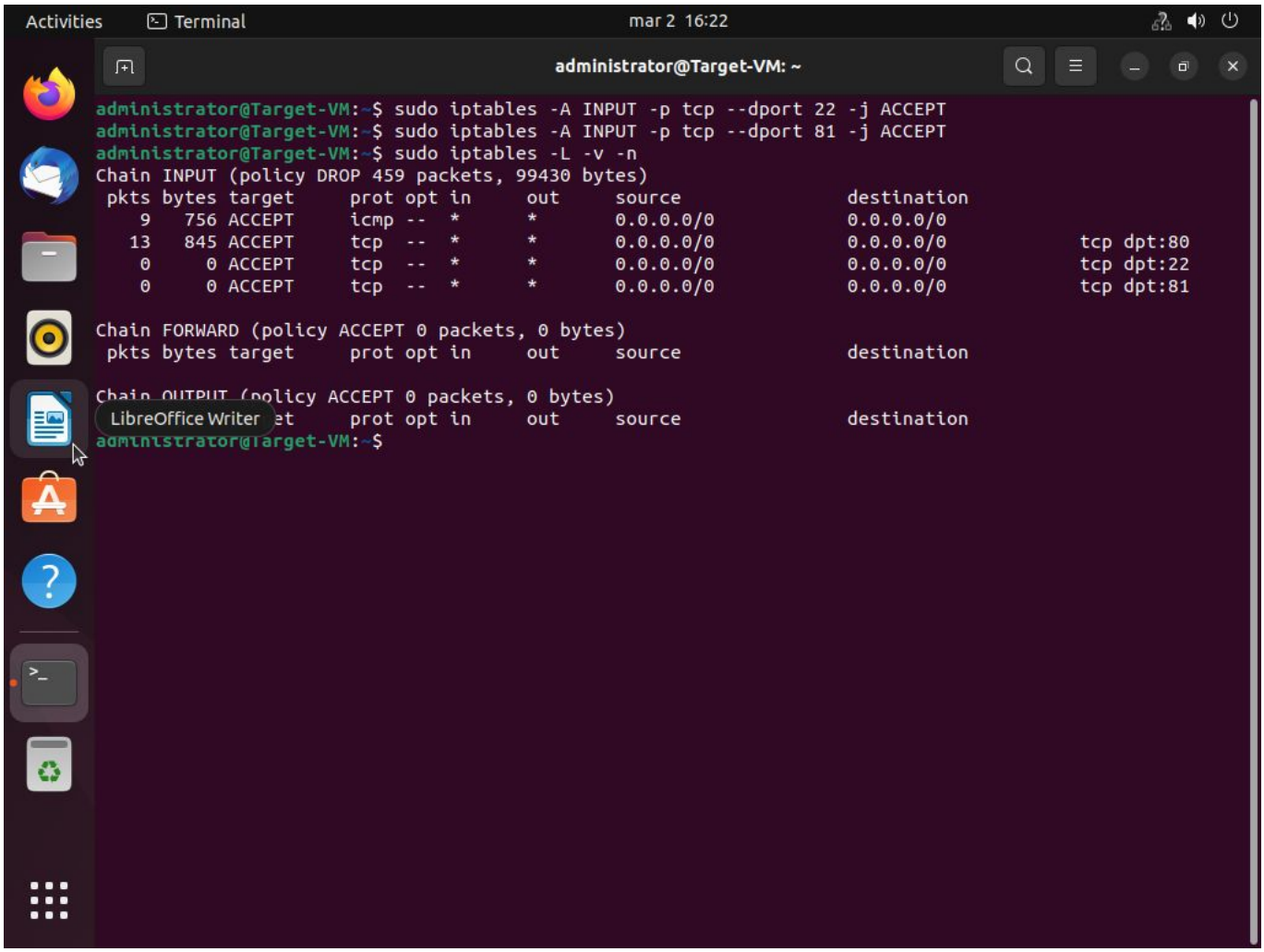
```
Wykonaj modyfikacje podane w zadaniach Z
ad 1.1, Z ad 1.2, Z ad 1.3 i sprawdź nową konfigurację programu
iptables. Sprawdź czy użytkownik maszyny B może wykonać ping
maszyny A oraz czy może połączyć się z
usługą WWW (port 80).
```



Verification of the iptables configuration on the target machine

```
kali@kali:~$ ping 10.10.10.254
PING 10.10.10.254 (10.10.10.254) 56(84) bytes of data.
64 bytes from 10.10.10.254: icmp_seq=1 ttl=64 time=0.372 ms
64 bytes from 10.10.10.254: icmp_seq=2 ttl=64 time=0.628 ms
64 bytes from 10.10.10.254: icmp_seq=3 ttl=64 time=0.409 ms
64 bytes from 10.10.10.254: icmp_seq=4 ttl=64 time=0.725 ms
^C
--- 10.10.10.254 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3079ms
rtt min/avg/max/mdev = 0.372/0.533/0.725/0.147 ms

kali@kali:~$ curl -I http://10.10.10.254
HTTP/1.1 200 OK #tutaj możemy zaobserwować że usługa HTTP jest przepuszczana
przez firewall na maszynie target
Date: Sun, 02 Mar 2025 15:17:50 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Sun, 02 Mar 2025 13:55:59 GMT
ETag: "29af-62f5c663f231e"
Accept-Ranges: bytes
Content-Length: 10671
Vary: Accept-Encoding
Content-Type: text/html
```

```
Następnie wykonaj poniższe kroki:
1. Użytkownik maszyny A wybiera dwa dowolne porty i otwiera do
nich dostęp (akceptuje ruch
przychodzący) np. Porty 22 i 81.
Składnię polecenie zapisz w spra wozdaniu.
2. Użytkownik maszyny B wyko
n uje skanowanie portów (np. Używając programu nmap) na maszynie A w
celu wykrycia otwartych portów np. nmap adres_IP_maszyny_A p 10 100
Wynik wywołania polecenia umieść w sprawozdaniu.
```



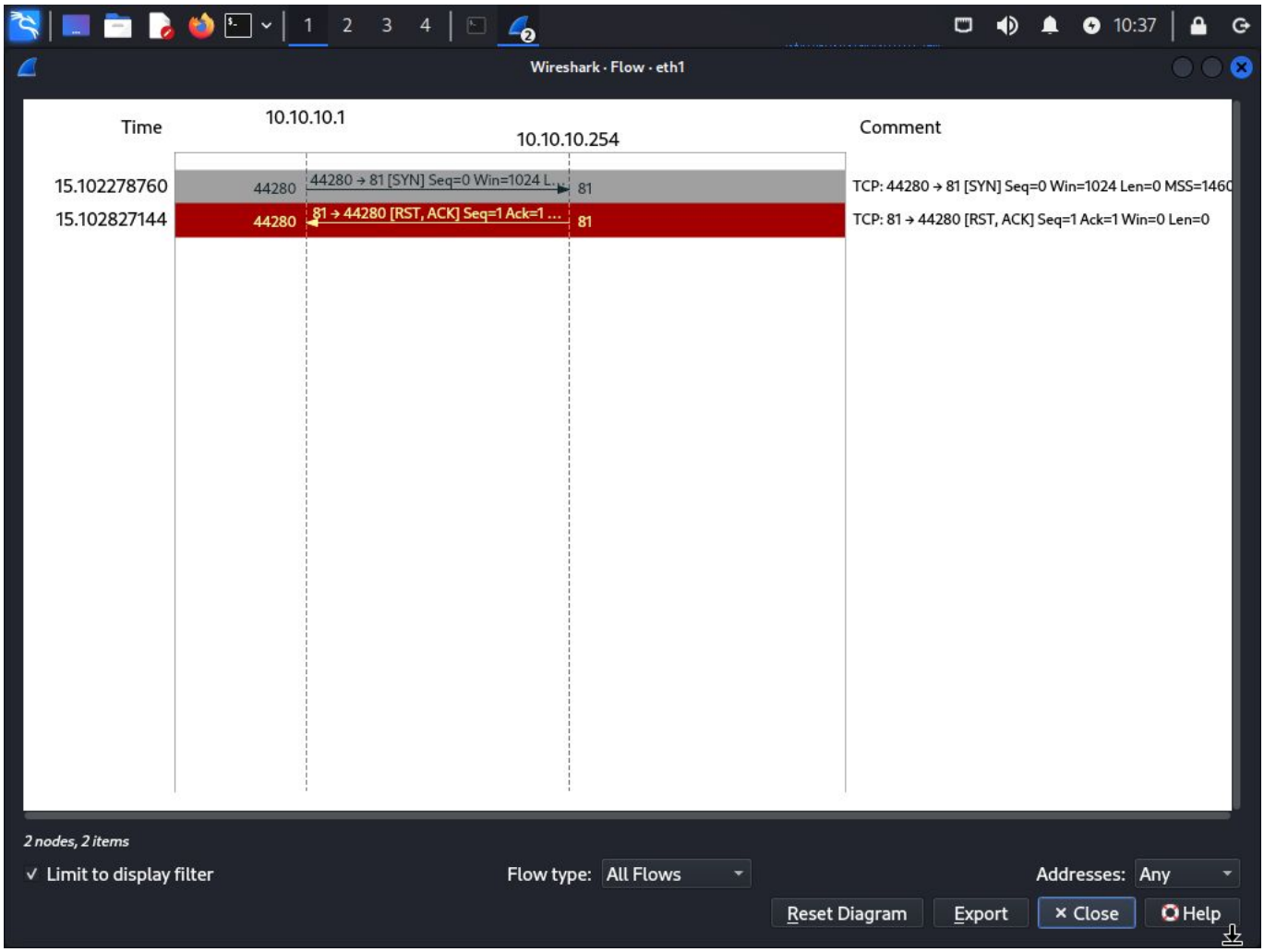
Perform iptables configuration on the target machine

```
kali@kali:~$ nmap 10.10.10.254 -p 10-100
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-02 10:24 EST
Nmap scan report for 10.10.10.254
Host is up (0.00052s latency).
Not shown: 88 filtered tcp ports (no-response)
PORT   STATE  SERVICE
22/tcp open   ssh # widzimy że port 22 jest otwarty
80/tcp open   http
81/tcp closed hosts2-ns # widzimy że port 81 jest otwarty
MAC Address: 00:15:5D:38:01:1C (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 1.91 seconds
```

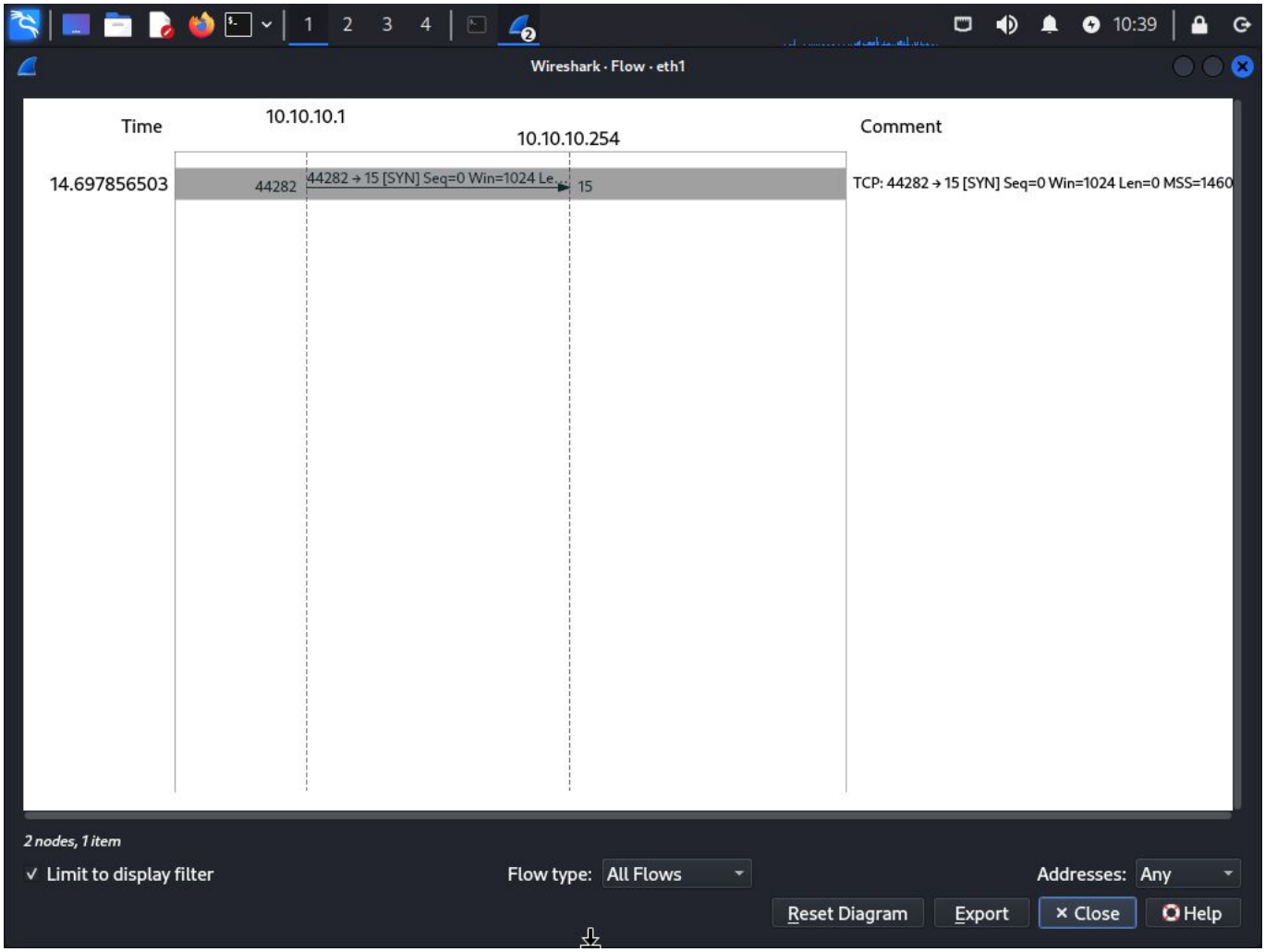
```
kali@kali:~$
```

Answer the questions

How can nmap distinguish between filtered ports and closed ports ? Try to determine this by analysing the traffic exchanged between two machines A and B (use wireshark or tcpdump). Include your answer with reasons in your report. Next, set up a web server (Apache) on machine B and try to connect to it using a browser from machine A.

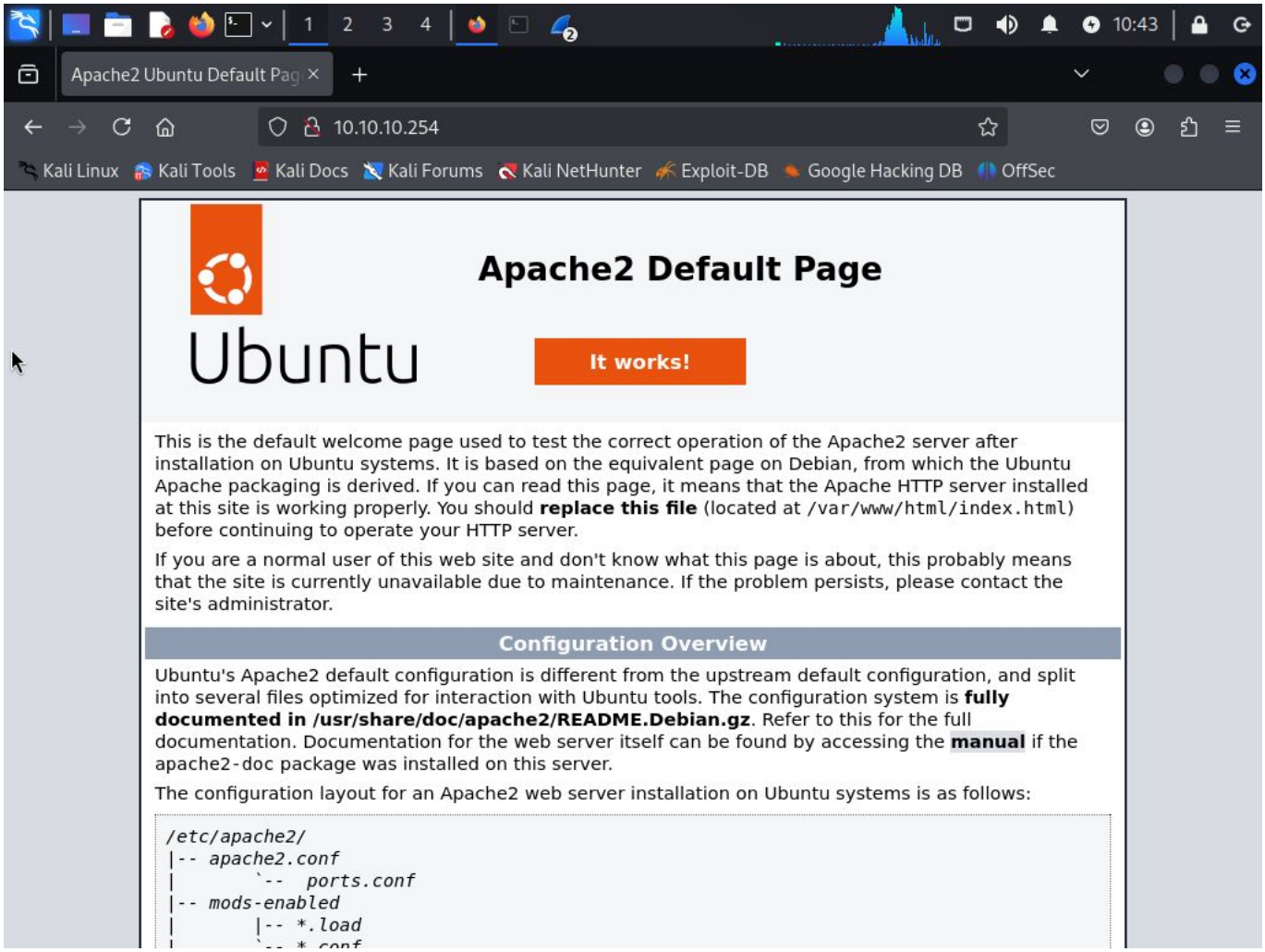


Analysis of the packets responsible for port scanning 81



Analysis of the packets responsible for scanning the port 15

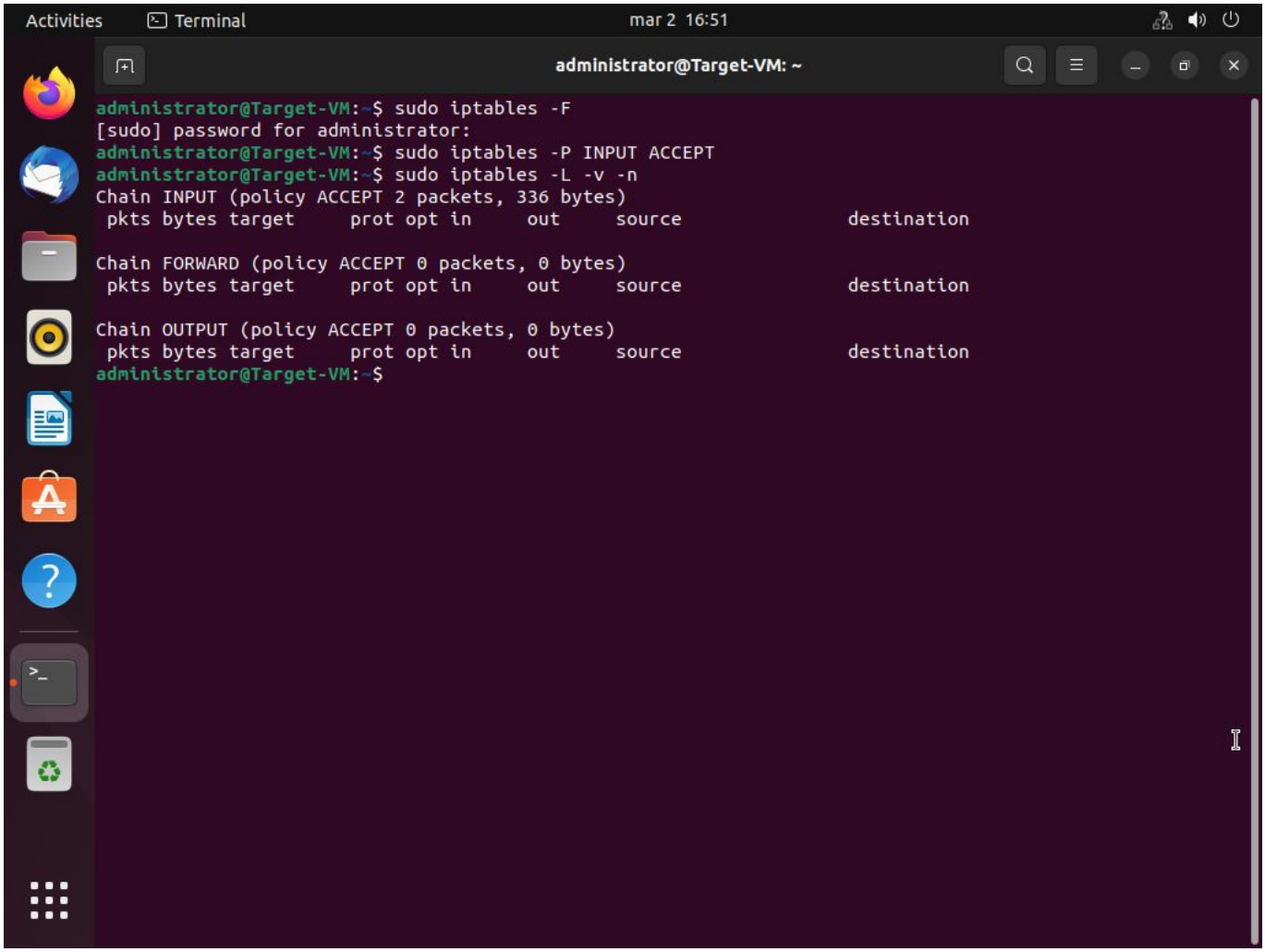
After comparing these two analyses, we can conclude that if the port is blocked, it does not reach the operating system itself and its network stack, whereas if the port is unblocked, then the system's network stack responds with a connection reset. Then we know that there is no service on the port but we know that it is open in the firewall.



Connecting via a browser as requested in the task

Why can't the user of machine A connect to the web server on machine B? Justify your answer by analysing the traffic exchanged between machines A and B. Include your answer in your report. Before proceeding, restore the policy on machine A that accepts all traffic (ACCEPT ALL) (hint: delete all current rules and then define a rule for the INPUT chain). Now check again the connectivity from machine A to the HTTP server on machine B. Include the result of the check in the report.

In task 1.3, you mentioned to add a rule that allows you to log in to the web server, I did not remove the rule so in my case the connection was successful. However if such a rule was not in the IP tables then the same situation as shown in figure 6 would occur, the Apache service running on the host would not get the information that someone wants to connect to it and the connection would not happen because before the packet would get to the web server it would be blocked by iptables



Restoring the state before the changes

```
kali@kali:~$ curl -I http://10.10.10.254
HTTP/1.1 200 OK # tutaj widzimy że HTTP działa poprawnie
Date: Sun, 02 Mar 2025 15:52:37 GMT
Server: Apache/2.4.52 (Ubuntu)
Last-Modified: Sun, 02 Mar 2025 13:55:59 GMT
ETag: "29af-62f5c663f231e"
Accept-Ranges: bytes
Content-Length: 10671
Vary: Accept-Encoding
Content-Type: text/html


kali@kali:~$
```

Task 2

Virtual infrastructure used to perform the task

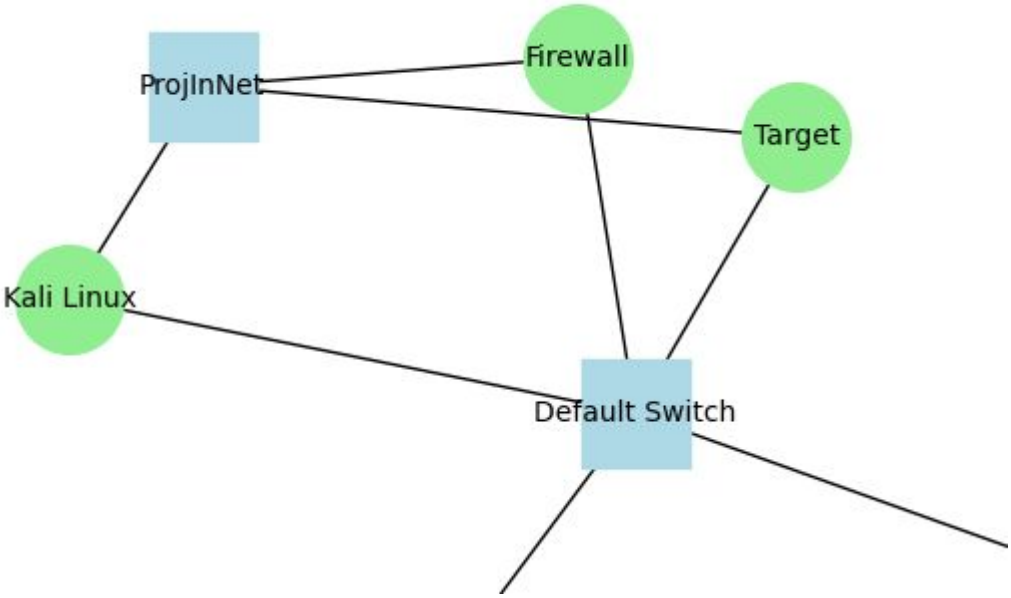


Diagram of the virtual environment

The diagram looks like this because it was made using my own python script which generates a network diagram in hyper-v by taking information about the environment from powershell.
The following virtual environment was used to perform this task:

- Hypervisor: Hyper-V 10.0.26100.1882.
- Target machine: Ubuntu 24.04 Noble Numbat
- Firewall machine: Ubuntu 24.04 Noble Numbat
- Kali Linux Machine: Kali GNU/Linux 2024.4
- ProjInNet Virtual Switch: a virtual switch added to hyper-v for the purpose of this item.
- Virtual Switch Default Switch: this is a switch which acts as a simple NAT router and allows access to the internet from virtual machines.

IP addressing from the ProjInNet switch side:

- Kali Linux -. `10.10.10.1` Machine B
- Target - `10.10.10.254` Machine A
- Firewall - `10.10.10.100` Machine C

Contents and solutions are intertwined as the task is extensive and consists of many annotations

```
    W tym ćwiczeniu udział biorą 3 maszyny (A, B i C), w tym maszyna C jako
    firewall. W celu przekazywania
    pakietów pomiędz y maszynami A I B za pośrednictwem maszyny C, konieczna
     jest modyfikacja tablic
    rutingu maszyn A I B. W tym celu należy:
    1. Na maszynie A skonfigu ruj ruting tak aby pakiety adresowane do
maszyny
     B przechodziły przez
    maszynę C:
    route add host ip_maszyny_B gw ip_maszyny_C
    2. Analogiczną zmian ę należy wprowadzić na maszynie B:
    route add
    host ip_maszyny_A gw ip_maszyny_C
```

```
    3. Na maszynie C należy wyłączyć przekierowywanie pakietów icmp oraz
umożliwić
     przekazywanie
    pakietów (forwarding) pomkiędzy interfejsami:
    echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
    echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
kali@kali:~$ sudo route add -host 10.10.10.254 gw 10.10.10.100
[sudo] password for kali:

kali@kali:~$ route
Kernel IP routing table # ARDU-STATION to nazwa hosta virtualizacji ...
trasa ta jest wykorzystana do dostępu do internetu
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         ARDU-STATION.ms 0.0.0.0         UG    100    0        0 eth0
10.10.10.0      0.0.0.0         255.255.255.0   U     101    0        0 eth1
10.10.10.254    10.10.10.100    255.255.255.255 UGH   0      0        0 eth1
172.30.0.0      0.0.0.0         255.255.240.0   U     100    0        0 eth0

kali@kali:~$
```

```
administrator@Target-VM:~$ sudo route add -host 10.10.10.1 gw 10.10.10.100
administrator@Target-VM:~$ route
Kernel IP routing table # tutaj tak samo jak w poprzednim trasa ta jest
wykorzystana do dostępu do internetu
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
default         ARDU-STATION.ms 0.0.0.0         UG    100    0        0 eth0
10.10.10.0      0.0.0.0         255.255.255.0   U     101    0        0 eth1
10.10.10.1      10.10.10.100    255.255.255.255 UGH   0      0        0 eth1
link-local      0.0.0.0         255.255.0.0     U     1000   0        0 eth0
172.30.0.0      0.0.0.0         255.255.240.0   U     100    0        0 eth0
administrator@Target-VM:~$
```

```
root@Firewall-VM:~> echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
root@Firewall-VM:~> echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
    Na maszynach A I B uruchom serwery WWW i ssh
    W przypadku problemów (brak efektu komunik
    acji maszyn A i B poprzez maszynę C) należy dodatkowo
    wykonać polecenie
    ip route flush cache
    iptables A INPUT p icmp icmp type redirect j DROP
    Zadanie 2 (ruch wychodz ą cy)
    Na maszynie C zastosowano politykę pozwalającą u
    ż y tkownikowi z maszyny A na połączenie z dowolną
    zewnętrzną stroną WWW (zewnętrzna oznacza stronę na maszynie B). W tym
celu należy
    skonfigurować politykę wydając następujące polecenia:
    iptables P FORWARD DROP
```

```
    iptables A FORWARD p tcp s ip_maszyny_A dport 80 j ACCEPT
    iptables A FORWARD p tcp d ip_maszyny_A sport 80 j ACCEPT
```

Web servers on Kali linux and Target launched

```
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -s 10.10.10.254
--dport 80 -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -d 10.10.10.254
--dport 80 -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination

Chain FORWARD (policy DROP 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination
0     0 ACCEPT     tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp dpt:80
0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.10.10.254
tcp dpt:80

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination
administrator@Firewall-VM:~$
```

Questions

On which chain do the above commands operate?

The commands in Listing 9 work on the FORWARD chain, which means that they apply to packets passing through the firewall machine

Is it possible to connect from machine A to the web server running on machine B ?

```
# z target VM możemy się dostać do kali
administrator@Target-VM:~$ curl -I 10.10.10.1
HTTP/1.1 200 OK                #działa
Date: Sun, 02 Mar 2025 17:13:49 GMT
Server: Apache/2.4.63 (Debian)
Last-Modified: Sat, 30 Nov 2024 12:33:01 GMT
ETag: "29cf-628208420f1c0"
Accept-Ranges: bytes
Content-Length: 10703
Vary: Accept-Encoding
```

```
Content-Type: text/html
```

What is the task of the last command above ?

The purpose of the last command is to allow communication on a TCP/80 port from all possible addresses to the address 10.10.10.254. In other words to allow everyone to access the web server on the target VM. The letter D stands for destination

Is it possible to connect from machine A to a web server running on some other machine if the traffic goes through machine C (firewall) ?

According to this command `sudo iptables -A FORWARD -p tcp -s 10.10.10.254 -dport 80 -j ACCEPT` yes, because this command allows traffic where the source is the target vm and the target is not specified so we can get anywhere. The letter S stands for source.

Is it possible to connect from machine B to the web server running on machine A (think of what happens if the user of machine B uses port 80 as the source port in this attempt) ?

```
kali@kali:~$ curl -v http://10.10.10.254
*   Trying 10.10.10.254:80...
```

Such an attempt is unsuccessful. If, on the other hand, the source port instead of the destination port were 80, then such traffic would pass through the firewall. Below is a demonstration of how to use the netcat tool to force the use of port 80 as the source port.

```
kali@kali:~$ sudo systemctl stop apache2

kali@kali:~$ sudo nc -v -p 80 10.10.10.254 80
10.10.10.254: inverse host lookup failed: Unknown host
(UNKNOWN) [10.10.10.254] 80 (http) open
GET / HTTP/1.1
HTTP/1.1 400 Bad Request
Date: Sun, 02 Mar 2025 17:32:29 GMT
Server: Apache/2.4.52 (Ubuntu)
Content-Length: 301
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
<p>Your browser sent a request that this server could not understand.<br />
</p>
```

```
<hr>
<address>Apache/2.4.52 (Ubuntu) Server at 127.0.1.1 Port 80</address>
</body></html>

kali@kali:~$
```

Such a message indicates that the server recognised the connection, but the HTTP request was invalid or did not meet the server's expectations at all, causing an error. However, the fact that the connection was established shows that the firewall did not block the connection because the source port was port 80, ensuring that such traffic was not rejected by firewalls.

Is it possible to connect from machine A to the ssh server on machine B and vice versa ? (think what happens if the user of machine B uses port 80 as the source port in this attempt) ?

```
kali@kali:~$ sudo nc -v -p 80 10.10.10.254 22

10.10.10.254: inverse host lookup failed: Unknown host
(UNKNOWN) [10.10.10.254] 22 (ssh) open
SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.11
^C

kali@kali:~$
```

We can see above that if we use port 80 as the source port then the SSH server is trying to agree to communicate with us. The very fact that the SSH-2.0 header appears.... indicates that the SSH server is responding.

Task 3

```
    Zadanie 3 (usługi publiczne)
    W poniższym zadaniu zastosowana polityka na maszynie C (firewall) ma
umożliwić zdalny dostęp do maszyny A
    poprzez ssh (na maszynie A działa serwer ssh, na maszynie B zdalny
klient ssh).
    W tym celu należy uzupełnić (w miejscu kropek) poniżej wypisane komend
    y iptables:
    iptables A FORWARD p tcp d ip_maszyny_A dport 22 j ACCEPT
    iptables A FORWARD p ... s ip_maszyny_A sport ... syn j DROP
    iptables A FORWARD p tcp s ip_maszyny_A sport 22 j ...
    Uzupełnione komendy zapisz w s prawozdaniu.
    Po uzupełnieniu w/w komend wprowadź je na maszynie C oraz sprawdź
możliwe jest poł
    ą czenie ssh z
    maszyny B do maszyny A ale nie w drugą stronę.
    Wyniki umieść w sprawozdaniu.
```

```
    Wyniki umieść w sprawozdaniu.
```

```
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -d 10.10.10.254
--dport 22 -j ACCEPT
[sudo] password for administrator:
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -s 10.10.10.254
--sport 22 --syn -j DROP
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -s 10.10.10.254
--sport 22 -j DROP
administrator@Firewall-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination

Chain FORWARD (policy DROP 55 packets, 3300 bytes)
pkts bytes target     prot opt in     out     source
destination
6   378 ACCEPT     tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp dpt:80
3   180 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.10.10.254
tcp dpt:80
0     0 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.10.10.254
tcp dpt:22
0     0 DROP       tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp spt:22 flags:0x17/0x02
0     0 DROP       tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp spt:22

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination
administrator@Firewall-VM:~$
```

```
administrator@Target-VM:~$ ssh kali@10.10.10.1
The authenticity of host '10.10.10.1 (10.10.10.1)' can't be established.
ED25519 key fingerprint is
SHA256:BKso31RSEejEwAenrTsxzs/xouoBf45WqBNlbbDVP2c.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.1' (ED25519) to the list of known
hosts.
kali@10.10.10.1's password:
Linux kali 6.11.2-amd64 1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1
(2024-10-15) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

```
Last login: Sun Mar  2 12:41:54 2025 from 10.10.10.100
kali@kali:~$ la
.ICEauthority             .dmrc       .ssh                         Documents
.Xauthority               .face       .sudo_as_admin_successful    Downloads
.bash_logout              .face.icon  .xsession-errors             Music
.bashrc                   .gnupg      .zprofile                    Pictures
.bashrc.kali-tweaks-orig  .java       .zsh_history                 Public
.bashrc.original          .local      .zshrc                       Templates
.cache                    .mozilla    .zshrc.kali-tweaks-orig      Videos
.config                   .profile    Desktop

kali@kali:~$
```

```
kali@kali:~$ ssh administrator@10.10.10.254
#brak wyników po chwili wyrzucenie z powrotem do promptu

kali@kali:~$
```

Task 4

```
    Zadanie 4 (ruch icmp)
    Zmień politykę na maszynie C tak, aby umoż
    liwić tylko wybrany ruch icmp (selektywnie) od i do maszyny A.
    Uzupełnij (w miejscu kropek) poniżej wypisane komendy iptables
(podpowiedź: listę wszystkich dostępnych
    wiadomości ICMP można sprawdzić za pomocą polecenia:
    iptables
    p icmp h
    iptables A FORWARD p icmp s ip_maszyny_A icmp type echo request j ACCEPT
    iptables A FORWARD p icmp d ... icmp type echo reply j ...
    iptables A FORWARD p icmp d ip_maszyny_A icmp type echo request j ...
    iptables A FORWARD p icmp s .. ... icmp type echo reply j ...
    Uzupełnione komendy zapisz w
    sprawozdaniu.
```

```
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p icmp -s
10.10.10.254 --icmp-type echo-request -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p icmp -d
10.10.10.254 --icmp-type echo-reply -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p icmp -d
10.10.10.254 --icmp-type echo-request -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p icmp -s
10.10.10.254 --icmp-type echo-reply -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination
```

```
Chain FORWARD (policy DROP 66 packets, 3880 bytes)
pkts bytes target     prot opt in     out     source
destination
6   378 ACCEPT     tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp dpt:80
3   180 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.10.10.254
tcp dpt:80
1    60 ACCEPT     tcp  --  *      *       0.0.0.0/0            10.10.10.254
tcp dpt:22
0     0 DROP       tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp spt:22 flags:0x17/0x02
11   660 DROP       tcp  --  *      *       10.10.10.254         0.0.0.0/0
tcp spt:22
0     0 ACCEPT     icmp --  *      *       10.10.10.254         0.0.0.0/0
icmptype 8
0     0 ACCEPT     icmp --  *      *       0.0.0.0/0            10.10.10.254
icmptype 0
0     0 ACCEPT     icmp --  *      *       0.0.0.0/0            10.10.10.254
icmptype 8
0     0 ACCEPT     icmp --  *      *       10.10.10.254         0.0.0.0/0
icmptype 0

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source
destination
administrator@Firewall-VM:~$
```

Questions

Why is it dangerous to let IMCP traffic through without any restrictions ?

Passing ICMP traffic without restrictions is dangerous because it can be used in DoS attacks (e.g. ping flood), Smurf attacks, can reveal information about the network structure and allow firewalls to be bypassed, increasing the risk of system compromise.

Assuming machine C filters traffic to network 1.2.3.0/24 instead of filtering only traffic addressed to machine A, what happens when machine C receives an ICMP E cho R equest packet with destination address 1.2.3.255 knowing that it is passing ICMP traffic ?

Machine C will let the ICMP packet pass because it is directed to a permitted network. A packet with a broadcast address will trigger ICMP responses from all devices on that network, which can lead to undesirable effects such as overloading the network or revealing too much information about devices on the network.

Task 5

```
        Wykorzystaj tą samą konfiguracją jak w punkcie 2.2 w zadaniu 2 tj:
        iptables P FORWARD DROP
        iptables A FORWARD p tcp s ip_maszyny_A dport 80 j ACCEPT
        iptables A FORWARD p tcp d ip_maszyny_A sport 80 j ACCEPT
        Sprawdź wynik po wydaniu po niższego polecenia :
        nmap sA Pn n p 22,25 source port 80 ip_maszyny_A
        Wynik wydania powyższego polecenia zapisz w sprawozdaniu.
```

```
kali@kali:~$ nmap -sA -Pn -n -p 22,25 --source-port 80 10.10.10.254
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-02 13:04 EST
Nmap scan report for 10.10.10.254
Host is up (0.00060s latency).

PORT   STATE      SERVICE
22/tcp unfiltered ssh
25/tcp filtered   smtp

Nmap done: 1 IP address (1 host up) scanned in 1.29 seconds

kali@kali:~$
```

Questions

How can this problem be solved using a package filter?

To solve the problem, adjust the filter rules in iptables, allowing traffic on port 25 only for selected addresses or blocking this port if it is not needed. It is important to specify precisely which connections should be allowed to avoid unauthorised traffic.

continuation of the task

```
    Określ numery (pozycje) reguł związanych z ruchem WWW
    i zmodyfikuj konfigurację na maszynie C
    poprzez wydanie następujących poleceń iptables:
    iptables D FORWARD position rule web
    iptables A FORWARD p tcp s ip_maszyny_A dport 80 j ACCEPT
    iptables A FORWARD m state state ESTABLISHED,RELATED j ACCEPT
    Pytania:Pytania:
    14. Pytanie 14: Pytanie 14: Jakie jest znaczenie nowych reguł ?
(podpowiedź: sprawdź
    informacje w sekcjiJakie jest znaczenie nowych reguł ? (podpowiedź:
sprawdź informacje
     w sekcji “MATCH “MATCH EXTENSIONS” poEXTENSIONS” podręcznika iptables
```

```
dostępnego po wydani
     u polecenia “man iptables”)dręcznika iptables dostępnego po wydaniu
polecenia “man iptables”)
    Odpowiedź zapisz w sprawozdaniu.
    Odpowiedź zapisz w sprawozdaniu.
    Dokonaj sprawdzenia konfiguracji za pomocą programu nmap tak jak to było
robione poprzednio.Dokonaj
     sprawdzenia konfiguracji za pomocą programu nmap tak jak to było
robione poprzednio.
    Wyniki opisz i zapisz w sprawozdaniu. Wyniki opisz i zapisz w
sprawozdaniu.
```

```
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -p tcp -s 10.10.10.254
--dport 80 -j ACCEPT
administrator@Firewall-VM:~$ sudo iptables -A FORWARD -m state --state
ESTABLISHED,RELATED -j ACCEPT
```

Questions

What is the meaning of the new rules ? (hint: check the information in the "MATCH EXTENSIONS" section of the iptables manual available after issuing the "man iptables" command)

The new rules in the 'iptables' configuration have the following meaning:

- The rule 'iptables -A FORWARD -p tcp -s 10.10.10.254 -dport 80 -j ACCEPT' allows TCP connections from the address '10.10.10.254' to port 80 (HTTP) of machine C.
- The 'iptables -A FORWARD -m state -state ESTABLISHED,RELATED -j ACCEPT' rule allows replies to existing connections and related connections to be accepted, ensuring that network sessions such as HTTP work correctly.

These rules allow access to the web server and correct management of network connections.

Task 6

```
    Zadanie 6
    . O graniczanie przepływności
    Wykorzystaj konfiguracje z zadania 4 (ruch icmp).
    Pytania:
    1 5 . Pytanie 15: Jaki jest efekt zastąpienia reguły:
    iptables A FORWARD p icmp d ip_maszyny_A icmp type echo request j ACCEPT
    poniższą regułą:
    iptables
    A FORWARD p icmp d ip_maszyny_A icmp type echo request m limit limit
20/minute limit
    burst 1 j ACCEPT
```

```
    Podpowiedź: Sprawdź w podręczniku iptables (man iptables ) znaczenie
opcji limit
    Wynik sprawdzenia opisz i zapisz w sprawozdaniu.
```

Questions

What is the effect of replacing the rules?

Rule replacement:

```
    iptables -A FORWARD -p icmp -d ip_maszyny_A --icmp-type echo-request -j
ACCEPT
```

Rule:

```
    iptables -A FORWARD -p icmp -d ip_maszyny_A --icmp-type echo-request
    -m limit --limit 20/minute --limit-burst 1 -j ACCEPT
```

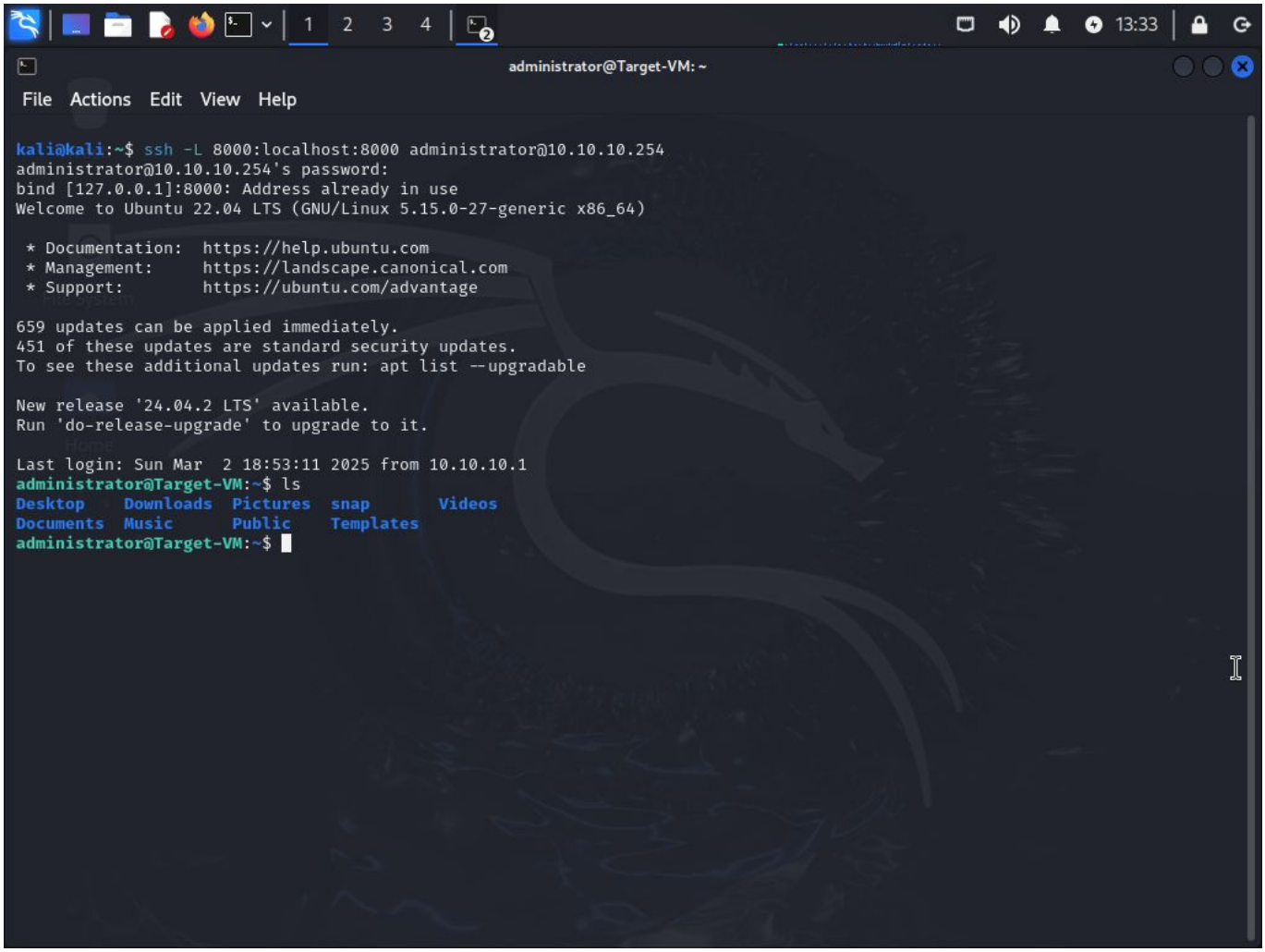
leads to the introduction of a bandwidth limitation on ICMP echo request packets. In the first case, all packets are accepted without limit, while in the second case, the number of ICMP packets accepted is limited to 20 per minute, and only one packet can be accepted immediately (thanks to the '-limit-burst 1' option). This limitation prevents the firewall from being overloaded by too many ICMP packets.

Task 7

```
    Zadanie 7
    Z ad 7.1 Zapisz pełną składnię wywołania programu ptunnel, która na
maszynie A uruchomi klienta
    nasłuchującego na porcie 8000 i wysyłającego odebrany na tym porcie ruch
do serwera ssh działającego
     na maszynie B poprzez proces pośrednika (wpisz odpowiednie wartości w
miejsce kropek):
    ptunnel lp 8000 p ... da ... dp 22
    Uzupełnioną komendę zapisz w sprawozdaniu.
    Z ad 7.2 Napisz składnię wywołania komendy ssh otwierająca połączenie
ssh z maszyny A (klient ssh )
    do maszyny B (serwer ssh)
    Uzupełnioną komendę zapisz w sprawozdaniu.
    Pytania:
    Zbierz ruch wymieniany pomiędzy maszynami A i B (za pomocą programu
wireshark lub tcpdump).
    Pytanie 16:
    Czy widać jakąś różnicę pomiędzy pakietami ICMP generowanymi przez
program ptunnel a
```

```
    “normalnymi” pakietami ICMP
    Odpow
    i edź na pytanie z uzasadnieniem zapisz w sprawozdaniu.
```

```
kali@kali:~$ sudo ptunnel -lp 8000 -p 10.10.10.254 -da 10.10.10.254 -dp 22
[inf]: Starting ptunnel v 0.72.
[inf]: (c) 2004-2011 Daniel Stoedle, <daniels@cs.uit.no>
[inf]: Security features by Sebastien Raveau, <sebastien.raveau@epita.fr>
[inf]: Relaying packets from incoming TCP streams.
```



Connecting SSH on a kali machine

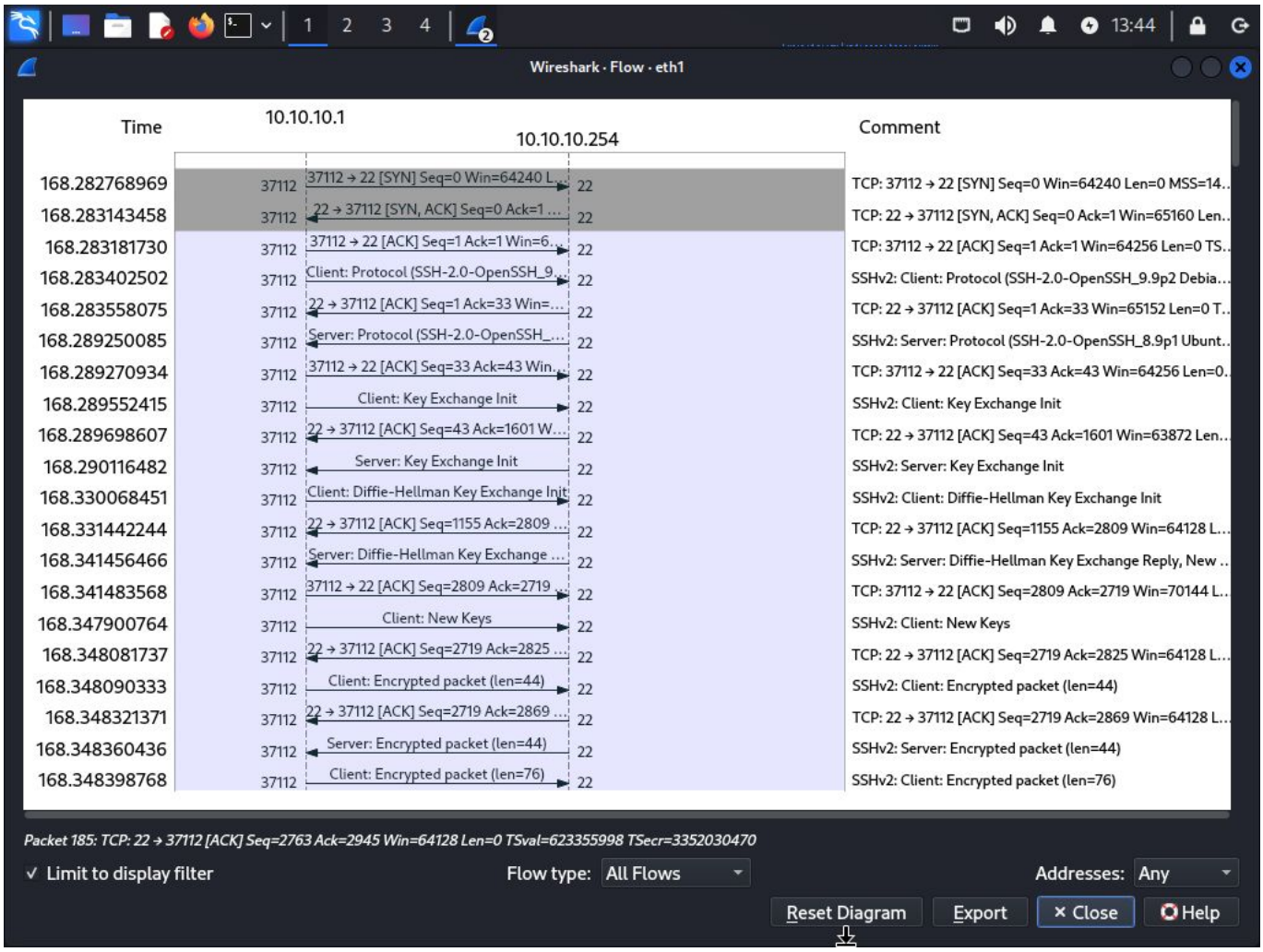
Questions

Is there any difference between ICMP packets generated by ptunnel and "normal" ICMP packets

Unfortunately but the tunnel is probably not working as it should because I'm making a mistake somewhere I've tried several different combinations but still nothing works as it should.

The SSH connection as seen in the picture sets up but does not go through the tunnel and the network traffic looks like this. It is no different from a normal SSH. If you have reached this point ;)

then I would be grateful if you could let me know what I was doing wrong at email `s10449ko@ms.wwsi.edu.pl`



SSH traffic and there should be ICMP instead