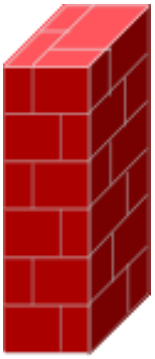
Ćwiczenia 11 – firewall, budowa i konfiguracja



1. Zaloguj się na swoje konto.
2. Na pierwszym terminalu:

```
andrzej@servubu:~$ sudo journalctl -f
```

3. Na 5 terminalu : man iptables
4. Wyczyścić wszystkie reguły w tablicy filter i nat oraz mangle

```
andrzej@servubu:~$ sudo iptables -F
andrzej@servubu:~$ sudo iptables -F -t nat
andrzej@servubu:~$ sudo iptables -F -t mangle
```

5. Sprawdź stan zapory.

```
andrzej@servubu:~$ sudo ufw status
Status: inactive
```

6. Ustawić dolne karty i ping do sąsiada. (Powinien działać)
7. Ustaw politykę na DROP w tablicy filter dla łańcuchów INPUT i FORWARD

```
andrzej@servubu:~$ sudo iptables -P INPUT DROP
andrzej@servubu:~$ sudo iptables -P FORWARD DROP
```

8. Ustaw politykę na ACCEPT w tablicy filter dla łańcucha OUTPUT

Sprawdzenie:

```
andrzej@servubu:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

```
andrzej@servubu:~$ sudo iptables -S
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
```

9. Dopuścić połączenia związane i ustanowione. Dopuścić ruch dla aplikacji działających na maszynie lokalnej. (loopback lo)

```
andrzej@servubu:~$ sudo iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACC
EPT
andrzej@servubu:~$ sudo iptables -S
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
```

10. Otworzyć możliwość sprawdzenia poleceniem ping (icmp echo reply request , kody 0 i 8) dla adresów z podsieci lokalnej.

```
andrzej@servubu:~$ sudo iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
andrzej@servubu:~$ sudo iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
```

Sprawdzenie:

```
andrzej@servubu:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere             icmp echo-reply
ACCEPT     icmp --  anywhere             anywhere             icmp echo-request
```

11. Sprawdź połączenie ssh:

```
administrator@administrator-VirtualBox:~$ ssh andrzej@10.20
.30.177
```

12. Na serwerze musi być zainstalowany pakiet openssh-server. Sprawdź działanie usługi ssh:

```
andrzej@servubu:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2023-01-27 08:54:30 CET; 18min ago
```

13. Otwórz port 22, na którym ma słuchać serwer ssh.

```
andrzej@servubu:~$ sudo iptables -A INPUT -p tcp -m state --state NEW --dport 22 -j ACCEPT
andrzej@servubu:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
```

14. Sprawdź połączenie na tym porcie z komputera sąsiada.

```
administrator@administrator-VirtualBox:~$ ssh andrzej@10.20
.30.177
The authenticity of host '10.20.30.177 (10.20.30.177)' can'
t be established.
ECDSA key fingerprint is SHA256:LwvsT952eSm1i8LeW5p5jt0oMgG
joc31Awoz0Pmdl2Q.
Are you sure you want to continue connecting (yes/no/[finge
rprint])? yes
Warning: Permanently added '10.20.30.177' (ECDSA) to the li
st of known hosts.
andrzej@10.20.30.177's password:
Welcome to Ubuntu 20.04.4 LTS (GNU/Linux 5.4.0-136-generic
```

15. Zapisz ustawienia w pliku /home/twoje_konto/iptables_rules_ddmmrrrr_hh:mm

```
andrzej@servubu:~$ sudo iptables-save > /home/andrzej/iptables_rules_26sty2023_14:35
andrzej@servubu:~$ cat /home/andrzej/iptables_rules_26sty2023_14:35
# Generated by iptables-save v1.8.4 on Thu Jan 26 14:35:17 2023
*filter
:INPUT DROP [61:9803]
:FORWARD DROP [0:0]
:OUTPUT ACCEPT [350:55079]
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
COMMIT
# Completed on Thu Jan 26 14:35:17 2023
```

16. Ruch wychodzący do portu 80 i 443 TCP ma być zablokowany.

```
-A OUTPUT -p tcp -m tcp --dport 80 -j DROP
-A OUTPUT -p tcp -m tcp --dport 443 -j DROP
```

17. Test w przeglądarce: lynx zsmeie.torun.pl (strona nie powinna się ładować)

18. Przywrócić ruch wychodzący po portach 80, 443.

```
Chain OUTPUT (policy ACCEPT)
num  target     prot opt source               destination
1    DROP       tcp  --  anywhere             anywhere
andrzej@servubu:~$ sudo iptables -D OUTPUT 1
```

19. Test w przeglądarce: lynx zsmeie.torun.pl (strona powinna się ładować)

20. Dopuścić ruch dla serwerów DNS dla cloudflare.

    Dla iptables:

```
-A INPUT -s 1.1.1.1/32 -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -s 1.0.0.1/32 -p udp -m udp --dport 53 -j ACCEPT
```

    Sprawdzenie:

```
andrzej@servubu:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLIS
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
ACCEPT     udp  --  one.one.one.one      anywhere             udp dpt:domain
ACCEPT     udp  --  one.one.one.one      anywhere             udp dpt:domain
```

21. Zapisz ustawienia w pliku /home/twoje_konto/iptables_rules_ddmmrrrr_hh:mm

```
andrzej@servubu:~$ sudo iptables-save > /home/andrzej/iptables_rules_26sty2023_15:10
```

22. Otworzyć port dla pracy serwera ftp-data, ftp, tftp, mysql, postfix(4 porty), dhcp i dhcpv6, http, https

```
andrzej@servubu:~$ sudo iptables -A INPUT -p tcp -m multiport --dports 20,21,69,3306,
,110,465,995,68,547,80,443 -j ACCEPT
andrzej@servubu:~$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISH
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
ACCEPT     udp  --  one.one.one.one      anywhere             udp dpt:domain
ACCEPT     udp  --  one.one.one.one      anywhere             udp dpt:domain
ACCEPT     tcp  --  anywhere             anywhere             multiport dports ftp-da
,ftp,69,mysql,smtp,pop3,submissions,pop3s,68,547,http,https
```

23. Zapisz ustawienia w pliku /home/twoje_konto/iptables_rules_ddmmrrrr_hh:mm

24. Zbuduj nat źródłowy dla sieci 10.11.12.0/24

Ustawienie karty:

```
andrzej@servubu:~$ sudo ip addr add 10.11.12.1/24 dev enp0s8
andrzej@servubu:~$ sudo ip link set enp0s8 up
andrzej@servubu:~$ ip -c a
```

Nat:

```
andrzej@servubu:~$ sudo iptables -t nat -A POSTROUTING -s 10.11.12.0/24 -j MASQUERADE
andrzej@servubu:~$ sudo iptables -t nat -S
-P PREROUTING ACCEPT
-P INPUT ACCEPT
-P OUTPUT ACCEPT
-P POSTROUTING ACCEPT
-A POSTROUTING -s 10.11.12.0/24 -j MASQUERADE
andrzej@servubu:~$ sudo iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
MASQUERADE  all  --  10.11.12.0/24        anywhere
```

25. Włącz forwardowanie pakietów tak, aby działało tylko do najbliższego restartu.

```
andrzej@servubu:~$ sudo su -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
andrzej@servubu:~$ sudo cat /proc/sys/net/ipv4/ip_forward
1
```

26. Wyczyścić wszystkie reguły w tablicy filter

```
andrzej@servubu:~$ sudo iptables -F
andrzej@servubu:~$ sudo iptables -S
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
```

27. Przywróć reguły z pliku:

```
andrzej@servubu:~$ sudo iptables-restore < /home/andrzej/iptables_rules_26sty2023_15:19
```

28. Sprawdzenie:

```
andrzej@servubu:~$ sudo iptables -F
andrzej@servubu:~$ sudo iptables -S
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
andrzej@servubu:~$ sudo iptables -S
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -s 1.1.1.1/32 -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -s 1.0.0.1/32 -p udp -m udp --dport 53 -j ACCEPT
-A INPUT -p tcp -m multiport --dports 20,21,69,3306,25,110,465,995,68,547,80,443 -j ACC
EPT
```

29. Zablokować ruch do Rosji i Chin. Zainstaluj pakiet dla whois.

    Sprawdź działanie:

```
andrzej@servubu:~$ whois 158.75.33.142
```

    Dla Chin blokowanie tylko 1 zakresu:

```
andrzej@servubu:~$ sudo iptables -A INPUT -p tcp -m iprange --src-range 42.0.0.0-42.255.255.25
5 -j DROP
```

30. Monitorować ruch narzędziem tcpdump. (W drugim terminalu uruchomić ping do dowolnej strony)

```
andrzej@servubu:~$ sudo tcpdump -i enp0s3
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
10:24:24.175509 IP servubu > www.wp.pl: ICMP echo request, id 1, seq 16, length 64
10:24:24.177205 IP servubu.41081 > netia-dns2.inetia.pl.domain: 53879+ [1au] PTR? 15.2.0.1
-addr.arpa. (51)
10:24:24.181270 IP www.wp.pl > servubu: ICMP echo reply, id 1, seq 16, length 64
```

31. Monitorować ruch narzędziem wireshark na stacji ubuntu-desktop dla karty dolnej.

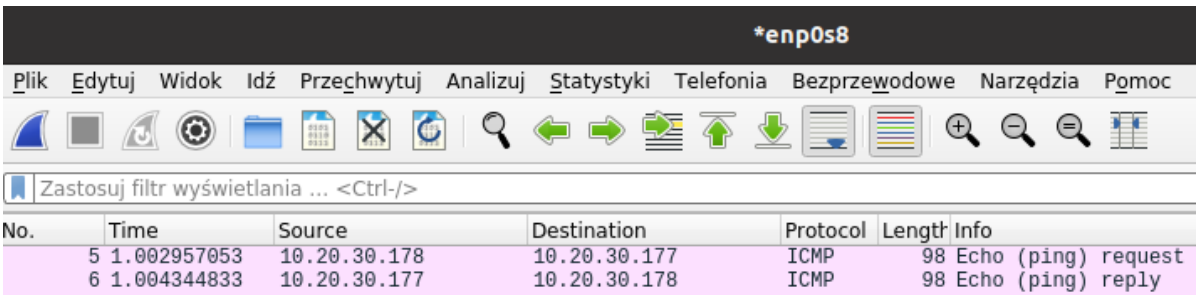
    Instalacja:

```
administrator@ubuntu:~$ sudo apt install wireshark-qt -y
[sudo] hasło użytkownika administrator:
Czytanie list pakietów... Gotowe
```

    Uruchomienie na stacji:

```
administrator@ubuntu:~$ sudo wireshark -i enp0s8
```

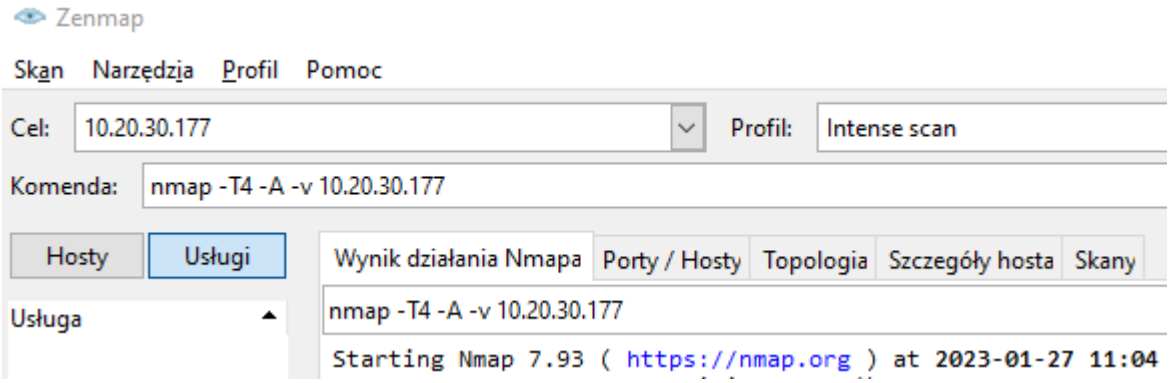
    Niebieska płetwa:



Zapisz ruchu do pliku o nazwie test.pcapng.

```
administrator@ubuntu:~$ ls
Dokumenty  Muzyka  Obrazy  Pobrane  Publiczny  Pulpit  Szablony  test.pcapng
```

32. Monitorować ruch narzędziem zen-map z poziomu stacji windows.



33. Sprawdzić otwarte porty na maszynie z pomocą narzędzia nmap np. port 22 dla ssh.

```
andrzej@servubu:~$ sudo nmap -sS -p 22 localhost
Starting Nmap 7.80 ( https://nmap.org ) at 2023-01-27 10:17 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000059s latency).

PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 0.11 seconds
```

34. Sprawdzić otwarte porty na maszynie z pomocą narzędzia netcat.

Na stacji ubuntu:

```
administrator@ubuntu:~$ nc -z -v 10.20.30.177 22
Connection to 10.20.30.177 22 port [tcp/ssh] succeeded!
```

35. Sprawdź pozostałe otwarte porty na swoim serwerze.
36. KONIEC.