

WYDANIE IV



ZAPORY
SIECIOWE
W SYSTEMIE
LINUX[®]

KOMPENDIUM WIEDZY O NFTABLES

STEVE SUEHRING

Helion 

Tytuł oryginału: Linux® Firewalls Enhancing Security with nftables and Beyond, Fourth Edition

Tłumaczenie: Lech Lachowski

ISBN: 978-83-283-1297-5

Authorized translation from the English edition, entitled: LINUX FIREWALLS: ENHANCING SECURITY WITH NFTABLES AND BEYOND, Fourth Edition; ISBN 0134000021; by Steve Suehring; published by Pearson Education, Inc, publishing as Addison-Wesley Professional. Copyright © 2015 by Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Polish language edition published by HELION S.A. Copyright © 2015.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Wydawnictwo HELION

ul. Kościuszki 1c, 44-100 GLIWICE

tel. 32 231 22 19, 32 230 98 63

e-mail: helion@helion.pl

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:

<ftp://ftp.helion.pl/przyklady/zasili.zip>

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/zasili>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Przedmowa	13
O autorze	15
Część I Filtrowanie pakietów i podstawowe środki bezpieczeństwa	17
Rozdział 1. Wstępne koncepcje dotyczące działania zapór sieciowych filtrujących pakiety	19
Model sieciowy OSI	22
Protokoły bezpołączeniowe i połączeniowe	23
Następne kroki	24
Protokół IP	24
Adresowanie IP i podsieciowanie	24
Fragmentacja IP	28
Broadcasting i multicasting	28
ICMP	29
Mechanizmy transportowe	31
Protokół UDP	32
Protokół TCP	32
Nie zapominajmy o protokole ARP	35
Nazwy hostów i adresy IP	36
Adresy IP oraz adresy ethernetowe	36
Routing: przekazywanie pakietu z jednego miejsca do drugiego	37
Porty usług: drzwi dla programów w Twoim systemie	37
Typowe połączenie TCP: odwiedzanie zdalnej witryny	39
Podsumowanie	42
Rozdział 2. Koncepcje związane z filtrowaniem pakietów	43
Zapora sieciowa filtrująca pakiety	45
Wybór domyślnej polityki filtrowania pakietów	47
Odrzucanie pakietu w porównaniu z blokowaniem pakietu	49

Filtrowanie pakietów przychodzących	50
Filtrowanie zdalnych adresów źródłowych	50
Filtrowanie lokalnych adresów docelowych	53
Filtrowanie zdalnego portu źródłowego	54
Filtrowanie lokalnego portu docelowego	54
Filtrowanie stanu przychodzących połączeń TCP	55
Sondy i skanowanie	55
Ataki DoS	60
Pakiety routowane źródłowo	67
Filtrowanie pakietów wychodzących	67
Filtrowanie lokalnych adresów źródłowych	68
Filtrowanie zdalnych adresów docelowych	68
Filtrowanie lokalnych portów źródłowych	69
Filtrowanie zdalnych portów docelowych	69
Filtrowanie stanów wychodzących połączeń TCP	70
Usługi sieci prywatnej i publicznej	70
Ochrona niezabezpieczonych usług lokalnych	71
Wybór uruchamianych usług	72
Podsumowanie	72

Rozdział 3. iptables: starszy program do administrowania zaporą sieciową systemu Linux73

Różnice pomiędzy mechanizmami zapory sieciowej IPFW i Netfilter	74
Trawersacja pakietów w zaporze sieciowej IPFW	75
Trawersacja pakietów w zaporze sieciowej Netfilter	76
Podstawowa składnia polecenia iptables	77
Funkcje programu iptables	78
Funkcje tabeli nat	81
Funkcje tabeli mangle	83
Składnia polecenia iptables	84
Polecenia tabeli filter	85
Rozszerzenia celu tabeli filter	90
Rozszerzenia dopasowywania tabeli filter	92
Rozszerzenia celu tabeli nat	103
Polecenia tabeli mangle	106
Podsumowanie	106

Rozdział 4. nftables: program do administrowania zaporą sieciową systemu Linux109

Różnice pomiędzy iptables i nftables	109
Podstawowa składnia nftables	109
Funkcje programu nftables	110
Składnia nftables	111
Składnia tabeli	112
Składnia łańcucha	113

Składnia reguły	114
Podstawowe operacje nftables	118
Składnia plików nftables	119
Podsumowanie	119
Rozdział 5. Budowa i instalacja samodzielnej zapory sieciowej	121
Programy do administrowania zaporą sieciową systemu Linux	122
Jądro systemu Linux: standardowe czy niestandardowe	124
Opcje adresowania źródłowego i docelowego	125
Inicjowanie zapory sieciowej	126
Symboliczne stałe używane w przykładach zapory sieciowej	127
Włączanie obsługi monitorowania w jądrze	128
Usunięcie wszelkich istniejących reguł	130
Resetowanie domyślnych polityk i zatrzymywanie zapory sieciowej	131
Włączanie interfejsu pętli zwrotnej	132
Definiowanie domyślnej polityki	133
Wykorzystywanie stanu połączenia do omijania sprawdzania reguł	135
Fałszowanie adresów źródłowych i inne złe adresy	136
Ochrona usług na przypisanych portach nieuprzywilejowanych	141
Typowe usługi lokalne TCP przypisane do nieuprzywilejowanych portów	142
Typowe usługi lokalne UDP przypisane do nieuprzywilejowanych portów	144
Włączenie podstawowych, wymaganych usług internetowych	147
Włączenie usługi DNS (UDP/TCP port 53)	147
Włączenie typowych usług TCP	152
E-mail (TCP SMTP port 25, POP port 110, IMAP port 143)	153
SSH (port TCP 22)	159
FTP (porty TCP 21, 20)	161
Ogólna usługa TCP	164
Włączanie typowych usług UDP	165
Dostęp do serwera DHCP dostawcy usług internetowych (porty UDP 67, 68)	166
Dostęp do zdalnych sieciowych serwerów czasu (port UDP 123) ...	168
Rejestrowanie porzuconych pakietów przychodzących	169
Rejestrowanie porzuconych pakietów wychodzących	170
Instalowanie zapory sieciowej	170
Wskazówki dla debugowania skryptu zapory sieciowej	171
Uruchamianie zapory sieciowej przy starcie systemu — Red Hat i SUSE	172
Uruchamianie zapory sieciowej przy starcie systemu — Debian	173
Instalowanie zapory sieciowej z dynamicznym adresem IP	173
Podsumowanie	174

Część II Zaawansowane zagrożenia, wiele zapór sieciowych oraz strefy ograniczonego zaufania ...175

Rozdział 6. Optymalizacja zapory sieciowej177

Organizacja reguł	177
Rozpocznij od reguł blokujących ruch na portach o dużych numerach	178
Użyj modułu stanu dla dopasowań ESTABLISHED i RELATED	178
Uwzględnij protokół transportowy	178
Reguły zapory sieciowej dla bardzo popularnych usług umieszczaj jak najwyżej w łańcuchu	180
Użyj przepływu ruchu do określenia, gdzie umieścić reguły dla wielu interfejsów sieciowych	180
Łańcuchy definiowane przez użytkownika	181
Zoptymalizowane przykłady	184
Zoptymalizowany skrypt iptables	184
Inicjowanie zapory sieciowej	186
Instalowanie łańcuchów	188
Budowanie definiowanych przez użytkownika łańcuchów EXT-input i EXT-output	190
Łańcuch tcp-state-flags	199
Łańcuch connection-tracking	200
Łańcuchy local-dhcp-client-query i remote-dhcp-server-response	200
Łańcuch source-address-check	201
Łańcuch destination-address-check	201
Rejestrowanie porzuconych pakietów za pomocą polecenia iptables	202
Zoptymalizowany skrypt nftables	204
Inicjowanie zapory sieciowej	204
Budowanie plików reguł	205
Rejestrowanie porzuconych pakietów za pomocą polecenia nftables	209
Co dała optymalizacja?	210
Optymalizacja iptables	210
Optymalizacja nftables	211
Podsumowanie	212

Rozdział 7. Przekazywanie pakietów213

Ograniczenia samodzielnej zapory sieciowej	213
Podstawowe konfiguracje bramy z zaporą sieciową	215
Kwestie bezpieczeństwa sieci LAN	217
Opcje konfiguracyjne dla zaufanej domowej sieci LAN	218
Dostęp sieci LAN do bramy z zaporą sieciową	220
Dostęp sieci LAN do innych sieci LAN: przekazywanie ruchu pomiędzy wieloma sieciami LAN	221

Opcje konfiguracyjne dla większych lub mniej zaufanych sieci LAN	223
Dzielenie przestrzeni adresowej w celu tworzenia wielu sieci	224
Selektywny dostęp wewnętrzny na podstawie hosta, zakresu adresów lub portu	226
Podsumowanie	231
Rozdział 8. Przekazywanie pakietów	233
Konceptyjne tło powstania translacji NAT	233
Semantyka translacji NAT w programach iptables i nftables	238
NAT źródłowy	240
NAT docelowy	242
Przykłady translacji SNAT i prywatnych sieci LAN	244
Maskowanie ruchu LAN kierowanego do internetu	244
Stosowanie standardowej translacji NAT do ruchu LAN kierowanego do internetu	245
Przykłady translacji DNAT, sieci LAN i serwerów proxy	246
Przekierowanie hostów	246
Podsumowanie	248
Rozdział 9. Debugowanie reguł zapory sieciowej	249
Ogólne wskazówki dotyczące tworzenia zapory sieciowej	249
Wyświetlanie listy reguł zapory sieciowej	251
Przykład wyświetlania zawartości tabel iptables	252
Przykład wyświetlania zawartości tabel nftables	255
Interpretacja wpisów dziennika systemowego	256
Konfiguracja syslog	256
Znaczenie komunikatów dziennika zapory sieciowej	259
Sprawdzanie otwartych portów	263
Polecenie netstat -a [-n -p -A inet]	263
Użycie polecenia fuser do sprawdzania procesu powiązanego z konkretnym portem	266
Nmap	266
Podsumowanie	267
Rozdział 10. Wirtualne sieci prywatne — VPN	269
Przegląd wirtualnych sieci prywatnych	269
Protokoły VPN	269
PPTP i L2TP	270
IPsec	270
System Linux i produkty VPN	273
Openswan/Libreswan	273
OpenVPN	273
PPTP	273
VPN i zapory	274
Podsumowanie	275

Część III Wykraczając poza iptables i nftables277**Rozdział 11. Wykrywanie włamań i reagowanie279**

Wykrywanie włamań	279
Objawy sugerujące, że system mógł zostać przejęty przez atakującego	281
Wskazania dziennika systemowego	281
Wskazania konfiguracji systemu	282
Wskazania systemu plików	282
Wskazania kont użytkowników	283
Wskazania narzędzi do audytu bezpieczeństwa	284
Wskazania wydajności systemu	284
Co zrobić, gdy Twój system zostanie przejęty	284
Zgłaszanie incydentów	286
Dlaczego zgłaszać incydenty?	287
Jakie rodzaje incydentów można zgłaszać?	288
Komu zgłaszać incydenty?	289
Jakie informacje należy dostarczyć?	290
Podsumowanie	291

Rozdział 12. Narzędzia do wykrywania włamań293

Zestaw narzędzi do wykrywania włamań: narzędzia sieciowe	293
Różnica między przełącznikami i koncentratorami	295
ARPCatch	295
Narzędzia do wykrywania rootkitów	295
Uruchamianie programu Chkrootkit	296
Co zrobić, gdy Chkrootkit zgłasza, że komputer został zainfekowany?	298
Ograniczenia programu Chkrootkit i innych podobnych narzędzi	299
Bezpieczne korzystanie z programu Chkrootkit	300
Kiedy należy korzystać z programu Chkrootkit?	300
Integralność systemu plików	301
Monitorowanie plików dziennika	301
Swatch	302
Jak ustrzec się przed atakami	303
Często weryfikuj zabezpieczenia	304
Często przeprowadzaj aktualizacje	304
Często testuj	305
Podsumowanie	307

Rozdział 13. Monitorowanie sieci i wykrywanie ataków	309
Nasłuchiwanie eteru	309
Trzy cenne narzędzia	311
Prosty przegląd programu TCPDump	312
Pobieranie i instalowanie narzędzia TCPDump	313
Opcje narzędzia TCPDump	314
Wyrażenia TCPDump	316
Zaawansowana obsługa narzędzia TCPDump	319
Korzystanie z narzędzia TCPDump do przechwytywania konkretnych protokołów	319
Korzystanie z narzędzia TCPDump w prawdziwym świecie	320
Ataki z perspektywy TCPDump	328
Rejestrowanie ruchu za pomocą narzędzia TCPDump	333
Zautomatyzowane monitorowanie włamań za pomocą pakietu Snort	335
Pobieranie i instalowanie pakietu Snort	336
Konfiguracja pakietu Snort	337
Testowanie działania pakietu Snort	339
Otrzymywanie alertów	340
Końcowe uwagi na temat pakietu Snort	340
Monitorowanie za pomocą programu ARPWatch	341
Podsumowanie	343
Rozdział 14. Integralność systemu plików	345
Zdefiniowanie integralności systemu plików	345
Integralność systemu plików w praktyce	345
Instalacja programu AIDE	347
Konfiguracja AIDE	347
Tworzenie pliku konfiguracyjnego AIDE	347
Przykładowy plik konfiguracyjny AIDE	350
Inicjowanie bazy danych AIDE	351
Ustalanie harmonogramu automatycznego uruchamiania AIDE	351
Monitorowanie nieprawidłowości za pomocą AIDE	352
Czyszczenie bazy danych AIDE	353
Zmiana wyjścia dla raportu AIDE	355
Uzyskanie dokładniejszych informacji	356
Definiowanie makr w AIDE	357
Rodzaje kontroli AIDE	359
Podsumowanie	362

Dodatki	363
Dodatek A Zasoby dotyczące bezpieczeństwa	365
Źródła informacji dotyczących bezpieczeństwa	365
Opracowania źródłowe i najczęściej zadawane pytania	366
Dodatek B Przykłady zapory sieciowej i skryptów do jej obsługi	367
Zapora sieciowa iptables z rozdziału 5. dla samodzielnego systemu	367
Zapora sieciowa nftables z rozdziału 5. dla samodzielnego systemu	380
Zoptymalizowana zapora sieciowa iptables z rozdziału 6.	384
Zapora sieciowa nftables z rozdziału 6.	397
Dodatek C Słowniczek	403
Dodatek D Licencja GNU Wolnej Dokumentacji	417
0. Preambuła	417
1. Zastosowanie i definicje	418
2. Kopiowanie dosłowne	419
3. Kopiowanie w dużej liczbie egzemplarzy	420
4. Modyfikacje	420
5. Łączenie dokumentów	423
6. Zbiory dokumentów	423
7. Agregacja z pracami niezależnymi	423
8. Tłumaczenie	424
9. Wygaśnięcie praw	424
10. Przyszłe wersje Licencji	425
11. Relicencjonowanie	425
Skorowidz	427

nftables: program do administrowania zaporą sieciową systemu Linux

W rozdziale 3. przedstawiłem iptables, wykorzystywany od dawna program administracyjny dla zapor sieciowych systemu Linux. Omówiliśmy składnię oraz różne opcje dostępne w iptables. W tym rozdziale przyjrzymy się nowym tabelom Netfilter, czyli programowi nftables. Program nftables jest dostępny w ramach jądra mainline systemu Linux, począwszy od wersji 3.13.

Różnice pomiędzy iptables i nftables

W jądrze program nftables stanowi znaczące odejście od systemu filtrowania iptables. Program nftables zastępuje funkcjonalność nie tylko iptables, ale także ip6tables dla IPv6, arptables dla filtrowania ARP oraz ebtables dla filtrowania pakietów na mostkach Ethernet. Składnie poleceń dla nftables i iptables różni się od siebie, a nftables umożliwia korzystanie z dodatkowych funkcji skryptowych. Program administracyjny dla nftables nazywa się nft i za pomocą tego polecenia budowane są zapory sieciowe.

W przeciwieństwie do iptables, program nftables nie zawiera żadnych wbudowanych tabel. To do administratora należy określenie, które tabele są potrzebne, i dodanie tych tabel wraz z regułami dla przetwarzania. W dalszej części tego rozdziału przyjrzymy się składni programu nftables i jego wykorzystaniu do tworzenia zapory sieciowej.

Podstawowa składnia nftables

Polecenie nft stanowi program administracyjny, który jest używany do budowy zapory sieciowej. Podstawowa składnia polecenia nftables zaczyna się od samego programu nft, po którym następują polecenia i podpolecenia oraz różne argumenty i wyrażenia. Oto przykład:

```
nft <polecenie> <podpolecenie> <łańcuch> <definicja_reguły>
```

Do typowych poleceń należą:

- add;
- list;
- insert;
- delete;
- flush.

Typowe pod polecenia to:

- table;
- chain;
- rule.

Funkcje programu nftables

Program nftables zawiera kilka charakterystycznych dla języków programowania wyższego poziomu funkcjonalności, takich jak możliwość definiowania zmiennych i załączania zewnętrznych plików. Program nftables może być również stosowany do filtrowania i przetwarzania różnych rodzin adresów. Te rodziny adresów to:

- **ip** — adresy IPv4;
- **ip6** — adresy IPv6;
- **inet** — adresy IPv4 i IPv6;
- **arp** — adresy protokołu ARP (ang. *Address Resolution Protocol*);
- **bridge** — przetwarzanie pakietów, które przechodzą przez interfejsy zmostkowane.

Jeśli nie zostało określone inaczej, domyślną rodziną adresów są adresy IP. Zdolność do przetwarzania różnych rodzin adresów oznacza, że nftables ma zastąpić inne mechanizmy filtrowania, takie jak ebtables i arptables.

Ogólna architektura przetwarzania dla nftables polega na określeniu rodziny adresów, do której dana reguła będzie stosowana. Następnie nftables wykorzystuje jedną tablicę lub kilka tablic, które zawierają jeden łańcuch lub kilka łańcuchów, a te z kolei zawierają reguły przetwarzania. Reguły przetwarzania dla nftables składają się z wyrażeń, takich jak adres, interfejs, porty (lub inne dane zawarte w aktualnie przetwarzanym pakiecie), oraz instrukcji, takich jak drop, queue oraz continue.

Wskazówka

Tabele zawierają łańcuchy. Łańcuchy zawierają reguły.

Niektóre rodziny adresów zawierają zaczepy (ang. *hooks*) umożliwiające programowi nftables uzyskanie dostępu do pakietów, gdy trawersują one przez stos sieciowy w systemie

Linux. Oznacza to, że można wykonać operację na pakiecie, zanim zostanie on przekazany do routowania lub po jego przetworzeniu. Dla rodzin adresów ip, ip6 oraz inet zastosowanie mają następujące zaczepy:

- prerouting — pakiety, które właśnie przybyły i nie zostały jeszcze preroutowane lub przetworzone przez inne części nftables.
- input — pakiety przychodzące, które zostały odebrane i wysłane poprzez zaczep prerouting.
- forward — jeśli pakiet będzie wysłany do innego urządzenia, będzie dostępny poprzez zaczep forward.
- output — pakiety wychodzące z procesów w systemie lokalnym.
- postrouting — tuż przed opuszczeniem systemu zaczep postrouting udostępni pakiet do dalszego przetwarzania.

Rodzina adresów ARP wykorzystuje tylko zaczepy input i output.

Składnia nftables

Samo polecenie nft ma kilka opcji, które są dostępne z poziomu wiersza poleceń i nie są związane bezpośrednio z definiowaniem reguł filtrowania. Do tych opcji wiersza poleceń należą:

- --debug <poziom, [poziom]> — dodanie debugowania na poziomie <poziom>, takim jak scanner, parser, eval, netlink, mnl, segtree, proto-ctx lub na wszystkich poziomach.
- -h | --help — wyświetla podstawową pomoc.
- -v | --version — wyświetla numer wersji nft.
- -n | --numeric — wyświetla informacje o adresie i porcie w postaci numerycznej, zamiast przeprowadzać rozwiązywanie nazw.
- -a | --handle — wyświetla uchwyty reguł.
- -I | --includepath <katalog> — dodaje <katalog> do ścieżki wyszukiwania dla załączonych plików.
- -f | --file <nazwa_pliku> — załącza zawartość pliku <nazwa_pliku>.
- -i | --interactive — odczytuje dane wejściowe z wiersza poleceń.

Jak już wspomniano, w nftables nie ma predefiniowanych tabel. Od Ciebie zależy zdefiniowanie tabel, których chcesz użyć w systemie nftables. Polecenia dostępne do definiowania danej reguły zależą od tego, czy pracujesz z tabelą, łańcuchem lub regułą.

Składnia tabeli

Podczas pracy z tabelą dostępne są cztery polecenia:

- `add` — dodaje tabelę;
- `delete` — usuwa tabelę;
- `list` — wyświetla wszystkie łańcuchy i reguły dla tabeli;
- `flush` — czyści wszystkie łańcuchy i reguły w tabeli.

Listę dostępnych tabel możesz wyświetlić za pomocą następującego polecenia (uruchomionego z poziomu użytkownika `root`):

```
nft list tables
```

Pamiętaj, że nie ma żadnych domyślnych tabel dla `nftables`, tak jak miało to miejsce w przypadku `iptables`. Dlatego polecenie `list tables` może nic nie zwrócić, jeśli nie zostały zdefiniowane żadne tabele. Jest to normalne zachowanie, jeżeli dopiero skonfigurowałeś `nftables` i nie zdefiniowałeś jeszcze żadnej zapory sieciowej za pomocą tego programu. Tabelę, która będzie przechowywać standardowe łańcuchy i reguły zapory sieciowej, można zdefiniować w następujący sposób:

```
nft add table filter
```

Po dodaniu tabeli zapory sieciowej polecenie `list tables` zwróci jej nazwę:

```
table filter
```

Dalsze informacje na temat tabeli można uzyskać za pomocą następującego polecenia:

```
nft list table filter
```

Spowoduje to wyświetlenie informacji na temat tabeli, łącznie ze wszystkimi zdefiniowanymi w niej łańcuchami:

```
table ip filter{
}
```

Jak widać na przykładzie, tabela `filter` używa rodziny adresów IP i aktualnie jest pusta.

W tym przykładzie tabela została nazwana `filter`, ale można użyć dowolnej innej nazwy, na przykład `firewall`. Zazwyczaj jednak dla tej tabeli stosuje się właśnie nazwę `filter` i taka nazwa została również użyta w przykładach zawartych w dokumentacji dla programu `nftables`.

W przypadku wyświetlania listy reguł całkiem pomocne jest dodanie opcji `-a`, która pozwala sprawdzić numery uchwytów. Uchwyt może być wykorzystywany do zmodyfikowania lub usunięcia reguły w prosty sposób. To zastosowanie zostanie przedstawione w dalszej części rozdziału, podczas dodawania reguł do zapory sieciowej.

W trakcie wyświetlania listy reguł zapory sieciowej `nftables` przeprowadza rozwiązywanie adresów i portów. To zachowanie można zmodyfikować za pomocą opcji `-n`. Można dodać dwie opcje `-n`, aby zapobiec rozwiązywaniu zarówno adresów, jak i portów:

```
nft list table filter -nn
```

Składnia łańcucha

Podczas pracy z łańcuchem dostępnych jest sześć poleceń:

- `add` — dodanie łańcucha do tabeli;
- `create` — tworzenie łańcucha w tabeli, chyba że łańcuch o tej samej nazwie już istnieje;
- `delete` — usunięcie łańcucha;
- `flush` — wyczyszczenie wszystkich reguł w łańcuchu;
- `list` — wyświetlenie wszystkich reguł w łańcuchu;
- `rename` — zmiana nazwy łańcucha.

Podczas dodawania łańcucha można zdefiniować wspomniany wcześniej zaczep. Ponadto do definicji łańcucha można dodać opcjonalny priorytet.

Istnieją trzy podstawowe rodzaje łańcuchów, które mogą zawierać reguły i mieć również podłączone zaczepy. Rodzaje łańcucha i zaczepu należy zdefiniować podczas tworzenia łańcucha i są one niezbędne do funkcjonowania łańcucha podczas normalnej pracy zapory sieciowej. Jeśli rodzaje łańcucha i zaczepu nie zostaną zdefiniowane, pakiety nie będą routowane do tego łańcucha.

Trzy podstawowe rodzaje łańcuchów są następujące:

- `filter` — używany do filtrowania pakietów;
- `route` — używany do routowania pakietów;
- `nat` — używany do translacji NAT.

Aby pogrupować podobne do siebie reguły, można również dodać inne łańcuchy. Gdy pakiety trawersują przez podstawowy łańcuch, mogą być kierowane do łańcucha zdefiniowanego przez użytkownika lub do kilku takich łańcuchów w celu dodatkowego przetwarzania.

Podczas dodawania łańcucha należy określić tabelę, w której dany łańcuch ma być zdefiniowany. Poniższe polecenie dodaje na przykład łańcuch `input` do zdefiniowanej uprzednio tabeli `filter`:

```
nft add chain filter input { type filter hook input priority 0 \; }
```

To polecenie powoduje dodanie łańcucha o nazwie `input` do tabeli o nazwie `filter`. Rodzaj łańcucha to podstawowy łańcuch `filter`, który zostanie dołączony do zaczepu `input` z priorytetem 0. Przy wprowadzaniu tego polecenia w wierszu poleceń należy dodać pomiędzy nawiasami pojedynczą spację oraz średnik. W przypadku stosowania polecenia w natywnym skrypcie `nft` można pominąć spację i lewy ukośnik.

Dodawanie łańcucha `output` wygląda podobnie, trzeba jedynie w odpowiednich miejscach zamienić `input` na `output`:

```
nft add chain filter output { type filter hook output priority 0 \; }
```

Składnia reguły

W regułach przeprowadzane są akcje filtrowania pakietów. Podczas pracy z regułami dostępne są trzy polecenia:

- `add` — dodanie reguły;
- `insert` — wstawienie reguły do łańcucha na początku lub w określonym miejscu;
- `delete` — usunięcie reguły.

W ramach reguły określone są kryteria dopasowywania dla pakietów oraz podjęta zostaje decyzja dotycząca tego, co powinno się zdarzyć z pakietem, który pasuje do danej reguły. Program `nftables` oraz budowane w nim reguły używają różnych instrukcji i wyrażeń do tworzenia definicji.

Instrukcje `nftables` są podobne do instrukcji `iptables` i zazwyczaj wpływają na sposób, w jaki pakiet będzie przetwarzany. Może to polegać na zatrzymaniu przetwarzania, wysłaniu przetwarzania do innego łańcucha lub po prostu na zarejestrowaniu pakietu. Instrukcje i werdykty mogą być następujące:

- `accept` — zaakceptowanie pakietu i zatrzymanie przetwarzania;
- `continue` — kontynuowanie przetwarzania pakietu;
- `drop` — zatrzymanie przetwarzania i porzucenie pakietu po cichu;
- `goto` — wysłanie przetwarzania do określonego łańcucha, ale niezwracanie go do łańcucha wywołującego;
- `jump` — wysłanie przetwarzania do określonego łańcucha i zwrócenie go do łańcucha wywołującego po zakończeniu operacji lub gdy zostanie wykonana instrukcja `return`;
- `limit` — przetwarzanie pakietu zgodnie z regułą, jeśli osiągnięty zostanie limit odebranych pasujących pakietów;
- `log` — zarejestrowanie pakietu i kontynuowanie przetwarzania;
- `queue` — zatrzymanie przetwarzania i wysłanie pakietu do procesu przestrzeni użytkownika;
- `reject` — zatrzymanie przetwarzania i odrzucenie pakietu;
- `return` — wysłanie przetwarzania z powrotem do łańcucha wywołującego.

Wyrażenia `nftables` mogą być charakterystyczne dla rodziny adresów lub rodzaju przetwarzanego pakietu. Program `nftables` używa wyrażeń bloku danych i metawyrażeń. Wyrażenia bloku danych (ang. *payload expressions*) są wyrażeniami zebranymi z informacji dotyczących pakietów. Istnieją na przykład określone wyrażenia nagłówka, takie jak `sport` i `dport` (odpowiednio port źródłowy i port docelowy), które mają zastosowanie do pakietów TCP i UDP, ale nie mają sensu w warstwach IPv4 i IPv6, ponieważ te warstwy nie używają portów. Metawyrażenia (ang. *meta expressions*) mogą być wykorzystywane

do reguł, które mają szerokie zastosowanie lub są związane z typowymi właściwościami pakietów lub interfejsów.

Dostępne metawyrażenia zostały przedstawione w tabeli 4.1.

Tabela 4.1. Metawyrażenia w nftables

Wyrażenie	Opis
<code>iif</code>	Indeks interfejsu, który odebrał pakiet.
<code>iifname</code>	Nazwa interfejsu, na którym został odebrany pakiet.
<code>iiftype</code>	Typ interfejsu, na którym został odebrany pakiet.
<code>length</code>	Długość pakietu w bajtach.
<code>mark</code>	Wartość mark pakietu.
<code>oif</code>	Indeks interfejsu, który wysłał pakiet.
<code>oifname</code>	Nazwa interfejsu, z którego pakiet zostanie wysłany.
<code>oiftype</code>	Typ interfejsu, z którego pakiet zostanie wysłany.
<code>priority</code>	Priorytet <code>tc</code> pakietu.
<code>protocol</code>	Protokół warstwy wyższej.
<code>rtclassid</code>	Obszar routingu dla pakietu.
<code>skgid</code>	Identyfikator grupy dla gniazda pochodzenia.
<code>skuid</code>	Identyfikator użytkownika dla gniazda pochodzenia.

Wyrażenia śledzenia połączenia (ang. *conntrack expressions*) wykorzystują metadane z pakietu w celu zapewnienia informacji do dalszego przetwarzania reguł. Wyrażenia śledzenia połączenia dodaje się za pomocą słowa kluczowego `ct`, po którym następuje jedna z poniższych opcji:

- `daddr;`
- `direction;`
- `expiration;`
- `helper;`
- `l3proto;`
- `mark;`
- `protocol;`
- `proto-src;`
- `proto-dst;`
- `saddr;`
- `state;`
- `status.`

Wyrażenie stanu (state) jest ważnym wyrażeniem dla zapory sieciowej. Normalna inspekcja pakietów i przetwarzanie reguł są bezstanowe, co oznacza, że przetwarzanie „nie wie nic” o wcześniej przetworzonym pakiecie. Każdy pakiet jest sprawdzany zgodnie z jego unikatowymi cechami charakterystycznymi, takimi jak adresy źródłowe i docelowe, porty i inne kryteria. Wymienione poniżej wyrażenia stanu umożliwiają zapisywanie informacji o pakiecie, tak aby reguła przetwarzania miała kontekst dotyczący trwającej wymiany powiązanego ruchu.

- `new` — nowy pakiet przybywający do zapory sieciowej, na przykład pakiet TCP z ustawioną flagą SYN;
- `established` — pakiet będący częścią połączenia, które jest już przetwarzane lub śledzone;
- `invalid` — pakiet, który nie jest zgodny z regułami protokołów;
- `related` — pakiet związany z połączeniem dla protokołu, który nie używa żadnych innych środków do śledzenia swojego stanu, na przykład protokół ICMP lub pasywny FTP;
- `untracked` — stan administracyjny wykorzystywany do omijania śledzenia połączenia, zazwyczaj stosowany tylko w szczególnych przypadkach.

W praktyce często używane są stany `new`, `related` i `established`, a stan `invalid` w razie potrzeby. Przedstawiona poniżej reguła dopuszcza na przykład połączenia SSH ustanowione (`established`) i powiązane (`related`). Dopuszczanie połączeń powiązanych jest istotne w przypadku, gdy pamięć stanu zostanie wyczyszczona, a tym samym zanegowane zostaną wszystkie ustanowione stany połączenia.

```
nft add rule filter input tcp dport 22 ct state established,related accept
```

Mechanizm stanu został omówiony szczegółowo w rozdziale 3., w podpunkcie „Rozszerzenie dopasowywania state tabeli filter”.

Wyrażenia bloku danych są wykorzystywane do budowania reguł, które dopasowują określone szczególne kryteria i są ściśle związane z rodzajem przetwarzanego pakietu.

Wyrażenia dla nagłówków IPv4 zostały przedstawione w tabeli 4.2.

Tabela 4.2. Wyrażenia bloku danych dla IPv4

Wyrażenie	Opis
<code>checksum</code>	Suma kontrolna nagłówka IP.
<code>daddr</code>	Docelowy adres IP.
<code>frag-off</code>	Przesunięcie fragmentacji.
<code>hdrlength</code>	Długość nagłówka IP razem z opcjami.
<code>id</code>	Identyfikator IP.
<code>length</code>	Całkowita długość pakietu.

Tabela 4.2. Wyrażenia bloku danych dla IPv4 — *ciąg dalszy*

Wyrażenie	Opis
Protocol	Protokół warstwy wyższej.
saddr	Źródłowy adres IP.
tos	Wartość typu usługi (TOS).
ttl	Wartość czasu życia (TTL).
version	Wersja nagłówka IP. W wyrażeniach IPv4 zawsze wartość 4.

Wyrażenia dla nagłówków IPv6 zostały przedstawione w tabeli 4.3.

Tabela 4.3. Wyrażenia nagłówka IPv6

Wyrażenie	Opis
daddr	Docelowy adres IP.
flowlabel	Etykieta przepływu.
hoplimit	Limit przeskoków.
length	Długość bloku danych.
nextthdr	Protokół następnego nagłówka.
priority	Wartość priorytetu.
saddr	Źródłowy adres IP.
version	Wersja nagłówka IP. W wyrażeniach IPv6 zawsze wartość 6.

Wyrażenia dla nagłówków TCP zostały przedstawione w tabeli 4.4.

Tabela 4.4. Wyrażenia nagłówka TCP

Wyrażenie	Opis
ackseq	Numer potwierdzenia.
checksum	Suma kontrolna pakietu.
doff	Długość nagłówka.
dport	Port, dla którego przeznaczony jest pakiet.
flags	Flagi TCP.
sequence	Numer sekwencyjny.
sport	Port, z którego pochodzi pakiet.
urgptr	Wartość wskaźnika priorytetu.
window	Szerokość okna TCP.

UDP jest zasadniczo prostszym protokołem, istnieje więc mniej wyrażeń dla nagłówek UDP. Wyrażenia te zostały przedstawione w tabeli 4.5.

Tabela 4.5. Wyrażenia nagłówka UDP

Wyrażenie	Opis
checksum	Suma kontrolna pakietów.
dport	Port, do którego przeznaczony jest pakiet.
length	Całkowita długość pakietu.
sport	Port, z którego pochodzi pakiet.

Wyrażenia nagłówka dostępne dla ARP zostały przedstawione w tabeli 4.6.

Tabela 4.6. Wyrażenia nagłówka ARP

Wyrażenie	Opis
hlen	Długość adresu sprzętowego.
htype	Typ warstwy fizycznej.
op	Operacja.
plen	Długość protokołu warstwy wyższej.
ptype	Typ protokołu warstwy wyższej.

Podstawowe operacje nftables

Przy dodawaniu reguły określone są: tabela i łańcuch oraz kryteria dopasowywania. Dodanie na przykład reguły przyjmowania połączeń SSH z określonego hosta będzie wyglądać tak, jak przedstawiono poniżej. Ta reguła jest dodawana do poprzednio utworzonego łańcucha input tabeli filter:

```
nft add filter input tcp dport 22 accept
```

Poszczególne instrukcje, takie jak accept, drop, reject czy log (omówione w poprzednim punkcie), w iptables były nazywane **rozszerzeniami** (ang. *extensions*). Wiele z tych samych opcji i trybów operacji, które działały dla tych rozszerzeń, działa również z nftables, na przykład do rejestrowania połączeń przychodzących używana jest instrukcja log. Ta instrukcja może być łączona ze śledzeniem połączenia w taki sposób, że rejestrowane będą tylko nowe połączenia z portem 22. Ponadto można dodać ograniczenie, aby mechanizm rejestrowania nie był przeciążony.

W nftables rejestrowanie wymaga modułów jądra nfnetlink_log lub xt_LOG albo wsparcia jądra dla tych modułów. Dodatkowo należy w katalogu *proc* włączyć rejestrowanie poprzez nadanie parametrowi nf_log wartości "ipt_LOG":

```
echo "ipt_LOG" > /proc/sys/net/netfilter/nf_log/2
```

Ostateczne polecenie nftables służące do rejestrowania nowych połączeń SSH (z ograniczoną prędkością) wygląda następująco:

```
nft add filter input tcp dport 22 ct state new limit rate 3/second log
```

Jako dalsze selektory w ramach reguły używane są metawyrażenia, takie jak te służące do wybierania interfejsów wejściowego i wyjściowego. Polecenie używane na przykład do rejestrowania nowych połączeń przychodzących na interfejs eth0 będzie wyglądać tak:

```
nft add filter input iif eth0 ct state new limit rate 10/minute log
```

Reguły i opcje składni dla różnych wyrażen zostały opisane w rozdziale 3.

Składnia plików nftables

Jedną z najlepszych cech nftables jest możliwość odczytu zewnętrznych plików zawierających reguły nftables. Te pliki umożliwiają importowanie zapisanych zestawów reguł i wykorzystywanie ich bez konieczności tworzenia długich i skomplikowanych skryptów powłoki. Mimo to skrypty powłoki nadal są pomocne jako główny kontener dla plików reguł zapory sieciowej, który importuje je w odpowiednim czasie.

Pliki są importowane poprzez dodanie do nftables opcji -f. Przedstawiony poniżej plik tworzy na przykład podstawową zaporę sieciową filtrującą pakiety, która rejestruje nowe pakiety SSH (z ograniczoną prędkością):

```
table filter {
    chain input {
        type filter hook input priority 0;
        tcp dport 22 ct state new limit rate 3/second log prefix "NEW packet: "
    }

    chain output {
        type filter hook input priority 0;
    }
}
```

Jeśli przyjmiemy, że plik został zapisany pod nazwą *firewall.nft*, można go załadować za pomocą następującego polecenia:

```
nft -f firewall.nft
```

Podsumowanie

Program nftables jest podobny do iptables pod tym względem, że reguły i opcje tych programów zazwyczaj dobrze się przekładają podczas budowania zapory sieciowej. Program nftables wykorzystuje tabele zawierające łańcuchy, które z kolei zawierają reguły. Reguły wskazują nftables, co zrobić z przetwarzanymi pakietami. Podobnie jak w przypadku iptables, program nftables może akceptować, porzucać, odrzucać i rejestrować pakiety oraz wykonywać na nich inne podobne działania. Program nftables

może również obejmować przetwarzanie oparte na stanie. Zastępuje programy arptables, iptables i ebtables.

Ponieważ wiele reguł i operacji z nftables jest podobnych do tych z iptables, w celu uzyskania informacji o wyrażeniach, które nie zostały bezpośrednio opisane w tym rozdziale, możesz odwołać się do rozdziału 3.

Skorowidz

A

administrowanie zaporą

sieciową, 73, 109

adres

sieciowy 0, 25, 138

sieciowy pętli zwrotnej
127, 25

adresowanie IP, 24

adresy

broadcastowe, 52

docelowe, 239

dynamiczne IP, 173

ethernetowe, 36

IP, 36, 51

IP klas A, B i C, 51

IP klasy D, 51

IP klasy E, 52

lokalne, 146

rozgłoszeniowe, 25

sieci LAN, 51

specjalne IP, 25

TEST-NET, 52

zdalne, 146

AIDE, 345–62

aktualizacje, 304

alerty, 340

ALG, Application-Level

Gateway, 44

APNIC, 290

architektura hosta

zatajonego, 214

architektura podsieci

zatajonej, 214

ARIN, 290

ARP, Address Resolution

Protocol, 35

ARPWatch, 295

atak

DoS, 60, 66

ping flood, 62

Ping of Death, 62

smerfów, 331

TCP SYN flood, 60

typu LAND, 332

UDP flood, 63

Xmas Tree, 331

z perspektywy

TCPDump, 328

automatyczne uruchamianie

AIDE, 351

B

baza danych AIDE, 351

bezpieczeństwo, 365

sieci LAN, 217

blokowanie pakietu, 50

problemowych

lokalizacji, 53

pakietu, 49

bomby

fragmentacyjne, 64

ICMP Redirect, 66

brama, 215

na poziomie aplikacji, 44

warstwy aplikacji, 44

broadcast, 25

ograniczony, 53

broadcasting, 28

budowanie plików reguł, 205

C

CIDR, Classless Inter

Domain Routing, 27

czyszczenie bazy danych, 353

D

debugowanie reguł, 249

debugowanie skryptu zapory

sieciowej, 171

definiowanie

domyślnej polityki, 133

makr, 357

demon, 37

pptpd, 273

syslogd, 258

DHCP, 29

diagram sekwencji, 46

dławik, 216

DMZ, Demilitarized Zone, 215

DNAT, 79, 242

DNS, 148, 191

dokument RFC 2647, 43

- domyślne
 - akceptowanie
 - wszystkiego, 49
 - blokowanie wszystkiego, 48
 - dopasowanie
 - ESTABLISHED, 178
 - iprange, 102
 - length, 103
 - RELATED, 178
 - dopasowywanie tabeli filter, 87
 - Dos, Denial of Service, 60
 - dostęp do
 - bramy z zaporą, 220
 - serwera czasu, 168
 - serwera DHCP, 166
 - serwera FTP, 163
 - serwera SSH, 160
 - DS, Differentiated Services, 80
 - dwukierunkowy NAT, 81
 - dwukrotny NAT, 82
 - dyrektywy konfiguracyjne programu AIDE, 348
 - działanie pakietu Snort, 339
 - dzielenie przestrzeni adresowej, 224
 - dziennik systemowy, 256, 281
- F**
- falszowanie adresu źródłowego, 51, 136
 - filtrowanie
 - IP TOS, 80
 - lokalnych
 - adresów docelowych, 53
 - adresów źródłowych, 68
 - portów docelowych, 54
 - portów źródłowych, 69
 - pakietów, 43
 - przychodzących, 50
 - wychodzących, 67
 - stanów
 - przychodzących
 - połączeń TCP, 55
 - wychodzących
 - połączeń TCP, 70
 - zdalnych
 - adresów docelowych, 68
 - adresów źródłowych, 50
 - portów docelowych, 69
 - portów źródłowych, 54
 - firewall, 19
 - flaga TCP, 146
 - flagi nagłówka TCP, 33, 331
 - format
 - nagłówka AH, 271
 - nagłówka ESP, 272
 - składni, 85
 - fragmentacja IP, 28
 - FTP, 161
 - funkcje
 - programu iptables, 78
 - programu nftables, 110
 - rejestrwania syslog, 256
 - tabeli mangle, 83
 - tabeli nat, 81
- H**
- hostowanie serwera
 - IMAP, 159
 - POP, 158
- I**
- ICMP, 29
 - IMAP, 157
 - implementacje sieci VPN, 275
 - incydent, 286
 - infekcja, 298
 - inicjowanie
 - bazy danych AIDE, 351
 - zapory sieciowej, 126, 186, 204
 - instalowanie
 - łańcuchów, 188
 - pakietu Snort, 336
 - programu AIDE, 347
 - TCPDump, 313
 - samodzielnej zapory sieciowej, 121
 - zapory sieciowej, 170, 173
 - integralność systemu plików, 301, 345
 - iptables, 73
 - ISP, 290
- J**
- jądro systemu, 124
- K**
- kanal
 - danych, 326
 - danych FTP, 164
 - sterowania, 326
 - klient
 - IMAP, 157
 - POP, 156
 - komunikaty
 - DHCP, 166
 - dziennika syslog, 257
 - dziennika zapory sieciowej, 259
 - ICMP, 30, 66
 - konsoli i terminala, 281
 - koncentrator, 295
 - konfiguracja
 - AIDE, 347
 - bramy z zaporą sieciową, 215
 - pakietu Snort, 337
 - syslog, 256, 258
 - systemu, 282
 - konta użytkowników, 283
 - krotka, 95
 - kwalifikator
 - kierunku TCPDump, 317
 - protokołu TCPDump, 318
 - typu TCPDump, 316

L

LAN, local area network, 19
 licencja GNU Wolnej
 Dokumentacji, 417
 lista reguł, 251

Ł

łańcuch, 86, 87
 connection-tracking, 200
 destination-address-
 check, 201
 EXT-icmp-in, 197
 EXT-icmp-out, 197
 EXT-input, 190
 EXT-output, 190
 FORWARD, 239
 local-dhcp-client-query,
 200
 local-dns-client-request,
 193
 local-dns-server-query,
 192
 local-tcp-server-
 response, 195
 local-udp-client-request,
 196
 log-tcp-state, 199
 remote-dhcp-server-
 response, 200
 remote-dns-server-
 response, 192
 remote-tcp-client-
 request, 195
 remote-udp-server-
 response, 196
 source-address-check,
 201
 tcp-state-flags, 199
 łańcuchy
 reguł, 73
 zdefiniowane przez
 użytkownika, 181, 183

M

makro, 357
 mapowanie nazw usług, 38
 maska
 domyślna, 26
 podsieci, 26
 maskarada, 74, 82, 239
 maskowanie ruchu LAN, 244
 MASQUERADE, 79
 mechanizmy transportowe, 31
 metawyrażenia w nftables, 115
 model
 referencyjny TCP/IP, 45
 sieciowy OSI, 22
 monitorowanie, 341
 alertów Snort, 340
 nieprawidłowości, 352
 plików dziennika, 301
 sieci, 309
 włamań, 335
 MSS, Maximum Segment
 Size, 35
 multicasting, 28

N

nagłówek
 AH, 271
 ESP, 272
 IPv4, 26
 TCP, 32
 NAT, Network Address
 Port Translation, 235
 narzędzia
 do audytu
 bezpieczeństwa, 283
 do wykrywania
 rootkitów, 295
 sieciowe, 293
 do wykrywania włamań,
 293
 narzędzie, *Patrz* program
 nasłuchiwanie eteru, 309
 NAT, Network Address
 Translation, 21, 213, 233

docelowy, 242
 docelowy typu
 REDIRECT, 243
 dwukierunkowy, 235
 dwukrotny, 236
 jednokierunkowy
 NAT, 235
 podstawowy, 235
 u operatora, 52
 źródłowy, 240
 źródłowy typu
 MASQUERADE, 242
 nazwy
 hostów, 36
 symboliczne, 125
 nftables, 109
 nielegalne adresy, 51
 numery portów usług, 141

O

obsługa monitorowania, 128
 ochrona
 niezabezpieczonych
 usług lokalnych, 71
 usług, 141
 odbieranie poczty, 156
 odrzucanie
 pakietu, 49, 50
 połączeń, 143
 odwiedzanie zdalnej
 witryny, 39
 ograniczanie pakietów
 przychodzących, 53
 ograniczenia
 programu Chkrootkit, 299
 samodzielnej zapory
 sieciowej, 213
 omijanie sprawdzania reguł,
 135
 opcje
 adresowania
 docelowego, 125
 źródłowego, 125
 fragmentacji, 253

- konfiguracyjne sieci domowej, 218
 - konfiguracyjne sieci LAN, 223, 226, 228
 - narzędzia TCPDump, 314
 - Openswan/Libreswan, 273
 - OpenVPN, 273
 - operacje
 - dopasowywania icmp, 89
 - dopasowywania tcp, 88
 - dopasowywania udp, 89
 - na łańcuchach, 86
 - na regule, 87
 - nftables, 118
 - reguł, 88
 - tabeli filter, 86
 - we-wy, 146
 - optymalizacja
 - iptables, 210
 - nftables, 211
 - zapory sieciowej, 177, 210
 - organizacja reguł, 177
 - otwarte porty, 263
- P**
- pakiet
 - Snort, 335, 340
 - Swatch, 302
 - pakiety
 - blokowanie, 49
 - odrzućanie, 49
 - pakiety routowane
 - źródłowo, 67
 - pętla zwrotna, 132
 - plik
 - arp.dat, 343
 - konfiguracyjny AIDE, 347, 350
 - localhost-policy, 133
 - pliki
 - dziennika systemowego, 281
 - nftables, 119
 - reguł, 205
 - pobieranie poczty, 156
 - poczta wychodząca, 154
 - podsielowanie, 24
 - bezklasowe, 27
 - polecenia tabeli
 - filter, 85
 - mangle, 106
 - polecenie
 - fuser, 266
 - iptables, 77, 84
 - iptables -L INPUT, 252
 - polityka domyślna, 133
 - połączenie TCP, 34, 40, 41
 - POP, 156
 - port
 - lokalny, 146
 - zdalny, 146
 - porty
 - nieuprzywilejowane, 37, 141
 - o dużych numerach, 178
 - serwera DNS BIND, 152
 - usług, 37, 56
 - porzucanie nieprawidłowego ruchu, 207
 - potwierdzenie żądania, 40
 - PPTP, 273
 - priorytety komunikatów, 257
 - procedura three-way
 - handshake, 34, 43
 - program
 - iptables, 73
 - nftables, 109
 - program
 - AIDE, 345–362
 - ARPWatch, 312, 341
 - Bastille Linux, 304
 - Chkrootkit, 296, 298, 300
 - hping3, 307
 - ipchains, 74
 - netstat, 263, 266
 - Nikto, 307
 - Nmap, 266, 306, 329
 - Snort, 311
 - Swatch, 302, 340
 - TCPDump, 311, 319, 333
 - programy
 - do administrowania
 - zaporą sieciową, 122
 - serwera, 37
 - protokoły
 - bezpółczeniowe, 23
 - pocztowe, 154
 - połączeniowe, 23
 - VPN, 269
 - protokół, 146, 253
 - AH, 270
 - ARP, 29, 35
 - DHCP, 167
 - DNS, 148, 149
 - ESP, 271
 - FTP, 162
 - GRE, 270
 - IKE, 272
 - IP, 24
 - IPsec, 270
 - L2TP, 270
 - PPTP, 270
 - SMTP, 153
 - TCP, 32, 193, 195
 - transportowy, 178
 - UDP, 32, 196
 - proxy na poziomie obwodu, 44
 - przechwytywanie
 - konwersacji HTTP, 320
 - konwersacji SMTP, 325
 - konwersacji SSH, 326
 - pakietów ping, 327
 - protokołów, 319
 - protokołów opartych na
 - TCP, 326
 - zapytań DNS, 327
 - przekazywanie pakietów, 37, 213, 233, 238
 - przekierowanie hostów, 246
 - przełącznik, 295
 - przepelnienie bufora, 65
 - przepływ ruchu, 180

R

raport AIDE, 355
 raportowanie danych, 266
 REDIRECT, 79
 reguły, 87
 blokujące ruch, 178
 dla wielu interfejsów sieciowych, 180
 domyślnej polityki, 134
 dynamiczne, 135
 iptables, 135
 spoofingu, 178
 statyczne, 135
 UDP, 179
 zapory sieciowej, 180
 rejestrowanie
 porzuconych pakietów, 202, 209
 przychodzących, 169
 wychodzących, 170
 ruchu, 333
 w zaporze sieciowej, 136
 resetowanie domyślnych polityk, 131
 RIPE, 290
 rodzaje
 incydentów, 288
 kontroli AIDE, 359
 rootkit, 295
 routing, 37
 międzydomenowy, 27
 rozszerzenie, 118
 celu
 DNAT, 104
 mark, 106
 LOG, 91
 MASQUERADE, 104
 REDIRECT, 105
 SNAT, 103
 tabeli filter, 90
 tabeli nat, 103
 ULOG, 91
 dopasowywania
 addrtype, 101

 limit, 94
 mac, 99
 mark, 100
 multiport, 92
 owner, 99
 state, 95, 97
 tos, 100
 unclean, 101
 ruch
 DNS, 191, 207
 ICMP, 197, 209
 lokalnego
 hosta, 206
 klienta, 193, 196, 208
 serwera, 195, 209
 sieci LAN, 219

S

segmenty TCP, 33
 selektywny dostęp, 226
 semantyka translacji NAT, 238
 serwer
 czasu, 168
 DHCP, 166
 DNS, 148
 DNS BIND, 152
 FTP, 161
 IMAP, 159
 POP, 158
 SMTP, 156
 serwery proxy, 246
 sieć
 klasy C, 225
 LAN, 19, 218, 223
 VPN, 269
 skanowanie, scan, 55, 329
 ogólne portów, 56
 portów, 58, 142
 skryte, 58
 ukierunkowane portów, 56
 składnia
 dopasowywania, 92
 łańcucha, 113

 nftables, 109, 111
 plików nftables, 119
 polecenia iptables, 77, 84
 reguły, 114
 tabeli, 112
 skomarzenia SA, 273
 skrypt, 367
 iptables, 184
 nftables, 204
 SMTP, 153, 154
 SNAT, 79, 241
 sonda, probe, 55
 splukanie, 130
 sprawdzanie adresu źródłowego, 197
 SSH, 159
 stałe symboliczne, 127
 standardowy
 DNAT, 242
 SNAT, 241
 stosowanie translacji NAT, 240, 245
 strefa ograniczonego zaufania, 215
 strumień standardowe, 355
 suma kontrolna, 362
 system
 dwuadresowy, 213
 plików, 282, 301, 345

Ś

śledzenie stanu połączenia, 206

T

tabela
 filter, 85, 88–103
 mangle, 79, 83, 106
 nat, 79, 81, 103–105
 tabele protokołów, 145
 TCP, Transmission Control Protocol, 32

- testowanie, 305
 - działania pakietu Snort, 339
 - translacja
 - adresów docelowych, 79
 - adresów źródłowych, 74, 79
 - DNAT, 246
 - NAT, 21, 213, 233
 - SNAT, 244
 - trawersacja
 - łańcucha, 182
 - pakietów, 75, 76
 - NAT, 83, 103
 - pętli zwrotnej, 181
 - w zaporze sieciowej, 181
 - zamaskowanych, 181
 - tryb
 - pasywny, 163
 - portu, 163
 - transportowy, 272
 - tunelowy, 272
 - tworzenie
 - łańcuchów, 188
 - tabel, 206
 - wielu sieci, 224
 - zapory sieciowej, 249
 - typy
 - gniazd, 264
 - komunikatów DHCP, 166
- U**
- UDP, User Datagram Protocol, 32
 - uruchamianie zapory sieciowej
 - Debian, 173
 - Red Hat, 172
 - SUSE, 172
 - usługi
 - DNS, 147
 - E-mail, 153
 - ICMP, 179
 - lokalne TCP, 142
 - lokalne UDP, 144
 - podstawowe, 147
 - sieci
 - prywatnej, 70
 - publicznej, 70
 - TCP, 152, 164, 178
 - UDP, 165 179
 - ustanowienie połączenia
 - TCP, 41
 - usuwanie reguł, 130
 - użycie dopasowań, 178
- V**
- VPN, Virtual Private Network, 269
- W**
- warstwy modelu OSI, 22
 - weryfikowanie zabezpieczeń, 304
 - wirtualne sieci prywatne, VPN, 269
 - włamania, 279, 293
 - włączanie
 - interfejsu pętli zwrotnej, 132
 - obsługi monitorowania, 128
 - ruchu DNS, 207
 - ruchu lokalnego hosta, 206
 - śledzenia stanu
 - połączenia, 206
 - typowych usług TCP, 152
 - usług internetowych, 147
 - usługi DNS, 147, 149, 151
 - usług UDP, 165
 - wskazania
 - dziennika systemowego, 281
 - konfiguracji systemu, 282
 - kont użytkowników, 283
 - narzędzi do audytu bezpieczeństwa, 283
 - systemu plików, 282
 - wydajności systemu, 284
 - wybór uruchamianych usług, 72
 - wydajność systemu, 284
 - wykrywanie
 - ataków, 309
 - rootkitów, 295
 - włamań, 279, 293
 - wyrażenia
 - bloku danych, 116
 - nagłówka ARP, 118
 - nagłówka IPv6, 117
 - nagłówka TCP, 117
 - nagłówka UDP, 118
 - pierwotne, 318
 - TCPDump, 316
 - wysyłanie poczty, 153, 155
 - wyszukiwanie DNS, 149, 151
 - wyświetlanie
 - listy reguł, 251
 - zawartości tabel, 252, 255
- Z, Ż**
- zapobieganie atakom, 303
 - zapora sieciowa, 19, 43
 - filtrująca pakiety, 45
 - IPFW, 75
 - iptables, 367
 - iptables
 - zoptymalizowana, 384
 - Netfilter, 74, 76
 - nftables, 380, 397
 - samodzielna, 121
 - typu twierdza, 19
 - zasada pierwszej
 - dopasowanej reguły, 134
 - zasoby systemowe, 66
 - zatrzymywanie zapory sieciowej, 131
 - zaufane sieci domowe LAN, 219
 - zgłaszanie incydentów, 286, 289
 - żądania FTP, 163

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄŻKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

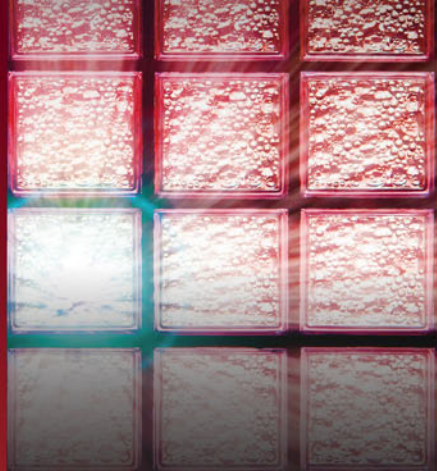
<http://program-partnerski.helion.pl>

ZAAWANSOWANE ZAPORY SIECIOWE DLA KAŻDEGO!

Zapora sieciowa (ang. *firewall*) to ostatni bastion chroniący Twój komputer przed atakiem. Dzięki odpowiedniej konfiguracji jesteś w stanie decydować, które pakiety mogą trafić do wnętrza sieci, a które nie. Możesz przesądzić o dostępie do określonych usług, zezwolić lub zabronić aplikacjom korzystać z dostępu do sieci Internet oraz ustalić limity na prędkość albo ilość przesłanych danych. Duże możliwości konfiguracji pozwalają na elastyczne podejście do tematu. Dzięki tej książce możesz wykorzystać wszystkie dostępne opcje!

Jednak na samym początku zapoznasz się z podstawowymi zasadami działania zapór sieciowych filtrujących pakiety. Przypomnisz sobie model sieciowy OSI, protokół IP, zasady routingu oraz sposób wyboru reguł filtrujących pakiety. Po tym wstępie przejdziesz do szczegółowego badania możliwości starszego narzędzia, jakim jest iptables. Składnia, omówienie dostępnych rozszerzeń dopasowywania, konfiguracja NAT — to tylko niektóre z poruszanych kwestii. Czas się zmieniają i obecnie popularnością cieszy się narzędzie znane pod nazwą nftables. Dlatego z kolejnych rozdziałów dowiesz się, jak zbudować i uruchomić własną zaporę sieciową, korzystając właśnie z nftables. W następnych krokach uruchomisz usługi takie jak: e-mail, SSH, FTP oraz DHCP. Dalej zajmiesz się zagadnieniami związanymi z optymalizacją działania zapory oraz usługą NAT, czyli udostępnianiem łącza innym komputerom. W tej książce zostały poruszone niezwykle ważne kwestie łączące się z wykrywaniem włamań i najlepszymi sposobami reagowania na te incydenty. Ta książka jest doskonałą lekturą zarówno dla administratorów sieci, jak i dla pasjonatów systemu Linux!

STEVE SUEHRING — ma ponad 20-letnie doświadczenie w administrowaniu sieci i bezpieczeństwa systemów Linux. Redaktor do spraw bezpieczeństwa w magazynie „LinuxWorld”. Konsultant, architekt technologii, autorytet w branży IT. Autor książek *JavaScript krok po kroku* oraz *MySQL Bible*.



Dzięki tej książce:

Poznasz podstawowe koncepcje związane z zaporami sieciowymi

Zaznajomisz się z narzędziem iptables

Skonfigurujesz i uruchomisz zaawansowaną zaporę dzięki nftables

Zoptymalizujesz działanie zapory sieciowej

Udostępnisz łącze innym urządzeniom w Twojej sieci

Prawidłowo zareagujesz na incydenty

W pełni wykorzystasz możliwości iptables oraz nftables!

Helion

37633 numer katalogowy

księgarnia internetowa

<http://helion.pl>

zamówienia telefonicznie



0 801 339900



0 601 339900

Sprawdź najnowsze promocje:
● <http://helion.pl/promocje>
Książki najchętniej czytane:
● <http://helion.pl/bestsellery>
Zamów informacje o nowościach:
● <http://helion.pl/nowosci>

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
<http://helion.pl>

Addison-Wesley

ISBN 978-83-283-1297-5



9 788328 312975

cena: 69,00 zł

Informatyka w najlepszym wydaniu

sięgnij po **WIĘCEJ**



KOD KORZYŚCI