

## 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

# Wykrywanie włamań i aktywna ochrona danych

Autor: Alex Lukatsky

Tłumaczenie: Przemysław Szeremiota

ISBN: 83-7361-666-7

Tytuł oryginału: [Protect Your Information  
with Intrusion Detection](#)

Format: B5, stron: 512



### Poznaj sposoby wykrywania ataków hakerów

- Wybierz i skonfiguruj odpowiedni system detekcji włamań
- Dowiedz się, w jaki sposób hakerzy mogą zaatakować Twój system
- Zareaguj na atak hakera

Wykrywanie włamań to stosunkowo nowa dziedzina, którą zainteresowało się już wielu specjalistów od zabezpieczeń systemów komputerowych. Wokół niej, podobnie jak wokół innych rozwijających się dziedzin, krąży wiele mitów. Książki poświęcone ochronie przed włamaniami przedstawiają ten temat z innej perspektywy. Trudno jest również znaleźć zestawienie i porównanie narzędzi do detekcji włamań oraz omówienie procedury instalacji i skonfigurowania takiego narzędzia.

Głównym założeniem książki „Wykrywanie włamań i aktywna ochrona danych” jest wypełnienie tej luki. Książka przedstawia systemy detekcji włamań i sposoby korzystania z nich. Pokazuje również rodzaje i techniki ataków hakerskich oraz metody reagowania na włamanie i próby włamań.

- Anatomia ataku
- Zasady wykrywania włamań do sieci
- Wykrywanie śladów ataku
- Źródła informacji o atakach
- Czynniki warunkujące bezpieczeństwo systemu
- Narzędzia do wykrywania ataków
- Instalacja i konfiguracja systemu detekcji włamań oraz administrowanie nim
- Bezpieczeństwo systemu wykrywania ataków
- Reagowanie na zdarzenia zachodzące w sieci

**Zabezpiecz swoją sieć za pomocą nowoczesnych technik i narzędzi.**



# Spis treści

<b>Wstęp .....</b>	<b>9</b>
Docelowy krąg odbiorców .....	10
Przegląd treści .....	11
<b>Rozdział 1. Wykrywanie włamań — wprowadzenie .....</b>	<b>15</b>
Znaczenie systemów wykrywania włamań.....	15
Poziomy systemu informatycznego.....	19
Tradycyjne środki bezpieczeństwa.....	20
Wady tradycyjnych środków bezpieczeństwa .....	20
Omijanie zapór sieciowych.....	22
Podsumowanie .....	34
<b>Rozdział 2. Anatomia ataku .....</b>	<b>35</b>
Zdarzenia związane z bezpieczeństwem.....	35
Luki i podatność na włamanie.....	36
Klasyfikacja luk .....	36
Ataki.....	41
Nieformalny model ataku.....	43
Model ataku tradycyjnego.....	44
Model ataku rozproszonego .....	45
Skutki ataku .....	54
Etapy realizacji ataku .....	56
Narzędzia realizacji ataku .....	66
Klasyfikacja ataków.....	67
Bazy danych luk i ataków .....	68
Incydenty.....	70
Włamywacze .....	71
Cele włamywaczy .....	74
Podsumowanie .....	79
<b>Rozdział 3. Wprowadzenie do wykrywania włamań.....</b>	<b>81</b>
Wspomaganie zapór sieciowych .....	84
Kontrola dostępu do plików .....	84
Kontrolowanie nierzetelnych pracowników i zapobieganie wyciekom informacji .....	85
Ochrona antywirusowa.....	85
Kontrolowanie poczynań administratorów .....	86
Kontrolowanie dostępu do internetu.....	87
Wykrywanie nieznananych urządzeń.....	88

Analiza efektywności konfiguracji zapory sieciowej .....	88
Analiza przepływu informacji .....	89
Analiza danych urządzeń sieciowych .....	89
Zbieranie dowodów i reagowanie na incydenty .....	89
Inwentaryzacja zasobów i tworzenie mapy sieci .....	90
Wykrywanie konfiguracji domyślnych .....	91
Kontrolowanie efektywności działu informatycznego .....	92
Systemy wykrywania włamań a inne narzędzia zabezpieczające .....	92
Podsumowanie .....	93
<b>Rozdział 4. Trzy podstawowe zasady wykrywania włamań .....</b>	<b>95</b>
Przesłanki ataku .....	95
Powtarzające się zdarzenia .....	96
Monitorowanie liczby powtórzeń .....	97
Monitorowanie odstępu czasowego pomiędzy zdarzeniami .....	98
Niepoprawne polecenia .....	101
Próby wykorzystywania luk .....	103
Nieprawidłowe (nietypowe) parametry ruchu sieciowego .....	104
Źródła informacji o atakach .....	125
Technologie wykrywania włamań .....	130
Wykrywanie włamań — dwa podejścia .....	132
Podsumowanie .....	140
<b>Rozdział 5. Wykrywanie śladów ataku .....</b>	<b>141</b>
Kontrola spójności plików i katalogów .....	143
Dane o ważnych plikach i katalogach .....	147
Kontrola spójności plików i katalogów .....	149
Analiza plików dziennika .....	149
Analiza ruchu sieciowego .....	153
Analiza powiadomień .....	153
Analiza procesów, usług i portów .....	154
Wykrywanie nieautoryzowanej instalacji urządzeń .....	155
Regularny przegląd urządzeń .....	155
Monitorowanie modemów .....	156
Kontrolowanie dostępu do zasobów fizycznych .....	157
Analiza zewnętrznych źródeł informacji i analiza zachowania systemu .....	157
Podsumowanie .....	158
<b>Rozdział 6. Klasyfikacja systemów wykrywania włamań .....</b>	<b>159</b>
Systemy zapewniania bezpieczeństwa .....	161
Systemy wyszukujące słabości projektu .....	162
Systemy wykrywające wady fazy konfiguracji .....	167
Klasyczne systemy wykrywania włamań i programy analizujące pliki dzienników .....	176
Rys historyczny .....	176
Klasyfikacja systemów wykrywania włamań — wstęp .....	178
Architektura systemu wykrywania włamań .....	181
Stanowiskowe systemy wykrywania włamań .....	185
Sieciowe systemy wykrywania włamań .....	190
Narzędzia wykrywania ataków odmowy obsługi .....	199
Systemy wykrywania włamań a zapory sieciowe .....	201
Systemy zwodzące (wabiki) .....	202
Podsumowanie .....	210

<b>Rozdział 7. Uprowadzenie ataków czyli tworzenie infrastruktury wykrywania włamań .....</b>	<b>211</b>
Szkolenie personelu.....	214
Ośrodki szkoleniowe.....	216
Szkolenia on-line.....	217
Seminaria on-line.....	217
Seminaria i konferencje.....	217
Gry symulacyjne .....	218
Certyfikacja.....	219
Definiowanie procedur i zasad bezpieczeństwa.....	219
Wybór i stosowanie systemowych i sieciowych mechanizmów rejestrowania zdarzeń.....	223
Rejestrowane informacje.....	224
Użyteczność wbudowanych mechanizmów rejestrowania informacji o zdarzeniach .....	224
Rejestrowanie zdarzeń.....	226
Ochrona plików dzienników .....	226
Plan zarządzania plikami dzienników .....	228
Generowanie informacji dla systemów kontroli spójności .....	230
Mapa sieci .....	231
Archiwizacja ważnych plików i katalogów.....	233
Podsumowanie .....	234
<b>Rozdział 8. Cykl życia i etapy wdrożenia systemu wykrywania włamań .....</b>	<b>235</b>
Cykl życia projektu wdrożenia infrastruktury wykrywania włamań .....	236
Planowanie.....	236
Wybór producenta.....	237
Testowanie .....	237
Projekt pilotażowy .....	238
Instalacja .....	238
Obsługa i konserwacja .....	238
Uzasadnianie zakupu.....	238
Łączny koszt posiadania .....	240
Stopa zwrotu z inwestycji .....	243
Uruchamianie systemu wykrywania włamań.....	247
<b>Rozdział 9. Wybór systemu wykrywania włamań .....</b>	<b>249</b>
Analiza wstępna .....	250
Co chronić?.....	250
Przed czym chronić? .....	252
Przed kim chronić?.....	252
Jak chronić? .....	253
Jakie narzędzia angażować do ochrony? .....	255
Nabywcy systemów wykrywania włamań.....	255
Małe firmy.....	256
Duże firmy z filiami .....	256
Korporacje międzynarodowe .....	256
Dostawcy usług internetowych .....	257
Dostawcy usług informatycznych .....	258
Kryteria oceny systemów wykrywania włamań .....	258
Miejsce instalowania .....	259
Źródła informacji i metody analizy .....	260
Tryby przetwarzania.....	260
Architektura .....	260
Obsługiwane platformy .....	262

Wykrywanie włamań .....	262
Liczba wykrywanych ataków .....	262
Aktualizacje sygnatur .....	263
Definiowanie własnych zdarzeń .....	267
Reakcje na ataki .....	278
Zarządzanie czujnikami .....	286
Zarządzanie zdalne .....	286
Zarządzanie hierarchiczne .....	288
Zarządzanie zdarzeniami .....	289
Wydajność systemu wykrywania włamań .....	309
Instalacja i konfiguracja .....	310
Dostępność interfejsów i zestawów programistycznych .....	311
Wsparcie techniczne .....	311
Cena .....	313
Elastyczność .....	313
Inne kryteria .....	313
Testowanie .....	314
Podsumowanie .....	318
<b>Rozdział 10. Umiejscowienie systemu wykrywania włamań .....</b>	<b>319</b>
Rozmieszczenie czujników systemu wykrywania włamań .....	319
Czujnik sieciowy pomiędzy zaporą sieciową a routerem .....	320
Czujnik sieciowy w strefie zdemilitaryzowanej .....	321
Czujnik za zaporą sieciową .....	322
Czujniki w kluczowych segmentach chronionej sieci lokalnej .....	323
Czujnik w pobliżu serwera dostępu zdalnego .....	323
Czujnik w sieci szkieletowej .....	324
Czujniki w oddziałach regionalnych .....	325
Równoważenie obciążenia .....	325
Przypadki szczególne .....	328
Czujniki systemu wykrywania włamań w sieciach przełączanych .....	331
Wykorzystanie portu analizatora .....	333
Podłączenie dodatkowego koncentratora sieciowego .....	336
Zastosowanie rozdzielacza .....	337
Zastosowanie urządzeń równoważących obciążenie .....	340
Integrowanie systemów wykrywania włamań z przełącznikami sieciowymi .....	341
Umiejscowienie skanera bezpieczeństwa .....	342
Umiejscowienie systemów kontroli spójności .....	344
Umiejscowienie systemów zwodzących .....	344
Umiejscowienie konsol sterujących .....	345
Czynniki wpływające na konfigurację systemu wykrywania włamań .....	346
Obciążenie sieci i intensywność ruchu .....	347
Rodzaj transmisji w kontrolowanym segmencie .....	347
Średnie i szczytowe obciążenie procesora .....	348
Średni rozmiar pakietu w chronionym segmencie .....	348
Czas odpowiedzi .....	350
Obecność sieci z segmentami przełączanymi .....	350
Obecność routerów asymetrycznych .....	350
Skalowalność i rozszerzalność systemu .....	351
Zgodność z istniejącym sprzętem i oprogramowaniem .....	351
Szyfrowanie .....	351
Odległość pomiędzy czujnikami a konsolą sterującą .....	351
Zabezpieczenia na linii konsola-czujnik .....	352
Dynamiczny przydział adresów w kontrolowanym segmencie sieci (DHCP) .....	352
Współpraca z działem informatycznym .....	353

<b>Rozdział 11. Obsługa systemów wykrywania włamań.....</b>	<b>355</b>
Wybór węzła dla systemu wykrywania włamań.....	356
Wybór platformy.....	356
Węzeł dedykowany systemowi obsługi włamań.....	359
Zakup systemu wykrywania włamań.....	360
Zakup wymaganego sprzętu i oprogramowania.....	360
Zakup dokumentacji i usług wsparcia technicznego.....	367
Instalacja i rozruch systemu wykrywania włamań.....	369
Wyznaczanie bazy wiedzy dla (klasycznych) systemów wykrywania włamań.....	375
Wyznaczanie bazy wiedzy dla (klasycznych) skanerów bezpieczeństwa.....	378
Strategia skanowania.....	379
Taktyki skanowania.....	381
Konfiguracja mechanizmu rejestrowania zdarzeń i mechanizmów ostrzegawczych.....	384
Pliki dzienników.....	384
Powiadamianie pocztą elektroniczną i za pomocą SNMP.....	387
Zabezpieczanie systemu wykrywania włamań.....	387
Obwód rezerowy.....	388
Zabezpieczanie przed nieuprawnionym dostępem.....	389
Ograniczanie liczby użytkowników systemu wykrywania włamań.....	389
Kontrola dostępu do komponentów systemu wykrywania włamań.....	390
Określenie zasad bezpieczeństwa.....	391
Działanie w trybie utajonym.....	391
Automatyzacja.....	392
Możliwe problemy.....	394
Podsumowanie.....	396
<b>Rozdział 12. Problemy typowe dla systemów wykrywania włamań.....</b>	<b>397</b>
Problemy natury ogólnej.....	398
Aktualizacje bazy danych sygnatur.....	398
Heterogeniczność sieci.....	398
Ujednolicenie zarządzania bezpieczeństwem.....	399
Podatność systemów operacyjnych.....	400
Brak matematycznych podstaw technologii.....	401
Falszywe alarmy.....	401
Falszywe pominięcia.....	401
Rozwój narzędzi automatyzujących włamanie.....	402
Trudności wykrywania włamań.....	405
Ataki na systemy wykrywania włamań.....	406
Brak współpracy pomiędzy producentami.....	407
Nieustanny łańcuch przejęć i fuzji.....	407
Aktywne reagowanie.....	413
Przywrócenie systemu do stanu sprzed ataku.....	414
Co robić w obliczu ataku?.....	415
Testowanie systemów wykrywania włamań.....	416
Kompresja semantyczna.....	417
Brak mechanizmów gromadzenia dowodów sądowych.....	417
Ograniczenia sieciowych systemów wykrywania włamań.....	418
Sieci przełączane.....	418
Sieci z komunikacją szyfrowaną.....	418
Modemy.....	419
Niedostatek zasobów.....	419
Prowokowanie awarii systemów wykrywania włamań.....	421
Oszustwa proste.....	423
Oszustwa wyrafinowane.....	425

---

Ograniczenia stanowiskowych systemów wykrywania włamań.....	426
Rozmiar pliku dziennika .....	426
Interwał archiwizacji dzienników.....	426
Ograniczona wydajność .....	426
Ochrona plików dzienników .....	427
Rodzaj i poziom szczegółowości rejestrowanych informacji.....	427
Brak jednolitego formatu zapisu danych .....	427
Podsumowanie .....	428
<b>Rozdział 13. Standaryzacja w dziedzinie wykrywania włamań .....</b>	<b>429</b>
Adaptive Network Security Alliance.....	429
Projekt Lincoln Laboratory .....	430
Projekt OSEC.....	431
Intrusion Detection Systems Consortium .....	431
OPSEC .....	432
Common Content Inspection.....	432
Common Intrusion Detection Framework .....	432
Intrusion Detection Working Group.....	433
Baza danych CVE .....	433
Baza danych ICAT .....	434
Projekty agencji DARPA .....	435
<b>Rozdział 14. Reagowanie na incydenty.....</b>	<b>437</b>
Nieuprawnione zmiany w systemie.....	441
Dokumentowanie wszystkich nieudanych ataków .....	442
Podsumowanie .....	442
<b>Dodatek A Porty okupowane przez konie trojańskie .....</b>	<b>443</b>
<b>Dodatek B Najczęściej skanowane porty .....</b>	<b>469</b>
<b>Dodatek C Zakresy adresów IP w internecie.....</b>	<b>473</b>
<b>Dodatek D Domeny najwyższego poziomu .....</b>	<b>475</b>
<b>Dodatek E Identyfikatory protokołów (IPv4) .....</b>	<b>483</b>
<b>Dodatek F Bibliografia .....</b>	<b>487</b>
<b>Skorowidz.....</b>	<b>497</b>

## Rozdział 4.

# Trzy podstawowe zasady wykrywania włamań

*Niewyciężoność leży w naszej gestii, a podatność na przegraną w gestii przeciwnika. A zatem biegły w sztuce wojennej może uczynić siebie niewyciężonym, ale nie może spowodować, aby wróg stał się podatny na porażkę*

— Sun Tzu, *Sztuka Wojny*

Aby wdrażana w systemie technologia wykrywania włamań była efektywna, należy udzielić odpowiedzi na następujące trzy pytania:

- ◆ **Co wykrywać?** Należy umieć rozpoznać przesłanki wskazujące na naruszenie reguł bezpieczeństwa.
- ◆ **Gdzie wykrywać?** Należy umieć wskazać źródła informacji, w których można szukać przesłanek co do naruszenia reguł bezpieczeństwa.
- ◆ **Jak wykrywać?** Należy poznać metody analizy informacji pozyskanych ze wspomnianych źródeł celem wykrycia na ich podstawie ewentualnego faktu naruszenia reguł bezpieczeństwa.

## Przesłanki ataku

Aby wykryć fakt naruszenia przyjętych w danym systemie reguł bezpieczeństwa (na podstawie analizy ruchu sieciowego albo plików dziennika), należy uprzednio nabyć wiedzę o sposobie identyfikowania takich faktów i odróżniania ich od normalnych zdarzeń związanych z bezpieczeństwem. Przesłankami ataku mogą być następujące zdarzenia [Edward1-99]:

- ◆ powtarzające się konkretne zdarzenia;
- ◆ niepoprawne polecenia albo polecenia niepasujące do bieżącego stanu systemu;
- ◆ próby wykorzystywania luk;

- ◆ nieprawidłowe (nietypowe) parametry ruchu sieciowego;
- ◆ niespodziewane atrybuty poleceń;
- ◆ niewytłumaczalne problemy;
- ◆ dodatkowe informacje o naruszeniach bezpieczeństwa.

Tradycyjne narzędzia zabezpieczające (zapora sieciowa, serwer uwierzytelniania, system kontroli dostępu itd.) opierają swoje działanie na obserwacji jednego lub najwyżej dwóch zdarzeń z powyższej listy; systemy wykrywania włamań (choć zależy to od ich implementacji) mogą brać pod uwagę wszystkie wymienione przesłanki.

Niniejszy rozdział powinien pomóc Czytelnikowi zrozumieć sposoby podejmowania decyzji implementowane we współczesnych systemach wykrywania włamań. Programistom mógłby zaś pomóc w tworzeniu własnych systemów wykrywania włamań i nieautoryzowanej działalności. Dotyczy to zwłaszcza tych programistów, którzy zajmują się tworzeniem aplikacji finansowych i zautomatyzowanych systemów dla potrzeb komunikacji. Opisy przesłanek nieuprawnionej działalności mogą okazać się przydatne również tym, którzy nie mają do dyspozycji zautomatyzowanych systemów wykrywania włamań i skazani są na samodzielną analizę stanu systemu.

## Powtarzające się zdarzenia

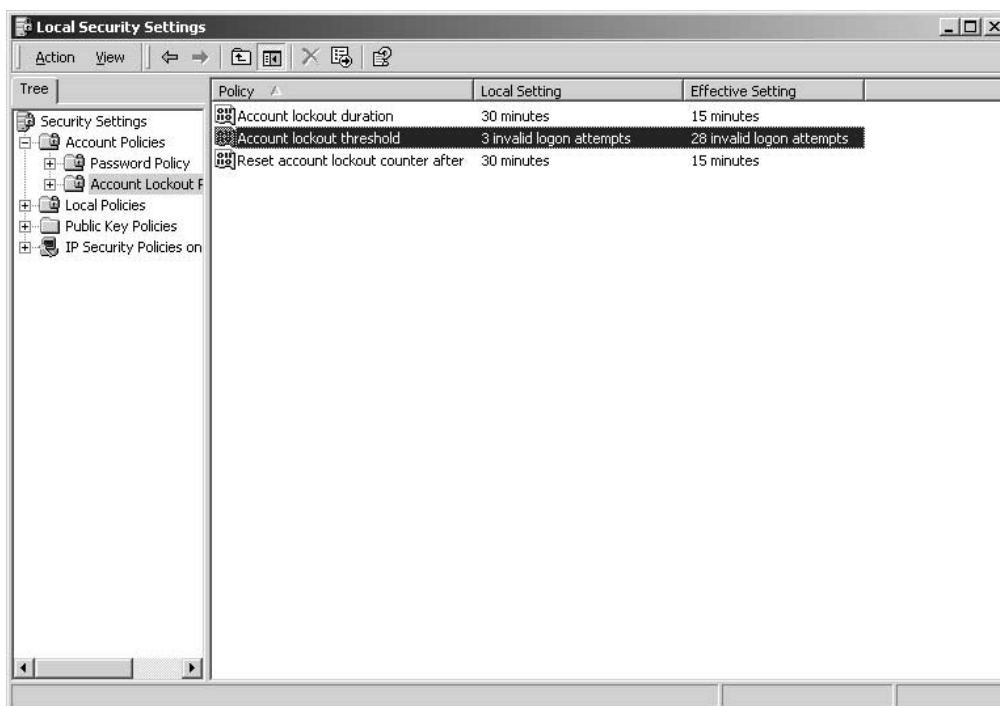
Rozpoznanie konkretnych czynności i zdarzeń, które zachodzą w systemie w sposób powtarzalny, to jedna z najlepszych metod wykrywania włamań. Mechanizm ten bazuje na założeniu, że w przypadku niepowodzenia pierwszej próby ataku intruz powtarza próbę odwołania się do pożądanego zasobu. Za taką próbę można uznać również skanowanie portów w poszukiwaniu dostępnych usług sieciowych i próby odgadnięcia hasła. Algorytmy wykrywania takich prób muszą rozpoznawać fakt powtarzalności zdarzenia i decydować, od jakiej liczby prób można mówić o ataku. Wypada jednak zaznaczyć, że jeśli intruz wie dokładnie, w jaki sposób powinien odwołać się do zasobu (albo, na przykład, zdoła przechwycić nazwę kont i hasło użytkownika uprawnionego) i nie popełni przy realizacji procedury żadnego błędu, jego próba uzyskania nieuprawnionego dostępu będzie praktycznie niemożliwa do wykrycia. Jeśli włamywacz zdoła dla własnych potrzeb utworzyć dokładny, działający model atakowanego celu i nabeździe nieco wprawy w imitowaniu czynności użytkowników autoryzowanych, jego poczynania również zostaną najprawdopodobniej niezauważone. Jednak utworzenie modelu atakowanego celu to operacja niezwykle kosztowna i znakomitej większości włamywaczy (tych działających na własną rękę) zwyczajnie nie stać na jej przeprowadzenie.

Wykrywanie powtarzających się zdarzeń to metoda efektywna, ponieważ pozwala na wykrycie również tych ataków, co do których nie ma żadnych dodatkowych informacji (czyli również na wykrywanie nieznanych dotąd ataków). Jako kryterium zaliczenia obserwowanych prób do ataku można uwzględnić dwa czynniki:

- ◆ Liczbę powtórzeń zdarzenia przekraczającą pewną założoną wartość ustaloną jako normę (jak w przypadku liczby prób logowania się do sieci).
- ◆ Okres, w którym obserwuje się powtarzalne zdarzenia.

## Monitorowanie liczby powtórzeń

W tym przypadku należy kontrolować liczbę powtórzeń zdarzenia i porównywać ją z pewnym przyjętym dla klasy podobnych zdarzeń progiem; przekroczenie tego progu odróżnia zdarzenia autoryzowane od nieautoryzowanych. W przypadku tych drugich może chodzić o zwykłe błędy, niekoniecznie o atak. Tak czy inaczej system wykryje każdy przypadek przekroczenia dozwolonej liczby powtórzeń. Dla różnych składników systemu i różnych kategorii zdarzeń wartości progowe należy dobrać doświadczalnie. Typowym przykładem kontrolowania dozwolonej liczby powtórzeń dla pewnej klasy zdarzeń jest ograniczenie liczby prób logowania użytkownika do sieci. Sposób ustawienia takiej wartości progowej dla liczby prób logowania w systemie Windows 2000 pokazuje rysunek 4.1.



**Rysunek 4.1.** Określanie maksymalnej liczby dozwolonych prób logowania w systemie Windows 2000

Trzeba pamiętać, że niewłaściwy dobór wartości progowej może dać efekt w postaci rejestrowania znacznej liczby fałszywych alarmów, albo odwrotnie — przeoczenia pewnej liczby faktycznych prób ataku. Jeśli wartość progowa jest zbyt niska, system wykrywania włamań będzie stosunkowo często stwierdzał zaistnienie ataku, również w przypadkach zupełnie typowych. Kiedy wartość progowa będzie zbyt wysoka, pewna liczba ataków pozostanie niewykryta.



### Atak na sieć Banku Westminsterkiego

Kasjer Banku Westminsterkiego nawiązał połączenie dial-up z siecią banku ze swojego komputera domowego. Następnie przeprowadził 1 200 transakcji, w ramach każdej z nich przelewając 10 funtów na swoje prywatne konto. Przelewy pozostały niewykryte, ponieważ tak niewielkie kwoty nie podlegały kontroli. Poczynania nieuczciwego kasjera zostały zauważone dopiero, kiedy ten spróbował przelać większą kwotę (984 funtów) na konto swojego przyjaciela. Bank ukarał kasjera i zażądał rekompensaty w kwocie 15 tysięcy funtów. Sąd odrzucił jednak roszczenia banku, obciążając bank za niedopełnienie obowiązku ochrony swoich zasobów.

## Monitorowanie odstępu czasowego pomiędzy zdarzeniami

Typowym przykładem zastosowania metody polegającej na pomiarze upływu czasu pomiędzy zdarzeniami jest wykrywanie ataku skanowania portów (patrz listing 4.1), czyli wykonywanych w zwykle krótkim przedziale czasu prób nawiązania połączenia z poszczególnymi portami węzła. Od tego momentu wszystkie elementy listingów, które zasługują na szczególną uwagę, będą wyróżniane pogrubieniem; większość listingów zamieszczanych tu w roli przykładów zaczerpniętych zostało albo z witryny <http://www.incidents.org/logs/>, albo wziętych z praktycznych ćwiczeń prowadzonych w ramach przygotowywania do egzaminu Global Information Assurance Certification Certified Intrusion Analyst ([www.giac.org/GCIA.php](http://www.giac.org/GCIA.php)). Wracając do skanowania portów, może być ono przeprowadzone za pośrednictwem rozmaitych programów różniących się implementacją mechanizmu skanowania. Najprostsze z nich, jak Haktek, po prostu sondują wszystkie porty z zakresu zadanego numerem początkowym i końcowym (listing 4.1).

### Listing 4.1. Skanowanie portów za pomocą programu Haktek (fragment pliku dziennika TCPdump)

```
17:17:21.966870 WS_LUKICH.2876 > WS_LUKA.1: S 713310:713310(0) win 8192 <mss 1460>
(DF)
17:17:21.967698 WS_LUKICH.2877 > WS_LUKA.2: S 713329:713329(0) win 8192 <mss 1460>
(DF)
17:17:21.968612 WS_LUKICH.2878 > WS_LUKA.3: S 713349:713349(0) win 8192 <mss 1460>
(DF)
17:17:21.969095 WS_LUKICH.2879 > WS_LUKA.4: S 713364:713364(0) win 8192 <mss 1460>
(DF)
17:17:21.969574 WS_LUKICH.2880 > WS_LUKA.5: S 713372:713372(0) win 8192 <mss 1460>
(DF)
17:17:21.970041 WS_LUKICH.2881 > WS_LUKA.6: S 713381:713381(0) win 8192 <mss 1460>
(DF)
17:17:21.970523 WS_LUKICH.2882 > WS_LUKA.7: S 713391:713391(0) win 8192 <mss 1460>
(DF)
17:17:21.971031 WS_LUKICH.2883 > WS_LUKA.8: S 713402:713402(0) win 8192 <mss 1460>
(DF)
17:17:21.971539 WS_LUKICH.2884 > WS_LUKA.9: S 713414:713414(0) win 8192 <mss 1460>
(DF)
17:17:21.972014 WS_LUKICH.2885 > WS_LUKA.10: S 713427:713427(0) win 8192 <mss 1460>
(DF)
```

```
17:17:21.973780 WS_LUKICH.2886 > WS_LUKA.11: S 713441:713441(0) win 8192 <mss 1460>
(DF)
17:17:21.973814 WS_LUKICH.2887 > WS_LUKA.12: S 713445:713445(0) win 8192 <mss 1460>
(DF)
17:17:21.973834 WS_LUKICH.2888 > WS_LUKA.13: S 713469:713469(0) win 8192 <mss 1460>
(DF)
```

---

Warto pamiętać, że również niepoprawne dobranie progowej częstotliwości zdarzenia może prowadzić do generowania przez system wykrywania włamań fałszywych alarmów albo pomijania faktycznych ataków.

Rozpoznanie ataku polegającego na skanowaniu portów jest stosunkowo proste. Jednak tak ograniczone ataki jak wykorzystujące skanery portów przeprowadzane są raczej rzadko i tylko przez użytkowników nie dość wykwalifikowanych, nieznających bardziej zaawansowanych narzędzi. Znakomitym przykładem takiego zaawansowanego narzędzia jest Nmap (<http://www.nmap.org>), produkt dostępny w wersjach dla większości odmian Uniksa i Windows. Program ten implementuje znaczną liczbę trybów skanowania — w wersji 3.00 obsługuje 10 różnych rodzajów skanowania (patrz listingi 4.2 do 4.4).

---

**Listing 4.2.** Skanowanie portów programem Nmap z opcją *-sT* (fragment pliku dziennika TCPdump)

---

```
17:26:48.031721 WS_LUKA.2797 > WS_LUKICH.371: S 26274004:26274004(0) win 8192 <mss
1460> (DF)
17:26:48.034553 WS_LUKA.2798 > WS_LUKICH.344: S 26330728:26330728(0) win 8192 <mss
1460> (DF)
17:26:48.035510 WS_LUKA.2799 > WS_LUKICH.919: S 26396113:26396113(0) win 8192 <mss
1460> (DF)
17:26:48.036466 WS_LUKA.2800 > WS_LUKICH.1155: S 26439772:26439772(0) win 8192 <mss
1460> (DF)
17:26:48.037421 WS_LUKA.2801 > WS_LUKICH.117: S 26476417:26476417(0) win 8192 <mss
1460> (DF)
17:26:48.038372 WS_LUKA.2802 > WS_LUKICH.625: S 26518671:26518671(0) win 8192 <mss
1460> (DF)
17:26:48.039338 WS_LUKA.2803 > WS_LUKICH.220: S 26552575:26552575(0) win 8192 <mss
1460> (DF)
17:26:48.040281 WS_LUKA.2804 > WS_LUKICH.770: S 26588454:26588454(0) win 8192 <mss
1460> (DF)
17:26:48.041235 WS_LUKA.2805 > WS_LUKICH.619: S 26633584:26633584(0) win 8192 <mss
1460> (DF)
17:26:48.042170 WS_LUKA.2806 > WS_LUKICH.1652: S 26670889:26670889(0) win 8192 <mss
1460> (DF)
17:26:48.043264 WS_LUKA.2807 > WS_LUKICH.403: S 26734852:26734852(0) win 8192 <mss
1460> (DF)
```

---

---

**Listing 4.3.** Skanowanie portów programem Nmap z opcją *-sS* (fragment pliku dziennika TCPdump)

---

```
17:22:32.224567 WS_LUKA.52735 > WS_LUKICH.1544: S 866284386:866284386(0) win 1024
17:22:32.225413 WS_LUKA.52735 > WS_LUKICH.427: S 866284386:866284386(0) win 1024
17:22:32.225413 WS_LUKA.52735 > WS_LUKICH.447: S 866284386:866284386(0) win 1024
17:22:32.224845 WS_LUKA.52735 > WS_LUKICH.496: S 866284386:866284386(0) win 1024
17:22:32.225009 WS_LUKA.52735 > WS_LUKICH.597: S 866284386:866284386(0) win 1024
17:22:32.225207 WS_LUKA.52735 > WS_LUKICH.659: S 866284386:866284386(0) win 1024
17:22:32.225413 WS_LUKA.52735 > WS_LUKICH.159: S 866284386:866284386(0) win 1024
```

```

17:22.32.225582 WS_LUKA.52735 > WS_LUKICH.529: S 866284386:866284386(0) win 1024
17:22.32.225782 WS_LUKA.52735 > WS_LUKICH.2017: S 866284386:866284386(0) win 1024
17:22.32.225913 WS_LUKA.52735 > WS_LUKICH.427: S 866284386:866284386(0) win 1024
17:22.32.225945 WS_LUKA.52735 > WS_LUKICH.1380: S 866284386:866284386(0) win 1024
17:22.32.226153 WS_LUKA.52735 > WS_LUKICH.1522: S 866284386:866284386(0) win 1024
17:22.32.226356 WS_LUKA.52735 > WS_LUKICH.1109: S 866284386:866284386(0) win 1024
17:22.32.131078 WS_LUKA.52735 > WS_LUKICH.306: S 866284386:866284386(0) win 1024
17:22.32.233200 WS_LUKA.52735 > WS_LUKICH.274: S 866284386:866284386(0) win 1024
17:22.32.235413 WS_LUKA.52735 > WS_LUKICH.447: S 866284386:866284386(0) win 1024
17:22.32.235925 WS_LUKA.52735 > WS_LUKICH.1663: S 866284386:866284386(0) win 1024

```

---

**Listing 4.4.** Skanowanie portów programem Nmap z opcją `-sU` (fragment pliku dziennika TCPdump)

---

```

17:30:03.034865 WS_LUKA.48796 > WS_LUKICH.670: udp 0
17:30:03.035066 WS_LUKA.48796 > WS_LUKICH.1248: udp 0
17:30:03.035269 WS_LUKA.48796 > WS_LUKICH.25: S udp 0
17:30:03.035448 WS_LUKA.48796 > WS_LUKICH.1017: S udp 0
17:30:03.035653 WS_LUKA.48796 > WS_LUKICH.1415: S udp 0
17:30:03.035815 WS_LUKA.48796 > WS_LUKICH.963: S udp 0
17:30:03.036006 WS_LUKA.48796 > WS_LUKICH.11: S udp 0
17:30:03.036160 WS_LUKA.48796 > WS_LUKICH.345: S udp 0

```

---

Wszystkie trzy fragmenty pliku dziennika programu TCPdump są ciekawe, ponieważ numery skanowanych portów nie są zwiększane o jeden (jak w przypadku zwykłego skanowania), ale dobierane w sposób losowy. Aby jeszcze bardziej utrudnić wykrycie skanowania, numery raz rosną, a raz maleją. Dodatkowo, aby ukryć fakt realizacji typowego skanowania, program Nmap niekiedy powtórnie sonduje te same porty (na listingach były to porty 427 i 447).

Poza klasycznymi atakami skanowania, łatwo wykrywanymi przez zapory sieciowe (patrz listing 4.5), atakujący stosują bardziej wyrafinowane metody wyszukiwania luk, których nie sposób wykryć i rozpoznać za pośrednictwem tradycyjnych mechanizmów zabezpieczających. Mowa między innymi o skanowaniu utajonym, w którym wykorzystywane są specyficzne cechy sposobu przetwarzania pakietów sieciowych, które nie spełniają wymogów standardu TCP/IP. Te metody skanowania zostaną omówione nieco później.

---

**Listing 4.5.** Wykrywanie skanowania portów (fragment pliku dziennika zapory sieciowej CheckPoint Firewall-1)

---

```

"421316" "29Dec2000" " 9:32:16" "daemon" "localhost" "alert" "accept"
"" "x.x.x.x" "x.x.x.x" "ip" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
"MAD" "additional: attack=blocked_connection_port_scanning"
"422255" "29Dec2000" " 9:33:59" "daemon" "localhost" "alert" "accept"
"" "x.x.x.x" "x.x.x.x" "ip" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
"MAD" "additional: attack=blocked_connection_port_scanning"
"427220" "29Dec2000" " 9:43:26" "daemon" "localhost" "alert" "accept"
"" "x.x.x.x" "x.x.x.x" "ip" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
"MAD" "additional: attack=blocked_connection_port_scanning"

```

---

Kolejną zaawansowaną metodą skanowania, utrudniającą jego wykrycie, jest zwiększenie odstępu czasowego pomiędzy kolejnymi próbami sondowania portów czy węzłów. Zwykle skanery portów działają z prędkością 5 do 10 portów na sekundę. Gdyby

zmienić domyślne opóźnienia czasowe (jak w trybie `-sI` skanera Nmap w wersji 3.00 albo trybie `-T` w wersjach 2.xx), większość systemów wykrywania włamań mogłaby nie zarejestrować podejrzanej działalności, interpretując ją jako mieszczącą się w granicach normy (listing 4.6).

**Listing 4.6.** Sondowanie węzłów (fragment pliku dziennika TCPdump)

```
12:01:38:234455 200.0.0.200 > 200.0.0.67: icmp: echo request
12:03:51:543524 200.0.0.200 > 200.0.0.87: icmp: echo request
12:05:04:655342 200.0.0.200 > 200.0.0.134: icmp: echo request
12:07:18:573256 200.0.0.200 > 200.0.0.23: icmp: echo request
12:09:31:676899 200.0.0.200 > 200.0.0.11: icmp: echo request
12:11:44:896754 200.0.0.200 > 200.0.0.104: icmp: echo request
12:13:57:075356 200.0.0.200 > 200.0.0.2: icmp: echo request
```

Monitorowanie odstępów czasowych pomiędzy zdarzeniami może być wykorzystywane również do wykrywania ataków odmowy obsługi. Na przykład należący do tej kategorii atak Smurf charakteryzuje się zastosowaniem pakietów rozgłoszeniowych transmitowanych z dość długim interwałem. Niekiedy pakiety takie były wysyłane przez kilka dni, co rzecz jasna musiało doprowadzić do załamania atakowanej sieci niezdolnej już do przetwarzania pakietów uprawnionych. Poniżej (na listingu 4.7) prezentowany jest fragment pliku dziennika routera Cisco, który zarejestrował ataki Smurf i Fraggle (Fraggle jest podobny do ataku Smurf, tyle że wykorzystuje protokół UDP). W praktyce liczba wpisów pliku dziennika zarejestrowanych podczas ataku znacznie przekraczała by setkę.

**Listing 4.7.** Zarejestrowane ataki Smurf i Fraggle (fragment pliku dziennika routera Cisco)

```
Dec 22 16:15:26: %SEC-6-IPACCESSLOGDP: list Internet denied icmp 172.20.20.1 ->
200.0.0.255 (8/0), 1 packet
Dec 22 16:16:26: %SEC-6-IPACCESSLOGDP: list Internet denied icmp 172.20.20.1 ->
200.0.0.255 (8/0), 24 packets
Dec 22 16:16:56: %SEC-6-IPACCESSLOGDP: list Internet denied udp 172.20.20.1 ->
200.0.0.255 (8/0), 1 packet
Dec 22 16:26:26: %SEC-6-IPACCESSLOGDP: list Internet denied icmp 172.20.20.1 ->
200.0.0.255 (8/0), 3 packets
Dec 22 16:27:26: %SEC-6-IPACCESSLOGDP: list Internet denied icmp 172.20.20.1 ->
200.0.0.255 (8/0), 4 packets
```

## Niepoprawne polecenia

Kolejna metoda wykrywania nieuprawnionych działań polega na wykrywaniu i analizie niepoprawnych poleceń, na przykład niepoprawnych żądań kierowanych do serwerów czy niepoprawnych odpowiedzi na żądania. Niepoprawność odpowiedzi w komunikacji pozwala wysnuć wniosek, że jedna ze stron wymiany informacji — albo żądający informacji, albo zwracający niespodziewaną odpowiedź — została podmieniona. Nie sposób byłoby zaprezentować wszystkie możliwe przypadki ataków polegających na wydaniu niepoprawnego polecenia czy udzielenia niepoprawnej odpowiedzi w najgrubszej nawet książce. W ramach przykładu możemy rozważyć podatny na ataki skrypt CGI serwera WWW (jak okazało się w rozdziale 2., dziurawe skrypty CGI to jedna

z najpowszechniejszych luk w zabezpieczeniach). W przykładzie prezentowanym poniżej, analizując wpisy pliku dziennika serwera (ręcznie albo za pośrednictwem systemu wykrywania włamań), można wykryć dwa odwołania do szeregu nieistniejących skryptów CGI (listing 4.8 i listing 4.9).

**Listing 4.8.** Wykrywanie skanowania w poszukiwaniu podatnych skryptów CGI (fragment pliku dziennika serwera WWW)

```
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/aglimpse
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/scripts/iisadmin/bdir.htr
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-dos/args.bat
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/AnyForm2
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/campas
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/Count.cgi
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/carbo.d11
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/finger
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/faxsurvey
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/htmlscript
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/handler
[Mon Dec 27 01:42:58 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/man.sh
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/jj
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/pfdisplay.cgi
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/nph-test.cgi
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/php.cgi
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/phf
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/search97.vts
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/test.cgi
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-win/uploader.exe
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/textcounter.pl
[Mon Dec 27 01:42:59 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/view-source
[Mon Dec 27 01:43:00 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/webdist.cgi
[Mon Dec 27 01:43:00 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/websendmail
```

```
[Mon Dec 27 01:43:00 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/webgais
[Mon Dec 27 01:43:00 1999] [error] [client 172.20.20.1] File does not exist:
/web/home/www/www_home/cgi-bin/www-sql
```

**Listing 4.9.** Wykrywanie odwołań do podatnych na ataki skryptów CGI (jak test-cgi) i Aglimpse (fragment pliku dziennika programu Snort)

```
22:00:08.952175 200.0.0.104:53558 > 200.0.0.110:80: P 1677621322:1677621391(69) ack
2335601879 win 8760 (DF) (ttl 242, id 12223)
0000: 4500 006d 2fbf 4000 f206 0465 80af 0d4a E..m/..@....e...J
0010: 0a00 0009 d136 0050 63fe 784a 8b36 74d7 .d...6.Pc.xJ.6t.
0020: 5018 2238 4af8 0000 504f 5354 202f 6367 P."8J...POST /cg
0030: 692d 6269 6e2f 7465 7374 2d63 6769 2048 i-bin/test-cgi H
0040: 5454 502f 312e 300a 436f 6e74 656e 742d TTP/1.0.Content-
0050: 7479 7065 3a20 2a0a 436f 6e74 656e 742d type: *.Content-
0060: 6c65 6e67 7468 3a20 300a 0a00 19 length: 0....

01:14:18.0427222 200.0.0.104.42930 > 200.0.0.110.80 P
3053993825:3053993920(95) ack 2009011357 win 8760 (DF) (ttl 242, id 57632)
0000: 4500 0087 e120 4000 f206 52e9 80af 0d4a E....@...R....J
0010: 0a00 0009 a7b2 0050 b608 3f61 77bf 149d .d....P..?aw...
0020: 5018 2238 8704 0000 4745 5420 2f63 6769 P."8...GET /cgi
0030: 2d62 696e 2f61 676c 696d 7073 652f 3830 -bin/aglimpse/80
0040: 7c49 4653 3d5f 3b43 4d44 3d5f 6563 686f /IFS=;CMD=_echo
0050: 5c3b 6563 686f 5f69 642d 6167 6c69 6d70 \;echo_id-aglimp
0060: 7365 5c3b 756e 616d 655f 2d61 5c3b 6964 se\;uname_-a\;id
0070: 3b65 7661 6c24 434d 443b 2048 5454 502f ;eval$CMD; HTTP/
0080: 312e 300a 0a00 20 1.0...
```

## Próby wykorzystywania luk

Zgodnie ze znanymi już Czytelnikowi klasyfikacjami i definicjami luk w zabezpieczeniach oraz definicją ataku, wszelkie przesłanki wskazujące na zajście ataku wskazują też na istnienie luki. Na przykład obecność niespodziewanych atrybutów w żądaniach czy pakietach można sklasyfikować jako luki projektowe albo luki fazy implementacji.

Jednak proces angażujący zautomatyzowane narzędzia wyszukiwania najpopularniejszych luk (czyli tak zwane skanery bezpieczeństwa) i proces samego ataku wykorzystującego ewentualnie odkryte luki to osobne kategorie ataków. Wśród tzw. skanerów bezpieczeństwa mamy i narzędzia darmowe, jak X-SPider czy ShadowSecurityScanner, i produkty komercyjne, jak Internet Scanner czy Retina. I choć narzędzia te powinny być wykorzystywane jako pomoc administratora i środek zabezpieczający, często jest dokładnie odwrotnie — zwłaszcza w przypadku narzędzi darmowych, niezabezpieczonych w żaden sposób przed takimi nadużyciami.

W wykrywaniu takich narzędzi wyróżniamy dwa aspekty:

- ♦ **Wykrywanie zastosowania skanerów bezpieczeństwa** — niektóre specjalizowane narzędzia, jak system Courtney opracowany przez CIAC (Computer Incident Advisory Capability) potrafią wykryć fakt użycia skanera SATAN.

- ◆ **Wykrywanie realizowanego ataku** — opracowany przez IIS system RealSecure Network Sensor potrafi wykryć imitowanie przez intruza czynności usprawiedliwiającej działania podejmowane w ramach skanowania węzła.

Samo wykrycie zastosowania skanera bezpieczeństwa nie jest równoznaczne z uzyskaniem dowodu przeprowadzenia ataku. Narzędzia sondujące system komputerowy pod kątem znanych luk mogą być przecież wykorzystywane również w sposób autoryzowany. Konieczne jest więc przeprowadzenie dodatkowej analizy wszystkich zarejestrowanych przypadków zastosowania skanerów bezpieczeństwa. Ale jeśli, na przykład, wkrótce po wykryciu skanowania węzła wykryta zostanie próba wykorzystania jednej ze znanych luk, można te zdarzenia powiązać, traktując oba jako dowód przeprowadzenia ataku. Taka analiza z korelacją faktów może być przeprowadzona albo ręcznie, albo za pośrednictwem specjalizowanych narzędzi, jak RealSecure, SiteProtector (ISS) czy netForensics (netForensics).

## Nieprawidłowe (nietypowe) parametry ruchu sieciowego

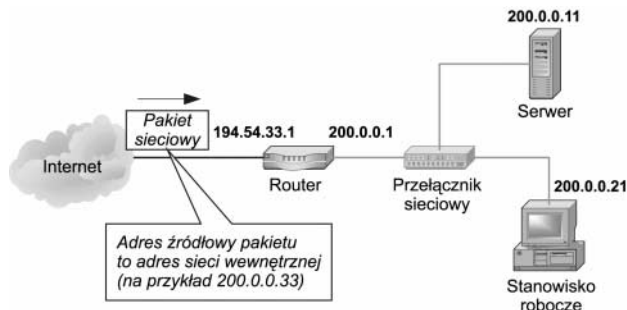
Przesłankami ataku mogą być też inne elementy. Na przykład cechy ruchu sieciowego, w tym:

- ◆ przychodzące do sieci wewnętrznej pakiety z adresami źródłowymi z sieci wewnętrznych;
- ◆ niespodziewane transmisje sieciowe (np. omijające zapórę sieciową);
- ◆ niespodziewane parametry ruchu sieciowego (np. niestandardowe kombinacje znaczników pakietów);
- ◆ wykrycie prób skanowania sieci;
- ◆ pojawiające się półotwarte połączenia charakterystyczne dla ataków odmowy obsługi;
- ◆ powtarzające się próby nawiązania połączeń z rzadko wykorzystywanymi albo nieistniejącymi w sieci usługami;
- ◆ powtarzające się z dużą częstotliwością próby nawiązania połączenia z węzłami lub usługami udostępnianymi przez węzły;
- ◆ pojawiające się połączenia z i do niespodziewanych lokacji;
- ◆ powtarzające się wielokrotnie błędne połączenia.

## Parametry transmisji przychodzących

Najbardziej obrazowym przykładem przesłanki ataku są przychodzące do chronionego segmentu sieci LAN pakiety z sieci zewnętrznych, których adresy źródłowe należą do zakresu adresów sieci wewnętrznej (jak na rysunku 4.2). Większość zapór sieciowych potrafi zidentyfikować tego typu przesłanki.

**Rysunek 4.2.**  
Podstawienie adresu  
źródłowego pakietu  
spoza sieci

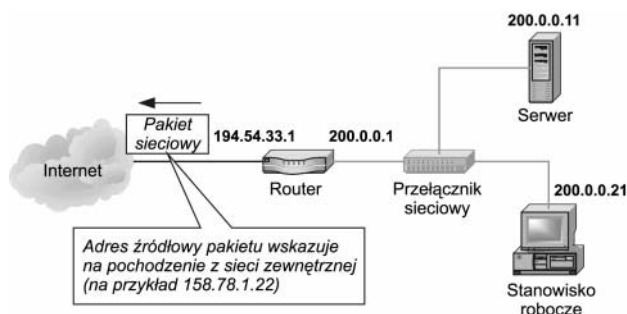


Jeśli system wykrywania włamań (albo inny mechanizm zabezpieczający czy mechanizm kontroli dostępu, jak zaporę sieciową czy router) nie potrafi rozpoznawać kierunku transmisji pakietu, wtedy sieć może ulec atakowi wykorzystującemu podszywanie. W tym ataku intruz może inicjować w sieci wewnętrznej transmisję, przy czym transmisje te będą interpretowane jako inicjowane przez jeden z węzłów sieci wewnętrznej, a więc transmisje autoryzowane. Tymczasem zwykle rygory reguł bezpieczeństwa dla węzłów sieci wewnętrznej są luźniejsze niż dla węzłów zewnętrznych.

## Parametry transmisji wychodzących

Przesłanką ataku może być również zaobserwowanie pakietu z wewnętrznej sieci LAN, którego adres źródłowy wskazuje na pochodzenie z sieci zewnętrznej (jak na rysunku 4.3). W takim układzie intruz mający dostęp do sieci wewnętrznej stara się ukryć swoje poczynania, próbując zafalszować ich źródło, stwarzając wrażenie, jakoby były one inicjowane z sieci zewnętrznej. W ten sposób odsuwa od siebie podejrzenia, utrudniając ewentualne śledztwo.

**Rysunek 4.3.**  
Podstawienie adresu  
źródłowego pakietu  
z wnętrza sieci



## Niespodziewane adresy pakietów

Przypadek niespodziewanego adresu źródłowego pakietu (albo niespodziewanego numeru portu docelowego w przypadku protokołów TCP i UDP) można uznać za przesłankę ataku. W ramach pierwszego przykładu rozważmy wykrycie pakietu pochodzącego z sieci zewnętrznej z niedostępnym adresem IP (jak w przypadku ataku Kevina Mitnicka) albo niedozwolonym adresem IP. Istnieje bowiem w sieci internet klasa adresów nieroutowalnych, na przykład 10.\*.\*.\*, 172.16.0.0 do 172.31.255.255 czy 192.168.\*.\* (patrz listingi 4.10 i 4.11). Wymienione zakresy adresów opisane są

w dokumencie RFC 1918 (*Address Allocation for Private Internets* — przydział adresów dla sieci prywatnych). Poza tą kategorią adresów jest jeszcze zakres adresów, których po prostu nie mogą mieć pakiety przychodzące do danej sieci z zewnątrz. Adresy te są bowiem zarezerwowane przez IANA. Wśród adresów zarezerwowanych są 0.0.0.0/8, 1.0.0.0/8, 2.0.0.0/8, 5.0.0.0/8, 7.0.0.0/8, 23.0.0.0/8, 31.0.0.0/8, 36.0.0.0/8, 37.0.0.0/8, 39.0.0.0/8, 41 do 42.0.0.0/8, 58 do 60.0.0.0/8, 67 do 127.0.0.0/8, 219 do 223.0.0.0/8, 240 do 255.0.0.0/8.

**Listing 4.10.** Wykrywanie zastosowania zarezerwowanych adresów IP (fragment pliku dziennika programu *TCPdump*)

---

```
03:38:18.285290 10.1.100.23.2483 > 192.168.1.2.21: S 21101636:21101636(0) win 8192 (DF)
03:38:18.287184 10.1.100.23.2483 > 192.168.1.2.21: S 21101636:21101636(0) win 8192 (DF)
```

---

**Listing 4.11.** Wykrywanie zastosowania zarezerwowanych adresów IP (fragment pliku dziennika routera *Cisco*)

---

```
May 10 09:20:33.328 UTC: %SEC-6-IPACCESSLOGP: list 100 denied tcp 10.1.2.73(0) -> 192.231.90.254(0), 1 packet
May 10 09:26:04.564 UTC: %SEC-6-IPACCESSLOGP: list 100 denied tcp 10.1.2.73(0) -> 192.231.90.254(0), 4 packets
May 10 09:26:34.260 UTC: %SEC-6-IPACCESSLOGP: list 100 denied tcp 10.0.0.57(0) -> 192.231.90.254(0), 1 packet
May 10 09:32:04.708 UTC: %SEC-6-IPACCESSLOGP: list 100 denied tcp 10.0.0.57(0) -> 192.231.90.254(0), 20 packets
```

---

Kolejny przykład to atak Land, w ramach którego adres oraz port źródłowy są ustawiane identycznie jak adres i port docelowy (patrz listing 4.12). Przetwarzanie takiego pakietu wprowadza węzeł w nieskończoną pętlę.

**Listing 4.12.** Atak Land (fragment pliku dziennika programu *TCPdump*)

---

```
10:56:32.395383 200.0.0.104.139 > 200.0.0.104.139: S
10:56:35.145383 200.0.0.104.139 > 200.0.0.104.139: S
10:56:36.265383 200.0.0.104.139 > 200.0.0.104.139: S
```

---

Następny przykład podejrzanego połączenia to połączenie Telnet inicjowane z zupełnie nieznanego węzła albo z węzła znanego, ale niepozostającego z adresatem transmisji w związku zaufania. Kolejny klasyczny przykład to niedopasowanie adresów MAC i IP. W ramach ostatniego przykładu przytoczmy atak o nazwie Setiri Trojan, omawiany na konferencji Def Con 10. Ów koń trojański, wykorzystujący lukę w przeglądarce Internet Explorer, nie jest wykrywany przez zapory sieciowe i nie może być przez nie zablokowany. Wykrycie tego konia trojańskiego jest o tyle utrudnione, że nie inicjuje on połączenia internetowego, jak większość innych tego typu programów. Zamiast tego otwiera niewidoczne okno przeglądarki Internet Explorer; transmisje inicjowane przez to okno nie są zatrzymywane przez zaporę sieciową. Wszelkie nieautoryzowane czynności — jak nawiązywanie połączenia z anonimowym serwerem proxy, pobieranie plików z bliżej nieokreślonych węzłów itd. — realizowane są właśnie za pośrednictwem owego niewidocznego okna.

## Nieoczekiwane parametry pakietów sieciowych

Nie byłoby trudno o listę ataków, których cechą charakterystyczną są niespodziewane parametry pakietów sieciowych. Ataki te nasilają się z upływem czasu, ponieważ włamywacze wciąż wyszukują nowe luki i błędy w implementacjach stosów protokołów TCP/IP we wszystkich niemal systemach operacyjnych. Na przykład, jeśli w sieci wykryty zostanie pakiet TCP z ustawionymi bitami SYN i ACK (reprezentujący drugi z trzech etapów nawiązywania połączenia TCP), można podejrzewać próbę ukrycia nieuprawnionej penetracji (listing 4.13). Metoda ta wykorzystywana jest między innymi w utajonym skanowaniu, ataku dość popularnym wśród włamywaczy [Northcutt1-99].

**Listing 4.13.** Skanowanie utajone pakietami SYN/ACK (fragment pliku dziennika programu TCPdump)

```
06:41:24.067330 stealth.bad.guy.org.113 > viper.infosec.ru.1004: S
4052190291:4052190291(0) ack 674711610 win 8192
06:42:08.063341 stealth.bad.guy.org.113 > www.infosec.ru.2039: S
2335925210:2335925210(0) ack 674711610 win 8192
06:41:24.067330 stealth.bad.guy.org.113 > un.infosec.ru.2307: S
2718446928:2718446928(0) ack 674711610 win 8192
```

Pojawienie się pakietu z ustawionymi znacznikami SYN i ACK bez poprzedzenia go pakietem SYN może być również dowodem nie tyle ataku, co obecności w sieci komputerowej asymetrycznych routerów. Jeśli w chronionej sieci zainstalowane są takie urządzenia, potwierdzenie przesłanki ataku wymaga dodatkowych badań. Poza parą znaczników SYN i ACK obserwuje się też pakiety z następującymi, podejrzanymi znacznikami: FIN (pakiet kończący połączenie przy tzw. skanowaniu pakietami FIN — listing 4.14), RESET (pakiet zerujący połączenie przy tzw. skanowaniu pakietami RESET — listing 4.15) albo wprost brak jakichkolwiek znaczników nagłówka jak w przypadku tzw. skanowania pakietami pustymi (listing 4.16).

**Listing 4.14.** Skanowanie pakietami FIN — opcja -sF programu Nmap (fragment pliku dziennika programu TCPdump)

```
18:18:03.436878 WS_LUKA.57239 > WS_LUKICH.9535: F 0:0(0) win 3072
18:18:03.437131 WS_LUKA.57239 > WS_LUKICH.1482: F 0:0(0) win 3072
18:18:03.437335 WS_LUKA.57239 > WS_LUKICH.617: F 0:0(0) win 3072
18:18:03.437501 WS_LUKA.57239 > WS_LUKICH.148: F 0:0(0) win 3072
18:18:03.437709 WS_LUKA.57239 > WS_LUKICH.638: F 0:0(0) win 3072
18:18:03.437872 WS_LUKA.57239 > WS_LUKICH.1467: F 0:0(0) win 3072
18:18:03.438089 WS_LUKA.57239 > WS_LUKICH.1475: F 0:0(0) win 3072
18:18:03.438286 WS_LUKA.57239 > WS_LUKICH.852: F 0:0(0) win 3072
18:18:03.438446 WS_LUKA.57239 > WS_LUKICH.653: F 0:0(0) win 3072
18:18:03.442145 WS_LUKA.57239 > WS_LUKICH.672: F 0:0(0) win 3072
```

**Listing 4.15.** Atak Xmas — opcja -sX programu Nmap (fragment pliku dziennika programu TCPdump)

```
18:18:28.038171 WS_LUKA.57407 > WS_LUKICH.1031: FP 0:0(0) win 3072 urg 0
18:18:28.038378 WS_LUKA.57407 > WS_LUKICH.1112: FP 0:0(0) win 3072 urg 0
18:18:28.038643 WS_LUKA.57407 > WS_LUKICH.2048: FP 0:0(0) win 3072 urg 0
18:18:28.038846 WS_LUKA.57407 > WS_LUKICH.6666: FP 0:0(0) win 3072 urg 0
18:18:28.039015 WS_LUKA.57407 > WS_LUKICH.906: FP 0:0(0) win 3072 urg 0
```

```
18:18:28.039180 WS_LUKA.57407 > WS_LUKICH.135: FP 0:0(0) win 3072 urg 0
18:18:28.039369 WS_LUKA.57407 > WS_LUKICH.1003: FP 0:0(0) win 3072 urg 0
18:18:28.039575 WS_LUKA.57407 > WS_LUKICH.3141: FP 0:0(0) win 3072 urg 0
18:18:28.039739 WS_LUKA.57407 > WS_LUKICH.1448: FP 0:0(0) win 3072 urg 0
```

---

**Listing 4.16.** Skanowanie pakietami pustymi — opcja `-sN` programu `Nmap` (fragment pliku dziennika programu `TCPdump`)

---

```
18:20:04.466572 WS_LUKA.53497 > WS_LUKICH.2766: . win 1024
18:20:04.466776 WS_LUKA.53497 > WS_LUKICH.534: . win 1024
18:20:04.466995 WS_LUKA.53497 > WS_LUKICH.206: . win 1024
18:20:28.467164 WS_LUKA.53497 > WS_LUKICH.119: . win 1024
18:20:04.467372 WS_LUKA.53497 > WS_LUKICH.636: . win 1024
18:20:04.467575 WS_LUKA.53497 > WS_LUKICH.313: . win 1024
18:20:04.467836 WS_LUKA.53497 > WS_LUKICH.372: . win 1024
18:20:04.468040 WS_LUKA.53497 > WS_LUKICH.378: . win 1024
18:20:04.468204 WS_LUKA.53497 > WS_LUKICH.532: . win 1024
```

---

Skanowanie pakietami pustymi można lepiej zilustrować innym przykładem zaczerpniętym z pliku dziennika systemu wykrywania włamań `Snort` (na listingu 4.17).

---

**Listing 4.17.** Wykrycie skanowania pakietami pustymi (fragment pliku dziennika programu `Snort`)

---

```
[**] NULL Scan [**]
05/28-21:09:23.686988 200.0.0.200:27025 -> 200.0.0.104:1186 TCP TTL:44 TOS:0x0
ID:64660 DF ***** Seq: 0xE4714 Ack: 0xFFFFFFFF Win: 0x0
```

---

Dla porównania można przeanalizować analogiczny wpis dziennika programu `Snort` dla skanowania pakietami `FIN` (listing 4.18).

---

**Listing 4.18.** Wykrycie skanowania pakietami `FIN` (fragment pliku dziennika programu `Snort`)

---

```
[**] FIN Scan [**]
02/02-04:49:15.135173 0:D0:58:4A:46:D0 -> 0:10:5A:6C:9A:55 type:0x800 len:0x104
195.11.50.204:2931 -> my-squid:53 TCP TTL:39 TOS:0x0 ID:2037 *F**** Seq: 0x32563E
Ack: 0x362C0000 Win: 0x0
```

---

Każdy pakiet, który nie spełnia standardów zapisanych w stosownych dokumentach RFC, może spowodować nieprawidłowe działanie urządzeń i oprogramowania komunikacyjnego przetwarzającego taki pakiet. Dotyczy to przy tym nie tylko routerów i przełączników sieciowych, ale również zapór sieciowych i systemów wykrywania włamań. W wielu atakach wykorzystuje się więc niepoprawne kombinacje znaczników nagłówka TCP. Niektóre kombinacje powodują zawieszenie węzła przetwarzającego spreparowany pakiet, inne powodują ominięcie zapór sieciowych i systemów wykrywania włamań. Sposób reakcji na pakiety TCP opisany jest w dokumencie RFC 793. Żaden dokument nie opisuje jednak zalecanych reakcji na niepoprawne pakiety TCP. Z tego względu różne urzędnienia i systemy operacyjne rozmaicie reagują na nadsyłane pakiety z niepoprawnymi znacznikami nagłówka. Nagłówki ten przewiduje sześć takich znaczników: `SYN`, `ACK`, `FIN`, `RST`, `PSH` i `URG`. Niepoprawne kombinacje mogą być wykryte na podstawie obserwacji następujących kombinacji [Frederick1-00]:

- ♦ SYN+FIN. Ta kombinacja jest niepoprawna, ponieważ znaczniki SYN i FIN wzajemnie się wykluczają. Pierwszy z nich to sygnał nawiązania połączenia, a drugi — jego zakończenia. Kombinacja ta jest bardzo często wykorzystywana przez oprogramowanie skanujące, jak choćby Nmap. Do niedawna systemy wykrywania włamań nie radziły sobie z wykrywaniem takich anormalnych pakietów. Dziś sytuacja się polepsza, ale wciąż łatwo oszukać system wykrywania włamań, dodając do kombinacji kolejne znaczniki (SYN+FIN+PSH, SYN+FIN+RST czy SYN+FIN+RST+PSH) — niektóre systemy wykrywania włamań dają się w ten sposób oszukać (listing 4.19). Niektórzy analitycy określają tego rodzaju ataki mianem „choinki”.

**Listing 4.19.** „Choinka” (fragment pliku dziennika programu Snort)

```
01/23-01:15:22.237103 195.11.212.180:30975 -> 192.0.97.80:49708
TCP TTL:49 TOS:0x0 ID:12207 DF
SFRPAU21 Seq: 0x78FFC22C Ack: 0x78FFC22C Win: 0xC22C
TCP Options => Opt 120 (40): C22C 78FF C22C 78FF C22C 78FF
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 78 FF C2 2C 78 FF
X...X
01/23-01:15:43.538590 195.11.212.180:30975 -> 92.0.97.80:49708
TCP TTL:49 TOS:0x0 ID:13449 DF
SFRPAU21 Seq: 0x78FFC22C Ack: 0x78FFC22C Win: 0xC22C
TCP Options => Opt 120 (40): C22C 78FF C22C 78FF C22C 78FF
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 78 FF C2 2C 78 FF
X...X
```



#### **Norton Personal Firewall nie wykrywa skanowania pakietami SYN+FIN**

W kwietniu 2002 roku pojawiły się doniesienia, że Norton Personal Firewall, zapora sieciowa dla systemu Windows 2000, nie zawsze wykrywa skanowanie pakietami SYN+FIN pomimo włączonej opcji wykrywania skanowania portów. Przy wykrywaniu skanowania zapora sieciowa przez pół godziny wykrywała jedynie skanowanie pakietami SYN, „zapominając” zupełnie o kombinacji SYN+FIN.

- ♦ Pakiet TCP nie powinien zawierać osamotnionego znacznika FIN. Jeśli w nagłówku ustawiony jest wyłącznie znacznik FIN, można podejrzewać atak polegający na skanowaniu utajonym.
- ♦ Pakiety TCP muszą zawierać w nagłówku przynajmniej jeden znacznik (ale nie FIN, jeśli miały to być jedyny znacznik).
- ♦ Jeżeli nagłówek pakietu nie zawiera znacznika ACK i nie jest to pierwszy z pakietów trójfazowej procedury nawiązywania połączenia, można podejrzewać atak, ponieważ znacznik ACK musi występować we wszystkich pakietach prawidłowej komunikacji (ze wskazanym wyjątkiem).
- ♦ Podejrzane są też kombinacje znaczników RST+FIN czy SYN+RST (listing 4.20).

**Listing 4.20.** Wykrycie podejrzanej komunikacji (fragment pliku dziennika systemu wykrywania Dragon)

```
20:42:17 [T] 172.20.255.135 10.168.74.96 [TCP-FLAGS] (flags:-1-SR---,
dp=11498,sp=0) (dragon-sensor) [PORT-ZERO] (tcp,dp=11498,sp=0)
```

Protokół TCP zakłada obecność bitów zarezerwowanych, których zastosowanie powinny określić przyszłe wersje protokołu. Bity te do dziś pozostawiono w spokoju. Z tego względu wykrycie pakietu, który ustawia owe zarezerwowane bity, może wskazywać na działalność nieautoryzowaną, w szczególności na próbę zdalnej identyfikacji systemu operacyjnego. Zwykle system operacyjny określany jest na podstawie nagłówków wysyłanych z systemu pakietów (analiza pasywna) albo na podstawie jego reakcji na specjalnie spreparowane pakiety (analiza aktywna). W czasie przygotowywania niniejszej książki w użyciu były trzy narzędzia zdalnej identyfikacji systemu operacyjnego: QueSO (Listing 4.21), hping i Nmap.

**Listing 4.21.** Identyfikacja systemu operacyjnego — program *QueSO*

```
luka# queso -d 200.0.0.253
Starting luka.infosec.com:6363 -> 200.0.0.253:80
IN #0 : 80->6363 S:1 A:+1 W:7C00 U:0 F: SYN ACK
IN #1 : 80->6364 S:0 A: 0 W:0000 U:0 F: RST
IN #3 : 80->6366 S:0 A: 0 W:0000 U:0 F: RST
IN #4 : 80->6367 S:1 A:+1 W:7C00 U:0 F: SYN FIN ACK
IN #6 : 80->6369 S:1 A:+1 W:7C00 U:0 F: SYN ACK XXX YYY
200.0.0.253:80      * Linux 1.3.xx, 2.0.0 to 2.0.34

luka# queso -d 200.0.0.200
Starting luka.infosec.ru:15690 -> 200.0.0.200:80
IN #0 : 80->15690 S:1 A:+1 W:2180 U:0 F: SYN ACK
IN #1 : 80->15691 S:0 A:+1 W:0000 U:0 F: RST
IN #2 : 80->15692 S:0 A:+1 W:0000 U:0 F: RST ACK
IN #3 : 80->15693 S:0 A:+1 W:0000 U:0 F: RST
IN #4 : 80->15694 S:1 A:+1 W:2180 U:0 F: SYN ACK
IN #5 : 80->15695 S:0 A:+0 W:0000 U:0 F: RST ACK
IN #6 : 80->15696 S:1 A:+1 W:2180 U:0 F: SYN ACK
200.0.0.200:80      * Windoze 95/98/NT
```

Po uruchomieniu program *QueSO* wykrywa otwarte porty skanowanego węzła i przesyła do niego kilka pakietów niespełniających rygorów RFC. Analiza odpowiedzi na te pakiety pozwala na określenie rodzaju i wersji systemu operacyjnego (patrz też listing 4.22).

**Listing 4.22.** Próba identyfikacji systemu operacyjnego programem *QueSO* (fragment pliku dziennika programu *TCPdump*)

```
02:35:10.11 WS_LUKA.29709 > WS_LUKICH.80: S 1173826820:1173826820(0) ack 0
02:35:10.13 WS_LUKA.29710 > WS_LUKICH.80: F 1173826820:1173826820(0)
02:35:10.15 WS_LUKA.29711 > WS_LUKICH.80: F 1173826820:1173826820(0) ack 0
02:35:10.17 WS_LUKA.29712 > WS_LUKICH.80: SF 1173826820:1173826820(0)
02:35:10.19 WS_LUKA.29713 > WS_LUKICH.80: P
02:35:10.21 WS_LUKA.29714 > WS_LUKICH.80: S 1173826820:1173826820(0)
      4500 0028 ee7b 0000 fc06 2f62 ab45 a42d
      ac15 a569 7412 0017 45f7 2d04 0000 0000
      50c2 1234 14d8 0000 0000 0000 0000
```

W pakietach przychodzących można spodziewać się następujących niepoprawnych kombinacji znaczników:

- ♦ SYN+ACK z nieprawidłowym numerem ACK
- ♦ FIN
- ♦ FIN+ACK z nieprawidłowym numerem ACK
- ♦ SYN+FIN
- ♦ PSH
- ♦ SYN+XXX+YYY, gdzie XXX i YYY to zarezerwowane znaczniki nagłówka TCP (dwa najbardziej znaczące bity 13. bajta nagłówka pakietu TCP)

Skaner Nmap również korzysta z podobnej metody identyfikowania systemu operacyjnego. Używa jednak nieco innych kombinacji znaczników:

- ♦ SYN
- ♦ brak znaczników
- ♦ SYN+FIN+URG+PSH
- ♦ ACK
- ♦ SYN (pakiet kierowany do zamkniętego portu)
- ♦ ACK (pakiet kierowany do zamkniętego portu)
- ♦ FIN+URG+PSH (pakiet kierowany do zamkniętego portu)

Wraz z opublikowaniem dokumentów RFC 2481 [RFC1-99] i RCF 2884 [RFC1-00] sytuacja jeszcze się pogorszyła. Zgodnie z wymienionymi standardami, bity zarezerwowane mogą być wykorzystywane do przekazywania informacji serwisowych. Według [Miller1-00], jeśli pakiet odpowiada temu z listingu 4.23, można podejrzewać, że sieć jest skanowana za pomocą programu QueSO, hping albo Nmap.

**Listing 4.23.** *Zastosowanie zarezerwowanych znaczników ECN w nagłówku pakietu TCP (fragment pliku dziennika programu TCPdump)*

```
12:25:38.650123 bad.guy.org.1641 > viper.infosec.ru.111: S 1533993767:1533993767(0)
win 512 (ttl 64, id 64461)
4500 0028 fbcd 0000 4006 91f5 xxxx xxxx
xxxx xxxx 0669 006f 5b6e e327 0000 0000
50c2 0200 7b16 0000
```

Aby poznać znaczenie fragmentu c2 pakietu (wartość ta korzysta z dwóch zarezerwowanych bitów), należałoby przeprowadzić dodatkową analizę pakietu wykluczającą albo potwierdzającą fakt ataku.

Przy uwzględnieniu możliwości zastosowania znaczników zarezerwowanych (ECN) nowa wersja programu TCPdump (3.5) wyświetla również wartości tych znaczników (patrz listing 4.24).

**Listing 4.24.** Zastosowanie zarezerwowanych znaczników ECN w nagłówku pakietu TCP (fragment pliku dziennika programu TCPdump)

```
12:25:38.650123 bad.guy.org.1641 > viper.infosec.ru.111: S [ECN-Echo, CWR]
380601688:380601688 (0) win 4660 (ttl 244, id 55411)
4500 0028 d873 0000 f406 85b4 xxxx xxxx
xxxx xxxx 0669 006f 16af 8558 0000 0000
50c2 1234 39e5 0000 753a 0000 e3a8
```

Owo „inteligentne” zachowanie wprowadza jednak pewne zamieszanie, ponieważ program TCPdump interpretuje pakiet jako zgodny ze standardem RFC 2481, podczas gdy w rzeczywistości pakiet został wygenerowany przez program Nmap realizujący zdalną identyfikację systemu operacyjnego. Mamy tu do czynienia z fałszywym braniem alarmu (ang. *false negative*, tzw. fałszywe pominięcie).

Wśród innych oznak nienormalności pakietu można wymienić:

- ◆ wyzerowane numery portów (źródłowego bądź docelowego) w pakietach TCP i UDP (listing 4.25);
- ◆ ustawienie znacznika ACK (w pakietach protokołu TCP) i wyzerowanie pola numeru potwierdzenia.

**Listing 4.25.** Wykrycie podejrzanej działalności (fragment pliku dziennika programu TCPdump)

```
21:20:11.084066 172.20.20.1.0 > ftp.infosec.com.1030: F 1310724:1310728(4) ack
3642168720 win 20496 urg 1250
```

Stosunkowo często przesłanką ataku może być również pakiet mający niestandardowy rozmiar. Na przykład większość pakietów Echo Request protokołu ICMP ma 8-bitowy nagłówek i 56-bitowe pole danych. Jeśli w sieci wykryte zostaną pakiety o rozmiarach niestandardowych, można podejrzewać, że dzieje się coś złego. Na przykład atak Loki (omawiany już w rozdziale 1.) pozwala atakującemu na tunelowanie rozmaitych poleceń w pakietach Echo Request protokołu ICMP. Reakcje na te polecenia odsyłane są zaś w pakietach Echo Reply protokołu ICMP. Tunelowanie danych w pakietach ICMP powoduje znaczne ich — w porównaniu z pakietami standardowymi — „rozduęcie”. Innym przykładem może być atak Ping of Death, w ramach którego atakujący generuje pakiet Echo Request protokołu ICMP o rozmiarze ponad 65 536 bajtów (listing 4.26).

**Listing 4.26.** Wykrycie ataku Ping of Death (fragment pliku dziennika programu TCPdump)

```
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@0+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@2656+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@3040+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@3416+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@376+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@3800+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@4176+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@760+)
...
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@63080+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@63456+)
```

```
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@63840+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@64216+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@64600+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@64976+)
12:43:431 big.pinger.org > 200.0.0.104: icmp: echo request (frag 4321:380@65360+)
```

---

Łączna długość pakietu po jego zmontowaniu wynosi 65 740 (380 + 65 360) bajtów.

Jeśli do sieci komputerowej napływają pakiety fragmentowane, możemy mówić o kolejnej przesłance ataku i w większości przypadków przesłanka ta jest prawdziwa. Dostępne współcześnie urządzenia i oprogramowanie zabezpieczające systemy komputerowe nie radzą sobie zbyt dobrze z asemblacją (montażem) takich pakietów. Często próba zmontowania fragmentowanego pakietu kończy się zawieszeniem programu; ewentualnie jego późniejsza analiza okazuje się nieskuteczna i pakiet mimo charakterystyk wskazujących na atak przenika do sieci. Pierwszy z tych efektów (załamanie narzędzia zabezpieczającego) można osiągnąć, ustawiając niepoprawne przesunięcia poszczególnych fragmentów pakietów (w prezentowanym na listingu 4.26 przykładowym ataku Ping of Death przesłany fragment nie ma rozmiaru będącego wielokrotnością 8 — pierwszy pakiet ma 380 bajtów. Drugi efekt osiąga się często atakiem określanym przez włamywaczy mianem „fragmentników” (ang. *tint fragment*) — trzeba w nim wygenerować dwa fragmenty pakietu TCP. Pierwszy powinien być tak mały, aby nie zawierał nawet kompletnego nagłówka TCP, w tym — to ważne — numeru portu docelowego. W drugim fragmencie należy przesłać resztę nagłówka. Większość zapór sieciowych przepuści pierwszy z fragmentów (a niekiedy oba) do chronionej sieci (listing 4.27).

**Listing 4.27.** *Atak z wykorzystaniem „fragmentników” (fragment pliku dziennika programu TCPdump)*

---

```
06:25:55.315 [|tcp] (frag 38783:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 38783:4@16+)
06:25:55.315 [|tcp] (frag 16422:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 16422:4@16+)
06:25:55.315 [|tcp] (frag 43143:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 43143:4@16+)
06:25:55.315 [|tcp] (frag 18554:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 18554:4@16+)
06:25:55.315 [|tcp] (frag 8231:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 8231:4@16+)
06:25:55.315 [|tcp] (frag 45846:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 45846:4@16+)
06:25:55.315 [|tcp] (frag 6245:16@0+)
06:25:55.315 bad.guy.org > viper.infosec.ru: (frag 6245:4@16+)
```

---

Etykieta [|tcp] sygnalizuje niemożność pozyskania naraz całego nagłówka pakietu TCP i zinterpretowania pełnego adresu docelowego pakietu.

## Anomalie ruchu sieciowego

Pod pojęciem anomalii ruchu sieciowego rozumiemy wszelkie odchylenia parametrów transmisji sieciowych od normy. Parametrami tymi mogą być: stopień przeciążenia sieci, średni rozmiar pakietu, średnia liczba pakietów z fragmentowanych i tak dalej. Odchylenia od normalnych wartości tych parametrów to jawne przesłanki ataku, ewentualnie sygnały niepoprawnego działania urządzeń sieciowych.

## Podważane cechy transmisji sieciowych

W ramach podważanych cech transmisji sieciowych można rozważyć następujące czynniki:

- ◆ **Podważane transmisje od konkretnego nadawcy albo do konkretnego odbiorcy.** W niektórych przypadkach określone transmisje czy ich treść są podważane. Na przykład, jeśli wiadomość poczty elektronicznej zawiera pewne słowa, kiedy następuje próba odwołania się do adresu takiego jak *http://www.playboy.com*, *http://www.recruitment.com* czy *http://www.vacation.com*, albo do węzła przychodzi transmisja sterowana protokołem, którego obsługi się od nadawcy nie oczekuje (na przykład żądanie zainicjowania połączenia Telnet przez dział operacji bankowych).
- ◆ **Podważane transmisje.** Niektóre transmisje są podważane same w sobie, niezależnie od tego, do którego z węzłów sieci są adresowane; mowa na przykład o transmisjach sterowanych nieoczekiwanymi protokołami albo inicjowanymi spod adresów nienależących do sieci wewnętrznej. Przykładowo, jeden z pracowników banku wykorzystującego system RealSecure Network Sensor wykrył nieautoryzowane połączenie modemowe z siecią banku; połączenie zestawione zostało przez innego pracownika banku, który za jego pośrednictwem „załatwił” sobie szybsze połączenie domowego komputera z siecią internet. Tego rodzaju transmisje mogą zostać wykryte przez monitorowanie transmisji sieciowych w poszukiwaniu określonych słów kluczowych (np. „dane poufne”, „ściśle tajne” itd.).

## Wartości domyślne rozmaitych atrybutów

Jeszcze jedną przesłanką ataku może być zaobserwowanie domyślnych wartości atrybutów takich jak nazwy plików czy procesów (np. koni trojańskich), numerów portów itd. Tę klasę przesłanek rozważymy na przykładzie ataku wykorzystującego program konia trojańskiego.

Większość takich programów posiada pewne określone nazwy plików wykonywalnych i procesów; obecność takich plików w systemie plików czy takich procesów na liście uruchomionych procesów może wskazywać na nieautoryzowany dostęp do danego węzła. Jeśli, na przykład, na dysku twardym komputera znajduje się plik *Patch.exe*, to można podejrzewać przeniknięcie do komputera koni trojańskich NetBus, Digital RotBeer, Krenx czy Solid Gold. Lista nazw plików wykonywalnych znanych programów koni trojańskich publikowana jest na stronie WWW pod adresem *http://www.simovits.com/trojans/trojans\_files.html*. Na tej samej witrynie można znaleźć informacje o numerach portów okupowanych przez te programy. W wykryciu koni trojańskich pomocna może być również lista procesów uruchomionych w systemie. W systemach uniksowych listę taką wyświetla się poleceniem *ps*; w systemach z rodziny Windows NT i 2000 analogiczną listę można zobaczyć po uruchomieniu *Menedżera zadań* (rysunek 4.4). Oczywiście obecność procesu konia trojańskiego na takiej liście uzależniona jest od tego, czy włamywacz nie zainstalował wraz z koniem trojańskim pakietu maskującego (ang. *rootkit*) mającego ukryć jego obecność w systemie.

**Rysunek 4.4.**  
 Widoczny na liście  
 zadań proces  
*Patch.exe* to proces  
 konia trojańskiego  
*NetBus*

Nazwa	PID (l...	CPU	Czas CPU	Użycie pa...
Systemowy proces ...	0	93	1:21:41	16 K
System	2	00	0:00:19	216 K
SMSS.EXE	20	00	0:00:00	396 K
CSRSS.EXE	24	00	0:00:00	1816 K
WINLOGON.EXE	34	00	0:00:00	140 K
SERVICES.EXE	40	00	0:00:00	2996 K
LSASS.EXE	43	00	0:00:00	2860 K
<b>Patch.exe</b>	<b>44</b>	<b>00</b>	<b>0:00:00</b>	<b>2388 K</b>
SPOOLSS.EXE	69	00	0:00:00	2416 K
RPCSS.EXE	75	00	0:00:00	1212 K
msdtc.exe	85	00	0:00:01	3444 K
sqlservr.exe	105	00	0:00:00	6756 K
NDDEAGNT.EXE	110	00	0:00:00	1058 K
PSTORES.EXE	115	00	0:00:00	140 K
fdon.exe	132	00	0:00:07	436 K
EXPLORER.EXE	145	00	0:01:02	3432 K
vi_gm.exe	146	00	0:00:00	1364 K
gg.exe	148	06	0:03:09	19792 K
WINCMD32.EXE	152	00	0:00:11	620 K
LOADWC.EXE	157	00	0:00:00	1256 K
DDHELP.EXE	161	00	0:00:00	1304 K
TASKMGR.EXE	166	01	0:00:03	1288 K

Kolejna metoda wykrywania koni trojańskich polega na analizie zestawu otwartych portów (tak na podstawie analizy żądań nawiązania połączenia kierowanych do danego węzła, jak i na podstawie wykonanego samodzielnie skanowania portów, zdalnego bądź za pośrednictwem stosownych narzędzi systemowych). W ten sposób można wykryć obecność większości koni trojańskich, które zwykle wykorzystują do komunikacji port o określonym numerze. Jeśli koń trojański korzysta z niestandardowych numerów portów, można wykryć jego obecność na podstawie obecności w transmisjach pewnych słów kluczowych przewidzianych w „protokole” komunikacyjnym konia trojańskiego. Na prezentowanych niżej listingach (listingi od 4.28 do 4.34) można zaobserwować zapisy wskazujące na obecność konia SubSeven (w wersji 2.1) wykorzystującego w komunikacji porty 1234 oraz 27374.

**Listing 4.28.** Wykrycie konia trojańskiego *SubSeven* (fragment pliku dziennika programu *Snort*)

```
[**] BACKDOOR Attempt- Subseven [**]
12/26-23:09:42.219109 0:90:27:F:22:A2 -> 0:40:5:F6:34:51 type:0x800 len:0x4E
216.192.29.30:3216 ->206.18.108.130:1243 TCP TTL:64 TOS:0xD0 ID:11841
S***** Seq: 0x4908C6 Ack: 0x0 Win: 0x2000
TCP Options => MSS: 536 NOP WS: 0 NOP NOP TS: 0 0 Opt 9 (40): 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

**Listing 4.29.** Wykrycie konia trojańskiego *SubSeven* (fragment pliku dziennika zapory sieciowej *IPCHAINS*)

```
Jan 5 02:56:39 input REJECT eth1 PROTO=TCP 152.166.212.218:2102 192.168.1.1:1243
L=48 S=0x00 I=38494 F=0x4000 T=108 SYN (#13)
```

**Listing 4.30.** *Wykrycie konia trojańskiego SubSeven (fragment pliku dziennika zapory sieciowej Ascend SecureConnect 3.03)*


---

```

Mar 30 19:44:02 209-30-73-81.flash.net ASCEND: wan5 tcp 209.30.73.80;27374 <-
24.29.78.48;1493 58 syn !pass (totcp-1)
Mar 30 19:44:02 209-30-73-81.flash.net ASCEND: wan6 tcp 209.30.73.81;27374 <-
24.29.78.48;1494 58 syn !pass (totcp-1)
Mar 30 19:44:02 209-30-73-81.flash.net ASCEND: wan5 tcp 209.30.73.82;27374 <-
24.29.78.48;1495 58 syn !pass (totcp-1)
Mar 30 19:44:02 209-30-73-81.flash.net ASCEND: wan6 tcp 209.30.73.83;27374 <-
24.29.78.48;1496 58 syn !pass (totcp-1)
Mar 30 19:44:02 209-30-73-81.flash.net ASCEND: wan6 tcp 209.30.73.95;27374 <-
24.29.78.48;1508 58 syn !pass (totcp-1)

```

---

**Listing 4.31.** *Wykrycie konia trojańskiego SubSeven (fragment pliku dziennika zapory sieciowej ZoneAlarm)*


---

```

Name Packet sent from 24.226.103.143 (TCP Port 3387)
to x.x.x.x (TCP Port 27374) was blocked
  Status Dropped
  Source IP Address 24.226.103.143
  Destination IP Address x.x.x.x
  Source Port 3387
  Destination Port 27374
  Link Layer Protocol 1
  Network Layer Protocol 1
  Transport Layer Protocol 2
  Count 1
  Status Code 100002
  Lock Level 0
  Security Information 0,1,0,2
  Operating System WIndows 98-4.10.1998- -SP
  Product ZoneAlarm
  ProductVersion 2.0.26
  Language 0409
  State Find Code 13

```

---

**Listing 4.32.** *Wykrycie konia trojańskiego SubSeven (fragment pliku dziennika programu NFR Back Officer)*


---

```

17:54:19.596269 ip 60: 24-216-141-52.hsacorp.net .1566 > my.box.1243:
S 12759610:12759610(0) win 8192 <mss 1460> (DF) (ttl 19, id 61249)
17:54:19.624034 ip 54: my.box.1243 > 24-216-141-52.hsacorp.net .1566:
R 0:0(0) ack 12759611 win 0 (ttl 128, id 4263)

```

---

**Listing 4.33.** *Wykrycie konia trojańskiego SubSeven (fragment pliku dziennika zapory sieciowej BlackICE Defender)*


---

```

59 2000-05-04 19:35:31 2003105 SubSeven port probe 172.142.109.211
AC8E6DD3.ipt.aol.com 152.207.70.106 port=1243&name=Sub_7 8

```

---

**Listing 4.34.** *Wykrycie konia trojańskiego SubSeven (fragment pliku dziennika zapory sieciowej Pix)*


---

```

Jun 03 00:06:26 [FW1] Jun 03 2000 00:08:00: %PIX-2-106001:
Inbound TCP connection denied from 216.58.19.218/3483
to server1/27374 flags SYN

```

---

```
Jun 03 00:06:26 [FW1] Jun 03 2000 00:08:00: %PIX-7-106001:
Deny inbound (No xlate) tcp src outside:216.58.19.218/3487
dst outside:global/27374
```

Program konia trojańskiego Satans korzysta w komunikacji z portu o numerze 666 (z tego samego numeru portu korzysta też koń trojański BackConstruction). Program Satans można też wykryć na podstawie obecności w pamięci systemu programu o nazwie WinVMM32 (listing 4.35).

**Listing 4.35.** Wykrycie konia trojańskiego Satans (fragment pliku dziennika programu Snort)

```
[**] BACKDOOR Attempt- Attack FTP / Satans Backdoor [**]
12/26-23:09:51.159109 0:0:0:0:0:0 -> 0:0:0:0:0:0 type:0x800 len:0x4E
216.192.29.30:3125 -> 206.18.108.130:666 TCP TTL:65 TOS:0xC8 ID:25409
S***** Seq: 0x4907EF Ack: 0x0 Win: 0x2000
TCP Options => MSS: 536 NOP WS: 0 NOP NOP TS: 0 0 Opt 9 (40): 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

Koń trojański BackOrifice komunikuje się za pośrednictwem portu o numerze 31337 (patrz listingi 4.36 i 4.37). Zostawia też ślady swojej obecności w rejestrze systemu Windows, w kluczu `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices`.

**Listing 4.36.** Wykrycie konia trojańskiego BackOrifice (fragment pliku dziennika programu SHADOW)

```
15:20:48.698626 172.20.20.1.31338 > 192.168.1.1.31337: udp 19
```

**Listing 4.37.** Wykrycie konia trojańskiego BackOrifice (fragment pliku dziennika zapory sieciowej IPCHAINS)

```
Jan 4 20:59:40 input REJECT eth1 PROTO=UDP 24.114.172.74:3764 192.168.1.1:31337
L=47 S=0x00 I=65460 F=0x0000 T=123 (#16)
```

Jak już wspomniałem, obecność koni trojańskich można wykrywać, nie tylko analizując żądania nawiązania połączeń z konkretnymi numerami portów, ale również wyszukując w treści transmisji pewnych określonych słów (technika ta sprawdza się zwłaszcza w przypadku koni trojańskich niekorzystających z jednego, domyślnego numeru portu). Na przykład obecność programu BackOrifice można wykryć, obserwując w pierwszych ośmiu bajtach pola danych pakietu UDP następującą sygnaturę: `ce 63 d1 d2 16 e7 13 cf` (patrz listing 4.38).

**Listing 4.38.** Wykrycie konia trojańskiego BackOrifice (fragment pliku dziennika programu TCPdump)

```
09:39:36.365 200.0.0.104.1613 > 200.0.0.200.31337: udp 19
4500 002f cea7 0000 7d11 cd56 ab45 a49b
ac15 a5c9 064d 7a69 001b 30e7 ce63 d1d2
16e7 13cf 38a5 a586 b275 4b99 ad32 58
09:39:36.695 200.0.0.200.31337 > 200.0.0.104.1613: udp 53
4500 0051 b408 0000 2011 44d4 ac15 a5c9
ab45 a49b 7a69 064d 003d bc7b ce63 d1d2
16e7 13cf 1ea5 a586 7a75 4b99 2d61 2196
c7fc 8502 192a
```

```
09:39:36.695 200.0.0.200.31337 > 200.0.0.104.1613: udp 48
4500 004c b508 0000 2011 43d9 ac15 a5c9
ab45 a49b 7a69 064b 0038 bb34 ce63 d1d2
16e7 13cf 1ba5 a586 7b75 4b99 6d71 2d69
c1fc 8656 5031 ...
```

W rozdziale 2. wspomniałem, że poza klasycznymi atakami, bazującymi na modelach „jeden na jednego” i „jeden na wielu” istnieje też klasa tzw. ataków rozproszonych realizujących modele „wielu na jednego” albo „wielu na wielu”. Modele te są też wdrażane w niektórych koniach trojańskich, choćby w WinTrin00 (listing 4.39), TFN2K, Stacheldraht, mstream (listing 4.40) i tak dalej. Owe konie trojańskie mogą być wykrywane za pośrednictwem numerów portów wykorzystywanych przez nie do komunikacji ze światem zewnętrznym.

**Listing 4.39.** Wykrycie konia trojańskiego WinTrin00 (fragment pliku dziennika routera Cisco)

```
Feb 25 21:14:38 134.161.1.101 21043: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4419) -> 134.161.67.71(34555), 1 packet
Feb 25 21:14:51 134.161.1.101 21044: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4420) -> 134.161.67.71(34555), 1 packet
Feb 25 21:14:57 134.161.1.101 21045: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4421) -> 134.161.67.71(34555), 1 packet
Feb 25 21:15:00 134.161.1.101 21046: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4422) -> 134.161.67.71(34555), 1 packet
Feb 26 00:35:40 134.161.1.101 22024: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4523) -> 134.161.67.71(34555), 1 packet
Feb 26 11:01:56 134.161.1.101 24967: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4531) -> 134.161.67.71(34555), 1 packet
Feb 26 11:09:24 134.161.1.101 25007: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4541) -> 134.161.67.71(34555), 1 packet
Feb 26 11:09:26 134.161.1.101 25008: %SEC-6-IPACCESSLOGP:
list ingress denied udp 38.29.63.57(4542) -> 134.161.67.71(34555), 1 packet
```

**Listing 4.40.** Wykrycie konia trojańskiego mstream (fragment pliku dziennika programu TCPdump)

```
00:34:38.530000 192.168.0.20.1081 > 192.168.0.100.6838: udp 9
4500 0025 ef75 0000 4011 098a c0a8 0014 E..%.u..@.....
c0a8 0064 0439 1ab6 0011 2b63 6e65 7773 ...d.9...+cnews
6572 7665 7200 0000 0000 0000 0000      erver.....
```

Wykrywanie koni trojańskich nie musi polegać na monitorowaniu całości transmisji w sieci komputerowej — wystarczy niekiedy ograniczyć się do analizy otwartych portów. Można wtedy zidentyfikować programy koni trojańskich okupujących zwykle ściśle określone domyślne numery portów. Sondowanie otwartych portów można zrealizować zarówno zdalnie (za pośrednictwem skanerów portów), jak i lokalnie, za pośrednictwem stosownych programowych narzędzi systemowych, jak netstat (patrz listing 4.41).

**Listing 4.41.** Wykrycie konia trojańskiego NetBus (w wykazie portów wyświetlanym poleceniem netstat -a)

Active Connections

Proto	Local Address	Foreign Address	State
TCP	ws_lukich:135	0.0.0.0:0	LISTENING
TCP	ws_lukich:901	0.0.0.0:0	LISTENING

---

```

TCP    ws_lukich:2998      0.0.0.0:0          LISTENING
TCP    ws_lukich:9991      0.0.0.0:0          LISTENING
TCP    ws_lukich:12345     0.0.0.0:0          LISTENING
TCP    ws_lukich:12346     0.0.0.0:0          LISTENING
TCP    ws_lukich:1026      0.0.0.0:0          LISTENING
TCP    ws_lukich:1026      localhost:1027      ESTABLISHED
TCP    ws_lukich:1027      localhost:1026      ESTABLISHED
TCP    ws_lukich:137       0.0.0.0:0          LISTENING
TCP    ws_lukich:138       0.0.0.0:0          LISTENING
TCP    ws_lukich:nbsession 0.0.0.0:0          LISTENING
TCP    ws_lukich:nbsession ws_luka:3055        ESTABLISHED
TCP    ws_lukich:135       *:*:
TCP    ws_lukich:nbname    *:*:
TCP    ws_lukich:nbdatagram *:*:

```

---

W przypadku z listingu 4.41 na węźle lokalnym o nazwie WS\_LUKICH rezyduje koń trojański NetBus. Program ten prowadzi nasłuch pod następującymi numerami portów: 12345 oraz 12346. Lista portów wykorzystywanych przez znane programy koni trojańskich publikowana jest pod adresem [http://www.simovits.com/trojans/trojans\\_files.html](http://www.simovits.com/trojans/trojans_files.html).

Wykrywanie obecności koni trojańskich można zrealizować również zdalnie, sondując podejrzany węzeł pod kątem dostępności określonych portów. Można do tego celu wykorzystać dowolny ze skanerów portów, choćby z programu Nmap, dostępnego dla większości odmian systemów operacyjnych z rodziny UNIX i systemów Windows NT (listing 4.22).

**Listing 4.22.** Zdalne wykrywanie konia trojańskiego NetBus (wynik działania polecenia `nmapnt -sS 200.0.0.200`)

---

```

Starting nmapNT V. 2.53 by ryan@eEye.com
eEye Digital Security ( http://www.eEye.com )
based on nmap by fyodor@insecure.org ( www.insecure.org/nmap/ )

We skillfully deduced that your address is 0.0.0.0
Interesting ports on WS_LUKICH (200.0.0.200):
(The 1518 ports scanned but not shown below are in state: closed)
Port      State  Service
135/tcp   open   unknown
139/tcp   open   unknown
901/tcp   open   unknown
12345/tcp open   NetBus
12346/tcp open   NetBus

Nmap run completed -- 1 IP address (1 host up) scanned in 1 second

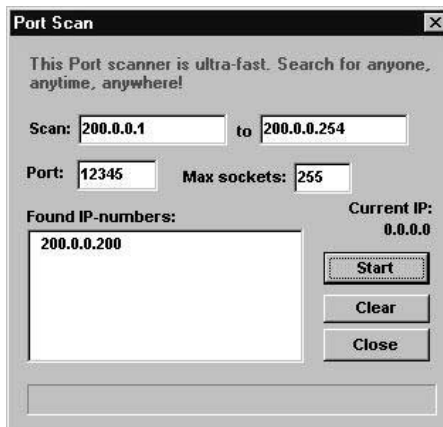
```

---

Zdalne skanowanie węzła WS\_LUKICH wykryło, że węzeł ten prowadzi nasłuch na portach o numerach 12345 i 12346, co pozwala na wysnucie wniosku, że węzeł ten zainfekowany jest koniem trojańskim NetBus. Wreszcie do wykrycia koni trojańskich można posłużyć się metodami nietradycyjnymi. Otóż niektóre programy do zarządzania zainstalowanymi już końmi trojańskimi pozwalają na przeprowadzenie skanowania zadanego zakresu adresów IP celem wykrycia zainstalowanych już składników programu konia trojańskiego. Funkcja taka dostępna jest również w programie zdalnie sterującym koniem trojańskim NetBus (rysunek 4.5).

**Rysunek 4.5.**

*Skanowanie sieci  
w poszukiwaniu  
rezydujących  
programów konia  
trojańskiego NetBus*

**Nieoczekiwane wartości atrybutów**

Żądanie kierowane z dowolnej sieci pochodzące od dowolnego użytkownika czy systemu charakteryzuje się pewnymi atrybutami opisującymi tę sieć, użytkownika czy system. Zestaw owych atrybutów nosi nazwę profilu. Profile wykorzystywane są do monitorowania i analizy działalności kontrolowanych obiektów. Najczęściej wykorzystywane w takiej analizie atrybuty (wymieniane w kolejnych punktach) mogą być analizowane również w ramach procesu wykrywania włamań.

**Data i czas**

Data i czas to najbardziej charakterystyczny atrybut obiektu, analizowany w ramach prób wykrywania przypadków naruszeń reguł bezpieczeństwa. Przypuśćmy na przykład, że w systemowym pliku dziennika zarejestrowany został fakt zalogowania się użytkownika do sieci o godzinie 18.30. Jeśli zalogowany jest jednym z administratorów systemu, jego działalność jest najprawdopodobniej uprawniona i reprezentuje pracę w godzinach nadliczbowych. Jeśli jednak system wykrywania włamań rejestruje zalogowanie się o takiej godzinie pracownika działu finansowego, to należałoby fakt reprezentowany takim wpisem rejestru dodatkowo zbadać (rozstrzygając, czy chodziło o pracę w nadgodzinach, czy może o udany atak polegający na próbie odgadnięcia nazwy konta i hasła). Podobnie należy rozpatrywać próby logowania i udane logowania w dniach świątecznych i w okresie weekendu. Przykładowo, w większości banków przelewy i płatności zlecane po pewnej godzinie odnotowywane są na kontach docelowych z pewnym opóźnieniem; jeśli na przykład wpłata następuje w piątek po południu, to jej realizacja jest zwykle odkładana do najbliższego poniedziałku. Jeśli pewna transakcja zostanie sfinalizowana wcześniej, można podejrzewać próbę defraudacji. Jeśli okres czasu pomiędzy złożeniem zlecenia a jego realizacją jest zbyt krótki (niewystarczający do przeprowadzenia zwykłej procedury potwierdzenia transakcji), można przypuszczać, że zlecenie jest niepoprawne albo realizujący je dopuszcza się nadużycia. Jak widać, określenie czasu pewnych operacji jest kluczowe dla stwierdzenia, czy są one operacjami uprawnionymi czy może próbami nadużyć i włamań. Szczególne znaczenie ma monitorowanie dat i czasu logowania, uruchamiania aplikacji, przeładowywania systemu, inicjowania rozruchu systemu i zatrzymywania systemu.

## Położenie

Zwykle użytkownik loguje się do sieci wciąż z tego samego komputera; w przypadku pracy zdalnej można mówić o stałości numeru telefonu, z którego następuje nawiązanie połączenia modemowego. Jeśli więc uda się zaobserwować logowanie z innego niż zwykle komputera albo próbę nawiązania połączenia modemowego spod innego niż zwykle numeru telefonu, należałoby uznać takie próby za nietypowe i przeprowadzić dodatkowe śledztwo. W rozległych sieciach korporacyjnych, łączących działy rozproszone po całym kraju (albo nawet świecie), systemy wykrywania włamań monitorują często geograficzne położenie węzła, z którego następuje próba logowania. Zmiana typowej lokalizacji jest wtedy przesłanką ataku. Weźmy kolejny przykład, typowy dla finansów. Otóż operacje takie jak wprowadzanie danych przelewów i potwierdzanie przelewów muszą być inicjowane z różnych lokacji. Rozdział funkcji poszczególnych pracowników ma tu zapobiegać nadużyciom uprawnień przez pojedynczych pracowników. Wykonywanie obu tych operacji z tego samego węzła może być przesłanką nadużycia.

## Zasoby systemowe

Przesłankami ataku mogą być też charakterystyki poszczególnych zasobów systemowych. Jeśli, na przykład, obciążenie procesora jest znacząco różne od rejestrowanego w ostatnim czasie obciążenia średniego, można podejrzewać realizowanie na węźle nieuprawnionych operacji. Zwiększone obciążenie procesora może nastąpić na przykład w wyniku nieprawidłowego działania systemu operacyjnego albo aplikacji użytkowych, albo, na przykład, w wyniku realizacji na danym węźle ataku na hasło metodą siłową. Wśród innych charakterystyk zasobów systemowych, których monitorowanie daje istotne z punktu widzenia wykrywania włamań informacje, należy wymienić zajętość pamięci, częstotliwość odwołań do dysku twardego i rozmiar transferowanych danych, częstotliwość odwołań do portów komunikacyjnych itp. Jeśli system niespodziewanie doświadczy znaczącego ubytku wolnej przestrzeni dyskowej, można podejrzewać, że np. do systemu podrzucono tzw. „bombę”. Bomby to pliki, które spakowane mają niewielkie rozmiary, ale których rozpakowanie powoduje zawłaszczenie znacznej ilości miejsca, na przykład kilku gigabajtów (tego typu „dowcipy” popularne były w sieci FIDO na przełomie lat osiemdziesiątych i dziewięćdziesiątych).

## Profile użytkowników i systemu

Zastosowanie profili to bardziej ogólne podejście do analizy odwołań do zasobów systemowych. Profile użytkowników i systemu charakteryzują właśnie te odwołania, jednak w ramach profili uwzględnia się dodatkowe parametry szczegółowo charakteryzujące użytkownika, proces czy węzeł. Wśród tych parametrów wyróżnić można, na przykład, obciążenie maksymalne i minimalne, czas trwania typowej sesji, regularne czasy logowania i wylogowania użytkownika i tak dalej. Odchylenia wartości tych zebranych w profilu parametrów jest przesłanką nienormalnej działalności systemu czy użytkownika.

Szczególnym przypadkiem przesłanki ataku może być kontrola usług, do których dany podmiot (użytkownik, proces itd.) odwołuje się najczęściej w swej regularnej, codziennej aktywności. Jeśli, na przykład, dany pracownik wykorzystuje do pracy sieć internet, to rejestrując przebieg jego pracy w ciągu, powiedzmy, miesiąca, można skonstruować wykaz najczęściej wykorzystywanych serwerów WWW i śledzić te odwołania, które kierowane są do serwerów spoza listy. Odwołania te mogą być przejawami naruszeń reguł bezpieczeństwa. Jako przypadki naruszenia bezpieczeństwa można też klasyfikować odwołania do konkretnych plików otrzymywanie poczty od określonych nadawców i jej wysyłanie do określonych odbiorców, korzystanie z pewnych usług (jak FTP czy Telnet) i inne działania, które stanowią odstępstwo od codziennej „rutyny” obserwowanego podmiotu.

### Pozostałe parametry

Potencjalne naruszenie reguł bezpieczeństwa można rzecz jasna wykrywać również, obserwując inne parametry. Wróćmy do omawianej już procedury wprowadzania (zlecenia) i realizacji przelewów. Jeśli, na przykład, pracownik, nadużywając swojej pozycji, realizuje obie operacje, korzystając z tego samego konta użytkownika, czy choćby tylko z tego samego węzła, można podejrzewać nadużycie. Zgodnie z regułami bezpieczeństwa akceptacja przelewu powinna być bowiem dokonywana przez podmiot inny od wystawiającego zlecenie przelewu.



#### Atak na system głosowania

W maju roku 2002 przedstawiciele korporacji Vivendi Universal podejrzewali, że elektroniczny, bezprzewodowy system głosowania wykorzystywany podczas zgromadzeń akcjonariuszy został przejęty przez włamywacza. Przesłanką ataku był wyjątkowo wysoki odsetek wstrzymujących się od głosu podczas jednego ze zgromadzeń odnośnie wypłaty dywidend.

Wykrycie konkretnej czynności realizowanej na konto byłego pracownika (którego konto nie zostało z jakichś przyczyn usunięte) albo pracownika, który aktualnie przebywa na urlopie, również należy traktować jako przesłankę naruszenia reguł bezpieczeństwa.

### Nieoczekiwane problemy

Wszelkie problemy pojawiające się w toku działania sieci korporacyjnej powinny stanowić asumpt do podjęcia dodatkowego śledztwa. Jeśli nawet w czasie takiego śledztwa okaże się, że przyczyna problemu nie ma związku z bezpieczeństwem, fakt wykrycia samej przyczyny będzie niewątpliwie korzystnie wpływał na niezawodność działania systemu. W kategorii nieoczekiwanych problemów należy wyróżnić:

- ◆ **Problemy ze sprzętem i oprogramowaniem.** Awarie routerów, przeładowanie serwera czy niemożność uruchomienia jednej z usług systemowych mogą świadczyć o próbie realizacji ataku odmowy obsługi.

- ♦ **Nieoczekiwane zachowania użytkowników.** Niespodziewane odwołania do nietypowych zasobów, do których dany użytkownik nigdy przedtem się nie odwoływał, mogą świadczyć o przechwyceniu albo złamaniu hasła konta uprawnionego użytkownika przez intruza i próbie realizacji odwołań do wrażliwych danych przedsiębiorstwa w imieniu tego użytkownika.



#### Kukułcze jajo

Klasyycznym przykładem wykrycia włamania jest przypadek opisany w [Stoll1-96] przez Clifforda Stolla, specjalisty informatyka i zawodowego astronoma, który dość przypadkowo wykrył niewielką niezgodność w systemie rozliczania czasu pracy systemu. Drobiazgowa analiza ujawniła, że jakiś nieznany użytkownik „pożyczył” na kilka sekund czas maszynowy, nie płacąc za niego (suma była niewielka, około jednego centa). Włamywacza udało się wykryć, po czym okazało się, że był odpowiedzialny również za włamania do kilkunastu ściśle tajnych systemów Pentagonu.

### Komunikaty powitalne usług

Analizą komunikatów powitalnych usług zajmują się specjalne programy, które zostaną omówione nieco później. Otóż większość usług sieciowych rozpoczyna sesję komunikacyjną z klientem od zaprezentowania tzw. banera, będącego zwykle formą komunikatu powitalnego. Analiza informacji przekazywanych takimi banerami (w tym numery wersji oprogramowania) powala niekiedy na stwierdzenie, czy dana usługa jest podatna na ataki określonego rodzaju.

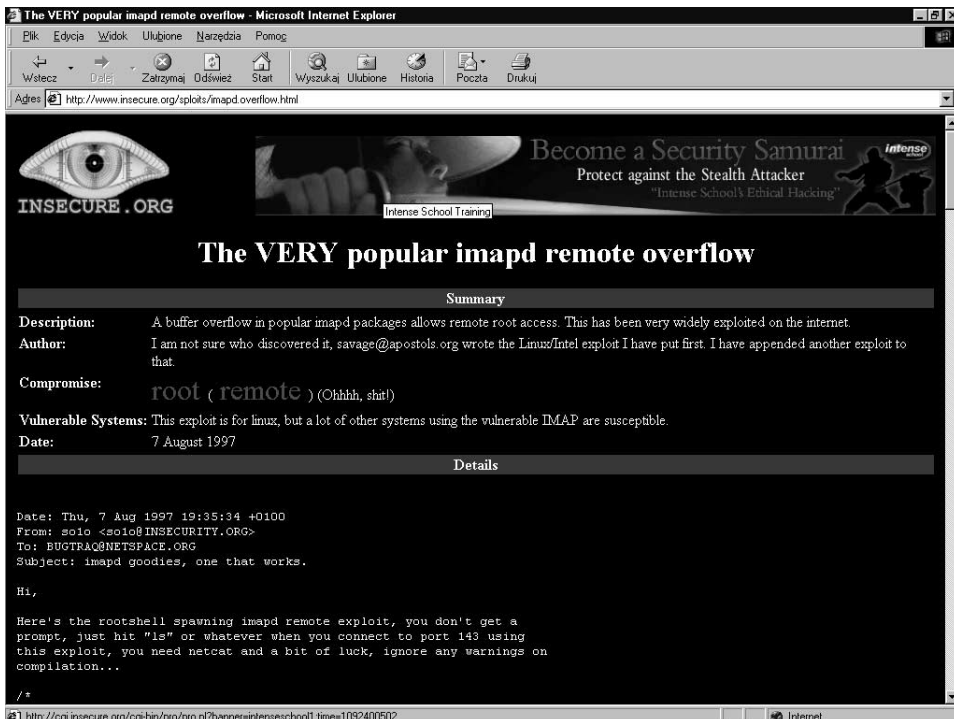
Analiza taka pozwala najczęściej określić numer wersji oprogramowania implementującego sondowaną usługę czy rodzaj systemu operacyjnego, co pozwala na zainicjowanie poszukiwań luk w oprogramowaniu systemu i usługi. Na przykład próba nawiązania połączenia Telnet z portem o numerze 143 (port usługi IMAP) może dać efekt w postaci wyświetlenia standardowego komunikatu powitalnego zawierającego informacje o wersji i producencie oprogramowania usługi IMAP, jak na listingu 4.43.

#### Listing 4.43. Analiza komunikatu zwróconego przez usługę IMAP

```
Trying x.x.x.x...
Connected to bank.ru
* OK bank IMAP4rev1 Service 9.0(157) at Wed, 14 Oct 1998 11:51:50 -0400
(EDT)
(Report problems in this server to MRC@CAC.Washington.EDU)
. logout
* BYE bank IMAP4rev1 server terminating connection
. OK LOGOUT completed
Connection closed by foreign host.
```

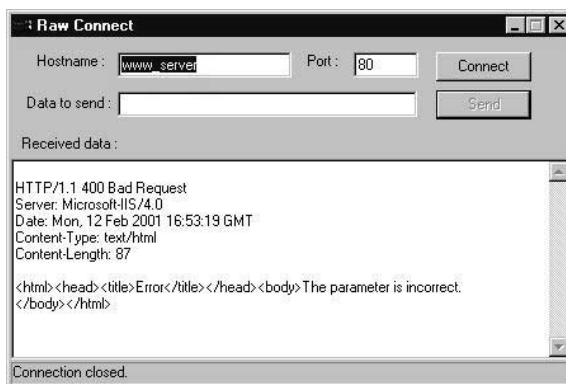
Powyższe informacje, choć na pierwszy rzut oka mało znaczące i nie bardzo przydatne, pozwalają na podjęcie poszukiwań w sieci WWW informacji o wszelkich znanych lukach w oprogramowaniu usługi IMAP; efekt krótkotrwałych poszukiwań ilustruje rysunek 4.6.

W podobny sposób można analizować podatność oprogramowania innych usług, jak choćby serwerów WWW (rysunek 4.7).



Rysunek 4.6. Poszukiwanie informacji o lukach w oprogramowaniu usługi IMAP na witrynie Insecure

Rysunek 4.7.  
Analiza nagłówka  
zwracanego przez  
serwer WWW



### Przykład praktyczny

Onegdaj miałem okazję mieć do czynienia z unikalnym koniem trojańskim (niewykrywanym przez oprogramowanie antywirusowe). Zgodnie z planem autora, program konia trojańskiego miał niepostrzeżenie wykradać hasła przesyłane do sieci internet. Autor nie wziął jednak pod uwagę możliwości nawiązywania połączenia z siecią internet za pośrednictwem sieci lokalnej LAN (koń trojański obsługiwał tylko połączenia dial-up). Uruchomiony na specjalnie do tego celu przygotowanym komputerze koń trojański bezskutecznie próbował odnaleźć usługę RAS. Ponieważ nie była ona zainstalowana, program konia trojańskiego wyświetlał komunikat o błędzie.

### Cyfrowy „odcisk”

Metoda porównywania cyfrowych odcisków aplikacji również ma zastosowanie w skanerach bezpieczeństwa. Polega ona na porównaniu fragmentu kodu wykonywalnego aplikacji z zadaniem wzorcem wersji tejże aplikacji, o której wiadomo, że zawiera pewne luki w zabezpieczeniach. Podobną metodę wykorzystują programy antywirusowe, porównując fragmenty plików wykonywalnych z sygnaturami wirusów. W odmianach tej metody przesłanką ataku może być również specyficzna data utworzenia pliku wykonywalnego czy rozmaite sumy kontrolne. Tę samą metodę stosuje się też do sprawdzania, czy plik nie został w sposób nieautoryzowany podmieniony — wskazuje na to różnica sum kontrolnych.

## Źródła informacji o atakach

Zanim na podstawie wymienionych przesłanek wyciągnie się wnioski o naruszeniu zasad bezpieczeństwa systemu, należałoby wiedzieć, gdzie szukać informacji, na podstawie których moglibyśmy taki wniosek wysnuć. Należy wyróżnić dwie kategorie źródeł takich informacji: źródła główne i źródła dodatkowe. Do źródeł głównych zaliczamy ruch sieciowy, pliki dziennika i informacje o bieżących stanach obiektów systemowych (użytkownikach, procesach, programach, dyskach itd.). Tradycyjne systemy wykrywania włamań korzystają głównie z tych właśnie zasobów. Skanery bezpieczeństwa, które służą nie tyle do wykrywania włamań, co do określania możliwości przeprowadzenia włamania, potrafią korzystać również ze źródeł informacji dodatkowych. Pozyскуją one informacje, analizując oprogramowanie (w tym oprogramowanie implementujące usługi sieciowe), procesy i pliki danych. Dodatkowe źródła informacji o rozmaitych przesłankach ataku to również informacje pozyskiwane od użytkowników systemu, z komunikatów publikowanych przez popularne biuletyny itp.

Informacje czerpane z pojedynczego źródła mogą zawierać niewystarczające dowody naruszenia reguł bezpieczeństwa. Różnicowanie źródeł informacji (najlepiej, aby źródła te były od siebie niezależne) zwiększa szansę niezawodnego wykrycia ataku.

### Pliki dzienników

Analiza plików dzienników to podstawowa i najstarsza metoda wykrywania naruszeń zasad bezpieczeństwa. Również współcześnie pozostaje metodą najbardziej efektywną. Cechą szczególną tej metody jest to, że niekiedy analiza plików dziennika (również ręczna) jest jedyną dostępną metodą wykrywania włamań. Większość publikacji na temat bezpieczeństwa systemów komputerowych (w tym słynna „Pomarańczowa księga” ISO 17799) podkreśla znaczenie systemu rejestrującego zdarzenia i wagę informacji zapisywanych przez te systemy w tzw. plikach dzienników. Nie będę w tej książce zajmował się formatami plików dzienników generowanych przez rozmaite programy i urządzenia, ponieważ byłoby to niecelowe. W każdym systemie przyjmowany jest inny format takich plików. Zresztą różnice w zapisie plików dziennika mogą się pojawiać również pomiędzy wersjami tego samego programu. Z tego względu ograniczę omówienie plików dziennika do prezentacji kilku przykładowych fragmentów takich plików wziętych z routera Cisco IOS (listing 4.44), systemu Windows 2000 (rysunek 4.8), zapory sieciowej CheckPoint Firewall-1 (listing 4.45) i serwera WWW Apache (listing 4.46 oraz listing 4.47).



**Listing 4.46.** *Fragment pliku dziennika odwołań (access\_log) serwera Apache*

```
193.56.123.47 - - [04/Apr/1991:16:39:06 -0500] "GET /etc/passwd HTTP/1.0" 404 139
```

**Listing 4.47.** *Fragment pliku dziennika błędów (error\_log) serwera Apache*

```
[Fri Apr 4 16:37:39 1997] HTTPd: access to /export/home/httpd_root/cgi-bin/phf  
failed for 193.56.123.47, reason: script does not exist from -
```

Chciałbym przykłady uzupełnić jedynie małym komentarzem. Zalecałbym uważne przestudiowanie formatów i specyficznych cech plików dzienników wykorzystywanych w danym systemie informatycznym. Zaniedbanie tego może doprowadzić do sytuacji, która niegdyś mi się zdarzyła. Otóż pewien administrator zabezpieczeń przysłał mi wiadomość, w której prosił o pomoc w rozstrzygnięciu sporu pomiędzy nim a działem informatycznym, w którym pracował. Wraz z prośbą otrzymałem fragment pliku dziennika, który — jak podejrzewał nadawca — został „podrasowany” przez administratora sieci. Wykrycie modyfikacji byłoby bez szczegółowej znajomości formatu pliku dziennika niemożliwe.

## Ruch sieciowy

Jednym z najważniejszych źródeł informacji dla systemów wykrywania włamań jest sam ruch sieciowy. Ruch ten tworzą pakiety przesyłane w sieci (zwane też ramkami). Bez zagłębiania się w szczegóły implementacji rozmaitych protokołów sieciowych przyjmijmy, że dla nas podstawową jednostką transmisji w sieci komputerowej będzie **pakiet**, który w przypadku ogólnym składać się będzie z trzech elementów:

- ♦ nagłówka pakietu (nagłówek zawiera informacje o usłudze, adres źródłowy, adres docelowy i inne pola);
- ♦ pola danych pakietu;
- ♦ „stopki” pakietu (sum kontrolnych i znaczników końca pakietu).

Nie wszystkie te elementy będą obecne we wszystkich protokołach sieciowych. Ogólnie można jednak przyjąć, że dzięki analizie trzech wymienionych elementów pakietu system wykrywania włamań może stwierdzić wystąpienie przypadku naruszenia zasad bezpieczeństwa. Na początku roku 2000 70 procent wszystkich sieci komputerowych stanowiły sieci oparte na protokołach TCP/IP; ich udział w łącznej liczbie sieci komputerowych stale się zwiększa. Z drugiej strony udział sieci korzystających z innych stosów protokołów (IPX/SPX, SMB/NetBIOS itd.) nie przekracza w najlepszym razie 20 procent. To wyjaśnia, dlaczego popularne systemy wykrywania włamań, wykorzystujące w roli źródła informacji ruch sieciowy, działają niemal wyłącznie w sieciach TCP/IP.

## Działalność podmiotów systemu

To źródło informacji odzwierciedla wszelkie działania realizowane przez obiekty (użytkowników, procesy itd.) kontrolowanego systemu w czasie rzeczywistym. Działalność podmiotów systemu można również analizować na podstawie zapisów plików dziennika. Jednak nie wszystkie zdarzenia zachodzące w kontrolowanym systemie są

rejestrowane. Z tego względu efektywna kontrola powinna zakładać analizę pewnej klasy zdarzeń w czasie rzeczywistym. Wybiegając nieco poza bieżący temat omówienia, powiedziałbym, że jedynie niewielki odsetek dostępnych systemów wykrywania włamań uwzględnia to źródło informacji. Komplikacje związane z implementacją mechanizmu bieżącej analizy zdarzeń systemowych zniechęca programistów; problem jest w istocie złożony, ponieważ wymaga zaprojektowania i utworzenia mechanizmu przechwytyjącego wszelkie wywołania systemowe. Tymczasem systemy wykrywania włamań zwykle nie potrafią podejmować bezzwłocznie czynności zmierzających do zablokowania pewnej podejrzanej działalności — najpierw działalność ta musi zostać przeanalizowana, a dopiero potem następuje zainicjowanie ewentualnej reakcji. Im bardziej skomplikowany byłby algorytm takiej analizy, tym większe opóźnienia wprowadzałyby, gdyby był realizowany w czasie rzeczywistym; opóźnienia te w oczywisty sposób zmniejszałyby wydajność kontrolowanego systemu.

## Dodatkowe źródła informacji

Informacje o atakach mogą być pozyskiwane nie tylko za pośrednictwem specjalnych programów, ale również ze źródeł leżących zupełnie poza danym systemem. Większość włamywaczy, zwłaszcza tych, którzy atakują sieci komputerowe celem podniesienia samooceny, chwali się swoimi wyczynami kolegom. Czasem publikują opisy swoich dokonań na stronach WWW. Istnieją nawet specjalne serwisy poświęcone tego rodzaju osiągnięciom (np. w witrynie <http://zone-h.org>). Informacje te można wykorzystać do wzbogacenia swojej wiedzy o atakach i lukach. Oto inne źródła, w których warto poszukać informacji o atakach:

- ◆ magazyny hakerskie (publikowane w sieci internet);
- ◆ listy dystrybucyjne poczty elektronicznej (jak te prowadzone w ramach witryny SecurityFocus);
- ◆ książki;
- ◆ repozytoria organizacji monitorujących doniesienia o atakach i lukach (<http://www.securityfocus.com> czy <http://packetstormsecurity.nl>);
- ◆ grupy dyskusyjne i konferencyjne sieci USENET i FIDONET;
- ◆ kanały IRC;
- ◆ konferencje i seminaria (jak DEFCON).

## Sygnaly od użytkowników

Dodatkowymi źródłami informacji, których nie sposób zlekceważyć, są niewątpliwie sygnały od użytkowników systemu. Choć otrzymywane tą drogą doniesienia są często wytworami wyobraźni niewykwalifikowanych odpowiednio osób, niekiedy mogą okazać się przydatne w wykryciu problemów, które w inny sposób nie zostałyby wcale wychwycone. Weźmy na przykład choćby doniesienie o pojawieniu się w sieci zdublowanego adresu IP. Jeśli nie istnieje możliwość natychmiastowego reagowania na tego rodzaju skargi i doniesienia, należałoby je przynajmniej pieczołowicie rejestrować. Zapiski owe mogą w przyszłości wspomóc ewentualne dochodzenie w sprawie naruszenia zasad bezpieczeństwa.

## Listy dystrybucyjne poczty elektronicznej

Listy dystrybucyjne poczty elektronicznej to popularne na całym świecie fora wymiany informacji w grupach zainteresowań. Koszt korzystania z takich źródeł informacji równy jest jedynie kosztowi dostępu do internetu — powiadomienia o nowych lukach i zagrożeniach są za pośrednictwem takich list nieodpłatnie rozprowadzane pomiędzy wszystkimi zainteresowanymi.

Tego rodzaju biuletyny są zwykle prowadzone przez tzw. zespoły szybkiego reagowania (ang. *response teams*). Najstynniejsze biuletyny prowadzone są przez Computer Emergency Response Team (CERT/CC), SecurityFocus, X-Force oraz SecuriTeam.

Działalność zespołów szybkiego reagowania polega na zbieraniu doniesień o atakach i lukach w zabezpieczeniach sprzętu i oprogramowania, opracowywaniu stosownych metod przeciwdziałania takim atakom i eliminowaniu wykrytych luk oraz publikowaniu pozyskanych informacji za pośrednictwem witryn WWW i właśnie list dystrybucyjnych poczty elektronicznej. Z reguły członkowie rzeczonych zespołów współpracują ściśle z programistami systemów wykrywania włamań i producentami oprogramowania i sprzętu, co zapewnia im możliwości odpowiednio szybkiego opracowywania łat i pakietów naprawczych blokujących możliwość wykorzystania wykrywanych luk. Informacje o właśnie wykrytych atakach i lukach nie są publikowane od razu — najpierw daje się producentom i zespołom programistów czas na opracowanie stosownych łat. Nieaktualność to podstawowa — i chyba niedająca się wyeliminować — wada informacji pozyskiwanych z takich źródeł — zainteresowani włamywacze mają zwykle do takich informacji dostęp nieco wcześniej, zwłaszcza kiedy luki wyszukują sami — dysponują wtedy pewną przewagą.

## Serwisy WWW

Serwisy WWW to jedne z najpopularniejszych i najłatwiej dostępnych źródeł informacji o nowych atakach i lukach. Źródła te można podzielić następująco:

- ♦ Serwisy wspomagane przez producentów oprogramowania i sprzętu — tutaj należy wymienić wszelkiego rodzaju witryny pomocy technicznej firm takich jak Microsoft, Novell, Sun, Hewlett-Packard, Cisco itd. Ich witryny zawierają zwykle informacje wyłącznie o lukach wykrytych w produktach tychże firm.
- ♦ Serwisy wyspecjalizowanych organizacji i firm — serwisy prowadzone bądź wspomagane przez firmy takie jak ISS, Symantec, MITRE wraz z serwisami organizacji takich jak NASA, uniwersytet Purdue itd.
- ♦ Serwisy niezależne — jak choćby Insecure (<http://www.insecure.org>). Serwisy tego rodzaju pojawiają się w sieci dziesiątkami; można je wyszukiwać za pośrednictwem specjalnych wyszukiwarek, jak <http://neworder.box.sk>.

## Konferencje internetowe

Wraz z rozwojem i rosnącą popularnością technologii wymiany informacji pozwalających na bardzo łatwe publikowanie i rozpowszechnianie dowolnych informacji, jak serwisy WWW i listy dystrybucyjne poczty elektronicznej, fora takie jak grupy dyskusyjne

sieci USENET czy konferencje FIDONET stopniowo tracą na popularności. Tymczasem kilka lat temu popularność sieci FIDO była porównywalna z popularnością sieci internet. Wraz z obniżaniem kosztów połączenia z siecią internet liczba uczestników konferencji FIDO drastycznie maleje. Dziś znaczenie FIDO jest raczej marginalne.

## Technologie wykrywania włamań

Skuteczne wykrywanie włamań wymaga spełnienia jednego z dwóch warunków: znajomości oczekiwanego działania obiektów kontrolowanego systemu bądź znajomości wszelkich możliwych ataków i ich odmian. W pierwszym przypadku wykrywanie włamań opiera się na wykrywaniu anomalii, czyli odstępstw od oczekiwanego zachowania obiektów kontrolowanego systemu; w drugim przypadku stosuje się metodę obserwacji znamion znanych ataków. W komercyjnych systemach wykrywania włamań implementowane są zwykle obie metody — pozwala to na połączenie ich zalet i wyeliminowanie najważniejszych wad.

### Wykrywanie anomalii

Wykrywanie anomalii opierane jest na założeniu, że ataki i wszelkie inne szkodliwe działania przejawiają się w niezwykłym bądź nieoczekiwanym zachowaniu systemu, programów i użytkowników. Na przykład nadzwyczaj duża liczba połączeń w krótkim czasie, wyższe niż przeciętne obciążenie procesorów i łącza czy próby odwołań do urządzeń zewnętrznych, które w codziennej pracy systemu nie są wykorzystywane — wszystko to to przejawy anormalnego zachowania systemu. Jeśli uda się opisać profil normalnego zachowania obiektów systemowych, wtedy wszelkie odchylenia parametrów uwzględnianych w profilu należy interpretować jako anormalne zachowanie obiektu. Nie musi ono od razu oznaczać ataku. Dotyczy to, na przykład, znacznej liczby odpowiedzi na żądania systemu zarządzającego siecią. Większość systemów wykrywania włamań taką aktywność zarejestrowałaby jako próbę ataku odmowy obsługi. Tymczasem zwiększona aktywność może być co prawda anormalna, ale zupełnie uprawniona, np. czynnościami konserwacyjnymi. Z tego względu w metodzie wykrywania anomalii należy uwzględniać dwie skrajności:

- ◆ **falszywe alarmy** (ang. *false positives*) — przypadki, kiedy system wykrywania włamań wykrył i zaliczył do ataków anomalie, która w istocie nie jest przejawem ataku;
- ◆ **falszywe pominięcia** (ang. *false negatives*) — przypadki, kiedy system wykrywania włamań nie wykrywa odchyłeń wystarczających do uznania zachowania systemu za anormalne, choć w rzeczywistości odchylenia te są efektem ataku. Ten przypadek jest oczywiście dużo groźniejszy od fałszywego alarmu.

W ramach dostosowywania systemu wykrywania włamań opartego na wykrywaniu anomalii do danego systemu administrator powinien przede wszystkim:

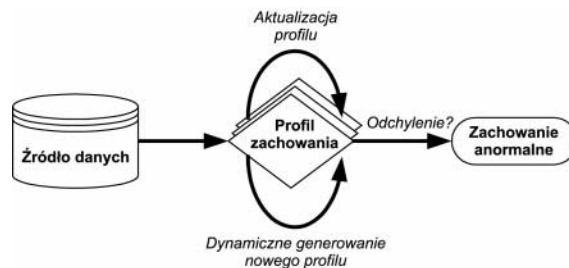
- ♦ **Utworzyć profil zachowania obiektu.** Zadanie to może być dość złożone, przede wszystkim jako trudne do sformułowania i czasochłonne. Wymaga też znacznego zaangażowania i wykonania znacznej liczby czynności przygotowawczych (w tym obserwacji stanów obiektu podczas niezakłóconego działania systemu).
- ♦ **Określić progowe wartości odchyień parametrów wchodzących w skład profilu.** Odpowiednie ustalenie owych wartości pozwala na zminimalizowanie liczby fałszywych alarmów i fałszywych pominięć.

Technika wykrywania anomalii jest w implementacji dość kosztowna, nie tylko z uwagi na konieczność wytypowania znacznej liczby parametrów i ustalenia wartości progowych na etapie wdrażania systemu. Również jego działalność jest kosztowna w tym sensie, że ciągle rejestrowanie bieżących wartości parametrów obiektów systemu może znacząco zmniejszyć wydajność systemu informatycznego. Zwykle systemy takie mocno obciążają procesory i wymagają do przechowywania gromadzonych danych znacznych przestrzeni dyskowych. Systemy wykrywania włamań oparte na detekcji anomalii nie sprawdzają się zwłaszcza w systemach czasu rzeczywistego, w których ze względu na ścisłe ograniczenia czasowe czas reakcji systemu ma znaczenie zasadnicze.

Weźmy, na przykład, system bankowy, w ramach którego codziennie realizowane są te same czynności. W takich mocno deterministycznych systemach metoda wykrywania anomalii może być stosunkowo łatwo wdrożona. Jednak w wielu innych środowiskach wykrywanie anomalii nie jest metodą odpowiednią. Do takich środowisk należałoby zaliczyć sieci uczelniane czy serwery WWW, z których za pośrednictwem pojedynczego konta systemowego korzystają tysiące użytkowników zewnętrznych.

Schemat typowego systemu wykrywania włamań opartego na wykrywaniu anomalii prezentowany jest na rysunku 4.9.

**Rysunek 4.9.**  
*Typowy system wykrywania włamań na bazie anomalii*



W systemach wykrywania anomalii głównym źródłem informacji są pliki dzienników oraz bieżąca działalność użytkowników. Istnieją jednak takie systemy bazujące na wykrywaniu anomalii, które analizują ruch sieciowy (jak w technologii sygnatur ruchu zaimplementowanej w systemie nGenius Performance Management System firmy NetScout [NetScout1-02]).

Ważne jest, aby uświadomić sobie, że we współczesnych systemach wykrywania włamań podejście oparte na wykrywaniu anomalii jest coraz powszechniejsze. Mechanizm ten stosuje też większość programistów zabezpieczeń. Szczególną popularnością (z racji skuteczności wykrywania ataków odmowy obsługi i rozproszonych ataków odmowy obsługi) cieszy się metoda monitorowania obciążenia sieci transmisjami.



### Wykrywanie anomalii w Departamencie Obrony

W kwietniu 2001 roku Departament Obrony Stanów Zjednoczonych podpisał z amerykańskim instytutem badawczym opiewający na 20 milionów dolarów kontrakt, w ramach którego ów instytut (American Institute for Research) miał opracować system zapobiegania włamaniom AIPS (*Advanced Intrusion Prevention System*) w oparciu właśnie o metodę wykrywania anomalii. System wciąż jest w budowie — ma zostać uruchomiony w grudniu 2004 roku.

## Wykrywanie podejrzanych działań

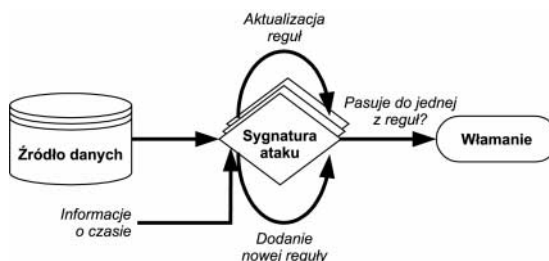
Druga metoda wykorzystywana w wykrywaniu włamań bazuje na wykrywaniu nieprawidłowości. Polega ona na poszukiwaniu w kontrolowanym obszarze cech znanych ataków; ataki opisywane są wzorcami albo sygnaturami. Technologia ta przypomina nieheurystyczne oprogramowanie antywirusowe — zresztą programy antywirusowe można by uznać za najprostsze systemy wykrywania włamań. W tej metodzie system wykrywania włamań zdolny jest jednak jedynie do wykrycia znanych ataków, nie ma natomiast możliwości wykrywania ataków nowych, których sygnatury jeszcze nie są znane.

Działanie takich systemów jest zwykle dość proste. Metoda wykrywania nieprawidłowości stosowana jest powszechnie we współczesnych, dostępnych na rynku systemach wykrywania włamań. Administratorzy mający obsługiwać takie systemy napotykają jednak trudności. Pierwsza z nich tkwi w konieczności opisu ataku sygnaturą, co zwykle wymaga zastosowania pewnego (najczęściej bardzo uproszczonego) języka opisu ataku. Trudność druga wynika z pierwszej i wyraża się następującym pytaniem: „Jak opisać znany atak tak, aby jego sygnatura pozwalała również na wykrycie wszelkich możliwych odmian tego ataku?”.

Działanie typowego systemu wykrywania włamań bazującego na wykrywaniu nieprawidłowości ilustruje schemat z rysunku 4.10.

### Rysunek 4.10.

Typowy system wykrywania nieprawidłowości



Podstawowymi źródłami danych dla systemów wykrywania nieprawidłowości są zwykle pliki dzienników i ruch sieciowy.

## Wykrywanie włamań — dwa podejścia

W następnych punktach chciałbym przedstawić dwa główne podejścia do wykrywania włamań: statystyczne i eksperckie. Chciałbym również przedstawić Czytelnikowi perspektywę rozwoju systemów wykrywania włamań, w tym raczkujące na razie technologie zakładające zaangażowanie sieci neuronowych, algorytmów genetycznych itp.

## Analiza statystyczna

Podejście statystyczne jest szeroko wykorzystywane w wykrywaniu anomalii. Tam bowiem właśnie odchylenie pewnego parametru systemu od statystycznie wyznaczonej normy jest przesłanką ataku. Dla każdego normalnego zachowania obiektu w systemie wyznaczane są średnie częstotliwości występowania i wartości pewnych parametrów (np. liczba zalogowanych użytkowników, liczba zdarzeń odmowy odstepu do zasobu, rozmaite czasy itp.). System informuje o prawdopodobnym ataku, kiedy wartości obserwowane nie mieszczą się w wyznaczonej normie, na przykład przekraczają pewne wartości progowe. Analiza statystyczna sprawdza się w wykrywaniu takich anormalnych przejawów działania systemu jak, na przykład, zalogowanie się uprawnionego użytkownika w niezwykłym dla niego czasie (na przykład między godziną szóstą a ósmą rano).

Parametry tworzące wzorzec zachowania można podzielić na następujące kategorie:

- ♦ parametry liczbowe (ilość danych transmitowanych przy użyciu rozmaitych protokołów, obciążenie procesora, liczba wykorzystywanych plików itd.);
- ♦ parametry rodzajowe (nazwy plików, polecenia użytkownika, otwarte porty itd.);
- ♦ parametry aktywności (liczba odwołań do pliku czy liczba połączeń w określonym przedziale czasu).

Szczególne znaczenie ma w podejściu statystycznym odpowiedni wybór parametrów kontrolowanych przez system wykrywania włamań. Zbyt ubogi zestaw parametrów, ewentualnie zestaw parametrów źle dobranych, może uniemożliwić pełne modelowanie zachowania obiektów systemu. To z kolei oznacza, że część ataków (a raczej ich przejawów) nie będzie w ogóle rozpoznawana. Z drugiej strony, nadmierna liczba monitorowanych parametrów skutecznie zmniejszy wydajność węzła i zwiększy zapotrzebowanie na zasoby systemowe (pamięć, przestrzeń dyskową, czas procesora itd.).

Metody statystyczne, choć stosunkowo efektywne i dość skuteczne w wykrywaniu niektórych typów ataków, nie są na dzień dzisiejszy szeroko stosowane właśnie z uwagi na wymienione wyżej wady. Dodatkowa trudność w stosowaniu systemów opartych na analizie statystycznej tkwi w odpowiednim dobraniu wartości progowych. Jeśli próg „normalności” ustawiony jest zbyt wysoko, wiele ataków pozostanie niezauważonych; jeśli zaś próg będzie zbyt niski, istnieje ryzyko, że również zwykłe czynności uprawnionych użytkowników będą uznawane za ataki. W niektórych systemach wykrywania włamań, jak choćby w RealSecure Network Sensor, użytkownik może dostosowywać wartości progowe do niektórych znanych rodzajów ataków. Mimo to zadanie ustalenia wartości progowych systemu wykrywania anomalii jest zadaniem trudnym i wymagającym szczegółowej wiedzy o kontrolowanym systemie. Inne wady systemów statystycznych wymienione zostały w tabeli 4.8.

**Tabela 4.8.** *Zalety i wady metod statystycznych w wykrywaniu włamań*

Zalety	Wady
Systemy oparte na analizie statystycznej potrafią wykrywać również nieznanne wcześniej ataki.	Włamywacze mogą mylić system wykrywania włamań, imitując swoim działaniem zwykle działania uprawnionych użytkowników systemu i wprowadzając stopniowe zmiany do profilu „normalnego” zachowania systemu — jest to możliwe dzięki „adaptacyjnym” możliwościom systemu statystycznego.
Metody statystyczne pozwalają na wykrywanie takich ataków, których przejawów nie da się prosto opisać wartościami pojedynczych parametrów (czego wymagają sygnatury ataków stosowane w wykrywaniu nieprawidłowości — <i>przykład</i> <i>tłum.</i> ).	Przy wykorzystaniu metod statystycznych zachodzi znacznie większe prawdopodobieństwo zgłaszania fałszywych alarmów niż w przypadku pozostałych metod wykrywania włamań.
Systemy statystyczne mogą być adaptowane do zmieniającego się zachowania systemu.	Metody statystyczne nie dają prawidłowych wyników, kiedy dochodzi do nagłych zmian w profilu zachowania systemu albo jego poszczególnych obiektów (kiedy, na przykład, szef danego działu wykonuje chwilowo obowiązki jednego ze swoich nieobecnych pracowników). Wada ta ma wielkie znaczenie w organizacjach, w których do takich zmian dochodzi stosunkowo często. Wraz ze zmianą, zanim system nie dostosuje się do nowego profilu zachowania, pojawia się szereg fałszywych alarmów; istnieje też ryzyko fałszywych pominięć.
	Metod statystycznych nie można stosować w wykrywaniu ataków realizowanych przez podmioty, których zachowania nie da się ująć w spójnym profilu zachowania (np. użytkowników o bardzo szerokich uprawnieniach, np. kontrolerów).
	Metody statystyczne mogą być nieodpowiednie do wykrywania ataków realizowanych przez podmiot, który od początku działania systemu (a więc również w fazie ustalania profilu zachowania „normalnego”) realizował nieuprawnione czynności — profil takiego podmiotu uwzględnia bowiem ataki jako normę.
	Metody statystyczne wymagają wstępnego dostosowania do cech danego systemu (w tym określenia wartości progowych poszczególnych parametrów), co jest zadaniem czasochłonnym i żmudnym.
	Metody statystyczne nie uwzględniają kolejności zdarzeń.

## Systemy eksperckie

Inaczej niż w systemach wykrywania anomalii, w których monitoruje się wartości pewnych parametrów w oczekiwaniu na przekroczenie przez nie wartości progowych, w wykrywaniu nieprawidłowości bazuje się na regułach opisujących scenariusz ataku. Mechanizm wykrywania nieprawidłowości identyfikuje potencjalne ataki, jeśli działalność użytkownika spowoduje zjawiska pasujące do jednej z reguł opisujących ataki. Podstawową cechą systemu wykrywania włamań opartego na wykrywaniu nieprawidłowości jest dostępność bazy danych znanych ataków. System ekspercki to taki system, który na bazie dostępnych reguł klasyfikuje zdarzenia jako ataki. Reguły opisujące ataki tworzone są na podstawie praktycznych doświadczeń specjalistów od zabezpieczeń systemów komputerowych; baza danych tych reguł nosi w systemie eksperckim miano bazy wiedzy. W zdecydowanej większości przypadków reguły zawierają tak zwane sygnatury ataków; system wykrywania włamań przeszukuje kontrolowany obszar pod kątem występowania zjawisk opisywanych sygnaturami.

Same sygnatury to wzorce cech ataków dopasowywane do obserwowanych na bieżąco zjawisk. Sygnatury mogą być i bardzo proste (np. ciąg znaków wyszukiwany w polu cenniku) i bardzo złożone (zmiana stanu bezpieczeństwa wyrażona matematycznie, wyrażona sekwencją określonych czynności albo zestawem wpisów plików dziennika).

Analiza sygnaturowa polega na dopasowaniu parametrów systemu i jego obiektów oraz ruchu sieciowego do bazy danych ataków. Większość komercyjnych systemów wykrywania włamań realizuje analizę sygnaturową w oparciu o bazę danych ataków opracowywaną przez producenta danego systemu. Klient może zwykle instalować dodatkowe, własne sygnatury w ramach konfiguracji i konserwacji systemu wykrywania włamań.

Metody eksperckie są co prawda wykorzystywane głównie w systemach wykrywania włamań opartych na wykrywaniu nieprawidłowości, ale system ekspercki da się też zastosować w wykrywaniu anomalii. Przykładem jest metoda predykcyjnego generowania wzorca, zakładająca, że przyszłe zdarzenia da się prognozować na podstawie zdarzeń już zaobserwowanych. Reguły takiego systemu eksperckiego mogą być zapisywane następująco:

$$\Pi_1 \rightarrow \Pi_2 \Rightarrow (\Pi_3 = 75\%, \Pi_4 = 20\%, \Pi_5 = 5\%)$$

Zapis ten oznacza, że jeśli po zdarzeniu  $\Pi_1$  nastąpi zdarzenie  $\Pi_2$ , to następnym zdarzeniem będzie — z prawdopodobieństwem 75 procent — zdarzenie  $\Pi_3$ ; prawdopodobieństwo, że trzecim zdarzeniem będzie  $\Pi_4$  wynosi jedynie 20 procent, a prawdopodobieństwo zajścia po  $\Pi_1$  i  $\Pi_2$  zdarzenia  $\Pi_5$  — jedynie 5 procent. Łatwo jednak dostrzec wady zastosowania systemów eksperckich. Jeśli, na przykład, pewien atak nie jest reprezentowany w bazie wiedzy systemu eksperckiego, jego wykrycie przez ten system jest niemożliwe. Wadę tę można wyeliminować, definiując wszystkie zdarzenia przebiegające według nieznanego scenariusza jako ataki (ale to zaowocuje znaczną liczbą fałszywych alarmów) albo jako zdarzenia normalne (wtedy system będzie obciążony znaczną liczbą błędów fałszywych pominięć) — oba rozwiązania są dalekie od ideału.

Siedemdziesiąt procent współczesnych systemów wykrywania włamań bazuje na metodach wyprowadzonych z metody eksperckiej, około 30 procent opiera się na metodach statystycznych. Wadą tak szeroko rozpowszechnionych systemów eksperckich jest konieczność ciągłej aktualizacji bazy wiedzy. Aktualizacje te nie zawsze dają się przeprowadzić automatycznie. Zwykle są albo instalowane ręcznie, albo po prostu ignorowane przez administratorów systemów. Nieaktualność bazy wiedzy redukuje skuteczność systemu eksperckiego [Cannady1-98], a w najgorszym przypadku brak odpowiedniego zaangażowania ze strony osób odpowiedzialnych za utrzymanie i konserwację systemu może doprowadzić do znacznego obniżenia poziomu bezpieczeństwa całej sieci, tym groźniejszego, że personel pozostaje w miłym poczuciu bezpieczeństwa zapewnianego przez działający przeciw system wykrywania włamań.

Systemy opierające się na wykrywaniu nieprawidłowości nie radzą sobie z wykrywaniem ataków, których przejawy obserwowane są w znacznych odstępach czasu. Wykrycie ataku utrudnia również jego rozproszenie — albo w czasie, albo pomiędzy kilku włamywaczy niepowiązanych ze sobą w wyraźny sposób. Wady i zalety systemów eksperckich w wykrywaniu włamań zebrane zostały w tabeli 4.9.

**Tabela 4.9.** *Zalety i wady metod eksperckich w wykrywaniu włamań*

<b>Zalety</b>	<b>Wady</b>
Prostota implementacji.	Niemożliwość wykrywania nieznanymi jeszcze ataków.
Szybkość działania systemów wykrywania włamań opartych na detekcji nieprawidłowości.	Niewielka skuteczność wykrywania odmian znanych ataków.
Wylimitowanie fałszywych alarmów.	Działanie systemu wykrywania włamań w znacznej mierze zależy od kwalifikacji i zaangażowania ekspertów budujących bazę wiedzy.

## Sieci neuronowe

Zaangażowanie do wykrywania włamań sieci neuronowych to pomysł stosunkowo nowy, a mimo to metoda ta odczekała się już implementacji w niektórych systemach wykrywania włamań.

Badania i prace nad zastosowaniem sieci neuronowych w dziedzinie wykrywania włamań wciąż trwają. Sieci neuronowe mogą potencjalnie znakomicie wspomóc rozwiązywanie szeregu problemów, z którymi nie radzą sobie współczesne technologie wykrywania włamań. Sieci neuronowe mają być alternatywą dla analizy statystycznej w systemach wykrywania anomalii.

Proponuje się zastosowanie sieci neuronowych do identyfikowania typowych cech użytkowników systemu i statystycznie znaczących odchyłeń obserwacji od standardowego trybu działania użytkownika.

Sieci neuronowe mają również wspomóc wykrywanie wirusów. Autorzy niektórych publikacji sugerują nawet, że sieci neuronowe mogą realizować analizę statystyczną pomocną przy wykrywaniu wirusów w systemach komputerowych. Architektura sieci

neuronowych wybrana do tego zadania nosi miano samoorganizującej się mapy (tzw. „sieci Cohonena”) z pojedynczą warstwą neuronów reprezentujących informacje z pojedynczych domen w formie zorganizowanego geometrycznie układu. Sieć taka miałaby analizować cechy normalnej działalności systemu, identyfikując statystyczne odchylenia od wartości normalnych, stanowiących przesłanki obecności wirusa.

Zmienna natura ataków na sieci komputerowe wymaga elastyczności zabezpieczeń i takiego systemu bezpieczeństwa, który zdolny byłby do analizowania ruchu sieciowego o dużym natężeniu za pośrednictwem metod mniej strukturalizowanych niż te stosowane w regułach wykrywających nieprawidłowości. Systemy wykrywania włamań oparte na sieciach neuronowych mogłyby rozwiązywać szereg istotnych problemów obciążających systemy bazujące na regułach. Przykładem systemu wykrywania włamań wykorzystującego sieci neuronowe jest system AUBAD (*Automated User Behavior Anomaly Detection*), opracowany na uniwersytecie w Melbourne w Australii.

Największą minusem zastosowania sieci neuronowych do wykrywania włamań jest to, że sieci neuronowe z natury są pewnego rodzaju „czarnymi skrzynkami”. Inaczej niż w systemach eksperckich, w których od początku dana jest pewna baza wiedzy umożliwiająca rozpoczęcie analizy, sieci neuronowe adaptuje się do działania danego systemu przez „naukę”. Początkowy dobór wag połączeń pomiędzy węzłami w takiej sieci i określenie funkcji przejścia staje się po osiągnięciu przez sieć neuronową dostatecznego poziomu adaptacji zupełnie nieaktualny i choć analiza sieciowa może dać znaczne prawdopodobieństwo skutecznego wykrycia anomalii, prawdopodobieństwa tego nie da się precyzyjnie ustalić. Problem „czarnej skrzynki” objawia się w większości zastosowań sieci neuronowych [Cannady1-98]. Ten aspekt praktycznego zastosowania sieci neuronowych powinien zostać dopiero zbadany. Tymczasem pozwolę sobie podsumować wady i zalety sieci neuronowych w tabeli 4.10.

**Tabela 4.10.** *Zalety i wady sieci neuronowych w wykrywaniu włamań*

<b>Zalety</b>	<b>Wady</b>
Zdolność wykrywania nieznanymi ataków.	Wyniki działania nieoparte wyjaśnieniami.
Możliwość funkcjonowania w środowiskach z dużym poziomem szumów.	Trudność „uczenia” sieci.
Możliwość skutecznego funkcjonowania w obliczu niekompletności albo uszkodzenia danych.	Brak komercyjnych systemów wykrywania włamań bazujących na sieciach neuronowych.
Zdolność prognozowania zachowania użytkowników i nowych ataków.	

## **Podjęcie kombinowane**

Istnieją już implementacje systemów wykrywania włamań, w których problem wykrywania włamań rozwiązywany jest przy użyciu różnych metod. W systemach tych, oprócz trzech metod już wymienionych, stosuje się też inne, które pozwolę sobie pokrótce opisać w kolejnych punktach.

## System NIDES

System wykrywania włamań NIDES, jako następca systemu IDES, to jeden z pierwszych przykładów zastosowania podejścia kombinowanego. W systemie tym, opracowanym w latach 1992 – 1994 na Uniwersytecie Stanford w laboratorium Stanford Research Institute (SRI), połączone zostały metody wykrywania anomalii i nieprawidłowości. Komponent wykrywania anomalii wykorzystuje tam podejście statystyczne, określając odchylenie obserwowanych parametrów od ustalonego wcześniej profilu normalnego zachowania użytkownika; profil składa się z ponad 30 parametrów (opisujących m.in. obciążenie procesora, operacje wejścia-wyjścia, błędy systemowe czy polecenia wydawane przez użytkownika). Profile te podlegają automatycznej aktualizacji, czyli adaptacji do zachowania użytkownika. W ramach komponentu „eksperyckiego” zaimplementowano w systemie NIDES bazę wiedzy o znanych atakach; ataki opisywane są w języku P-BEST. Zaletą takiego połączenia jest zwiększona ogólna skuteczność systemu: ataki niewykryte przez jeden komponent są często wykrywane przez drugi i odwrotnie. Analiza przeprowadzana jest w czasie rzeczywistym. System NIDES różni się od swojego poprzednika, systemu IDES, obecnością specjalnego modułu o nazwie RESOLVER, którego zadaniem jest łączenie danych przekazywanych z komponentów analizy statystycznej i analizy eksperckiej.

## System EMERALD

System wykrywania włamań EMERALD (*Event Monitoring Enabling Responses to Anomalous Live Disturbances*) również został opracowany w instytucie SRI. Powstał jednak nieco później niż NIDES. EMERALD również łączy dwa podejścia do wykrywania włamań; jest przewidziany do zastosowania w dużych, rozległych sieciach korporacyjnych. Cechą szczególną tego systemu jest to, że ma on zdolność do analizowania danych z każdego czujnika z osobna, jak i z dowolnych kombinacji każdej liczby czujników. System może też być integrowany z rozwiązaniami firm trzecich.

## Inne rozwiązania

Ciekawym pomysłem jest propozycja Jamesa Kennedy’ego ze School of Computer and Information Sciences na uniwersytecie Nova Southeastern University w Fort Lauderdale na Florydzie. Kennedy zaproponował system łączący trzy podejścia do wykrywania włamań. Połączył on technologię sieci neuronowych z systemem wykrywania włamań RealSecure Network Sensor firmy ISS. Efekty połączenia okazały się nadzwyczajne. Prawdopodobieństwo wykrycia włamania sięgnęło 98 procent, a liczba błędnych rozpoznań zmniejszyła się do 5 procent.

Rozwiązanie zaproponowane na uniwersytecie Nova Southeastern University dotyczy wykrywania włamań na podstawie analizy ruchu sieciowego; w Austin, na Texas University narodził się pomysł wykrywania ataków na podstawie analizy poleceń wpisywanych przez użytkownika. Zaproponowany mechanizm wykrywania włamań, bazujący na mechanizmie nauczania ze wsteczną propagacją, otrzymał nazwę NNID — *Neural Network Intrusion Detector*. Algorytm działania powstał przy okazji analizy problemu identyfikacji ataku i został wstępnie przetestowany w systemie z dziesięcioma użytkownikami.

System GASSATA (*Genetic Algorithm for Simplified Security Audit Trail Analysis*) to system wykrywania nieprawidłowości opracowany na uniwersytecie Rennes we Francji. System ten powstał jako narzędzie analizy zdarzeń rejestrowanych w pliku dziennika systemu operacyjnego AIX. W roli mechanizmu rejestracji danych zaangażowano w nim algorytm genetyczny.

Kolejny system — AID (*Adaptive Intrusion Detection*) — powstał jako wynik projektu badawczego prowadzonego na Politechnice Brandenburskiej w Niemczech. Projekt ten sponsorowany był przez niemieckie ministerstwo nauki, edukacji i kultury w latach 1994 – 1996. System AID wykorzystuje architekturę klient-serwer. Miał docelowo wykrywać podejrzane działania użytkowników sieci lokalnej. Czujnik systemu AID pobiera dane z systemu operacyjnego, tłumaczy je na własny format i przekazuje do konsoli sterującej, która zajmuje się analizą danych otrzymanych z poszczególnych czujników. Sama analiza odbywa się przy wykorzystaniu systemu eksperckiego czasu rzeczywistego RTworks i wykorzystuje automaty skończone. W pierwszej wersji systemu AID działał pod kontrolą systemu operacyjnego Solaris na platformie SPARC firmy Sun. Twórcy systemu postawili przed sobą następujące cele:

- ♦ Opracowanie agentów na potrzeby analizy nie tylko ruchu sieciowego, ale również działalności w obrębie węzłów.
- ♦ Opracowanie wersji dla Windows NT.
- ♦ Integracja z mechanizmem korzystającym z sieci neuronowych.

System wykrywania włamań NetSTAT to produkt z rodziny oprogramowania STAT powstającej na Uniwersytecie Kalifornijskim w Santa Barbara. System ten miał być działającym w czasie rzeczywistym systemem wykrywania włamań bazującym na tzw. kontroli zmian stanu. Otóż, aby wykonać atak, włamywacz musi sprowokować zmianę stanu systemu — przejście ze stanu początkowego do stanu naruszenia zasad bezpieczeństwa. W przeciwieństwie do innych stanowiskowych systemów wykrywania włamań, które analizują bezpośrednio zapisy plików dziennika, systemy z rodziny STAT dysponują komponentem pośrednim, tzw. analizatorem zapisów zdarzeń. Komponent ów odpowiedzialny jest za przetwarzanie wpisów plików dzienników i ich tłumaczenie na postać sygnatur. Otrzymane tak sygnatury określają przejścia pomiędzy stanami systemu. Przejścia te są następnie analizowane przez właściwy system wykrywania włamań. Z jego punktu widzenia atak stanowi jedno z przejść pomiędzy stanami. Główną zaletą tego podejścia jest zdolność do wykrywania ataku, zanim system znajdzie się w stanie oznaczającym skuteczne włamanie.

Pierwszym systemem z rodziny STAT był USTAT. Miał on wykrywać włamania do węzłów działających pod kontrolą systemu UNIX. Jego następcą, system NSTAT miał już chronić nie pojedyncze stanowiska, ale całe grupy połączonych siecią węzłów. System NetSTAT miał być zaś typowo sieciowym systemem wykrywania włamań.

Niektórzy programiści stosują jeszcze inne podejście do problemu wykrywania włamań. Na przykład brytyjska firma ProCheckUp (<http://www.procheckup.com>) udostępnia usługę ProCheckNet, która to usługa polega na realizacji testu penetracyjnego z użyciem algorytmów sztucznej inteligencji, imitujących działania włamywaczy. Zgodnie z deklaracjami producenta implementacja sztucznej inteligencji pozwala na ominięcie

w ramach testu penetracyjnego znacznej liczby zabezpieczeń. Zupełnie inne podejście obrała firma PROMIA (<http://www.promia.com>); po analizie większości problemów, jakimi obciążone są współczesne systemy wykrywania włamań (w tym problemu znacznej liczby fałszywych alarmów czy niezdolności do wykrywania nieznanymi jeszcze ataków), firma opracowała produkt o nazwie Intelligent Agent Security Module (ISAM) angażujący zarówno tradycyjny mechanizm porównywania sygnatur w wykrywaniu nieprawidłowości, jak i nowe technologie wykrywania anomalii (w tym sieci neuronowe i logikę rozmytą). We wrześniu 2001 roku na wdrożenie tego systemu zdecydowało się centrum SPAWAR (Space and Naval Warfare Systems Command).

## Podsumowanie

Brak matematycznych podstaw technologii wykrywania włamań uniemożliwia badaczom tej dziedziny opracowywanie efektywnych metod wykrywania włamań. Wszystkie istniejące metody zostały wyprowadzone albo na podstawie indywidualnych doświadczeń poszczególnych osób, albo stanowią efekty osiągnięć naukowych w dziedzinach pokrewnych. Istniejące narzędzia i mechanizmy nie mają solidnej bazy naukowej, co nie pozwala ani na zatwierdzanie poszczególnych rozwiązań jako poprawnych, ani na ich negowanie jako niepoprawnych. Mimo pewnych postępów na naukowym zapleczu interesującej nas dziedziny, sytuacja niepewności szybko nie ulegnie zmianie.

W niniejszym rozdziale udało się mimo to opisać trzy podstawowe elementy wszystkich technologii wykrywania włamań. Wiemy już co, gdzie i jak wykrywać. Wiedzę tę w praktyce stosuje się na dwa sposoby. Pierwszy z nich polega na ręcznej analizie dostępnych informacji w poszukiwaniu przesłanek ataku. Zostanie on omówiony w rozdziale 5. Sposób drugi polega na zaangażowaniu do tych samych czynności analitycznych narzędzi automatyzujących analizę. Ta metoda, jak i stosowane w niej narzędzia, opisywane będą w rozdziałach od 6. do 12.