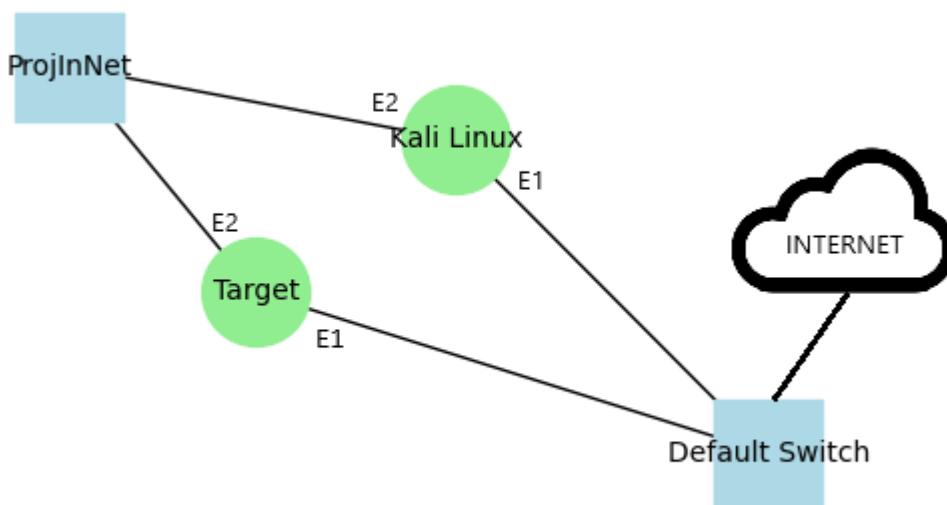


Security: iptables firewall na linux'ie

Wstęp

Infrastruktura wirtualna wykorzystana do wykonania zadania



Schemat środowiska

wirtualnego

Do wykonania tego zadania wykorzystano następujące środowisko wirtualne:

- Hypervisor: Hyper-V 10.0.26100.1882
- Maszyna Target: Ubuntu 24.04 Noble Numbat
- Maszyna Kali Linux: Kali GNU/Linux 2024.4
- Switch wirtualny ProjInNet: wirtualny switch dodany do hyper-v na cele tego przedmiotu
- Switch wirtualny Default Switch: jest to switch który działa jak prosty router NAT i pozwala na dostęp do internetu z maszyn wirtualnych

Adresacja IP od strony switcha ProjInNet:

- Kali Linux - 10.10.10.1
- Target - 10.10.10.254

Zadanie wstępne

Treść:

2.1 Osobisty Firewall

Sprawdź konfigurację programu iptables na komputerze wydając polecenie

```
iptables -L -v -n
```

Pytania:

1. Pytanie 1: jaka polityka włączona jest domyślnie na maszynie ?
2. Pytanie 2: Który łańcuch należy z modyfikować , aby chronić maszynę przed połączeniami zewnętrznymi ?

Odpowiedzi z wyjaśnieniami umieść w sprawozdaniu.

Rozwiązanie:

```
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:~$
```

Odpowiedzi na pytania:

jaka polityka włączona jest domyślnie na maszynie ?

Jest uruchomiona polityka ACCEPT czyli wszystkie pakiety będą przyjmowane przez maszynę ponieważ nie ma nic zablokowanego na firewallu.

Który łańcuch należy z modyfikować , aby chronić maszynę przed połączeniami zewnętrznymi ?

INPUT gdyż jest to łańcuch odpowiedzialny za ruch przychodzący do maszyny.

Zadanie 1

Treść i rozwiązania

Treść i rozwiązania przeplatają się ze sobą gdyż zadanie jest obszerne i składa się z wielu adnotacji

```
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

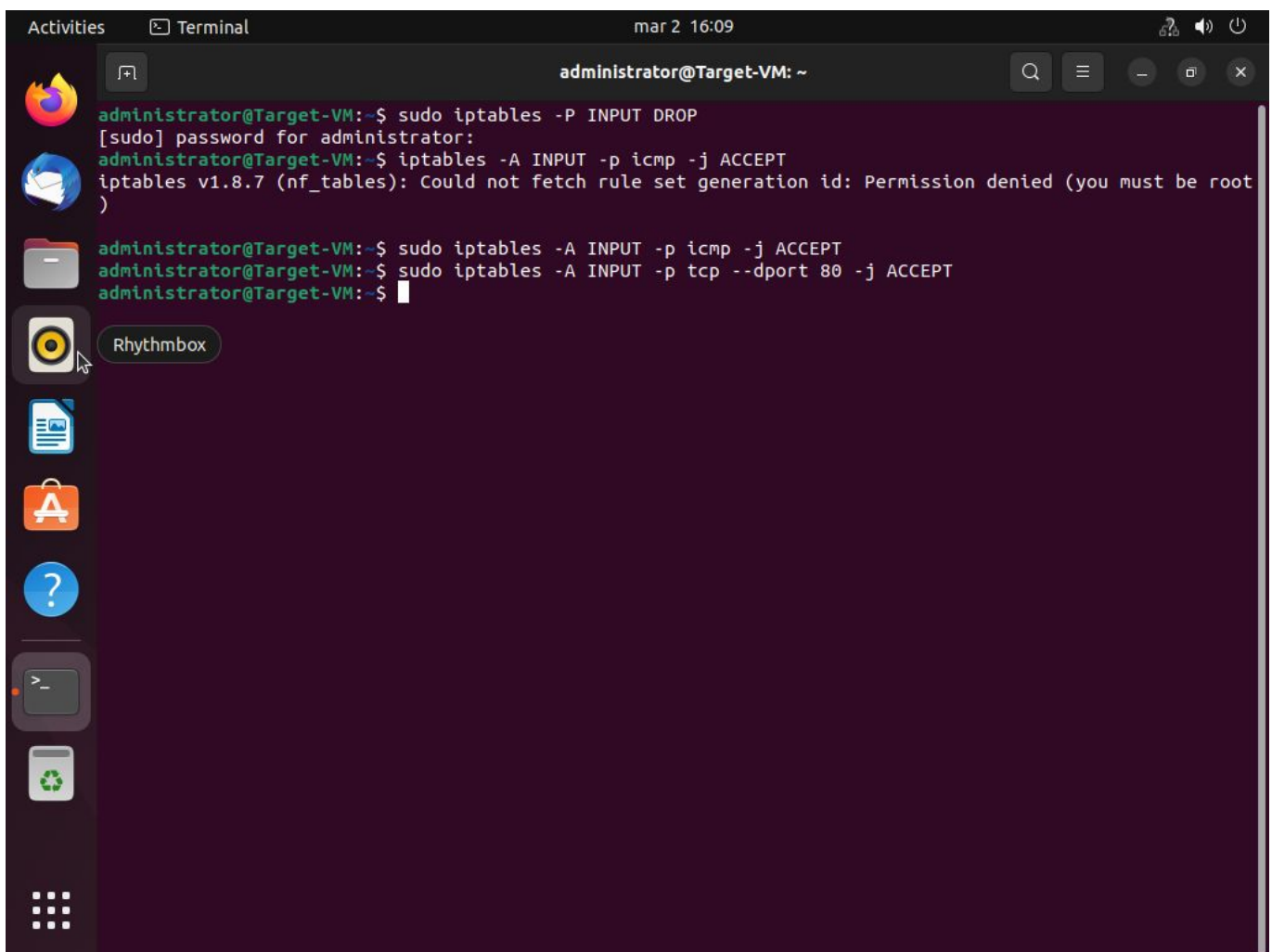
Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.02 seconds

kali@kali:~$
```

```
Zad 1.1. Napisz wywołanie iptables modyfikujące domyślną politykę
maszyny A
na taką, która będzie
odrzucać jakkolwiek ruch przychodzący (podpowiedź: należy zmodyfikować
domyślną politykę dla łańcucha
wejściowego 'INPUT' z ACCEPT na DROP). Wywołanie ma składnię:
iptables -P chain target
Zad 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 się poniżej):
iptables A ... p tcp dport ... j ...

Tutaj wykorzystano zdjęcie pulpitu zamiast listingu dlatego że połączenie po SSH zostało zablokowane więc nie mogłem skopiować wyników poleceń z maszyny Target

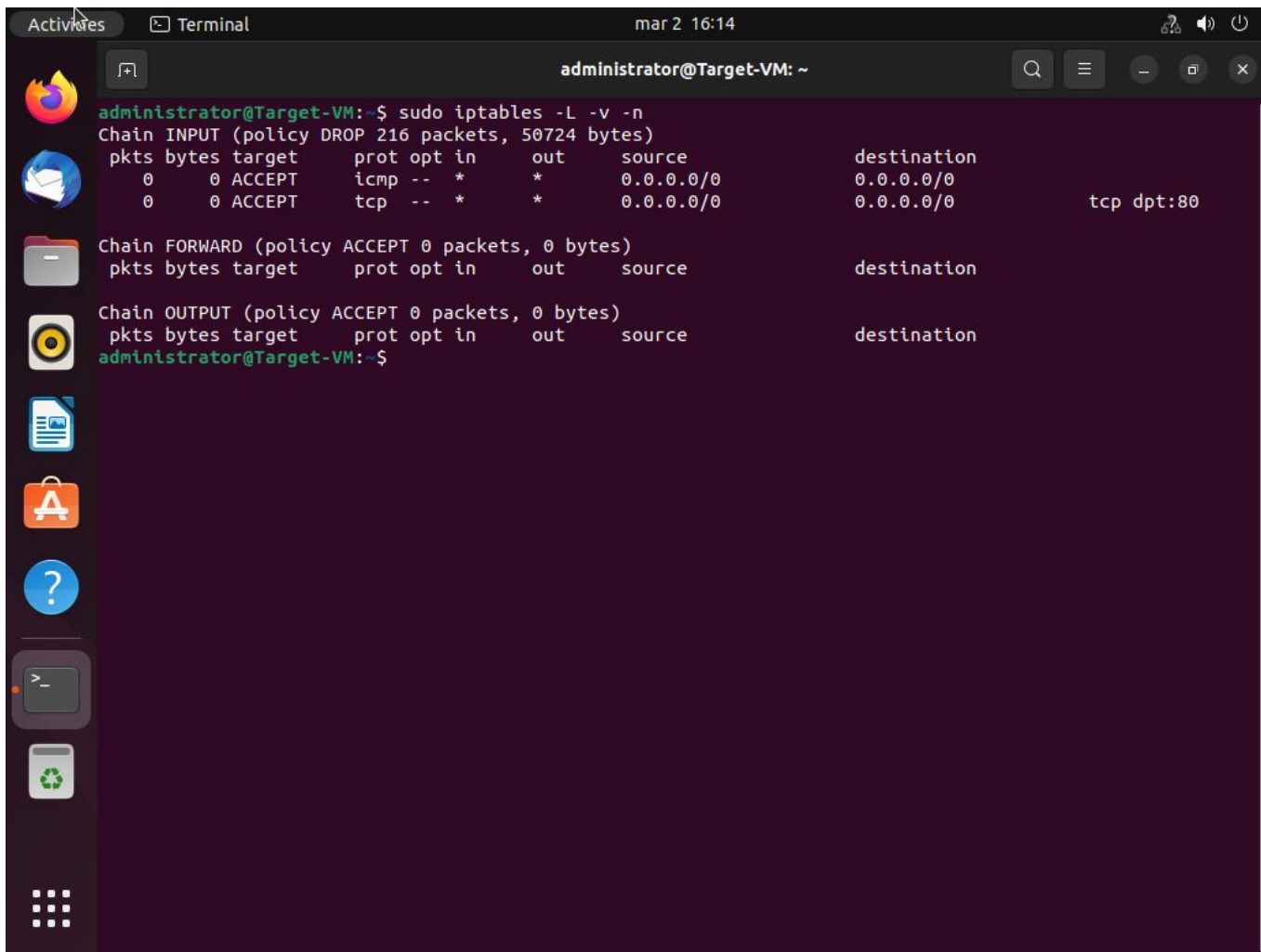
A screenshot of a Linux terminal window titled "administrator@Target-VM: ~". The terminal shows the following commands and output:

```
administrator@Target-VM:~$ sudo iptables -P INPUT DROP
[sudo] password for administrator:
administrator@Target-VM:~$ iptables -A INPUT -p icmp -j ACCEPT
iptables v1.8.7 (nf_tables): Could not fetch rule set generation id: Permission denied (you must be root)
administrator@Target-VM:~$ sudo iptables -A INPUT -p icmp -j ACCEPT
administrator@Target-VM:~$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
administrator@Target-VM:~$
```

The terminal window is part of a desktop environment with a sidebar on the left containing icons for Activities, Terminal, Rhythmbox, and other applications. The system clock at the top right shows "mar 2 16:09".

Wykonanie poleceń na maszynie Target

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).

A screenshot of a terminal window titled 'administrator@Target-VM: ~'. The terminal shows the output of the command 'sudo iptables -L -v -n'. The output displays three chains: INPUT, FORWARD, and OUTPUT. The INPUT chain has a policy of DROP and shows 216 packets and 50724 bytes. It lists two rules: one for ICMP and one for TCP. The FORWARD and OUTPUT chains both have a policy of ACCEPT and show 0 packets and 0 bytes. The terminal also shows the user prompt 'administrator@Target-VM:~\$' at the end.

```
administrator@Target-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy DROP 216 packets, 50724 bytes)
 pkts bytes target     prot opt in     out     source            destination
    0    0 ACCEPT     icmp -- *     *       0.0.0.0/0         0.0.0.0/0
    0    0 ACCEPT     tcp  -- *     *       0.0.0.0/0         0.0.0.0/0          tcp dpt:80

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:~$
```

Weryfikacja konfiguracji iptables na maszynie target

```
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 sprawozdaniu.

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.

```

administrator@Target-VM: ~
administrator@Target-VM:~$ sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
administrator@Target-VM:~$ sudo iptables -A INPUT -p tcp --dport 81 -j ACCEPT
administrator@Target-VM:~$ sudo iptables -L -v -n
Chain INPUT (policy DROP 459 packets, 99430 bytes)
 pkts bytes target     prot opt in     out     source         destination
    9   756 ACCEPT     icmp -- *     *       0.0.0.0/0      0.0.0.0/0
   13   845 ACCEPT     tcp  -- *     *       0.0.0.0/0      0.0.0.0/0      tcp dpt:80
    0     0 ACCEPT     tcp  -- *     *       0.0.0.0/0      0.0.0.0/0      tcp dpt:22
    0     0 ACCEPT     tcp  -- *     *       0.0.0.0/0      0.0.0.0/0      tcp dpt:81

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:~$
  
```

Wykonanie konfiguracji iptables na maszynie target

```

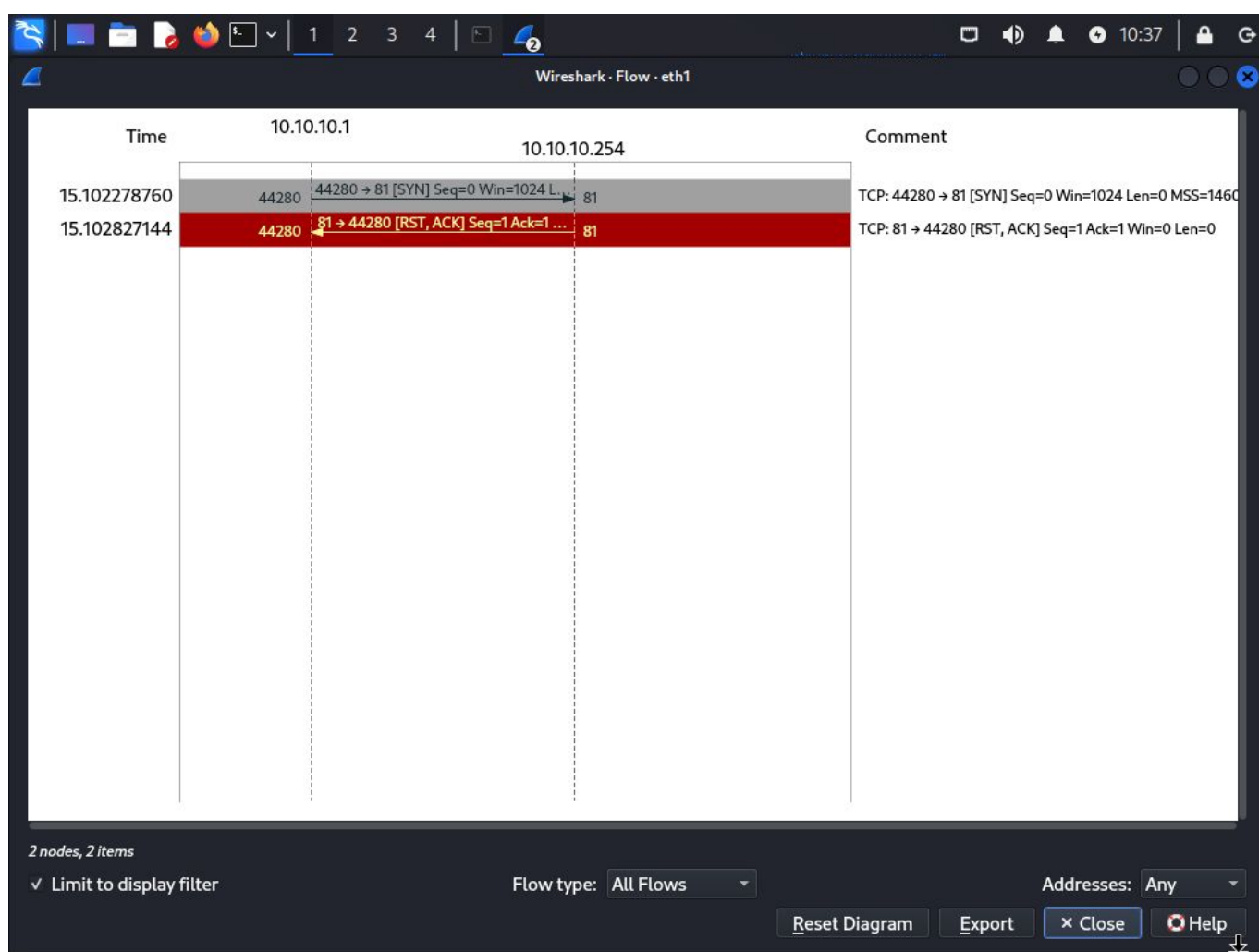
kali@kali:~$ nmap 10.10.10.254 -p 10-100
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-02 10:24 EST
Nmap scan report for 10.10.10.254
Host is up (0.00052s latency).
Not shown: 88 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh # widzimy że port 22 jest otwarty
80/tcp    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
  
```

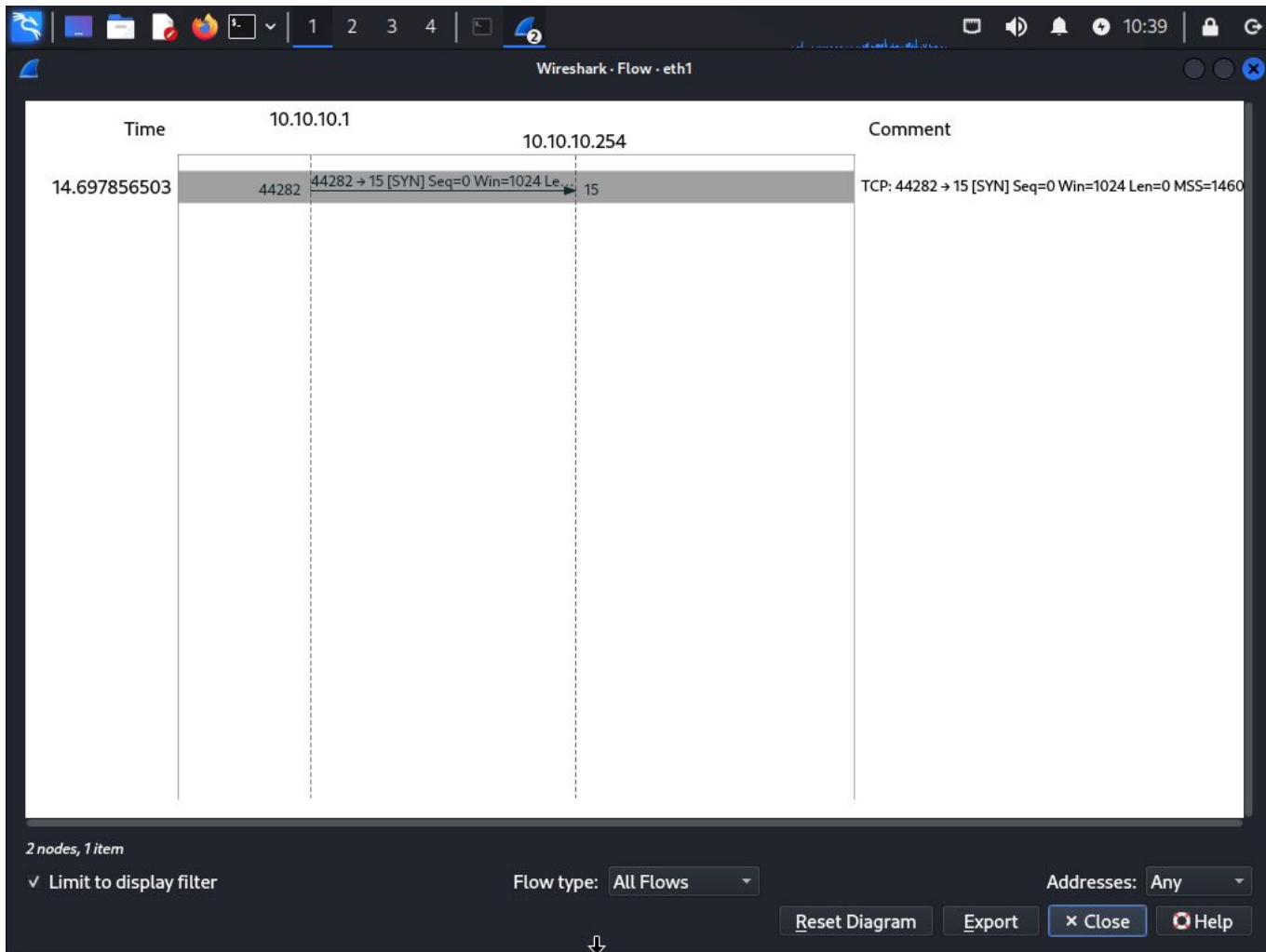
```
kali@kali:~$
```

Odpowiedz na pytania

W jaki sposób program nmap może odróżnić porty filtrowane od portów zamkniętych tych? Spróbuj określić to poprzez analizę ruchu wymienianego pomiędzy dwoma maszynami A i B (użyj programu wireshark lub tcpdump). Odpowiedź z uzasadnieniem umieść w sprawozdaniu. Następnie uruchom serwer WWW (Apache) na maszynie B i spróbuj połączyć się z nim za pomocą przeglądarki z maszyny A.

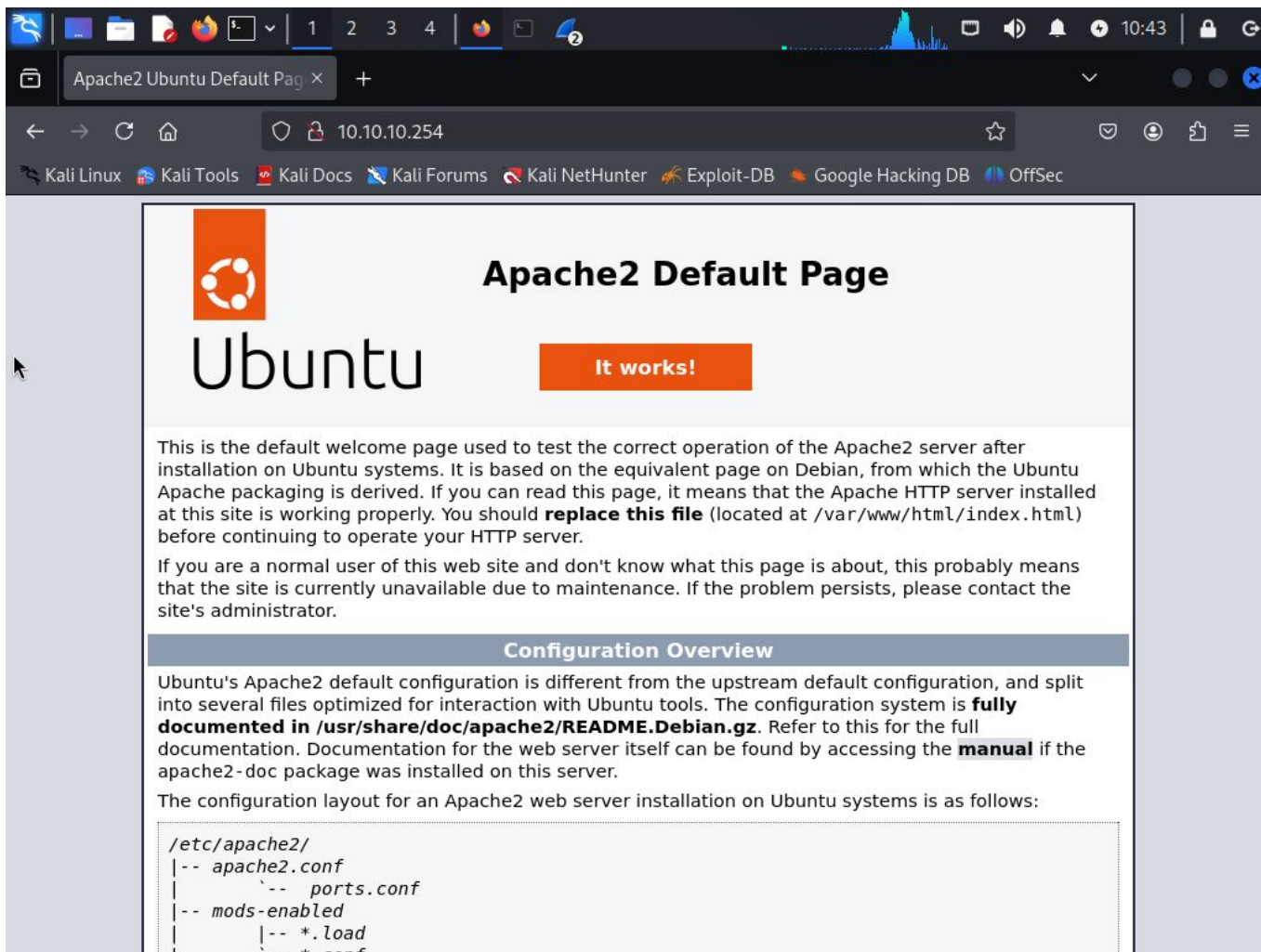


Analiza pakietów odpowiedzialnych za skanowanie portu 81



Analiza pakietów odpowiedzialnych za skanowanie portu 15

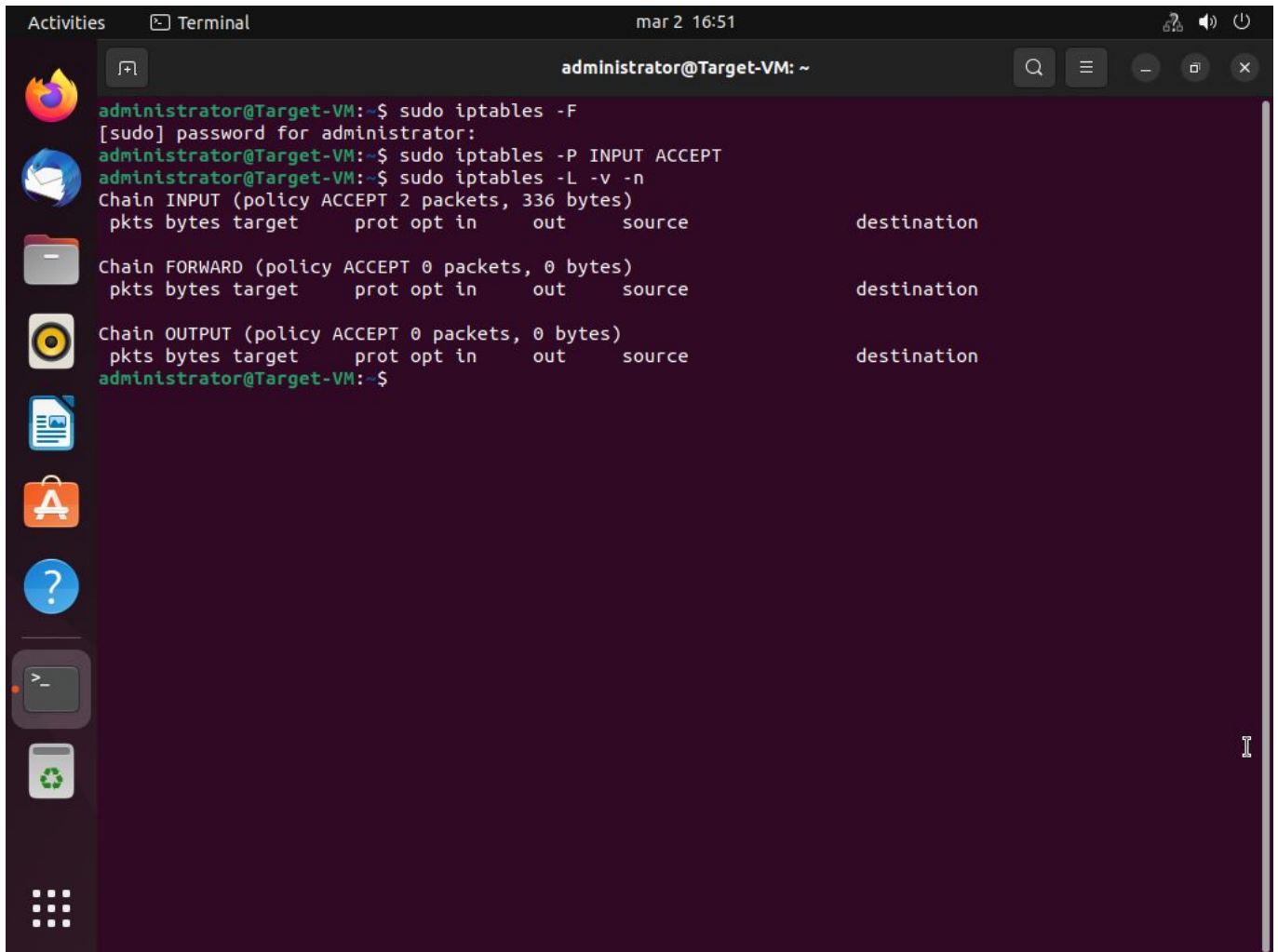
Po porównaniu tych dwóch analiz możemy wywnioskować że jeżeli port jest zablokowany to nie dociera on do samego systemu operacyjnego i jego stosu sieciowego natomiast jeżeli port jest odblokowany to wtedy stos sieciowy systemu odpowiada connection reset. Wtedy wiemy że na porcie nie ma żadnej usługi ale wiemy że jest on otwarty w firewallu.



Podłączenie przez przeglądarkę tak jak zażądano w zadaniu

Dlaczego użytkownik maszyny A nie może połączyć się z serwerem WWW na maszynie B ? Uzasadnij odpowiedź analizując ruch wymieniany pomiędzy maszynami A i B. Odpowiedź z uzasadnieniem umieść w sprawozdaniu. Zanim przejdziesz do kolejnych zadań, przywróć na maszynie A politykę akceptującą cały ruch (ACCEPT ALL) (podpowiedź: należy skasować wszystkie bieżące reguły a następnie określić regułę dla łańcucha INPUT). Teraz ponownie sprawdź możliwość połączenia z maszyny A do serwera HTTP na maszynie B. Wynik sprawdzenia zamieść w sprawozdaniu.

W zadaniu 1.3 wspominał Pan żeby dodać zasadę która pozwoli się zalogować na serwer WWW nie usuwałem tej zasady więc w moim przypadku połączenie nastąpiło poprawnie. Natomiast w przypadku kiedy takowa zasada by nie była w IP tables to nastąpiłaby identyczna sytuacja jak pokazana na rysunku 6, usługa Apache działająca na hoście nie dostała by informacji że ktoś chce się z nią łączyć i nie nastąpiłoby połączenie ponieważ zanim pakiet dostałby się do serwera WWW to zostałby zablokowany przez iptables



```
administrator@Target-VM: ~  
administrator@Target-VM:~$ sudo iptables -F  
[sudo] password for administrator:  
administrator@Target-VM:~$ sudo iptables -P INPUT ACCEPT  
administrator@Target-VM:~$ sudo iptables -L -v -n  
Chain INPUT (policy ACCEPT 2 packets, 336 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:~$
```

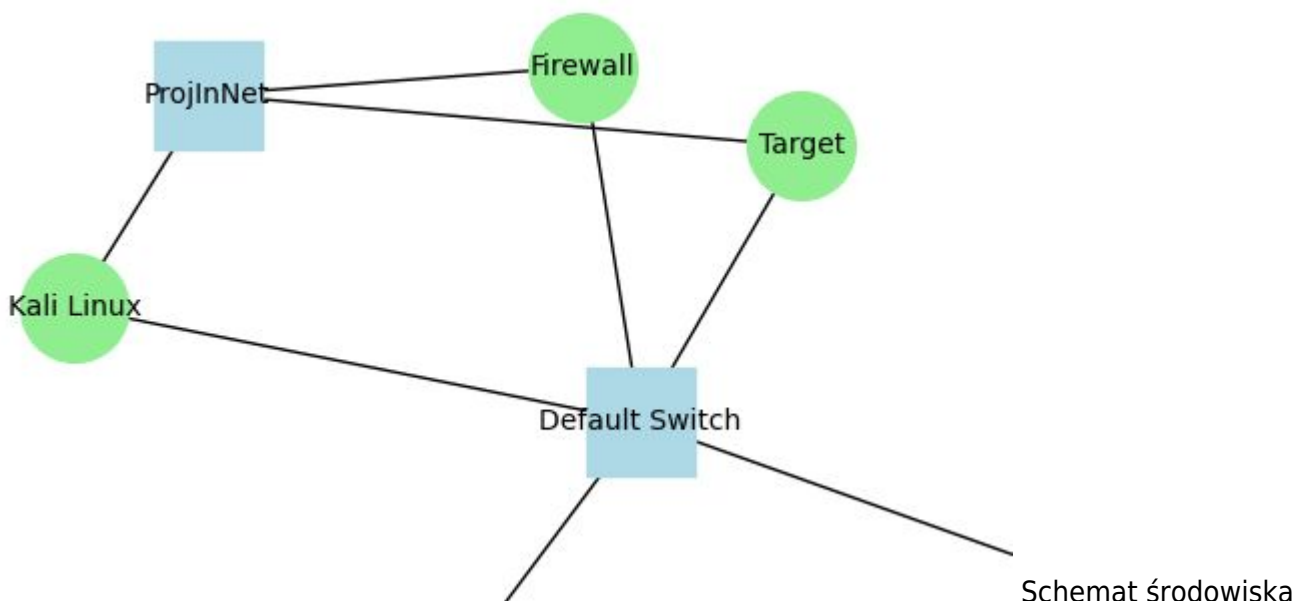
Przywrócenie stanu sprzed zmian

```
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:~$
```

Zadanie 2

Infrastruktura wirtualna wykorzystana do wykonania

zadania



wirtualnego

Schemat wygląda w taki sposób dlatego że został wykonany za pomocą mojego autorskiego skryptu w pythonie który generuje schemat sieci w hyper-v pobierając informację na temat środowiska z powershella.

Do wykonania tego zadania wykorzystano następujące środowisko wirtualne:

- Hypervisor: Hyper-V 10.0.26100.1882
- Maszyna Target: Ubuntu 24.04 Noble Numbat
- Maszyna Firewall: Ubuntu 24.04 Noble Numbat
- Maszyna Kali Linux: Kali GNU/Linux 2024.4
- Switch wirtualny ProjInNet: wirtualny switch dodany do hyper-v na cele tego przedmiotu
- Switch wirtualny Default Switch: jest to switch który działa jak prosty router NAT i pozwala na dostęp do internetu z maszyn wirtualnych

Adresacja IP od strony switcha ProjInNet:

- Kali Linux - 10.10.10.1 Maszyna B
- Target - 10.10.10.254 Maszyna A
- Firewall - 10.10.10.100 Maszyna C

Treść i rozwiązania przeplatają się ze sobą gdyż zadanie jest obszerne i składa się z wielu adnotacji

W tym ćwiczeniu udział biorą 3 maszyny (A, B i C), w tym maszyna C jako firewall. W celu przekazywania pakietów pomiędzy maszynami A i B za pośrednictwem maszyny C, konieczna jest modyfikacja tablic routingu maszyn A i B. W tym celu należy:

1. Na maszynie A skonfigurować routing tak aby pakiety adresowane do maszyny B przechodziły przez maszynę C:

```
route add host ip_maszyny_B gw ip_maszyny_C
```

2. Analogiczną zmianę 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) pomiędzy 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 komunikacji 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 wychodzący)

Na maszynie C zastosowano politykę pozwalającą u

żyć 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
```

Uruchomiono serwery WWW na Kali linux oraz na Target

```
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:~$
```

Pytania

Na którym łańcuchu działają powyższe komendy?

Komendy wykonane w listingu nr. 9 działają na łańcuchu FORWARD co oznacza że dotyczą pakietów przechodzących przez maszynę firewall

Czy z maszyny A można połączyć się z serwerem WWW działającym na maszynie 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
```

Jakie jest zadanie ostatniego z powyższych poleceń ?

Ostatnie polecenie ma na celu umożliwienie komunikacji na porcie TCP/80 z wszystkich możliwych adresów na adres 10.10.10.254. Innymi słowy żeby wszyscy mieli dostęp do serwera WWW na target VM. Literka D oznacza destination

Czy z maszyny A można połączyć się z serwerem WWW działającym na jakiejś innej maszynie jeżeli ruch ten przechodzi przez maszynę C (firewall) ?

Zgodnie z tym poleceniem `sudo iptables -A FORWARD -p tcp -s 10.10.10.254 -dport 80 -j ACCEPT` tak, ponieważ to polecenie pozwala na ruch gdzie źródłem jest target vm a cel nie jest podany więc możemy się dostać wszędzie. Literka S oznacza source.

Czy z maszyny B można połączyć się z serwerem WWW działającym na maszynie A (pomyśl co stanie się gdy użytkownik maszyny B w tej próbie wykorzysta port 80 jako port źródłowy) ?

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

Taka próba jest nie udana. Gdyby natomiast port źródłowy zamiast docelowy by był jako 80, to wtedy taki ruch zostałby przepuszczony przez firewall. Poniżej prezentacja jak wykorzystać narzędzie netcat aby wymusić użycie portu 80 jako źródłowego.

```
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:~$
```

Taki komunikat wskazuje, że serwer rozpoznał połączenie, ale żądanie HTTP było niepoprawne lub w ogóle nie spełniało oczekiwań serwera, co spowodowało błąd. Jednak fakt, że połączenie zostało nawiązane, pokazuje, że firewall nie zablokował połączenia, ponieważ port źródłowy był portem 80, co sprawiło, że taki ruch nie został odrzucony przez zapory sieciowe.

Czy z maszyny A można się łączyć z serwerem ssh na maszynie B i vice versa ? (pomyśl co stanie się gdy użytkownik maszyny B w tej próbie wykorzysta port 80 jako port źródłowy) ?

```
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:~$
```

Powyżej możemy zauważyć że jeżeli wykorzystamy port 80 jako port źródłowy to serwer SSH próbuje uzgodnić z nami komunikację. Sam fakt tego że pojawia się nagłówek SSH-2.0... oznacza że serwer SSH odpowiada.

Zadanie 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 komendy 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 połączenie 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/xouBf45WqBNlbbDVP2c.
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:~\$

Zadanie 4

Zadanie 4 (ruch icmp)

Zmień politykę na maszynie C tak, aby moż

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_maszyzny_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_maszyzny_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
icmp type 8
0    0  ACCEPT      icmp --  *      *      0.0.0.0/0          10.10.10.254
icmp type 0
0    0  ACCEPT      icmp --  *      *      0.0.0.0/0          10.10.10.254
icmp type 8
0    0  ACCEPT      icmp --  *      *      10.10.10.254      0.0.0.0/0
icmp type 0

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

Pytania

Dlaczego przepuszczanie ruchu ICMP bez żadnych restrykcji jest niebezpieczne ?

Przepuszczanie ruchu ICMP bez restrykcji jest niebezpieczne, ponieważ może być wykorzystywane w atakach DoS (np. ping flood), atakach typu Smurf, może ujawniać informacje o strukturze sieci oraz umożliwiać obejście zapór ogniowych, co zwiększa ryzyko kompromitacji systemu.

Zakładając, że maszyna C filtruje ruch do sieci 1.2.3.0/24 zamiast filtrowania tylko ruchu adresowanego do maszyny A, co się stanie gdy maszyna C odbierze pakiet ICMP E cho R equest z adresem przeznaczenia 1.2.3.255 wiedząc że przepuszcza ona ruch ICMP ?

Maszyna C przepuści pakiet ICMP, ponieważ jest on skierowany do dozwolonej sieci. Pakiet z adresem rozgłoszeniowym wywoła odpowiedzi ICMP od wszystkich urządzeń w tej sieci, co może prowadzić do niepożądanych efektów, takich jak przeciążenie sieci lub ujawnienie zbyt wielu informacji o urządzeniach w sieci.

Zadanie 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:~$
```

Pytania

Jak można rozwiązać ten problem przy użyciu filtra pakietów ?

Aby rozwiązać problem, należy dostosować reguły filtrujące w iptables, zezwalając na ruch na porcie 25 tylko dla wybranych adresów lub blokując ten port, jeśli nie jest potrzebny. Ważne jest precyzyjne określenie, które połączenia powinny być dozwolone, aby uniknąć nieautoryzowanego ruchu.

kontynuacja zadania

```
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 wydaniu

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
```

Pytania

Jakie jest znaczenie nowych reguł ? (podpowiedź: sprawdź informacje w sekcji "MATCH EXTENSIONS" po dręcznika iptables dostępnego po wydaniu polecenia "man iptables")

Nowe reguły w konfiguracji 'iptables' mają następujące znaczenie:

- Reguła 'iptables -A FORWARD -p tcp -s 10.10.10.254 -dport 80 -j ACCEPT' zezwala na połączenia TCP z adresu '10.10.10.254' na port 80 (HTTP) maszyny C.
- Reguła 'iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT' umożliwia przyjmowanie odpowiedzi na istniejące połączenia oraz połączenia powiązane, zapewniając poprawne działanie sesji sieciowych, takich jak HTTP.

Te reguły umożliwiają dostęp do serwera WWW oraz poprawne zarządzanie połączeniami sieciowymi.

Zadanie 6

Zadanie 6

. 0 graniczenie 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_maszyzny_A icmp type echo request j ACCEPT
poniższą regułą:
iptables
A FORWARD p icmp d ip_maszyzny_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.
```

Pytania

Jaki jest efekt zastąpienia reguły?

Zastąpienie reguły:

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

regułą:

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

proceedzi do wprowadzenia ograniczenia przepustowości na pakiety ICMP typu „echo request”. W pierwszym przypadku wszystkie pakiety są akceptowane bez limitu, podczas gdy w drugim przypadku liczba przyjmowanych pakietów ICMP jest ograniczona do 20 na minutę, a także tylko jeden pakiet może być przyjęty natychmiastowo (dzięki opcji ‘-limit-burst 1’). Ograniczenie to zapobiega nadmiernemu obciążeniu zapory przez zbyt dużą liczbę pakietów ICMP.

Zadanie 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:

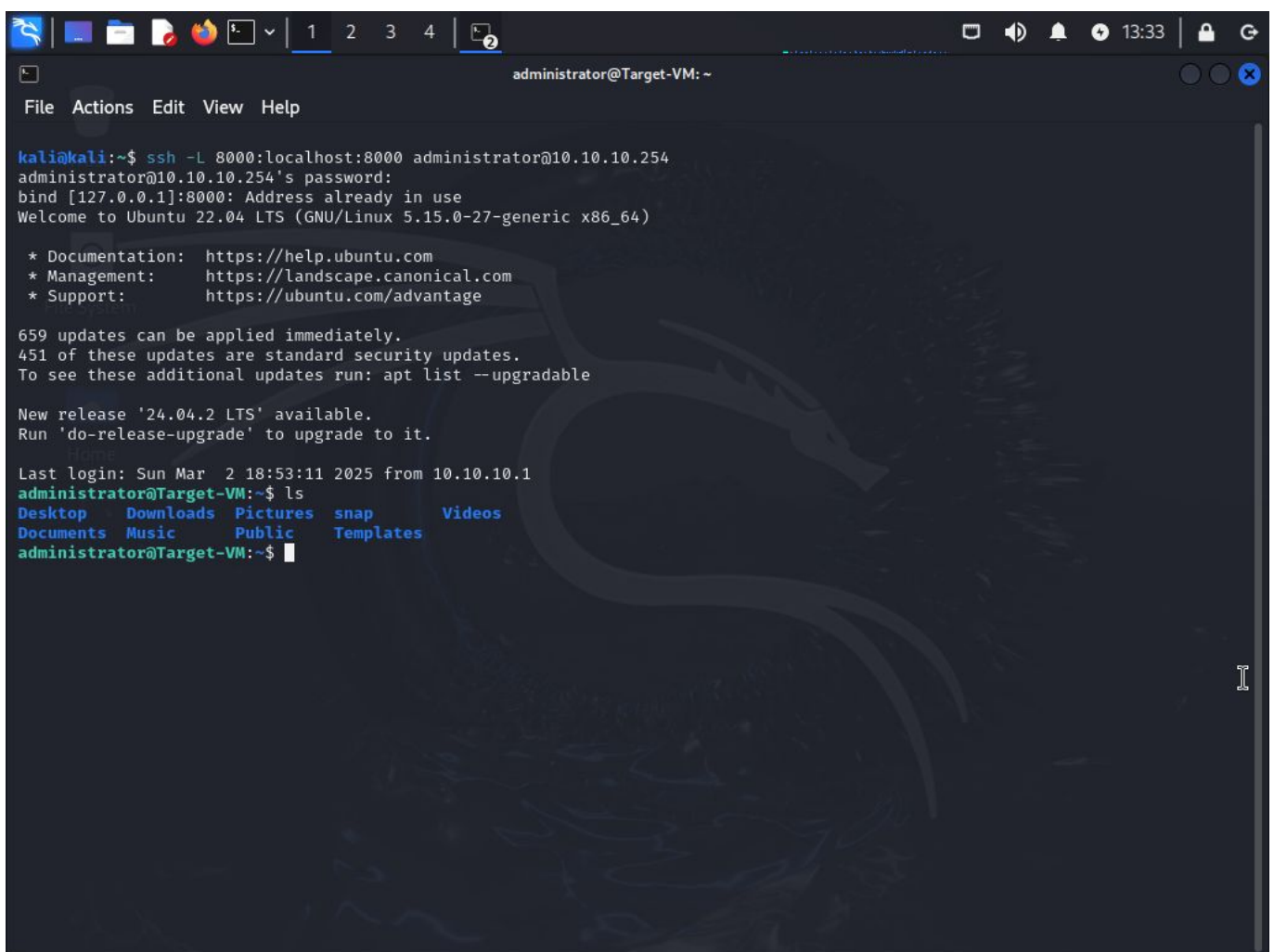
Czy widać jakąś różnicę pomiędzy pakietami ICMP generowanymi przez program ptunnel a

“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 Stuedle, <daniels@cs.uit.no>
[inf]: Security features by Sebastien Raveau, <sebastien.raveau@epita.fr>
[inf]: Relaying packets from incoming TCP streams.
```



```
administrator@Target-VM: ~
File Actions Edit View Help

kali@kali:~$ ssh -L 8000:localhost:8000 administrator@10.10.10.254
administrator@10.10.10.254's password:
bind [127.0.0.1]:8000: Address already in use
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-27-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

659 updates can be applied immediately.
451 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

New release '24.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Mar  2 18:53:11 2025 from 10.10.10.1
administrator@Target-VM:~$ ls
Desktop  Downloads  Pictures  snap      Videos
Documents Music      Public   Templates
administrator@Target-VM:~$
```

Podłączenie SSH na maszynie kali

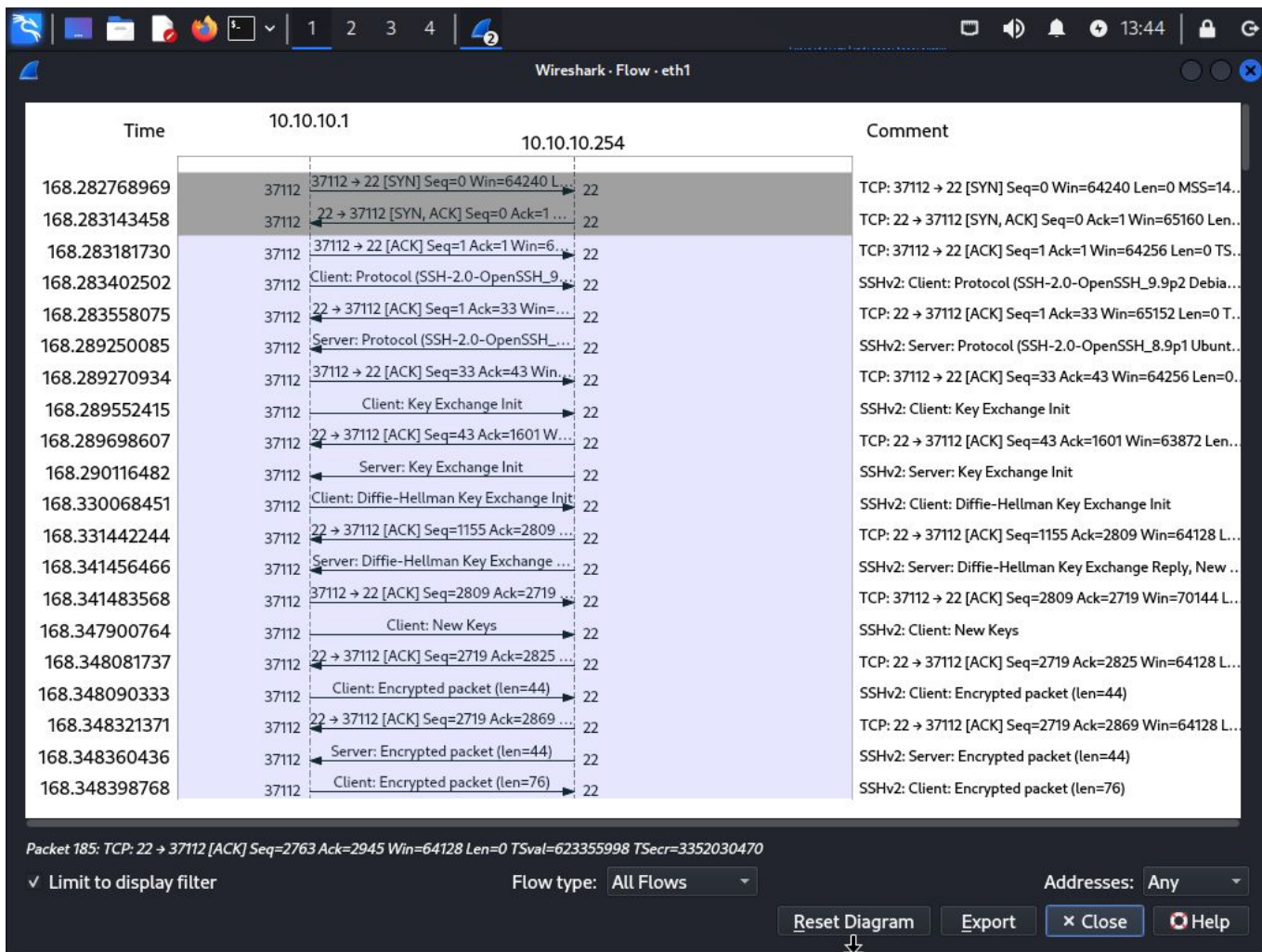
Pytania

Czy widać jakąś różnicę pomiędzy pakietami ICMP generowanymi przez

program ptunnel a "normalnymi" pakietami ICMP

Niestety ale tunel chyba nie działa tak jak powinien ponieważ popełniam gdzieś błąd próbowałem kilka różnych kombinacji ale nadal nic nie działa tak jak powinno.

Połączenie SSH tak jak widac na zdjęciu zestawia się ale nie przechodzi przez tunel i ruch sieciowy wygląda tak jak poniżej. Nie różni się niczym od zwykłego SSH. Jeżeli dotrwał Pan do tego momentu ;) to będę wdzięczny jak dałby Pan znać co robiłem nie tak na maila s10449ko@ms.wysi.edu.pl



Ruch SSH a zamiast tego powinien być ICMP