

IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK

DODAJ DO KOSZYKA

CENNIK I INFORMACJE

ZAMÓW INFORMACJE
O NOWOŚCIACH

ZAMÓW CENNIK

CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

101 zabezpieczeń przed atakami w sieci komputerowej

Autorzy: Maciej Szmit, Marek Gusta,
Mariusz Tomaszewski
ISBN: 83-7361-517-2
Format: B5, stron: 560



Chyba każda sieć komputerowa na świecie była już atakowana przez hakerów. Niektóre z ataków były skuteczne, inne nie. Efekty skutecznego ataku hakerów mogą być różne – od braku szkód, aż po utratę ważnych danych lub, co często okazuje się znacznie gorsze – wydostanie się takich danych na zewnątrz. Co sprawia, że niektóre sieci opierają się atakom hakerów, a inne nie? Sekret tkwi w zabezpieczeniach i pracy administratora.

W książce „101 zabezpieczeń przed atakami w sieci komputerowej” każdy, kto chce zabezpieczyć swoją sieć przed niepowołanym dostępem, znajdzie niezbędną do tego wiedzę. Książka przedstawia różne rodzaje ataków, sposoby ich wykrywania i metody ochrony sieci przed nimi. Opisuje ataki na różne warstwy i elementy sieci oraz zasady korzystania z zapór sieciowych.

- Wykrywanie sniffingu i ochrona przed nim
- Skanowanie portów i IP-spoofing
- Ataki typu DoS
- Wirusy, robaki i programy szpiegujące
- Zabezpieczanie procesu logowania
- Ochrona przed atakiem przez przepełnienie bufora
- Technologie i architektury zapór sieciowych
- Systemy wykrywania ataków typu IDS

Jeśli chcesz, aby administrowana przez Ciebie sieć była bezpieczna, skorzystaj ze sposobów przedstawionych w tej książce.



Spis treści

Wstęp	11
Rozdział 1. Metody obrony przed atakami prowadzonymi w warstwie dostępu do sieci	15
Sniffing w sieci o fizycznej topologii magistrali i w sieci wykorzystującej koncentratory.....	16
Programy wykorzystywane do podsłuchu.....	16
Tcpdump	16
Ethereal.....	17
Sniff v. 1.4	18
Konfiguracja sieci testowej	18
Przeprowadzenie ataku.....	20
Sniffing w sieci zbudowanej przy wykorzystaniu przełączników	22
Konfiguracja sieci testowej	23
Sniffing w sieci zbudowanej przy wykorzystaniu przełączników — ARP-spoofing....	24
Przeprowadzenie ataku	25
Sniffing w sieci zbudowanej z wykorzystaniem przełączników — MAC-flooding.....	26
Przeprowadzenie ataku	27
Sniffing w sieci zbudowanej przy wykorzystaniu przełączników — duplikacja adresu fizycznego	29
Przeprowadzenie ataku	29
Antysniffing	30
Zabezpieczenie nr 1. Wykrywanie sniffingu za pomocą testu ARP.....	31
Zabezpieczenie nr 2. Wykrywanie sniffingu za pomocą testu ARP-Cache	34
Zabezpieczenie nr 3. Wykrywanie sniffingu za pomocą testu ICMP.....	37
Zabezpieczenie nr 4. Wykrywanie sniffingu za pomocą testu DNS	39
Zabezpieczenie nr 5. Wykrywanie sniffingu za pomocą pomiarów czasów latencji.....	41
Zabezpieczenie nr 6. Wykrywanie podsłuchu metodami reflektometrycznymi..	45
Zabezpieczenie nr 7. Wykrywanie ataku ARP-spoofing za pomocą programu arpwatc	46
Zabezpieczenie nr 8. Ochrona przed atakiem ARP-spoofing za pomocą statycznej tablicy ARP.....	48
Zabezpieczenie nr 9. Wykrywanie ataku ARP-spoofing za pomocą programu ARP-Analyzer	51
Zabezpieczenie nr 10. Lokalne wykrywanie sniffingu.....	54
Zabezpieczenie nr 11. Ochrona przed podsłuchem przy użyciu technologii VLAN.....	55

Zabezpieczenie nr 12. Przełączniki zarządzalne jako zabezpieczenie przed podsłuchem	59
Zabezpieczenie nr 13. Wirtualne sieci prywatne jako zabezpieczenie przed podsłuchem	59
Zabezpieczenie nr 14. Wykrywanie ataku MAC-flooding za pomocą programu MACManipulator	66
Zabezpieczenie nr 15. Szyfrowanie połączenia sieciowego z wykorzystaniem protokołu SSL	68
Zabezpieczenie nr 16. Szyfrowanie połączenia sieciowego z wykorzystaniem protokołu TLS	77

Rozdział 2. Metody obrony przed atakami prowadzonymi w warstwach internetu i host-to-host	79
Skanowanie portów	79
Nmap	80
Instalacja Nmapa	80
Instalacja Nmapa w systemie Linux	80
Instalacja Nmapa w systemie Windows	81
Techniki skanowania portów	82
Skanowanie TCP-connect	83
Zabezpieczenie nr 17. NAT jako składnik zapory sieciowej	83
Zabezpieczenie nr 18. Usługi pośredniczenia (proxy) w roli zapory sieciowej	87
Pośredniczenie za pomocą Socks	88
Konfiguracja komputera B	88
Konfiguracja komputera A	92
Testowanie działania usługi pośredniczącej	94
Zabezpieczenie nr 19. Wykorzystanie zapory sieciowej IPTables do blokowania prób skanowania TCP-connect	96
Skanowanie TCP SYN	99
Zabezpieczenie nr 20. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania TCP SYN	99
Skanowanie TCP FIN	100
Zabezpieczenie nr 21. Wykorzystanie systemu IDS Snort do wykrywania prób skanowania TCP FIN	100
Zabezpieczenie nr 22. Wykorzystanie zapory sieciowej IPTables do blokowania prób skanowania TCP FIN	101
Skanowanie TCP ACK	103
Zabezpieczenie nr 23. Wykorzystanie systemu IDS Snort do wykrywania prób skanowania TCP ACK	104
Zabezpieczenie nr 24. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania TCP ACK	104
Skanowanie TCP NULL	105
Zabezpieczenie nr 25. Wykorzystanie systemu IDS Snort do wykrywania prób skanowania TCP NULL	105
Zabezpieczenie nr 26. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania TCP NULL	105
Skanowanie TCP XMAS	106
Zabezpieczenie nr 27. Wykorzystanie systemu IDS Snort do wykrywania prób skanowania TCP XMAS	106
Zabezpieczenie nr 28. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania TCP XMAS	107
Skanowanie FTP-bounce	107
Zabezpieczenie nr 29. Obrona przed skanowaniem FTP-bounce	110

Zabezpieczenie nr 30. Narzędzie Portsentry jako obrona przed skanowaniem portów w systemie Linux	111
Przeprowadzenie próby skanowania portów komputera zabezpieczonego przez Portsentry	115
Zabezpieczenie nr 31. Osobiste zapory sieciowe jako obrona przed skanowaniem portów w systemach Windows.....	116
Skanowanie UDP	128
Zabezpieczenie nr 32. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania pakietami UDP	129
Skanowanie ICMP	129
Zabezpieczenie nr 33. Wykorzystanie zapory sieciowej iptables do blokowania prób skanowania pakietami ICMP.....	129
Techniki ukrywania przez napastnika skanowania portów.....	130
Metody wykrywania systemu operacyjnego (ang. OS fingerprint)	131
Pasywne wykrywanie systemu operacyjnego.....	131
Pasywna analiza stosu TCP/IP.....	132
Aktywne wykrywanie systemu operacyjnego	134
Zabezpieczenie nr 34. Zmiana parametrów stosu TCP/IP w systemie Linux w celu utrudnienia fingerprintingu.....	136
IP-spoofing	137
Zabezpieczenie nr 35. Filtrowanie ruchu na zaporze sieciowej jako zabezpieczenie przed atakami wykorzystującymi IP-spoofing	137
Zabezpieczenie nr 36. Weryfikacja adresu źródłowego za pomocą funkcji rp_filter	141
Atak wyboru trasy (ang. Source routing).....	142
Zabezpieczenie nr 37. Wyłączenie opcji source routing	142
Zabezpieczenie nr 38. Wykorzystanie uwierzytelniania RIPv2 jako ochrona przed atakami na routery	143
Rozdział 3. Metody obrony przed atakami DoS i DDoS	155
Ataki DoS.....	155
Ping of Death	156
Zabezpieczenie nr 39. Ochrona przed atakiem Ping of Death za pomocą filtrowania na zaporze sieciowej.....	156
Teardrop.....	157
Zabezpieczenie nr 40. Ochrona przed atakiem teardrop za pomocą systemu Snort	158
Zabezpieczenie nr 41. Ochrona przed atakiem teardrop za pomocą filtrowania pakietów	158
Atak SYN-flood	158
Zabezpieczenie nr 42. Ochrona przed atakiem SYN-flood wychodzącym z naszej sieci za pomocą zapory sieciowej.....	159
Zabezpieczenie nr 43. Ochrona przed atakami SYN-flood i Naptha na usługi w naszej sieci za pomocą iptables	160
Atak Land.....	160
Zabezpieczenie nr 44. Ochrona przed atakiem Land za pomocą programu Snort	161
Zabezpieczenie nr 45. Ochrona przed atakiem Land za pomocą programu Snort (reguła systemu IDS)	162
Atak Naptha	162
Zabezpieczenie nr 46. Wykorzystanie systemu IDS Snort do wykrywania ataku Naptha.....	164

Atak Smurf.....	165
Zabezpieczenie nr 47. Ochrona od strony pośrednika przed atakiem Smurf za pomocą zapory sieciowej	167
Zabezpieczenie nr 48. Ochrona przed atakiem Smurf od strony ofiary za pomocą zapory sieciowej	168
UDP-flood („Pepsi”).....	168
Zabezpieczenie nr 49. Ochrona przed atakiem Pepsi za pomocą zapory sieciowej	169
Zabezpieczenie nr 50. Ochrona przed atakiem Pepsi za pomocą programu Snort	170
Smbnuke	170
Zabezpieczenie nr 51. Ochrona przed atakiem Smbnuke za pomocą filtra pakietów.....	170
Zabezpieczenie nr 52. Ochrona przed atakiem Smbnuke za pomocą programu Snort.....	171
Zalewanie maszyny połączeniami na określonym porcie — Connection-flood.....	171
Zabezpieczenie nr 53. Ochrona przed atakiem Connection-flood za pomocą zapory sieciowej	173
Fraggle	173
Zabezpieczenie nr 54. Ochrona przed atakiem fraggle za pomocą zapory sieciowej	174
Jolt.....	175
Zabezpieczenie nr 55. Ochrona przed atakiem Jolt za pomocą zapory sieciowej	175
Zabezpieczenie nr 56. Ochrona przed atakiem Jolt za pomocą programu Snort	175
Rozproszony atak typu „odmowa usługi” (DDoS).....	175
Faza powstawania sieci DDoS	178
Właściwa faza ataku	179
Szczegółowa charakterystyka ataków DDoS	179
Atak Trinoo.....	179
Atak Tribe Flood Network.....	180
Atak TFN 2000	180
Atak Stacheldraht (druć kolczasty).....	181
Atak Shaft	181
Atak Mstream	182
Obrona przed atakami DDoS.....	182
Zabezpieczenie nr 57. Ręczne wykrywanie i usuwanie demona Wintrinoo.....	183
Zabezpieczenie nr 58. Wykrywanie demona Wintrinoo za pomocą programu wtrinscan	184
Zabezpieczenie nr 59. Wykrywanie narzędzi DDoS za pomocą programu Zombie Zapper	184
Zabezpieczenie nr 60. Wykrywanie demona trinoo za pomocą programu wtrinscan	186
Zabezpieczenie nr 61. Wykrywanie i unieszkodliwianie demona trinoo narzędziem netcat	187
Zabezpieczenie nr 62. Wykrywanie demona i węzła Trinoo za pomocą programu find_ddos.....	191
Zabezpieczenie nr 63. Zdalne i lokalne usuwanie z systemu demona Trinoo...	192
Zabezpieczenie nr 64. Wykrywanie narzędzi DDoS za pomocą programu DDoSPing.....	193
Zabezpieczenie nr 65. Wykrywanie narzędzi DDoS przez analizę ruchu sieciowego.....	195

Zabezpieczenie nr 66. Wykorzystanie systemu IDS Snort do wykrywania ataku Trinoo	200
Zabezpieczenie nr 67. Wykorzystanie systemu IDS Snort do wykrywania ataku Tribe Flood Network	204
Zabezpieczenie nr 68. Wykorzystanie systemu IDS Snort do wykrywania ataku Tribe Flood Network 2000.....	204
Zabezpieczenie nr 69. Wykorzystanie systemu IDS Snort do wykrywania ataku Stacheldraht	205
Zabezpieczenie nr 70. Wykorzystanie systemu IDS Snort do wykrywania ataku Shaft.....	205
Zabezpieczenie nr 71. Wykorzystanie systemu IDS Snort do wykrywania ataku Mstream	206

Rozdział 4. Obrona przed atakami w warstwie procesów i aplikacji oraz atakami przeciwko systemom i aplikacjom sieciowym 209

Robaki, wirusy, spyware	210
Zabezpieczenie nr 72. Wykrywanie programów typu rootkit w systemie Linux	211
Zabezpieczenie nr 73. Lokalne wykrywanie koni trojańskich	212
Zabezpieczenie nr 74. Wykrywanie modyfikacji plików z zapisem logowania użytkowników w systemie Linux	216
DNS-spoofing i ataki Man-in-the-Middle na sesje szyfrowane.....	217
DNS-spoofing	218
Zabezpieczenie nr 75. Ochrona przed atakiem DNS-spoofing za pomocą statycznych odwzorowań nazw.....	220
Zabezpieczenie nr 76. Ochrona przed niechcianymi banerami, plikami cookies za pomocą odwzorowań w pliku hosts	221
Zabezpieczenie nr 77. Obrona przed atakiem Man-in-The-Middle.....	221
Łamanie hasel.....	229
Proces logowania	229
Narzędzia do łamania hasel.....	230
LOpht Crack	230
John the Ripper	230
Łamacz 1.1.....	230
Advanced ZIP Password Recovery	231
Zdalne odgadywanie hasel użytkownika	234
Zabezpieczenie nr 78. Polityka silnych hasel.....	235
Zabezpieczenie nr 79. Hasła jednorazowe	238
Przykład implementacji hasel jednorazowych w systemie Knoppix 3.4 z wykorzystaniem usługi SSH	243
Zabezpieczenie nr 80. Bezpieczne uwierzytelnianie za pomocą serwera RADIUS	247
Zabezpieczenie nr 81. Bezpieczne uwierzytelnianie za pomocą protokołu Kerberos	249
SPAM i ataki na usługi pocztowe.....	251
Zabezpieczenie nr 82. Uwierzytelnianie użytkownika końcowego SMTP oraz ograniczenia na wysyłane listy.....	252
Zabezpieczenie nr 83. Szyfrowana transmisja POP (IMAP) i SMTP	253
Zabezpieczenie nr 84. Zamykanie przekaźnika (relay)	254
Zabezpieczenie nr 85. Filtrowanie poczty przychodzącej.....	255
Zabezpieczenie nr 86. Programy kontroli rodzicielskiej	257
Przykładowa konfiguracja programu Cyber Patrol	259
Zabezpieczenie nr 87. Usuwanie lub fałszowanie etykiet wyświetlanych przez usługi sieciowe	264

Protokół DHCP	267
Zabezpieczenie nr 88. Zabezpieczenie klienta przed nielegalnym serwerem DHCP w sieci	268
Zabezpieczenie nr 89. Alokacja manualna adresów DHCP	270
Zabezpieczenie nr 90. Honeypot	274
Zabezpieczenie nr 91. Zabezpieczenie przed atakiem buffer overflow za pomocą biblioteki libsafe	287
Rozdział 5. Dziesięć dobrych rad dla administratora	291
Zabezpieczenie 92. Wykonuj regularnie kopie bezpieczeństwa	291
Uaktualnianie systemu Windows	294
Zabezpieczenie 93. Uaktualnij swój system	294
Uaktualnianie systemu Linux (dystrybucja Knoppix 3.4)	304
APT — narzędzie do zarządzania pakietami	309
Uaktualnianie systemów Novell NetWare	314
Integralność systemu plików	322
Zabezpieczenie 94. Sprawdź integralność systemu plików	322
Program FastSum w systemach Windows	338
Zabezpieczenie 95. Ogranicz fizyczny dostęp do serwera	340
Zabezpieczenie 96. — wykonuj i czytaj logi systemowe	342
Analiza logów systemowych w systemie Linux	342
Zabezpieczenie 97. Wykonaj audyt bezpieczeństwa	353
Zabezpieczenie 98. Zasztyfuj swój system plików	375
Zabezpieczenie 99. Skorzystaj z internetowych serwisów skanujących	378
Kontrola dostępu do usług	382
Zabezpieczenie 100. Ogranicz zakres świadczonych usług	382
Zabezpieczenie 101. Zarządzaj pasmem	389
Podsumowanie	400
Dodatek A Podstawy komunikacji sieciowej	401
Pojęcia podstawowe	401
Modele łączenia systemów	404
Model referencyjny ISO/OSI — warstwy: fizyczna i łączenia danych, protokoły z rodziny Ethernet	407
Model referencyjny ISO/OSI — warstwa sieciowa, protokół IP, hermetyzacja	419
Model referencyjny ISO/OSI — warstwa transportowa, protokoły TCP i UDP, stos protokołów TCP/IP	432
Model referencyjny ISO/OSI — warstwy: sesji, prezentacji i aplikacji, protokoły warstw wyższych	439
Dodatek B Zapory sieciowe	451
Technologie zapór sieciowych	453
Filtrowanie pakietów (ang. packet filtering)	453
Usługi pośredniczenia (proxy)	458
Proxy warstwy aplikacji (ang. application-level proxies)	459
Proxy obwodowe (ang. circuit level gateway)	460
Translacja adresów sieciowych (ang. Network Address Translation — NAT)	461
Wirtualne sieci prywatne (ang. Virtual Private Network — VPN)	463
Architektury zapór sieciowych	467
Router ekranujący (ang. screening router)	467
Host dwusieciowy (ang. dual-homed host)	467
Host bastionowy (ang. bastion host)	468
Ekranowany host (ang. screened host)	469
Ekranowana podsieć (ang. screened subnet)	470

Host trzysięciowy (ang. tri-homed host)	471
Wiele ekranowanych podsieci (ang. split-screened subnet)	471
Dwa popularne filtry pakietów: Ipfiler i Iptables	472
Ipfiler	472
Iptables	484
Przygotowanie skryptu z regułami filtrowania	491
Konfiguracja systemu linux Redhat 9.0	493
Konfiguracja systemu Linux Knoppix	494
Dodatek C Systemy wykrywania intruzów IDS	497
Techniki wykrywania intruzów stosowane w systemach IDS	498
Sygnatury (dopasowywanie wzorców)	498
Badanie częstości zdarzeń i przekraczania ich limitów w określonej jednostce czasu ...	499
Wykrywanie anomalii statystycznych	499
Zaawansowane techniki detekcji intruzów	499
Budowa, działanie i umieszczanie systemu IDS w sieci	500
Budowa i działanie systemu	500
Umieszczanie systemu w sieci	501
Klasyfikacja systemów IDS	502
Budowa i zasada działania programu Snort	503
Zasada działania	503
Preprocesory	504
Możliwości wykrywania ataków oferowane przez SNORT-a	504
Zasady tworzenia reguł dla programu Snort	504
Podział ataków na klasy	509
Reakcja na ataki	511
Reakcja na typowy atak, wykrycie i zapis skanowania portów	511
Zdalne logowanie na użytkownika root	512
Statystyki oferowane przez Snorta	512
Konfiguracja systemu dynamicznie reagującego na włamania	516
Konfiguracja i uruchomienie Snorta	516
Sniffer	516
Logowanie pakietów	517
NIDS	518
Wykrywanie i dynamiczna reakcja	521
Podsumowanie	524
Dodatek D Instalowanie systemu Knoppix 3.4. na dysku twardym	525
Zakończenie	529
Bibliografia	531
Skorowidz	539

Rozdział 3.

Metody obrony przed atakami DoS i DDoS

Jednym z najbardziej niebezpiecznych, a zarazem popularnych, zagrożeń w sieciach komputerowych są ataki DoS (ang. *Denial of Service*, odmowa usług) w tym ich odmiana — ataki DDoS (ang. *Distributed Denial of Service*, rozproszone ataki „odmowa usługi”). Ich celem jest unieruchomienie atakowanego serwisu (na przykład serwera WWW, DNS czy całej atakowanej sieci) za pomocą różnych mechanizmów wykorzystujących zarówno luki konkretnych systemów operacyjnych, jak i niedociągnięcia stosu TCP/IP. Ataki DoS (pominąwszy pobudki czysto chuligańskie) mogą być podejmowane jako część szerszych działań mających na celu oszustwo, spenetrowanie cudzych zasobów bądź przejęcie nad nimi kontroli (na przykład zablokowanie prawdziwego DNS-a może być pomocne podczas prowadzenia ataku DNS-spoofing).

Obrona przed atakami DoS sprowadza się przede wszystkim do zainstalowania odpowiednich łąt na znane luki w systemach operacyjnych oraz do odfiltrowania na zaporce sieciowej pakietów niosących atak. Do wykrycia ataków DoS i DDoS najlepiej posłużyć się analizatorami ruchu sieciowego oraz systemami IDS. W tym rozdziale zaprezentujemy szereg najpopularniejszych ataków DoS i DDoS oraz sposoby obrony przed nimi. Jako zaporę sieciową wykorzystywaliśmy linuksowy program iptables, zaś jako systemu IDS używaliśmy Snorta (odpowiednie reguły zostały wzięte z aktualnych w chwili pisania książki baz Snorta). Opis instalacji i wykorzystania obu pakietów można znaleźć w dodatkach. Zachęcamy Czytelnika, aby na podstawie niniejszego rozdziału spróbował samodzielnie skonfigurować zaporę sieciową wraz z systemem detekcji intruzów i dynamicznej reakcji (za pomocą opisanego w dodatku C. programu Guardian).

Ataki DoS

Ataki typu DoS generują duży ruch, co powoduje zaburzenie lub całkowite zablokowanie pracy normalnych aplikacji sieciowych, lub też wykorzystują słabe punkty stosu protokołów TCP/IP dla unieruchomienia atakowanego systemu lub zaburzenia jego

działania. Niektóre z nich są możliwe do przeprowadzenia, ponieważ hosty sieciowe przy uwierzytelnianiu polegają tylko na źródłowym adresie IP. Inne istnieją, dlatego że niektóre mechanizmy kontrolne i większość protokołów routingu stosuje słabe metody uwierzytelniania źródła, z którego pochodzi informacja, bądź w ogóle ich nie używa.

Ataki DOS możemy podzielić na trzy grupy:

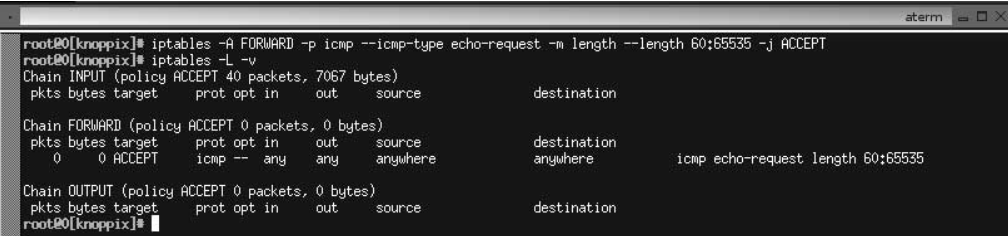
- ◆ ataki bazujące na implementacji stosu TCP/IP. Ataki tego typu wykorzystują słabości w specyfikacji TCP/IP w konkretnym systemie operacyjnym. Przykładami ataków z tej grupy jest *Ping of Death*, *Teardrop*, *Smbnuke*;
- ◆ ataki bazujące na standardach TCP/IP. Ataki tego typu wykorzystują słabości w samych standardach stosu TCP/IP. Przykładami ataków z tej grupy są *SYN attack* oraz *Land*;
- ◆ Ataki wykorzystujące tzw. „brutalną siłę” (ang. *brute force*). Ataki tego typu generują duży ruch, który zajmuje pasmo sieciowe. Przykładami ataków tego typu są *Smurf*, *Fraggle*.

Ping of Death

Ping of Death jest dość prostym atakiem. Standardy opisujące stos TCP/IP określają maksymalną wielkość datagramu IP na 65536 bajtów. Wiele systemów, szczególnie starsze wersje systemów uniksowych i linuksowych, może ulec uszkodzeniu, zawiesić się albo zrestartować, jeśli otrzyma datagram IP większy od możliwego maksymalnego rozmiaru. Podczas tego ataku tworzony jest i wysyłany do systemu ofiary pakiet ping przekraczający 65536 bajtów.

Zabezpieczenie nr 39. Ochrona przed atakiem Ping of Death za pomocą filtrowania na zaporze sieciowej

Jeżeli w sieci istnieją starsze wersje systemów operacyjnych, podatnych na atak Ping of Death, powinniśmy zadbać o ściągnięcie odpowiednich aktualizacji. Jeżeli z jakichś względów system taki nie może być uaktualniony do nowszej wersji, najlepszą metodą obrony jest zabezpieczenie dostępu do takiej stacji za pomocą reguły zapory sieciowej. Reguła ta może być zastosowana na routerze, przez który można uzyskać dostęp z zewnątrz do tej stacji. Regułę tej składni pokazano na rysunku 3.1.



```

root@0[knoppix]# iptables -A FORWARD -p icmp --icmp-type echo-request -m length --length 60:65535 -j ACCEPT
root@0[knoppix]# iptables -L -v
Chain INPUT (policy ACCEPT 40 packets, 7067 bytes)
 pkts bytes target    prot opt in      out     source                   destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in      out     source                   destination
  0    0 ACCEPT    icmp -- any      any     anywhere                 anywhere             icmp echo-request length 60:65535

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in      out     source                   destination
root@0[knoppix]#

```

Rysunek 3.1. Reguła zapory sieciowej dopuszczająca pakiety ICMP Echo request o wielkości od 60 do 65535 bajtów

Zabezpieczenie nr 40. Ochrona przed atakiem teardrop za pomocą systemu Snort

Najlepszą ochroną przeciwko temu atakowi jest sprawdzenie, czy dla naszego systemu operacyjnego istnieją aktualizacje przeciwdziałające atakowi i czy są one zainstalowane. Oczywiście można zastosować do ochrony odpowiednią regułę systemu IDS. Poniżej przedstawiono regułę wykrywającą pofragmentowane pakiety pochodzące z ataku teardrop:

```
alert udp any any -> any any (msg:"Przeprowadzono atak Teardrop"; fragbits:M; id:242; sid:270; rev:6;)
```

Wykrycie ataku przez Snorta, a następnie dynamiczna reakcja za pomocą zapory sieciowej pozwoli na zabezpieczenie się przed tego typu pakietami. Poszczególne opcje w regule przytoczonej wcześniej oznaczają, że alarm ma być ogłoszony w przypadku wykrycia pofragmentowanego (*fragbits:M;*) ruchu UDP z dowolnego adresu na dowolny adres (*alert udp any any -> any any*). Pola *Id* oraz *Sid* wskazują indeks reguły w bazie danych Snorta (ta reguła, jak i kolejne przedstawione poniżej, zostały zaczerpnięte z najnowszej w chwili pisania książki bazy programu Snort).

Zabezpieczenie nr 41. Ochrona przed atakiem teardrop za pomocą filtrowania pakietów

Drugą metodą ochrony przed atakiem Teardrop jest odpowiednia filtracja pakietów. W celu zabezpieczenia się przed tym atakiem należy odrzucić wszystkie przychodzące do naszej sieci lub komputera pofragmentowane datagramy UDP. Odpowiednią regułę zapory sieciowej zabezpieczającą przechodzenie przez bramę do internetu pofragmentowanych datagramów UDP zaprezentowano na rysunku 3.4.

Rysunek 3.4.
Reguła zapory sieciowej zabezpieczająca przed atakiem teardrop

```
root@0[knoppix]# iptables -A FORWARD -p udp -s 0/0 -f -j DROP
root@0[knoppix]# iptables -L -v
Chain INPUT (policy ACCEPT 3 packets, 447 bytes)
pkts bytes target      prot opt in      out     source         destination
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source         destination
 0    0 DROP        udp  -f  any    any     anywhere      anywhere
Chain OUTPUT (policy ACCEPT 5 packets, 376 bytes)
pkts bytes target      prot opt in      out     source         destination
root@0[knoppix]#
```

Atak SYN-flood

Atak *SYN-flood* wykorzystuje moment nawiązania połączenia TCP między dwoma hostami. Kiedy klient w sieci z implementowanym stosem TCP/IP chce nawiązać połączenie z serwerem, następuje tzw. *three-way-handshake*. Składa się on z trzech faz:

1. Klient wysyła segment tcp z flagą SYN do serwera.

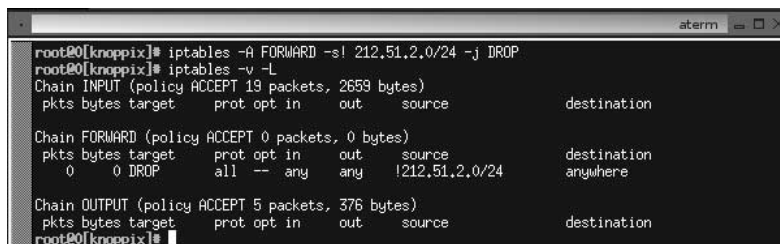
2. Serwer potwierdza odebranie segmentu SYN od klienta przez wysłanie segmentu z ustawionymi flagami SYN i ACK.
3. Klient odpowiada serwerowi segmentem z ustawioną flagą ACK.

Wykonanie tych trzech kroków powoduje ustanowienie połączenia i od tej chwili dane mogą być już wymieniane między nadawcą i odbiorcą. Każde wysłanie segmentu z flagą SYN na otwarty port u odbiorcy wymusza na nim odpowiedź przez odesłanie SYN i ACK i oczekiwanie na potwierdzenie od nadawcy flagą ACK. SYN-flood polega na zalewaniu systemu odbiorczego segmentami TCP z ustawioną flagą SYN spod fałszywych adresów IP. W tym przypadku ostateczna odpowiedź (ACK od klienta) nigdy nie nadejdzie, ponieważ odbiorca wysłał segment z flagami SYN i ACK pod fałszywy adres IP. Podczas gdy system odbiorcy czeka na potwierdzenie ACK, które nigdy nie przyjdzie, zapisuje wszystkie segmenty SYN i ACK, na które nie odpowiedział w swojej kolejce, zwykle dość małej. Po wypełnieniu tej kolejki system odbiorcy będzie ignorował wszystkie nowe próby nawiązania połączenia SYN. Segmenty z flagami SYN-ACK opuszczają kolejkę w momencie, gdy zostanie odebrane potwierdzenie ACK od nadawcy albo upłynie czas oczekiwania na to potwierdzenie — *timeout*. Przez ten czas system nie pozwoli na nawiązywanie nowych połączeń.

Zabezpieczenie nr 42. Ochrona przed atakiem SYN-flood wychodzącym z naszej sieci za pomocą zapory sieciowej

W celu przeciwdziałania tego typu atakom należy tak skonfigurować zaporę sieciową, aby pakiety opuszczające ją i wychodzące z naszej wewnętrznej sieci zawierały adres IP źródłowy pochodzący tylko z naszej wewnętrznej sieci. To spowoduje, że adresy IP nie będą mogły być *falszowane* (podmieniane). Aby zabezpieczyć się przed wychodzeniem tego typu pakietów z wnętrza naszej sieci, przy założeniu, że w naszej sieci są adresy publiczne z sieci 212.51.2.0/24, należy zastosować na routerze regułę zapory sieciowej przedstawioną na rysunku 3.5.

Rysunek 3.5.
Reguła blokująca wychodzenie z naszej sieci pakietów ze zmienionymi adresami źródłowymi



```

root@0[knoppix]# iptables -A FORWARD -s ! 212.51.2.0/24 -j DROP
root@0[knoppix]# iptables -v -L
Chain INPUT (policy ACCEPT 13 packets, 2653 bytes)
pkts bytes target      prot opt in      out     source         destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source         destination
 0    0 DROP        all  --  any     any     !212.51.2.0/24 anywhere

Chain OUTPUT (policy ACCEPT 5 packets, 376 bytes)
pkts bytes target      prot opt in      out     source         destination
root@0[knoppix]#

```

Znak (!) oznacza negację. W tym przypadku będą przepuszczane tylko pakiety z adresami źródłowymi pochodzącymi z sieci 212.51.2.0/24, pozostałe będą odrzucane.

W przypadku sieci lokalnej z prywatną pulą adresów np. 10.13.0.0/16, która jest ukryta za NAT-em, aby uchronić się przed atakiem SYN z wnętrza naszej sieci, należy dokonywać translacji tylko dla adresów z naszej sieci lokalnej. Sposób dokonywania translacji adresu źródłowego tylko dla adresów z naszej sieci przedstawiono na rysunku 3.6.

Rysunek 3.6.
*Translacja
 adresów tylko
 z prywatnej puli*

```

root@0[knoppix]# iptables -t nat -A POSTROUTING -o eth0 -s! 10.13.0.0/16 -j MASQUERADE
root@0[knoppix]# iptables -v -L -t nat
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target    prot opt in     out     source         destination
Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target    prot opt in     out     source         destination
  0      0 MASQUERADE all  --  any    eth0    110.13.0.0/16  anywhere
Chain OUTPUT (policy ACCEPT 1 packets, 69 bytes)
  pkts bytes target    prot opt in     out     source         destination
root@0[knoppix]#

```

W celu ograniczenia użytkownikom w sieci możliwość nawiązywania dużej liczby połączeń (segmentów tcp z ustawioną flagą SYN) możemy użyć reguły, która spowoduje, że każdy użytkownik o danym adresie IP będzie mógł nawiązać z wnętrza naszej sieci tylko 9 połączeń w ciągu 1s.

```
iptables -A INPUT -p tcp --syn -m iplimit --iplimit-above 9 -j DROP
```

Zabezpieczenie nr 43. Ochrona przed atakami SYN-flood i Naptha na usługi w naszej sieci za pomocą iptables

Jeżeli chcemy ochronić nasze serwisy dostępne z internetu przed atakami SYN-flood oraz Naptha, powinniśmy zastosować reguły filtrujące ograniczające liczbę możliwych połączeń w ciągu sekundy z naszymi serwerami. Na rysunku 3.7 pokazano regułę pozwalającą na wykonanie 3 połączeń w ciągu sekundy.

```

root@0[knoppix]# iptables -A INPUT -p tcp --syn -m limit --limit 3/s -j ACCEPT
root@0[knoppix]# iptables -L -v
Chain INPUT (policy ACCEPT 26 packets, 4006 bytes)
  pkts bytes target    prot opt in     out     source         destination
  0      0 ACCEPT    tcp  --  any    any    anywhere       anywhere       limit: avg 3/sec burst 5
  0      0 ACCEPT    tcp  --  any    any    anywhere       anywhere       tcp flags:SYN,RST,ACK/SYN limit: avg 3/sec burst 5
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target    prot opt in     out     source         destination
Chain OUTPUT (policy ACCEPT 8 packets, 582 bytes)
  pkts bytes target    prot opt in     out     source         destination
root@0[knoppix]#

```

Rysunek 3.7. *Ograniczenie liczby połączeń z usługami do 3 na sekundę*

Atak Land

Atak *Land* jest podobny do ataku SYN. Tak jak w przypadku ataku SYN, wysyłane są do systemu odbiorcy segmenty z ustawioną flagą synchronizacji, jednak, podczas gdy w ataku SYN adres IP nadawcy jest nieosiągalny lub jego komputer jest wyłączony, atak Land podmienia adres źródłowy na taki sam, jak adres odbiorcy. Serwer odbierając tak spreparowany pakiet wysyła potwierdzenie (pakiet z bitem ACK) na własny adres i nawiązuje nieaktywne połączenie — zapętla swoje działanie. Likwidacja nieaktywnego połączenia następuje dopiero po upływie czasu ustalonego w systemie operacyjnym serwera dla nieaktywnych połączeń.

Na rysunku 3.8 zaprezentowano sposób przeprowadzenia tego ataku.

Rysunek 3.8.

Przeprowadzenie ataku Land

```
root@0[dos]# ./land 172.16.30.1 22
land.e by m3lt, FLC
172.16.30.1:22 landed
root@0[dos]#
```

W wyniku przeprowadzenia ataku został wysłany charakterystyczny pakiet. Jak widać, adres źródłowy i adres docelowy wskazują na ten sam komputer (rysunek 3.9).

Rysunek 3.9.

Pakiet generowany podczas ataku Land

```
root@1[root]# tcpdump -i eth0 port 22
tcpdump: listening on eth0
22:06:31.263171 172.16.30.1.ssh > 172.16.30.1.ssh: S 3868:3868(0) win 2048
^
```

Zabezpieczenie nr 44. Ochrona przed atakiem Land za pomocą programu Snort

Chociaż atak tego typu nie jest stary, większość systemów operacyjnych jest wyposażona w łatwy zabezpieczający przed nim. Jak w przypadku wszystkich ataków DoS, należy pamiętać o instalowaniu zawsze odpowiednich łat bądź o sprawdzeniu, czy system nie jest już wyposażony w odpowiednie zabezpieczenie. Inną metodą obrony przed atakiem Land jest ustawienie odpowiednich filtrów na zaporze sieciowej. Mają one za zadanie odrzucenie wszystkich przychodzących datagramów IP z błędnymi adresami IP, czyli przychodzących z zewnątrz do naszej zapory sieciowej datagramów, których adres źródłowy wskazuje, że pochodzą z naszej wewnętrznej sieci lokalnej. Zabezpieczeniem będzie więc odpowiednia filtracja pakietów. Wśród znanych źródłowych adresów IP powinniśmy odfiltrowywać następujące:

- ♦ 10.0.0.0 to 10.255.255.255,
- ♦ 172.16.0.0 to 172.31.255.255,
- ♦ 192.168.0.0 to 192.168.255.255,

czyli adresy należące do sieci prywatnych oraz

- ♦ 127.0.0.0 to 127.255.255.255 (adresy pętli zwrotnej).

Przykładową regułą odrzucającą przychodzące z zewnątrz do naszej sieci pakiety z prywatnymi adresami z klasy 10.0.0.0/8 przedstawiono na rysunku 3.10.

Rysunek 3.10.

Reguła zapory sieciowej, która nie wpuszcza do wnętrza naszej sieci pakietów z adresami IP z prywatnej sieci 10.0.0.0/8

```
root@0[knoppix]# iptables -A FORWARD -s 10.0.0.0/24 -j DROP
root@0[knoppix]# iptables -L -v
Chain INPUT (policy ACCEPT 40 packets, 5754 bytes)
pkts bytes target      prot opt in      out     source         destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source         destination
 0      0 DROP        all  --  any     any     10.0.0.0/24    anywhere

Chain OUTPUT (policy ACCEPT 8 packets, 582 bytes)
pkts bytes target      prot opt in      out     source         destination
root@0[knoppix]#
```

Zabezpieczenie nr 45. Ochrona przed atakiem Land za pomocą programu Snort (reguła systemu IDS)

Aby chronić się przed atakiem Land, możemy też wykorzystać regułę systemu IDS. Regułę tę pokazano poniżej:

```
alert tcp any any -> any any (msg:"Przeprowadzono atak Land"; flags:S; id:3868; seq:3868; flow:stateless; sid:269; rev:9;)
```

Atak Naptha

Atak ten działa podobnie do ataku SYN-flood. Również wykorzystuje moment nawiązywania połączenia TCP między klientem i serwerem i, analogicznie do ataku SYN-flood, jego zadaniem jest zajęcie całej kolejki połączeń serwera i uniemożliwienie korzystania z danej usługi. Różnica w stosunku do zalewania segmentami SYN polega na tym, że atakujący wysyła pakiety SYN na określony port komputera ofiary i oczekuje na powracające pakiety SYN/ACK (w przypadku ataku SYN-flood atakujący wysyła tylko pakiety SYN i nic więcej nie robi). Jeśli atakujący zauważy pakiety powrotne SYN/ACK, wysyła do ofiary segment z ustawionymi flagami FIN/ACK. Powoduje to przestawienie serwera w stan tzw. pasywnego zamknięcia. Wynika on stąd, że serwer po odebraniu segmentu FIN/ACK odsyła pakiet FIN i czeka na pakiet ACK, którego nigdy nie otrzymuje. Serwer przechodzi w stan LAST_ACK i oczekuje na przyjęcie ostatniego segmentu ACK. W systemie Windows 98 stan ten utrzymywany jest przez około dwie minuty. W tym czasie serwer odrzuca wszelkie próby nawiązania z nim połączenia sieciowego (dopóty będzie ignorował wszystkie pakiety SYN, dopóki nie zwolni się miejsce w kolejce). Zalewając w ten sposób ofiarę sfałszowanymi segmentami SYN atakujący skutecznie blokuje możliwość korzystania z danej usługi.

Ograniczeniem tego ataku jest fakt, że napastnik musi — generując fałszywe pakiety SYN — podszywać się w nich pod adresy IP z własnej sieci. Jest to wymóg konieczny, aby powracające od zaatakowanej maszyny pakiety SYN/ACK mogły być zauważone przez napastnika i tym samym, by mógł on wysłać poprawnie zbudowane pakiety FIN/ACK. Napastnik musi w pakiecie SYN/ACK odczytać numer sekwencyjny i na jego podstawie zbudować poprawny dla ofiary pakiet FIN/ACK. Wysyłanie pakietów FIN/ACK z ustawionym niepoprawnym numerem sekwencyjnym zakończy się niepowodzeniem, ponieważ port ofiary stwierdzi, że pakiety takie nie należą do danego połączenia.

Na poniższym listingu pokazano sposób przeprowadzania ataku:

```
napastnik -> ofiara: SYN  
ofiara -> napastnik: SYN/ACK  
napastnik -> ofiara: ACK/FIN
```

Systemy podatne na ten atak to:

- ◆ Microsoft Windows 95,
- ◆ Microsoft Windows 98,

- ♦ Microsoft Windows 98SE,
- ♦ Microsoft Windows Millennium,
- ♦ Windows NT 4.0,
- ♦ HP-UX 11,
- ♦ IBM AIX 4.3,
- ♦ Sun Solaris 7-8,
- ♦ FreeBSD 4.0 wersja RELEASE,
- ♦ RedHat Linux 6.1 — 7.0,
- ♦ Systemy Linux oparte na jądrze z serii 2.0.

Na rysunku 3.11 przedstawiono stany połączeń na porcie 139. serwera po wykonaniu ataku SYN-flood w systemie Windows 98. Na kolejnym (rysunek 3.12) dla porównania widać stany połączeń, ale po wykonaniu ataku Naptha. Listę takich stanów uzyskano po wydaniu polecenia `netstat -na`.

Rysunek 3.11.
*Stany połączeń
serwera
po wykonaniu
ataku SYN-flood*

```

Tryb MS-DOS
Auto
TCP 172.16.30.137:139 172.16.30.150:53688 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:57787 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:16573 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:42690 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:61122 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:9924 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:5837 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:55245 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:720 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:20944 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:62932 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:7895 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:65501 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:31456 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:2020 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:56549 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:42470 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:56296 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:50664 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:80716 SYN_RECEIVED
TCP 172.16.30.137:139 172.16.30.150:5629 SYN_RECEIVED
UDP 172.16.30.137:137 *:*
UDP 172.16.30.137:138 *:*
C:\WINDOWS\Pulpit>

```

Rysunek 3.12.
*Stany połączeń
serwera
po wykonaniu
ataku Naptha*

```

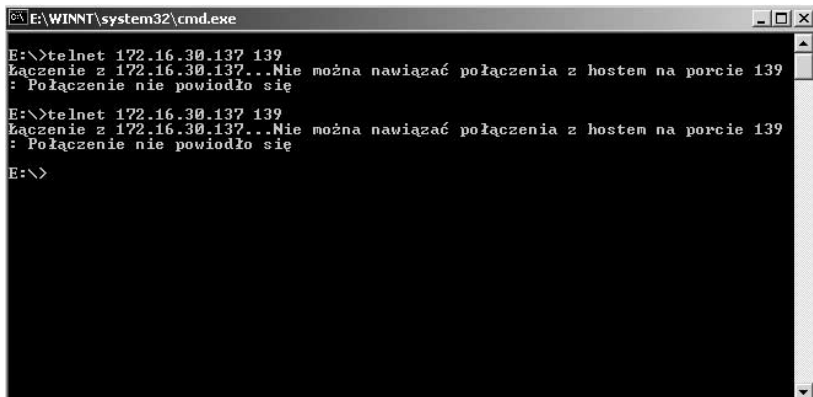
Tryb MS-DOS
Auto
TCP 172.16.30.137:139 172.16.30.150:36299 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:62411 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:16076 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:40909 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:53968 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:18386 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:49106 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:14292 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:9429 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:58586 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:1499 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:43742 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:3044 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:19428 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:49636 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:47079 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:50923 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:55593 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:39418 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:40188 LAST_ACK
TCP 172.16.30.137:139 172.16.30.150:56061 LAST_ACK
UDP 172.16.30.137:137 *:*
UDP 172.16.30.137:138 *:*
C:\WINDOWS\Pulpit>

```

Po wykonaniu ataku Naptha próba połączenia się z usługą na porcie 139. zakończy się niepowodzeniem (rysunek 3.13).

Rysunek 3.13.

*Próba
ustanowienia
połączenia
z portem 139
ofiary po
wykonaniu
ataku naptha*



```
E:\WINNT\system32\cmd.exe
E:\>telnet 172.16.30.137 139
Łączenie z 172.16.30.137...Nie można nawiązać połączenia z hostem na porcie 139
: Połączenie nie powiodło się

E:\>telnet 172.16.30.137 139
Łączenie z 172.16.30.137...Nie można nawiązać połączenia z hostem na porcie 139
: Połączenie nie powiodło się

E:\>
```