Trasowanie i Filtrowanie w Linuxie

Autor: Pawel Ossowski IVFDS

STRESZCZENIE

Niniejsze opracowanie stanowi zebranie podstawowych informacji, opisujacych mechanizm: trasowania i filtrowania w Linux. W szczególnosci przedstawione tu zostaly metody trasowania statycznego i dynamicznego. W kolejnych rozdzialach przedstawione sa narzedzia umozliwiajace filtrowanie. Sa to miedzy innymi: IP Firewall Administration, IP-Chains i IP-Tables. Uzupelnienie teorii stanowia ilustracje poszczególnych zagadnien i przykladowe listingi.

SPIS TRESCI

Streszczenie 1
1. Trasowanie 3
1.1 Wstep teoretyczny 3
1.2 Trasowanie statyczne 3
1.3 Trasowanie dynamiczne 5
2. Filtrowanie 6
2.1 Narzedzie IP Firewall Administration 6
2.2 Narzedzie IP Chains 7
2.3 Narzedzie IP Tables 9
Literatura 11

1. TRASOWANIE

Trasowanie – zwane równiez marszrutowaniem, routingiem, czy wyznaczaniem trasy – to proces przenoszenia pakietu danych z jednego fizycznego segmentu sieci do innego fizycznego segmentu sieci, którego celem jest dostarczenie pakietu do stacji docelowej. Jest ono realizowane w warstwie sieciowej modelu OSI [3].
Mechanizm wybierania tras pakietów funkcjonowac moze w oparciu o dwa schematy:

- Trasowanie dynamiczne (dynamic routing) – automatyczne konfigurowanie ukladu tras w sieci, realizowane za posrednictwem protokolów trasowania, przenoszacych informacje o zmianach w topologi sieci
- Trasowanie statyczne (static routing) – reczne konfigurowanie wszystkich sciezek trasowania pomiedzy sieciami. Stosowane w zasadzie wylacznie w mniejszych sieciach.

1.1 Wstep teoretyczny

Kryteria – zwane równiez metrykami, czy miernikami – sa to parametry wedlug których wybierana jest optymalna trasa dla pakietów. Do w/w kryteriów mozna zaliczyc [3]:

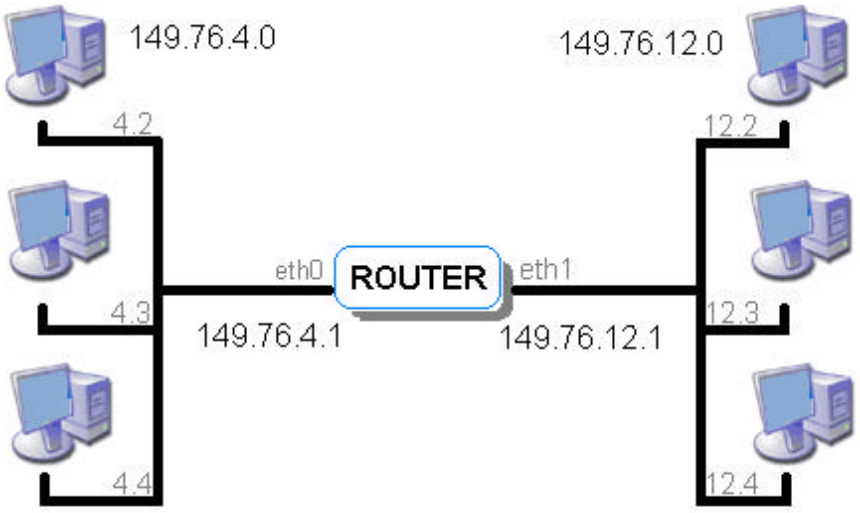
- liczba przeskoków – (hopów) wyraza liczbe routerów, przez które przechodzi pakiet w drodze pomiedzy siecia zródlowa a docelowa, stanowi najpowszechniejszy miernik trasowania
- opóznienie – czas przesylania pakietu z sieci zródlowej do docelowej, czynnikami wplywajacymi na wielkosc opóznienia sa miedzy innymi: szerokosc pasma sieci posredniczacych, wielkosc kolejek trasowania oczekujacych przy poszczególnych routerach, przeciazenia sieci posredniczacych, odleglosci miedzy sieciami
- przepustowosc – mozliwa do uzyskania wydajnosc lacza sieciowego (mierzona w bajtach na sekunde)
- niezawodnosc – pozwala na porównanie niezawodnosci laczy sieciowych
- koszt komunikacji – brany pod uwage, gdy celem jest utrzymanie kosztów przesylania na mozliwie niskim poziomie

W jadrze linuxa 2.0 byla dostepna tylko jedna tablica rutingu (podstawowa). W jadrze 2.2 tablic zdefiniowanych moze byc do 250, z czego domyslnie aktywne sa trzy [1]:

- local – (255) zawiera trasy dodawane automatycznie przez kernel, takie jak trasy do lokalnych interfejsów oraz trasy broadcastowe. Trasy w tej tablicy maja z reguly zasieg host lub link
- main – (254) odpowiednik starej tablicy rutingu (jadro 2.2) i do niej trafiaja trasy dodawane przez uzytkownika, jesli nie wskaze inaczej. Do niej dodawane sa równiez trasy tworzone automatycznie w momencie aktywacji interfejsu przez jadro
- default – tablica domyslna
- cache – jej zawartosc jest uzupelniana automatycznie przez jadro i nie jest ona dostepna do zapisu przez uzytkownika.

1.2 Trasowanie statyczne

Trasowanie statyczne – wymaga od administratora zdefiniowania pelnego zbioru tras. Jest najczesciej uzywane w sieciach mniejszych – kazda zmiana w topologii sieci prowadzi do modyfikacji statycznych tabel trasowania.



Rys 1 Trasowanie statyczne

Kiedy komputer otrzymuje pakiet z interfejsu sprawdza adres docelowy w naglówku:

- Jezeli adresem docelowym jest adres lokalnego komputera, to pakiet taki zostaje przekazany do odpowiedniego portu,
- Jezeli adres docelowy znajduje sie w sieci, która jest bezposrednio dolaczona, to pakiet przekazywany jest bezposrednio do docelowego hosta,
- Jezeli adres docelowy znajduje sie w zdalnej sieci (nie podlaczonej bezposrednio), to taki pakiet zostanie wyslany dalej w siec, poprzez ‘domyslna brame’ (default gateway)

Jednak aby komputer mógl przekazac pakiet innemu hostowi musi miec wlaczone przekazywanie pakietów IP (IP forwarding). Najprosciej jest to zrealizowac komenda:

echo ”1” > /proc/sys/net/ipv4/ip_formard

Ogólna skladnia polecenia route [6]:

route add|del [-net|-host] <cel> [gw <bramka>] [netmask <maska>] [dev <interfejs>]

- add|del – definiuje|usuwa droge w tablicy rutowania
- net|host – okresla czy droga prowadzi do calej sieci, czy tylko do pojedynczego komuptera
- cel – okresla adres sici lub komputera, do którego pakiety beda trafiac dana droga
- gateway – pozwala okreslic przez, która bramke nalezy przeslac pakiety kierowane do okreslonego celu
- maska – maska sieci adresu docelowego
- dev – okresla, za posrednictwem którego interfejsu beda przesylane pakiety korzystajace z danej drogi

Przyklad:

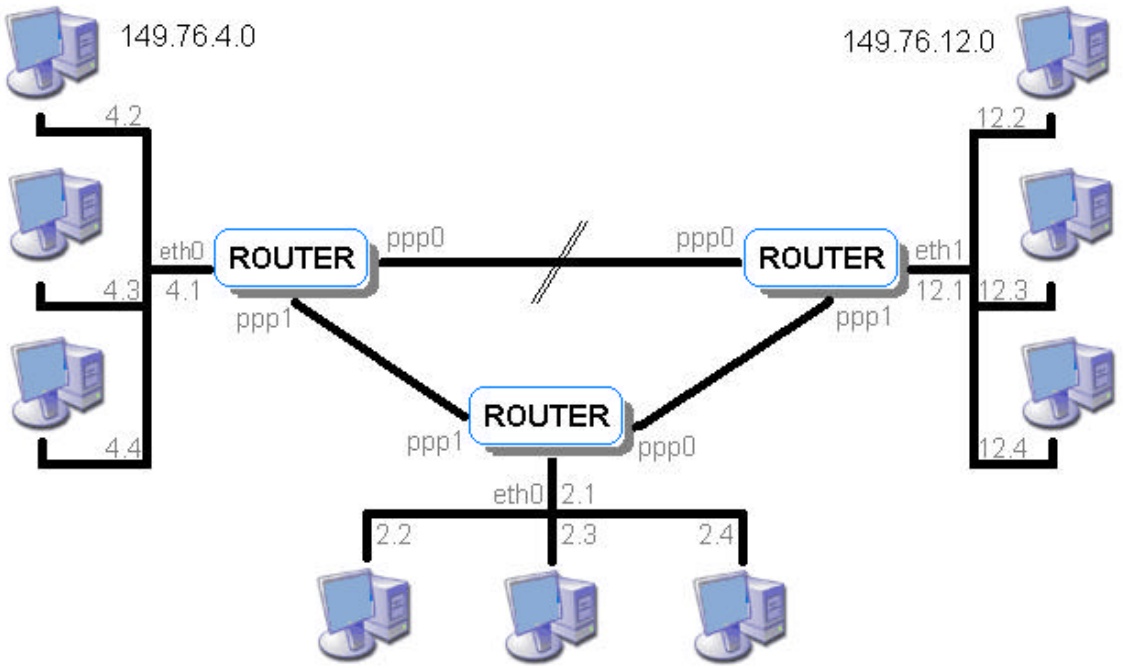
```
route add -net 213.184.9.64 netmask 255.255.255.192 eth0
route add default gw 213.184.9.65 eth0
```

1.3 Trasowanie dynamiczne

Siec w której do przesylania danych przeznaczonych dla okreslonego hosta mozna wybrac wiecej niz jedna trase, powinna zostac skonfigurowana do pracy z rutowaniem dynamicznym. Tablica rutowania jest tworzona na podstawie informacji wymienianych przez protokoly rutujace. Sluza one do rozsylania informacji, które automatycznie uaktualniaja sciezki, tak aby trasy przesylania danych odpowiadaly zmieniajacej sie konfiguracji sieci. Protokoly rutujace pozwalaja na szybsze oraz dokladniejsze dostrajanie rutowania, niz móglby to zrobic recznie administrator. Rutowanie dynamiczne umozliwia zarówno skorzystanie z zapasowej sciezki, jezeli pierwotnie wybrana jest niedostepna, jak i wybór najlepszej z posród wielu tras [4].

Najpopularniejszymi protokolami rutowania dynamicznego sa [4]:

- RIP – (Routing Information Protocol) – bardzo popularny protokól nadajacy sie do obslugi malych sieci korporacyjnych lub sieci miedzy budynkami, zaimplementowany w systemie operacyjnym linux jako ‘routed’ i ‘gated’
- OSPF – (Open Shortest Path First Protocol) – nowoczesniejszy i bardziej sprawny protokól, nadajacy sie do obslugi duzych konfiguracji sieci (duza liczba mozliwych tras przeslania pakietu), zaimplementowany w systemie operacyjnym linux jako ‘gated’



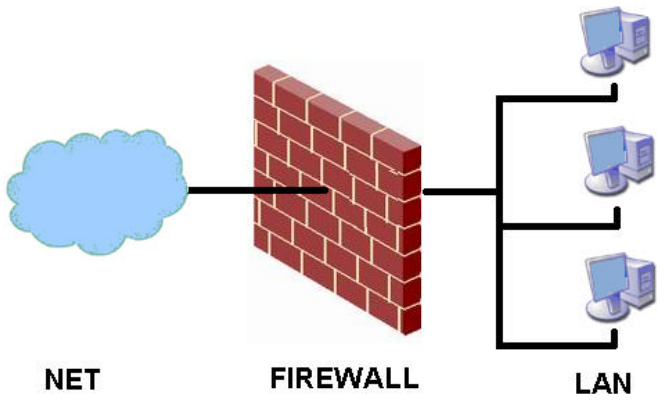
Rys 2 Trasowanie dynamiczne

Aby uruchomic w systemie rutowanie dynamiczne, nalezy: uruchomic demona ‘routed’ i dodac sciezke rutowania do wlasnej sieci, np.:

```
route add -net 213.184.9.0 netmask 255.255.255.192 eth0
routed
```

2. FILTROWANIE

Filtrowanie to termin nierozerwalnie zwiazany z tzw. ‘zapora ogniowa’.
Zapora ogniowa – to zwykle dedykowana maszyna, która sprawdza (filtruje) kazdy przeplywajacy przez nia pakiet i przepuszcza lub blokuje go zgodnie z regulami ustalonymi przez administratora. Istnieja dwa podstawowe typy firewalli (mozna je ze soba laczyc): dzialajace na poziomie aplikacji (czyli serwery posredniczace, proxy) oraz dzialajace na poziomie pakietów [3]. Firewall dzialajacy na poziomie pakietów po prostu przekazuje lub odrzuca pakiety w zaleznosci od ich zawartosci. Wiekszosc rozwiazan tego typu bazuje na danych o adresie komputera nadajacego pakiet, komputera, dla którego jest on przeznaczony, odpowiednio portu zródlowego i docelowego oraz faktu, czy pakiet jest czescia jakiejs dluszej konwersacji miedzy komputerami. Filtrowanie IP jest funkcja warstwy sieciowej. Oznacza to, ze nie ma ono nic wspólnego z aplikacja wykorzystujaca polaczenia sieciowe, a dotyczy tylko samych polaczen. Zapora oddziela siec bezpieczna (na przyklad siec lokalna) od sieci obcej (na przyklad Internet) [1].



Rys 3 Firewall

W systemie operacyjnym Linux, istnieje mozliwosc zrealizowania tzw. ‘zapory ogniowej’ przy uzyciu narzedzi, które pokrótce przedstawiaja kolejne podrozdzialy.

2.1 Narzedzie IP Firewall Administration

Narzedzie IP FIrewall Administration (ipfwadm) jest uzywane do konfiguracji (tworzenie regul) dla drugiej generacji firewalla IP w Linux. Jest dostepne poczawszy od wersji 2.2.0 jadra. Skladnia polecenia umozliwia realizowanie wielu skomplikowanych zadan [1].

Ogólna skladnia ipfwadm:
ipfwadm kategoria polecenie parametry [opcje]

kategoria – okresla jakiego typu reguly dotyczy, jedna i tylko jedna z ponizszych:
- -I – regula wejsciowa
- -O – regula wyjsciowa
- -F – regula przekazywania

polecenie – przynajmniej jedna z ponizszych:
- -a [*polityka*] – dodanie nowej reguly
- -i [*polityka*] – wstawienie nowej reguly
- -d [*polityka*] – usuniecie istniejacej reguly
- -p *polityka* – ustawienie polityki domyslnej
- -l – wylistowanie wszystkich istniejacych regul
- -f – usuniecie wszystkich istniejacych regul

polityka – przyjmuje jedna z ponizszych:
- accept – pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów
- deny – nie pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów
- reject – nie pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów, wysyla dodatkowo komunikat bledu ICMP do hosta, który przeslal datagram

parametry – przynajmniej jeden z ponizszych:
- -P protokól – precyzuje dla jakiego protokolu(-ów) jest przewidziana konstruowana regula, moze miec wartosc: TCP, UDP, ICMP, ALL
- -S adres[/maska] [port] – okresla adres zródlowy, maske, i port, gdzie maska domyslnie jest '/32', zas domyslnie parametr dotyczy wszystkich portów, niezbedne jest uprzednie sprecyzowanie protokolu '-P'
- -D adres/maska [port] – jak powyzej, dotyczy adresu docelowego
- -V adres – okresla adres interfejsu sieciowego, na którym pakiet jest odbierany (-I) lub z którego jest wysylany (-O).
- -W urzadzenie – jak powyzej, lecz precyzowane poprzez nazwe urzadzenia

opcje – przyjmuje jedna z ponizszych:
- -b – umozliwia tworzenie regul dwustronnych
- -o – pozwala na zapisywanie pasujacych datagramów do logu jadra
- -y – filtruje polaczeniowe datagramy TCP
- -k – filtruje datagramy-potwierdzen TCP

Przyklad:

```
# Usuniecie wszystkich dotychczasowych regul przekazywania
ipfwadm –F –f
# Domyslna polityka przekazywania na nie pozwalaj
ipfwadm –F –p deny
# Pozwala na wysylanie datagramów o adresie zródlowym nalezacym do naszej sici
# i gniezdzie docelowym 80 (korzystanie z protokolu http)
ipfwadm –F –a accept –P tcp –S 172.16.1.0/24 –D 0/0 80
# Pozwala na przekazywanie przez firewaall odpowiedzi przesylanych z powrotem
ipfwadm –F –a accept –P tcp –S 0/0 80 –D 172.16.1.0/24
```

2.2 Narzedzie IP Chains

Narzedzie IP Chains (ipchains) udostepnia cala elastycznosc IP Firewall Administration, ale za pomoca nieco uproszczonej skladni. Ponadto, jak sama nazwa wskazuje umozliwia równiez laczenie zestawu regul w tzw. 'lancuchy', przez co umozliwia wygodne konfigurowanie bardziej zlozonych srodowisk [1].

Ogólna skladnia ipchains:

ipchains polecenie reguly [opcje]

polecenie – istnieja rózne sposoby operowania pojedynczymi regulami jak i ich zestawami, oto niektóre z nich:

- -A *lancuch* – dodanie jednej lub kilku regul na koniec zadanego lancucha
- -I *lancuch* – wstawienie jednej lub kilku regul na poczatek zadanego lancucha
- -D *lancuch* – usuwa jedna lub kilka regul z zadanego lancucha, który pasuje do reguly
- -D *lancuch* numer – usuwa reguly z pozycji 'numer' w zadanym lancuchu
- -C *lancuch* – testuje konfiguracje firewalla, sprawdzajac zadanym lancuchem datagramu opisana regule
- -L [*lancuch*] – listowanie regul zadanego lancucha, domyslnie wszystkich
- -F [*lancuch*] – usuniecie regul z zadanego lancucha, domyslnie wszystkich
- -Z [*lancuch*] – wyzerowanie liczników datagramów i bajtów dla zadanego lancucha, domyslnie wszystkich
- -N *lancuch* – stworzenie lancucha o zadanej nazwie
- -X [*lancuch*] – usuniecie lancucha o zadanej nazwie, domyslnie wszystkich
- -P *lancuch polityka* – definiuje domyslna polityke dla zadanego lancucha

polityka – przyjmuje jedna z ponizszych:

- accept – pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów
- deny – nie pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów
- reject – nie pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów, wysyla dodatkowo komunikat bledu ICMP do hosta, który przeslal datagram
- redir – powoduje niewidoczne przekierowanie datagramu na port firewalla
- return – sprawia, ze kod firewalla IP powraca do tego lancucha firewalla, który wywolal lancuch zawierajacy te regule i kontynuuje dalszcze dzialanie poczawszy od nastepnej jej reguly

reguly – sklada sie z wielu parametrów, okreslajacych jakie typy pakietów maja do niej pasowac, dla parametrów pominietych w regule, zaklada sie jego wartosc domyslna:

- -p [!] protokól – okresla, którego z protokolów (! nie-)dotyczy regula, mozliwe wartosci to: tcp, udp, icmp, czy domyslnie all
- -s [!]adres[/maska][!][port] – okresla adres zródlowy, maske i port w datagramie, który (! nie-)bedzie pasowal do reguly, port mozna wskazac tylko z regula 'p'
- -d [!]adres[/maska][!][port] – okresla adres docelowy, identycznie jak powyzej
- -j cle – okresla dzialanie do wykonania w sytuacji gdy datagram bedzie sie zgadzal z regula, dopuszczalne cele to: accept, deny, reject, redir, return, nazwa wlasnego lancucha
- -i [!]interfejs[+] – (! nie-)okresla interfejsu przez który przechodzi datagram, znak '+' moze uogólnic sposób okreslania interfejsu (np. '–i eth+' – dowolne urzadzenie ethernetowe)

opcje – przyjmuje jedna z ponizszych:

- -b – okresla, ze polecenie generuje dwie reguly (jedna uwzglednia podane parametry, zas druga uwzglednia je w odwrotnym kierunku)
- -v – powoduje wylistowanie dodatkowych wyników
- -n – wylistowane wyniki maja forme liczbowa nie nazwowa (adresy i porty)
- -l – wlacza zapisywanie przez jadro pasujacych datagramów

- -y – jest uzywana do filtrowania zadan nawiazania polaczenia TCP

Przyklad:

```
# Usuniecie wszystkich dotychczasowych regul z zestawu forward
ipchains –F forward
# Definiuje domyslna polityke zestawu regul forward na DENY
ipchains –P forward DENY
# Zapobiega przyjmowaniu przychodzacych polaczen TCP z portem zródlowym
ipchains –A forward –s 0/0 80 –d 172.16.1.0/24 –p tcp –y –j DENY
# Pozwala datagramom kierowanym do I z serwerów www na przechodzenie
# do sieci wewnetrznej
ipchains –A forward –s 172.16.1.0/24 –d 0/0 80 –p tcp –b –j ACCEPT
# Reguly pozwalajace na dostep do zewnetrznych serwerów FTP w trybie biernym
ipchains –A forward –s 0/0 20 –d 172.16.1.0/24 –p tcp –y –j DENY
ipchains –A forward –s 172.16.1.0/24 –d 0/0 20 –p tcp –b –j ACCEPT
ipchains –A forward –s 0/0 21 –d 172.16.1.0/24 –p tcp –y –j DENY
ipchains –A forward –s 172.16.1.0/24 –d 0/0 21 –p tcp –b –j ACCEPT
```

2.3 Narzedzie IP Tables

W netfilter udostepniono piec wbudowanych lancuchów. Lancuchy INPUT i FORWARD sa dostepne dla tablicy filter, PREROUTING i POSTROUTING sa dostepne dla tablicy nat, natomiast lancuch OUTPUT jest dostepny dla obu tablic [1].

Ogólna skladnia iptables:
iptables polecenie reguly rozszerzenia

polecenie – istnieja rózne sposoby operowania pojedynczymi regulami jak i ich zestawami, oto niektóre z nich:

- -A *lancuch* – dodanie jednej lub kilku regul na koniec zadanego lancucha
- -I *lancuch* – wstawienie jednej lub kilku regul na poczatek zadanego lancucha
- -D *lancuch* – usuwa jedna lub kilka regul z zadanego lancucha, który pasuje do reguly
- -D *lancuch* numer – usuwa reguly z pozycji ‘numer’ w zadanym lancuchu
- -C *lancuch* – testuje konfiguracje firewalla, sprawdzajac zadanym lancuchem datagramu opisana regule
- -L [*lancuch*] – listowanie regul zadanego lancucha, domyslnie wszystkich
- -F [*lancuch*] – usuniecie regul z zadanego lancucha, domyslnie wszystkich
- -Z [*lancuch*] – wyzerowanie liczników datagramów i bajtów dla zadanego lancucha, domyslnie wszystkich
- -N *lancuch* – stworzenie lancucha o zadanej nazwie
- -X [*lancuch*] – usuniecie lancucha o zadanej nazwie, domyslnie wszystkich
- -P *lancuch polityka* – definiuje domyslna polityke dla zadanego lancucha

polityka – przyjmuje jedna z ponizszych:

- accept – pozwala na odbiór, przekazywanie lub wysylanie pasujacych datagramów
- drop – powoduje, ze datagram jest odrzucany
- queue – powoduje, ze datagram jest przekazywany do przestrzeni uzytkownika w celu jego dalszego przetwarzania
- return – sprawia, ze kod firewalla IP powraca do tego lancucha firewalla, który wywolal lancuch zawierajacy te regule i kontynuuje dalszcze dzialanie poczawszy od nastepnej reguly

opcje – przyjmuje jedna z ponizszych:
- -v – powoduje wylistowanie dodatkowych wyników
- -n – wylistowane wyniki maja forme liczbowa nie nazwowa (adresy i porty)
- -x – sprawia, ze wszelkie liczby sa pokazywane dokladnie (bez zaokraglania)
- - -numery-wierszy – wyswietla numery wierszy (pozycji w lancuchu)

rozszerzenia TCP – uzywane z '-m tcp' i '-p tcp', przyjmuje jedna z ponizszych:
- --sport [!][port] – okresla port z którego (! nie-)musi pochodzic datagram
- --dport [!][port] – okresla port do którego (! nie-)musi byc skierowany datagram
- --tcp-flags [!] maska lista – okresla wartosci jakie musza miec znaczniki pakietu aby ten pasowal do reguly. maska to lista oddzielonych przecinkami znaczników które sa sprawdzane, lista to lista oddzielonych przecinkami znaczników które (! nie-)musza byc ustawione zeby regula pasowala, dopuszczalne wartosci to SYN, ACK, FIN, FST, URG, PSH, ALL, NONE
- [!] --syn – powoduje, ze regula pasuje tylko do datagramów z ustawionym bitem SYN i wyzerowanymi bitami ACK i FIN

rozszerzenia UDP – uzywane z '-m udp' i '-p udp', przyjmuje jedna z ponizszych:
- --sport [!][port] – okresla port z którego (! nie-)musi pochodzic datagram
- --dport [!][port] – okresla port do którego (! nie-)musi byc skierowany datagram

rozszerzenia ICMP – uzywane z '-m icmp' i '-p icmp', przyjmuje jedna z ponizszych:
- --icmp-type [!] nazwa-typu – okresla typ komunikatu ICMP pasujacego do reguly (okreslony przez nazwe lub numer, np. echo-request, echo-reply, source-quench, time-exceeded, destination-unrechable, i inne)

Przyklad:
```
# Jak poprzednio z wewnetrznej sieci istnieje mozliwosc korzystania z uslugi www
# zas pozostaly ruch nie jest przepuszczany
iptables –F FORWARD
iptables –P FORWARD DROP
iptables –A FORWARD –m tcp –p tcp –s 0/0 –sport 80 –d 172.16.1.0./24 --syn –j DROP
iptables –A FORWARD –m tcp –p tcp –s 172.16.1.0/24 --sport 80 –d 0/0 –j ACCEPT
iptables –A FORWARD –m tcp –p tcp –d 172.16.1.0/24 --dport 80 –s 0/0 –j ACCEPT
```

LITERATURA

[1] Olaf Kirch, Terry Dawson, LINUX Podrecznik administratora sieci, O'Reilly 2000

[2] Dawid Pitts, Bill Ball Red Hat Linux 6 Ksiega Eksperta, Helion 2000

[3] Brian Komar Administracja TCP/IP dla kazdego, Helion 2000

[4] Internet: http://www.republika.pl/tht/