

T: Instalacja i konfigurowanie firewalla (iptables).

Cel ogólny lekcji: Nauczenie się instalacji, konfiguracji i uruchamiania firewalla (iptables) w systemie operacyjnym Linux.

Cele szczegółowe lekcji:

1. Wyjaśnienie pojęć związanych z iptables.
2. Zainstalowanie iptables.
3. Uruchomienie lub zatrzymanie iptables.
4. Konfigurowanie iptables.
5. Korzystanie z iptables.
6. Wyjaśnienie procedury instalacji i konfiguracji oraz uruchomienia iptables.
7. Wykonanie testowania uruchomionego iptables.
8. Wyświetlenie domyślnej bramy dla interfejsów sieciowych serwera.
9. Opisanie wykonanych poleceń w zeszycie.

Podczas wykonywania poniższych zadań w zeszycie w sprawozdaniu

1. podaj i wyjaśnij polecenia, które użyjesz, aby:

- wyjaśnić pojęcia związane z iptables,
- zainstalować iptables,
- uruchomić lub zatrzymać iptables,
- konfigurować iptables,
- korzystać z iptables.

2. podaj odpowiedzi na pytania zadane w treści zadań.

Przywróć migawkę „Migawka 1” zawierającą przygotowane do ćwiczeń maszyny Ubuntu serwer i desktop (klient) oraz Windows desktop (klient)

Przed przystąpieniem do ćwiczenia sprawdź czy ustawienie maszyny wirtualnej pozwala na dostęp do Internetu, jeżeli ustawienia są niezgodne wykonaj konfigurację pierwszej i drugiej karty sieciowej według instrukcji, a następnie uruchom Ubuntu serwer i klienty zgodnie z wymaganiami w instrukcji.

<p>Ubuntu serwer Adapter 1</p> <p>Sieć</p> <p>Karta 1 Karta 2 Karta 3</p> <p><input checked="" type="checkbox"/> Włącz kartę sieciową</p> <p>Podłączona do: NAT</p> <p>Nazwa: <input type="text"/></p> <p>Zaawansowane</p>	<p>Ubuntu serwer Adapter 2</p> <p>Sieć</p> <p>Karta 1 Karta 2 Karta 3 Karta 4</p> <p><input checked="" type="checkbox"/> Włącz kartę sieciową</p> <p>Podłączona do: Sieć wewnętrzna</p> <p>Nazwa: <input type="text" value="intnet"/></p> <p>Zaawansowane</p>
<p>Windows Adapter 1</p> <p>Sieć</p> <p>Karta 1 Karta 2 Karta 3 Karta 4</p> <p><input checked="" type="checkbox"/> Włącz kartę sieciową</p> <p>Podłączona do: Sieć wewnętrzna</p> <p>Nazwa: <input type="text" value="intnet"/></p>	

Po uruchomieniu Ubuntu serwer

podaj **login: root Password: 1234** lub **login: ubuntu Password: ubuntu**

Jeśli zalogowałeś się do ubuntu wpisz **sudo -s Password: ubuntu**

```
ubuntu@ubuntusrv:~$ sudo -s
[sudo] password for ubuntu:
root@ubuntusrv:~/home/ubuntu#
```

Przygotowanie do ćwiczenia. Przywróć migawkę z ustawieniami sieci jak poniżej lub wykonaj poniższe ustawienie adresu dynamicznego przydzielanego z NAT i statycznego adresu IP.

1. Za pomocą polecenia **ifconfig -a** lub **ip a** ustal dostępne interfejsy sieciowe.

```
root@ubuntusrv:~# ifconfig -a
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
  inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
  inet6 fe80::a00:27ff:febe:d52b prefixlen 64 scopeid 0x20<link>
  ether 08:00:27:be:d5:2b txqueuelen 1000 (Ethernet)
  RX packets 195929 bytes 258698845 (258.6 MB)
  RX errors 0 dropped 0 overruns 0 frame 0
  TX packets 16647 bytes 1045394 (1.0 MB)
  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
```

Plik **/etc/netplan/00-installer-config.yaml** - opisuje interfejsy sieciowe dostępne w systemie i jak je aktywować.

2. Zmień adres IP dla Ubuntu na Adapter 2 na statyczny.

Otwórz plik, który opisuje interfejsy sieciowe **nano /etc/netplan/0** tabulator – nazwa pliku zostanie uzupełniona do postaci ***.yaml**

Pozostaw zalecane wpisy w tym pliku jak poniżej pamiętaj o dokładności wpisów

```
# This is the network config wri
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      addresses: [10.0.0.30/24]
```

3. Zastosuj ustawienia

```
root@ubuntusrv:~# netplan apply
```

4. Zmień nazwę hosta na stałe

```
root@ubuntusrv:~# hostnamectl set-hostname dlp
```

```
root@ubuntusrv:~# init 6
```

```
root@dlp:~# netplan apply
```

5. Wyświetl domyślną bramę (adres routera) dla interfejsów sieciowych serwera

```
root@dlp:~# ip route show default
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
```

Jeżeli jeszcze nie masz wykonaj migawkę z skonfigurowanymi interfejsami sieciowymi a będzie łatwiej kolejnym razem.

Zapisz w zeszycie co się stało po wykonaniu poleceń. Wpisz kolejno polecenia.

Opisz w zeszycie:

- procedurę instalacji i konfiguracji oraz uruchomienia iptables,
- testowania uruchomionego iptables,

Iptables to program sterujący filtrem pakietów (głównie używanym jako zaporą sieciową bądź NAT) opracowany dla systemu operacyjnego Linux.

Program może być używany jako filtr pakietów, bądź tzw. stanowa zaporą dla systemów Linux z jądrem począwszy od serii 2.4.x, kontrolujący połączenia wchodzące i wychodzące do sieci komputerowej lub stacji roboczej.

Wymaga jądra skompilowanego z modułem ip_tables.

Iptables wymaga uprawnień roota do uruchomienia.

W większości dystrybucji Linuksowych iptables jest instalowane w katalogu /usr/sbin/iptables, jednakże w niektórych z nich można go znaleźć w /sbin/iptables.

A. Zapoznanie z iptables

Po zainstalowaniu Ubuntu firewall - którym jest iptables jest nie skonfigurowany i cały ruch przychodzący i wychodzący jest otwarty.

W terminalu wpisz `iptables -L`

zobaczysz

Chain INPUT (policy ACCEPT)

target prot opt source destination

Chain FORWARD (policy ACCEPT)

target prot opt source destination

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

Elementy ćwiczenia:

1. Opis funkcji
2. System V czy SystemD - określenie programu inicjalizującego
3. Tworzenie skryptu firewalla dla System V
4. Usuwanie skryptu
5. Błędy skryptu
6. Skrypt zaawansowany
7. Tworzenie skryptu firewalla dla Systemd
8. Ufw - frontend dla iptables

Zapisz w zeszycie **opis funkcji**

INPUT – to pakiety odbierane, wchodzące do naszego komputera

OUTPUT– to pakiety wygenerowane, wychodzące z naszego komputera

FORWARD – to pakiety przekazywane, przechodzące przez nasz komputer

ACCEPT - akceptuje pakiet, wpuszcza pakiet do komputera

DROP - odrzuca pakiet, nie wpuszcza pakietu do komputera.

REJECT - odrzuca pakiet i odsyła z powrotem do źródła, powiadamiając o tym nadawcę

System V czy SystemD - określenie programu inicjalizującego

Sysvinit (System V) - jest to pierwszy program uruchamiany w systemach Linux przez jądro w trakcie procesu uruchamiania systemu operacyjnego. Następnie na podstawie plików konfiguracyjnych lub skryptów startowych init uruchamia pozostałe procesy systemowe. Sysvinit był domyślnym w Debianie, łącznie z wersją Wheezy (7.0).

Debian Jessie (8.0) posiada Systemd - program inicjalizujący, zarządzający systemem i usługami (units - programy działające w tle, uruchomione jednorazowo lub cyklicznie)

Sprawdź czy Ubuntu używa Systemd wpisz w terminalu **systemd --version**

Otrzymasz podobny wynik

```
systemd 237
```

```
+PAM +AUDIT +SELINUX +IMA +SYSVINIT +LIBCRYPTSETUP +GCRYPT +ACL +XZ -SECCOMP -  
APPARMOR
```

Oznacza to, że mamy zainstalowany Systemd w wersji 237. W takim wypadku tworzysz skrypt dla Systemd ([patrz poniżej - Tworzenie skryptu firewala dla Systemd](#))

Możesz użyć polecenia

```
dpkg -S /sbin/init
```

otrzymasz wynik

```
systemd-sysv: /sbin/init
```

Oznacza to, że Ubuntu startuje używając Systemd, jeśli

```
sysvinit: /sbin/init
```

Oznacza, że Ubuntu startuje używając System V

Uzasadnij odpowiedz. Jak jest program inicjalizujący.

B. Tworzenie skryptu firewala dla System V

Skrypt firewala, będzie się uruchamiał wraz ze startem komputera.

(z materiałów pobierz firewall.iso

Podłącz Urządzenia > Napędy optyczne > firewall.iso

Podmontuj firewall.iso `mount /dev/cdrom /media/cdrom`

```
root@dlp:~# mount /dev/cdrom /media/cdrom
```

Przeanalizuj przykładowy skrypt firewala na początku musi być ten kod

```
#!/bin/bash
```

```
### BEGIN INIT INFO
```

```
# Provides:          firewall.sh
```

```
# Required-Start:   $local_fs $remote_fs
```

```
# Required-Stop:   $local_fs $remote_fs
```

```
# Default-Start:   2 3 4 5
```

```
# Default-Stop:    0 1 6
```

```
# Short-Description: Start daemon at boot time
```

```
# Description:      Enable service provided by daemon.
```

```
### END INIT INFO
```

Przekopiuj skrypt firewala z pliku firewall1.txt do pliku firewall (zapisz plik nadając mu nazwę firewall).

Skopiujesz plik firewall do katalogu init.d. W katalogu tym znajdują się skrypty startowe uruchamiane podczas ładowania systemu przez inita.

Wpisz ponownie ls by zobaczyć czy tam jest nasz plik firewall

```
root@d1p:~# cp /media/cdrom/firewall1.txt /etc/init.d/firewall
root@d1p:~# ls /etc/init.d/firewall
/etc/init.d/firewall
```

Wyświetl przykładowa konfiguracja iptables blokująca połączenia przychodzące

```
cat /etc/init.d/firewall
```

```
#!/bin/bash
```

```
### BEGIN INIT INFO
```

```
# Provides: firewall
```

```
# Required-Start: $local_fs $remote_fs
```

```
# Required-Stop: $local_fs $remote_fs
```

```
# Default-Start: 2 3 4 5
```

```
# Default-Stop: 0 1 6
```

```
# Short-Description: Start daemon at boot time
```

```
# Description: Enable service provided by daemon.
```

```
### END INIT INFO
```

```
# Czyszczenie reguł
```

```
iptables -F
```

```
iptables -X
```

```
# Polityka bezpieczeństwa
```

```
iptables -P INPUT DROP
```

```
iptables -P FORWARD DROP
```

```
iptables -P OUTPUT DROP
```

```
# Reguły dla pętli zwrotnej
```

```
iptables -A INPUT -i lo -j ACCEPT
```

```
iptables -A FORWARD -o lo -j ACCEPT
```

```
iptables -A OUTPUT -o lo -j ACCET
```

```
# Reguły dla łańcucha wejściowego
```

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# Reguły dla łańcucha wyjściowego
```

```
iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
```

Następnie, jeśli nie ma błędów nadaj plikowi firewall prawa uruchamiania

```
chmod +x /etc/init.d/firewall
```

Utwórz dowiązanie do uruchomienia serwisu w trybach pracy 2345 oraz zatrzymania serwisu w trybach pracy 016 - wpisując

```
update-rc.d firewall defaults 90
```

update-rc.d - instaluje lub usuwa dowiązania do skryptów startowych w stylu System-V

Uruchom ponownie komputer (**init 6**). Wraz ze startem zestaw reguł filtrujących powinien wystartować.

Po ponownym uruchomieniu, aby to sprawdzić wpisz

```
iptables -L
```

Zobaczysz reguły

Możesz użyć również komendy do zatrzymania firewalla

```
/etc/init.d/firewall stop
```

lub do jego uruchomienia

```
/etc/init.d/firewall start
```

Błędy skryptu

Jeśli stworzysz skrypt firewall dla sysvinit, używając go w systemd otrzymasz przy starcie systemu komunikat - systemd-sysv-generator [179]: Ignoring creation of an alias firewall.service for itse. Oznacza to, że skrypt nie został stworzony dla systemd i firewall.service został zignorowany. Mimo to skrypt ten został uruchomiony i działa.

Aby to sprawdzić wpisz w terminalu komendę

```
systemctl status firewall
```

Otrzymasz podobny wynik w terminalu

```
firewall.service - LSB: Start daemon at boot time
```

```
Loaded: loaded (/etc/init.d/firewall)
```

```
Active: active (exited) since nie 2016-11-06 09:13:49 CET; 1h 5min ago
```

```
Process: 462 ExecStart=/etc/init.d/firewall start (code=exited, status=0/SUCCESS)
```

```
lis 06 09:13:44 debian systemd[1]: Starting LSB: Start daemon at boot time...
```

```
lis 06 09:13:49 debian systemd[1]: Started LSB: Start daemon at boot time.
```

```
lis 06 10:17:18 debian systemd[1]: Started LSB: Start daemon at boot time.
```

Active: active (exited) - oznacza, że skrypt działa, ale nie wie, gdzie jest jego demon by go monitorować. Jeśli istnieje musimy go zdefiniować w pliku unita, konfigurując opcje Type i ExecStart, zgodnie z dokumentacją systemd.

Poproś o sprawdzenie tej części zadania

C. Usuwanie skryptu i skrypt zaawansowany

Aby usunąć skrypt musisz wejść do katalogu init.d i usunąć plik firewall wpisując w terminalu

```
rm /etc/init.d/firewall
```

Następnie musisz usunąć dowiązanie do skryptu

```
update-rc.d firewall remove
```

Poproś o sprawdzenie tej części zadania

Wykonaj skrypt zaawansowany korzystając z pliku `firewall2`.

```
#!/bin/bash
```

```
### BEGIN INIT INFO
```

```
# Provides:      firewall.sh
```

```
# Required-Start: $local_fs $remote_fs
```

```
# Required-Stop:  $local_fs $remote_fs
```

```
# Default-Start:  2 3 4 5
```

```
# Default-Stop:   0 1 6
```

```
# Short-Description: Start daemon at boot time
```

```
# Description:    Enable service provided by daemon.
```

```
### END INIT INFO
```

```
# CZYSZCZENIE STARYCH REGUŁ
```

```
iptables -F
```

```
iptables -X
```

```
iptables -F -t nat
```

```
iptables -X -t nat
```

```
iptables -F -t filter
```

```
iptables -X -t filter
```

```
# USTAWIENIE POLITYKI DZIAŁANIA
```



```

iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-request -j REJECT --reject-with icmp-host-unreachable
# OCHRONA PRZED SKANOWANIEM ACK SCAN
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH ACK -j LOG
--log-prefix "ACK scan: "
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH ACK -j
DROP # Metoda ACK (nmap -sA)
#OCHRONA PRZED SKANOWANIEM FIN SCAN
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH FIN -j LOG -
-log-prefix "FIN scan: "
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH FIN -j DROP
# Skanowanie FIN (nmap -sF)
#OCHRONA PRZED SKANOWANIEM XMAS TREE SCAN
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH PSH -j LOG
--log-prefix "Xmas scan: "
iptables -A INPUT -m conntrack --ctstate NEW -p tcp --tcp-flags SYN,RST,ACK,FIN,URG,PSH
FIN,URG,PSH -j DROP # Metoda Xmas Tree (nmap -sX)
#OCHRONA PRZED SKANOWANIEM NULL SCAN
iptables -A INPUT -m conntrack --ctstate INVALID -p tcp ! --tcp-flags SYN,RST,ACK,FIN,PSH,URG
SYN,RST,ACK,FIN,PSH,URG -j LOG --log-prefix "Null scan: "
#OCHRONA PRZED ATAKIEM Dos
iptables -A INPUT -m conntrack --ctstate INVALID -p tcp ! --tcp-flags SYN,RST,ACK,FIN,PSH,URG
SYN,RST
iptables -N syn-flood
iptables -A INPUT -p tcp --syn -j syn-flood
iptables -A syn-flood -m limit --limit 1/s --limit-burst 4 -j RETURN
iptables -A syn-flood -m limit --limit 1/s --limit-burst 4 -j LOG --log-prefix "SYN-flood: "
iptables -A syn-flood -j DROP
# OCHRONA PRZED ATAKIEM PING OF DEATH

```

```
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j LOG --log-prefix "Ping: "
```

```
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT # Ping of death
```

ZABLOKOWANIE PINGOWANIA

```
iptables -A INPUT -p icmp --icmp-type echo-request -j REJECT --reject-with icmp-host-unreachable
```

BLOKOWANIE TELNETU

```
iptables -A OUTPUT -p tcp --dport telnet -j REJECT
```

```
iptables -A INPUT -p tcp --dport telnet -j REJECT
```

#ZAPIS DO LOGA ODRZUCONYCH PAKIETÓW PRZYCHODZĄCYCH W KATALOGU

var/log/messages

```
iptables -N LOGGING
```

```
iptables -A INPUT -j LOGGING
```

```
iptables -A LOGGING -m limit --limit 2/min -j LOG --log-prefix "IPTables-Dropped: " --log-level 4
```

```
iptables -A LOGGING -j DROP
```

Poproś o sprawdzenie

D. Tworzenie skryptu firewala dla Systemd

Utwórz skrypt firewala dla systemd wykorzystując tym razem sam systemd.

Utwórz skrypt firewala, który będzie się uruchamiał wraz ze startem komputera.

Tam wklej przykładowy skrypt firewala **firewall1** - możesz użyć innego, lecz pamiętaj do niego dołączyć na początku ten kod

```
#!/bin/bash
```

```
### BEGIN INIT INFO
```

```
# Provides:      firewall.sh
```

```
# Required-Start: $local_fs $remote_fs
```

```
# Required-Stop:  $local_fs $remote_fs
```

```
# Default-Start:  2 3 4 5
```

```
# Default-Stop:   0 1 6
```

```
# Short-Description: Start daemon at boot time
```

```
# Description:    Enable service provided by daemon.
```

```
### END INIT INFO
```

Przykładowa konfiguracja iptables blokująca połączenia przychodzące

```
#!/bin/bash
```

```
### BEGIN INIT INFO
# Provides: firewall
# Required-Start: $local_fs $remote_fs
# Required-Stop: $local_fs $remote_fs
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start daemon at boot time
# Description: Enable service provided by daemon.
### END INIT INFO
```

```
# Czyszczenie reguł
```

```
iptables -F
```

```
iptables -X
```

```
# Polityka bezpieczeństwa
```

```
iptables -P INPUT DROP
```

```
iptables -P FORWARD DROP
```

```
iptables -P OUTPUT DROP
```

```
# Reguły dla pętli zwrotnej
```

```
iptables -A INPUT -i lo -j ACCEPT
```

```
iptables -A FORWARD -o lo -j ACCEPT
```

```
iptables -A OUTPUT -o lo -j ACCEPT
```

```
# Reguły dla łańcucha wejściowego
```

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# Reguły dla łańcucha wyjściowego
```

```
iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
```

Sprawdź czy zamontowałeś cdrom. Jeśli widzisz plik firewall1.txt to skopiuj plik firewall do katalogu /etc/systemd/system/ W terminalu wpisz

```
cp /media/cdrom/firewall1.txt /etc/systemd/system/firewall
```

Następnie, jeśli nie ma błędów nadaj plikowi firewall prawa uruchamiania

```
chmod 755 /etc/systemd/system/firewall
```

Następnie utwórz usługę firewall.service w terminalu wpisz

```
nano /etc/systemd/system/firewall.service
```

W edytorze nano wyświetli się pusty dokument tam wklejamy skrypt usługi dla systemud.

```
[Unit]
```

```
Description=firewall
```

```
After=network.target
```

```
[Service]
```

```
RemainAfterExit=yes
```

```
ExecStart=/etc/systemd/system/firewall start
```

```
ExecStop=/etc/systemd/system/firewall stop
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Następnie zapisz plik.

Lub przekopiuj plik unit.txt jako /etc/systemd/system/firewall.service

```
cp /media/cdrom/unit.txt /etc/systemd/system/firewall.service
```

Uruchom usługę firewall.service wpisując

```
systemctl enable firewall.service
```

Następnie uruchom firewalla wpisując

```
systemctl start firewall.service
```

Następnie sprawdź, czy działają reguły w iptables wpisując

```
iptables -L
```

Powinny wyświetlić się nam reguły. Ponadto sprawdź status firewalla wpisując

```
systemctl status firewall.service
```

Otrzymasz mniej więcej komunikat

```
firewall.service - firewall
```

```
Loaded: loaded (/etc/systemd/system/firewall.service; enabled; vendor preset:
```

```
"Active: active (exited)" since Thu 2017-03-16 16:24:03 CET; 1min 44s ago
```

```
Main PID: 525 (code=exited, status=0/SUCCESS)
```

```
Tasks: 0 (limit: 4915)
```

```
CGroup: /system.slice/firewall.service
```

```
"mar 16 16:24:03 debian systemd[1]: Started firewall."
```

Komunikaty Active: active (exited) i mar 16 16:24:03 debian systemd[1]: Started firewall. oznaczają że usługa firewall.service działa prawidłowo.

Poproś o sprawdzenie

E. Ufw - frontend dla iptables

UFW jest graficzną nakładką dla IPtables, jego zaletą jest to, że jest bardzo prosty w obsłudze i jest mniej plików do konfiguracji. Aby go zainstalować, wpisujemy komendę:

```
apt-get install ufw
```

po instalacji możesz ustawić jego autostart w pliku konfiguracyjnym:

```
nano /etc/ufw/ufw.conf
```

zmień wartość z:

```
ENABLED=no
```

na

```
ENABLED=yes
```

możesz także dodać obsługę protokołu IPv6:

```
IPV6=yes
```

Poproś o sprawdzenie

Przywróć pierwszą migawkę.

Podsumowanie:

Po wykonaniu wszystkich czynności z powyższej instrukcji przeczytaj ponownie z zrozumieniem cel ogólny i cele szczegółowe, które znajdują się na pierwszej stronie instrukcji. Jeżeli one zostały niezrealizowane to powtarzaj wykonać tej instrukcji w szkole lub/i w domu do momentu zrealizowania.