Biuletyn WAT
Vol. LX, Nr 4, 2011

# Optymalizacja pewnego algorytmu znajdowania anomalii w ruchu sieciowym

ADAM E. PATKOWSKI

Wojskowa Akademia Techniczna, Wydział Cybernetyki, Instytut Teleinformatyki,
00-908 Warszawa, ul. S. Kaliskiego 2, aep@ita.wat.edu.pl

**Streszczenie.** Przedmiotem artykułu jest pewna metoda wykrywania anomalii w ruchu sieciowym. Jest to „czysta" metoda histogramowa, której szczególną cechą jest niezwłoczne rozpoznawanie, pozwalające na zablokowanie, ataków sieciowych. Zaprezentowano możliwość optymalizacji profilów ruchu sieciowego za względu na szybkość działania implementacji tej metody.
**Sława kluczowe:** informatyka, sieci komputerowe, ataki sieciowe, anomalie, algorytmy, optymalizacja

## 1. Wstęp

Sposobem na zdalne ataki teleinformatyczne prowadzone za pomocą ruchu sieciowego jest blokowanie tego ruchu za pomocą wyspecjalizowanych urządzeń nazywanych w ogólności filtrami sieciowymi. Najczęściej filtry sieciowe dostrzegane są jako zapory sieciowe (*firewalls*), chociaż modelowi filtra odpowiadają także proste ACLe (*Access Control Lists* — zarówno na routerach, jak i przy dostępie zdalnym do plików), a nawet mechanizmy uwierzytelniania (sprawdzanie hasła) przy dostępie do aplikacji sieciowych.

Model mechanizmu filtrującego to dwa bloki funkcjonalne: sensor (czujnik) wykrywający niepożądane cechy ruchu sieciowego oraz bramka, przekazująca lub nie, informacje między dwoma interfejsami. Do sterowania służy tzw. konfiguracja, czyli uporządkowany zbiór pewnych prostych reguł o postaci: „jeżeli wystąpił *symptom*, to *reakcja*". Po rozpoznaniu przez sensor *symptomu* bramka realizuje *reakcję*. Dla sformułowania reguł blokujących ataki muszą być znane wzorce objawów ataków — nazywane sygnaturami ataku (por. CAPEC [3]).

Filtry sieciowe zwykle działają w pewnych ustalonych warstwach modelu ISO/OSI (tzn. analizują jednostki przesyłanej informacji według protokołów zaliczanych do ustalonej warstwy) i w rezultacie różne rodzaje filtrów pozwalają blokować różne ataki. Ataki polegające na podszywaniu się mogą zostać rozpoznane przez proste filtry działające w niskich warstwach (drugiej i trzeciej), natomiast rozpoznanie kodu *wirusa* przesyłanego w spakowanym załączniku e-maila wymaga już zapory sieciowej działającej w warstwie aplikacji. W każdym jednak przypadku działanie filtra sprowadza się do wyszukiwania w ruchu sieciowym sygnatury ataku — poszukiwania pewnego „wzorca zła". Wzorzec ten musi być znany i dostarczony zaporze w postaci rozumianej przez nią reguły (lub zbioru reguł) filtrowania.

Od dawna informatyków straszy widmo tzw. *zero-day exploit*, czyli niezawodnego sposobu przeprowadzenia skutecznego ataku — odkrytego i zastosowanego powszechnie, zanim ktokolwiek opracuje sposób obrony. Mogłoby to spowodować globalną klęskę. A jednak, chociaż *zero-day exploit* może dotknąć wielu komputerów na całym świecie i spowodować w przyszłości poważne straty, to specjaliści wiedzą, że problem nieznanego im ataku może ich dotknąć w każdej chwili — wystarczy, że zostanie wykorzystany sposób działania opracowany specjalnie lub zmodyfikowany, by zaatakować właśnie ich system (*targeted attack*). Takie modyfikacje powodują, że atak nie ma znanych symptomów i jest nierozpoznawalny. Dla zaatakowanego systemu to znacznie gorsze niż globalne uderzenie *zero-day exploit*, bo nie można skorzystać z cudzych/wspólnych doświadczeń.

Podobnie nawet w przypadku ataków o rozpoznanej sygnaturze, wprowadzenie odpowiednich poprawek w zabezpieczeniach w wielkich organizacjach nie może być zrealizowane natychmiast. Każda aktualizacja (podobnie jak każda inna modyfikacja systemu) zgodnie z ustalonymi procedurami wymaga testów, czy nie powoduje negatywnych skutków w aplikacjach biznesowych. A zatem w takich systemach nawet znane ataki mają przez pewien czas cechy ataku nieznanego zabezpieczeniom.

Sposobem na zagrożenia o nieznanych sygnaturach jest zmiana koncepcji. Zamiast poszukiwać w ruchu sieciowym sygnatur ataków — „wzorców zła", można opracować opis normalnego zachowania się ruchu sieciowego i uznać go za „wzorzec dobra". Od tej pory odstępstwa od tego „wzorca dobra" można uznawać za objawy nowych ataków, a ruch sieciowy o takich cechach blokować. Taki wzorzec normalnego ruchu nazwano profilem ruchu normalnego, zaś odstępstwo od niego — anomalią[1].

Zwięzłe przedstawienie trendów w zakresie profilowania ruchu z wykorzystaniem tzw. histogramów zaprezentowano w [7]. Na podkreślenie zasługuje fakt, że właściwie wszystkie koncepcje polegają na opisywaniu ruchu normalnego (profilu) za pomocą różnych wielkości statystycznych ([4-6], [8], czy [12]), co w praktyce powoduje,

[1] Należy zwrócić uwagę, że „anomalia" jest tu rozumiana dość wąsko — jako odstępstwo od wzorca, a nie jako jakikolwiek zdefiniowany wcześniej wzorzec niewłaściwego ruchu.

że wynik badania ruchu sprowadza się do odnotowania zmian tych charakterystyk, ale użyteczność wyników do blokowania agresywnego ruchu jest niewielka. Z drugiej strony publikacje przedstawiające rozwiązania użytkowe IDS posługujące się pojęciem profilów (np. [2, 1]) są w istocie bliższe rozwiązaniom sygnaturowym. We wszystkich jednak przypadkach opisywane rozwiązania mogą znaleźć zastosowanie co najwyżej do detekcji ataków, ale nie do ich natychmiastowego blokowania.

Należy zwrócić uwagę, że w przypadku profilowania podstawą działania jest dysponowanie profilem ruchu sieciowego. Stąd podobieństwo do systemów samouczących (i przenoszenie narzędzi i doświadczeń między tymi dziedzinami): najpierw trzeba przeprowadzić budowanie profilu, a dopiero po jego zakończeniu można użyć systemu do wykrywania odstępstw od profilu.

W [10] zaproponowano metodę profilowania z wykorzystaniem histogramów nastawioną na implementację w filtrach sieciowych. Przyjęto w niej m.in., że wielkości statystyczne mogą być wykorzystywane co najwyżej do budowania profilu, ale nie do sprawdzania zgodności ruchu z profilem. Wykrycie anomalii powinno pozwolić na identyfikację ruchu stanowiącego medium ataku i w konsekwencji jego blokowanie. Dodatkowo przyjęto, że profil powinien mieć możliwość uwzględniania wielu warstw modelu ISO/OSI oraz że podczas budowania profilu nie będzie jawnie uwzględniana interpretacja zawartości ruchu sieciowego (np. budowa i znaczenie pól nagłówka IP).

Dalej zaprezentowano metodę, a ponadto przedstawiono sposób wykorzystania histogramów nie tylko do budowania profilu, lecz także do sformułowania funkcji kryterium pozwalającej na optymalizację profilu dla uzyskania maksymalnej szybkości działania. Ta funkcja może pozwolić również, w przypadku implementacji metody w filtrach sieciowych, na ocenę obciążenia sprzętu sensora operacjami badania ruchu sieciowego.

## 2. Profile

Najważniejszym działaniem w proponowanej metodzie wykrywania anomalii jest oczywiście wytworzenie „wzorca dobra", czyli profilu ruchu sieciowego w obserwowanym punkcie. Dla jego zdefiniowania istotne jest przyjęcie założeń dotyczących opisu ruchu sieciowego. Zwykle w sieciach stosowany jest siedmiowarstwowy (rzadziej czterowarstwowy) model ISO/OSI, polegający na założeniu, że wszelkie przesyłane w sieci informacje mogą być interpretowane zgodnie z opisami protokołów zawartych w tzw. RFCs[2] (*Request For Comments*, publikowane na [11]).

[2] RFC (ang. *Request for Comments*) — zbiór technicznych oraz organizacyjnych dokumentów mających formę memorandum związanych z Internetem oraz sieciami komputerowymi. Publikacją RFC zajmuje się Internet Engineering Task Force.

Protokoły przypisane są dość jednoznacznie do poszczególnych warstw modelu. W praktyce można uznać, że każdy protokół posługuje się jednostkami przesyłanej informacji i że jednostki te można łatwo wyodrębnić ze strumienia przesyłanej informacji. Każda jednostka jest pewnym ciągiem binarnym. Podczas obserwacji ruchu w pewnym punkcie sieci można zarejestrować zbiór tych jednostek. Jest on uporządkowany chronologicznie, według czasu końca transmisji każdej jednostki i z jednostką tą można powiązać pewne wartości charakteryzujące czas ich obserwacji (dokładniej chwilę obserwacji zakończenia transmisji każdej jednostki). Dodatkowo każdą zarejestrowaną jednostkę można rozszerzyć (poprzedzić) zapisem dotyczącym pewnych szczególnych właściwości historii ruchu sieciowego, a także np. dnia tygodnia, pory dnia itp. Informacje rozszerzające zapisy jednostek informacji powinny być stałej długości i zapisane na początku ciągu binarnego.

W zależności od urządzenia można dokonywać rejestracji w różnych warstwach modelu ISO/OSI (w szczególności — we wszystkich). Można też dokonywać selekcji rejestrowanych jednostek ruchu. W każdym jednak przypadku za model ruchu sieciowego będący podstawą profilowania uznaje się pewien zbiór ciągów binarnych o zmiennej długości. Dzięki ujęciu właściwości czasowych i historii w każdym opisie jednostki ruchu, można abstrahować od porządku tych jednostek/ /ciągów binarnych w zbiorze, traktując ów zbiór jako nieuporządkowany. Można też dokonać normalizacji: przyjąć pewną maksymalną długość rozpatrywaną ciągów binarnych równą $n$: dłuższe jednostki ruchu sieciowego będą po prostu obcinane do tej długości. W rozważaniach formalnych można też przyjąć, że jednostki krótsze będą uzupełniane zerami do długości $n$ bitów.

Profilem nazywa się abstrakcyjny opis ciągów binarnych uznanych za „normalne", tzn. niezawierający ciągów symptomatycznych dla zdarzeń niepożądanych (np. ataków sieciowych). Zbiór wszystkich możliwych do wygenerowania ciągów binarnych o długości nie większej od $n$ można podzielić na zbiory ciągów:

- zabronionych przez zapisy standaryzujące protokołów (zapewne w rzeczywistości takie ciągi nie wystąpią);
- niepożądanych — które zawierają symptomy zdarzeń niepożądanych (a zatem ich wystąpienie powinno powodować alarm);
- normalnych;
- nierozstrzygalnych — takich, które mogą wystąpić zarówno w przypadku, gdy nie zachodzą żadne zdarzenia niepożądane, jak i podczas trwania takich zdarzeń.

W ruchu sieciowym za dopuszczalne należy uznać ciągi normalne i nierozstrzygalne. Niestety, dla opracowania opisu, na podstawie którego można by rozstrzygać o pojawieniu się symptomów zdarzeń niepożądanych o nieznanych cechach, dostępna będzie tylko pewna ograniczona próba ruchu sieciowego. Taka próba, pozyskana na potrzeby budowy profilu, stanowi niewielki podzbiór możliwych ciągów poprawnych i nierozstrzygalnych.

Podstawową funkcją profilu jest jego użyteczność do rozstrzygania o poprawności kolejnych elementów ruchu sieciowego. Zatem można przyjąć, że:

- profil powinien być budowany na podstawie pewnej próby ruchu sieciowego, co do której istnieje pewność, że nie zawierają one żadnych symptomów działań nieuprawnionych,
- profil ma być rodzajem zapisu warunku,
- profil powinien by tak zbudowany, aby zapewnić szybkie rozpoznawanie dopuszczalnych ciągów,
- rozpoznawanie ma być realizowane *on-line*, zatem aby jak najwcześniej rozpocząć interwencję, np. blokowanie agresywnego ruchu, czas rozpoznawania powinien być jak najkrótszy,
- budowanie profilu może być realizowane *off-line*, nie podlega więc szczególnym wymaganiom czasowym.

## 3. Model ruchu sieciowego

Przyjęto, że ruch sieciowy jest rozpatrywany jako zbiór $R$ składający się z ciągów binarnych (wektorów) $r$ o stałej długości $n$ pozycji binarnych $c_i$:

$$r = \langle c_{n-1}, ..., c_i, ..., c_1, c_0 \rangle. \tag{3.1}$$

Dalej rozważane będą pewne wybrane podzbiory pozycji binarnych $c$. Przyjęto, że dla określenia takich pozycji używane będą $n$-bitowe ciągi nazywane maskami. Maska o wartości $m$ to ciąg binarny — rozwinięcie binarne $m$, w którym wartości „1" oznaczają wybrane pozycje, zaś „0" — pozostałe; $n$-bitowa maska może przyjmować $2^n - 1$ różnych wartości, wyznaczając tyleż różnych podzbiorów pozycji.

**Przez sygnaturę zbioru ciągów binarnych rozumie się taki podzbiór pozycji tych ciągów, które przyjmują takie same wartości w każdym z ciągów**. Jeśli rozpatrywany zbiór jest częścią większej przestrzeni ciągów, to sygnatura zbioru nie powinna wystąpić w żadnym elemencie należącym do dopełnienia tego zbioru.

Sygnaturą $s = \langle m, w \rangle$ pewnego zbioru $R$ nazywa się parę ciągów: maskę $m$ i wartość $w$, dla każdego $r \in R$ spełniających następujący warunek:

$$(r \oplus w) \wedge m = 0, \tag{3.2}$$

gdzie: $\oplus$ oznacza operację różnicy symetrycznej (zanegowaną różnoważność);
$\wedge$ oznacza operację iloczynu logicznego;
0 oznacza wartość wynikowego ciągu interpretowanego jako liczba stałoprzecinkowa.

Najdłuższą sygnaturę pewnego zbioru $R$ można określić, wykonując operację równoważności na wszystkich elementach zbioru — wynikiem jest wartość maski:

$$m = r_{n-1} \equiv r_{n-2} \equiv \ldots \equiv r_1 \equiv r_0, \tag{3.3}$$

gdzie ≡ oznacza operację równoważności,
wówczas: $w = r_0 \wedge m$.

Jeśli operacja równoważności wykonana na wszystkich $r \in R$ daje w wyniku zero, to $R$ nie posiada sygnatury.

Znalezienie **jednej** wspólnej sygnatury dla wszystkich „dopuszczalnych" w ruchu sieciowym ciągów wydaje się nieprawdopodobne. Można jednak przypuszczać, że zbiór „dopuszczalnych" jednostek ruchu $R$ można podzielić na szereg mniejszych podzbiorów, z których każdy będzie miał własną sygnaturę. Taki podział może okazać się użyteczny na potrzeby algorytmu rozpoznawania anomalii. Wystąpienie bowiem ciągu niezawierającego żadnej z sygnatur oznacza rozpoznanie niedopuszczalnego ciągu. Podziały można kontynuować „w głąb" podzbiorów, uzyskując dalsze sygnatury.

Dla dowolnie wybranej pary $\langle m, w \rangle$ można w zbiorze wybrać podzbiór $R(s)$ odpowiadający tej parze:

$$R(s) = \left\{ r \in R : \left( (r \oplus w) \wedge m \right) = 0 \right\}, \tag{3.4}$$

jeżeli $R(s)$ jest zbiorem niepustym, to $s = \langle m, w \rangle$ jest sygnaturą $R(s)$.

W ogólnym przypadku za profil $Q$ pewnego zbioru $R$ można uznać uporządkowany zbiór sygnatur wybierających w zbiorze podzbiory ciągów uznawanych za należące do tego profilu:

$$Q = \langle s_0, s_1, \ldots, s_z \rangle, \quad R = \bigcup_{s \in Q} R(s). \tag{3.5}$$

## 4. Budowanie profilu — analiza ruchu

Budowanie profilu odbywa się w trakcie analizy ruchu zaobserwowanego w pewnym punkcie sieci i opisanego zbiorem ciągów $R$. Pojedyncza operacja analizy ruchu jest wykonywana na zbiorze $R$. Istotą tej operacji jest wybór pewnej wartości maski $m$ stanowiącej podstawę do podziału zbioru $R$. Wynikiem operacji analizy ruchu jest podział zbioru $R$ na podzbiory $R(s_m)$ jednostek $m$ zawierających różne sygnatury o tej samej masce:

$$R = \bigcup_{s_m} R(s_m), \tag{4.1}$$

gdzie: $s_m$ — sygnatury o ustalonej wartości maski $m$ (różniące się wartościami $w$).

W ramach analizy ruchu wyznaczone podzbiory $R(s_m)$ jednostek ruchu mogą być poddawane kolejnym podziałom, aż do chwili podjęcia decyzji o zaniechaniu

kolejnych podziałów. Zapis wyniku tych podziałów jest jedną z form rekurencyjnego zapisu profilu — opisu zachowania się analizowanego ruchu, np.:

$$Q(R) = \bigcup_{s_m} \left( s_m \wedge R(s_m) \right) \tag{4.2}$$

gdzie: $Q(R)$ — profil ruchu $R$;
$s_m$ — każda sygnatura o masce $m$, wyznaczająca niepusty zbiór $R(s_m) : \left| R(s_m) \right| \neq 0$.

Równanie (4.2) z jednej strony opisuje właściwości zbioru $R$, przez podanie jednego ze sposobów podziału tego zbioru na kolejne podzbiory, a z drugiej strony opisuje warunki, które musi spełniać ciąg binarny $r$, aby został uznany za „należący do profilu", czyli „normalny". W tej drugiej interpretacji, dla każdego ciągu $r$, profil $Q$ przyjmuje wartość logiczną (1 lub 0), zaś operatory mają interpretację właściwą algebrze Boole'a. Sygnatury $s$ w wyrażeniu oznaczają warunek „$r$ zawiera sygnaturę $s$" (3.2).

Zabieg wyznaczania profilu $Q(R)$ sprowadza się do rekurencyjnego wykonywania działań analizy ruchu:

A. wyboru pewnej maski $m$ dla analizowanych ciągów binarnych $r \in R$;
B. wyznaczenia spośród możliwych wartości $w$ dla tej maski wartości sygnatur wyznaczających niepuste zbiory $R(s_m)$;
C. dla każdego ze zredukowanych zbiorów ruchu $R(s_m)$
   — wykonane analizy ruchu, czyli wyznaczenie profilu $Q\left(R(s_m)\right)$ lub
   — zaniechanie dalszych działań, co oznacza przyjęcie, że wszystkie ciągi $r$ o sygnaturze $s_m$ uznaje się za normalne bez badania dalszych warunków.

Dokonując różnych wyborów w punkcie A tego algorytmu, **można wyznaczyć wiele różnych profili**. Ponadto w punkcie C można w dowolnym momencie podjąć decyzję o zakończeniu „zagłębiania się" w rekurencję, co oznacza uznanie wszystkich jednostek o sygnaturze $s_m$ za „należące do profilu analizowanego ruchu" (oraz $Q\left(R(s_m)\right) = 1$).

Można zauważyć, że opis profilu powinien zawierać: maskę oraz wykaz wszystkich wartości sygnatur wyznaczających niezerowe podzbiory i profile odpowiadające tym sygnaturom. W pseudokodzie można zapisać deklarację struktury implementującej opis profilu:

```
struct PROFIL profil =                    \\deklaracja profilu
{  BITFIELD maska;                        \\wartość maski
   int liczba_wartości;                   \\liczba wartości
   BITFIELD wart[liczba_wartości];        \\wartości sygnatur
   (PROFIL*) profil[liczba_wartości] };   \\wskaźniki podprofilów
```

Dodano zmienną „liczba_wartości" reprezentującą liczbę sygnatur. Ostatni wiersz to tablica wskaźników do struktur opisu podprofilów zbiorów wyznaczanych tymi sygnaturami, przy czym wartość NULL w tej tablicy odpowiada przyjęciu wartości „1" (lub „PRAWDA" w zależności od konwencji) przez odpowiedni profil. Zatem wszystkie pliki o odpowiedniej sygnaturze będą uznawane za poprawne. BITFIELD oznacza tu ciąg binarny o ustalonej długości.

## 5. Wykorzystanie profilu

Profil ma być używany jako warunek do rozstrzygania, czy pewien ciąg binarny jest dopuszczalny („należy do profilu"), czy też stanowi anomalię. Sprawdzanie odbywa się w kolejnych krokach, w porządku kolejnych sygnatur. Jeśli badany ciąg zawiera jedną z wartości sygnatury, to sprawdzeniu podlega, czy ciąg ten należy do odpowiedniego podprofilu. Jeśli ciąg nie zawiera żadnej z sygnatur — zostaje uznany za niepożądany, „niemieszczący się w profilu", a zatem za anomalię.

W rezultacie wygenerowany podczas analizy pewnej próby ruchu sieciowego jej profil (por. (4.2)) może być rozpatrywany jako wyrażenie logiczne, w którym oprócz sygnatur występują operatory sumy logicznej, iloczynu logicznego oraz dopuszczalne są znaki nawiasów. To wyrażenie logiczne może być wykorzystane dla sprawdzenia, czy dowolna jednostka ruchu sieciowego mieści się w profilu (nie zawiera anomalii). Dla dowolnego $r \in R$ sprawdzenie, czy nie zawiera on anomalii, polega na wyliczeniu wartości tego wyrażenia logicznego i wartość „prawda" oznacza zmieszczenie się w profilu, zaś wartość „fałsz" oznacza anomalię. Obecność w zapisie profilu sygnatury $s$ oznacza pewien elementarny warunek, polegający na wystąpieniu tej sygnatury w $r$. Sprawdzenie tego warunku daje wynik „prawda" lub „fałsz".

Algorytm rozpoznawania, czy ciąg binarny $r \in R$ należy do profilu $Q(R)$, można zapisać w pseudokodzie w postaci rekurencyjnej funkcji jak następuje (na początku powtórzono deklarację elementarnej struktury opisu profilu):

```
struct PROFIL profil =                          \\deklaracja profilu
{  BITFIELD maska;                              \\wartość maski
   int liczba_wartości;                         \\liczba wartości
   BITFIELD wart[liczba_wartości];              \\wartości sygnatur
   (PROFIL*) profil[liczba_wartości] };         \\wskaźniki podprofilów

BOOL wynik (BITFIELD r, struct PROFIL pro) {\\deklaracja funkcji
   rob:= r && pro.maska;                        \\wart. sygnatury w r∈R
   for (i=0; i<pro.liczba_wartości; i++)        \\dla każdej wartości
      {
```

```
    if (rob == pro.wart[i])                \\gdy znaleziono wartość
       {
       if(pro.profil[i])                   \\jest adres podprofilu
          return wynik (r, pro.profil[i]); \\to jego wartość
       else                                \\   jest wynikiem
          return TRUE;                     \\koniec – dopuszczalny
       }
    }
return FALSE                               \\koniec – anomalia
};
```

Rozsądne wydaje się uporządkowanie elementarnych działań sprawdzających obecność sygnatur tak, aby sprawdzenia rokujące największe szanse na powodzenie znalazły się na początku. Oznacza to występowanie najbardziej prawdopodobnych sygnatur na początku tablicy.

Dobrą propozycją jest wykonywanie elementarnych sprawdzeń warunków składających się na $Q(R)$ w porządku zapewniającym największe szanse na powodzenie. W takim przypadku średnia liczba sprawdzeń elementarnych, i w konsekwencji średni czas wyznaczania wartości $Q(R)$, będą minimalne.

Receptą na znalezienie właściwego porządku sprawdzeń, minimalizującego średni czas sprawdzania pojedynczej jednostki ruchu sieciowego, jest odpowiednie budowanie zapisu profilu, a następnie zachowanie tak zbudowanej formy profilu jako reguły postępowania podczas sprawdzania. W związku z tym wyznaczanie sygnatur (odpowiadających potem elementarnym sprawdzeniom) powinno być realizowane tak, by:

— sprawdzeń było możliwie niewiele,
— najczęściej występujące wartości powinny być sprawdzane w pierwszej kolejności.

## 6. Szybkość sprawdzania

W wyniku różnych decyzji o wyborze podzbioru, którego wartości będą stanowić sygnatury podziału zbioru (por. punkt A na str. xx), powstają różne opisy profilu. Ocena jakości profilu powinna uwzględniać jego zdolność do rozpoznawania niepożądanych jednostek ruchu (skuteczność), pewność oceny (unikanie fałszywych alarmów) oraz szybkość działania (rozpoznawania poprawnych[3] jednostek ruchu sieciowego). W niniejszym opracowaniu rozważana jest tylko szybkość. W trakcie rozpoznawania zachowania się strumienia ruchu sieciowego na potrzeby

[3] Wykrycie anomalii będzie zdarzeniem incydentalnym.

formułowania profilu, w [10] wykorzystuje się funkcję liczebności wystąpienia poszczególnych wartości na podzbiorach pozycji.

Dla dowolnego zbioru ciągów binarnych $R$ można wyznaczyć funkcję liczebności. Każda sygnatura o masce $m$, w której liczba jedynek wynosi $k$, może przyjmować $2^k$ różnych wartości $w$.

Dla zbioru jednostek (ciągów binarnych o długości $n$) $R$ i dla wybranej maski $m$ można określić liczności wystąpienia każdej z wartości $w$, tworząc histogram wartości podzbioru. Inaczej mówiąc, istnieje funkcja liczności wartości:

$$f : M \times W \times R \to C, \tag{6.1}$$

gdzie: $M$ — zbiór $n$-bitowych rozwinięć binarnych liczb naturalnych (reprezentuje wartości masek) $M = \{1,...,2^n - 1\}$;
$W$ — zbiór $n$-bitowych rozwinięć binarnych liczb naturalnych z zerem (reprezentuje wartości sygnatur) $W = \{0,1,...,2^n - 1\}$
$R$ — zbiór $n$-bitowych ciągów binarnych;
$C$ — zbiór liczb całkowitych.

Funkcja ta określa, w ilu elementach $R$ wystąpiła wartość $w$ i może być zapisana następująco:

$$f(m,w,R) = \left| r \in R : ((w \oplus r) \wedge m) = 0 \right|, \tag{6.2}$$

gdzie $|X|$ oznacza liczność zbioru $X$.

Funkcja liczności wartości wyznacza histogramy występowania wartości sygnatur. Histogram jest podstawą wyboru podejmowanego w punkcie A algorytmu ze str. **????**. W [10] proponuje się wybór takich podzbiorów, dla których liczba wartości o niezerowych licznościach jest możliwie niewielka (ale większa niż 1) i dla których rozkład histogramu jest możliwie równomierny.

Należy pamiętać, że dla każdego zbioru $R$ zaobserwowanych ciągów $r$ o długości $n$ suma liczności wartości dla jednej maski $m$ jest stała i równa liczności zbioru $R$. To pozwala na określenie prawdopodobieństwa (według definicji częstościowej) występowania wartości $w$ na wybranych maską $m$ pozycjach ciągów ze zbioru $R$:

$$p(m,w) = f(m,w,R) / |R|. \tag{6.3}$$

Szybkość działania profilu można mierzyć liczbą elementarnych sprawdzeń wykonywanych podczas rozpoznawania ruchu. Zakłada się, że podczas rozpoznawania:

- charakterystyka ruchu sieciowego jest taka sama jak podczas budowania profilu,
- wystąpi tylko nieznacząca liczba rozpoznań anomalii,
- każde sprawdzenie obecności pojedynczej wartości sygnatury daje stałą, jednostkową wartość obciążenia obliczeniowego.

Przy tych założeniach szybkość działania algorytmu, mierzona oczekiwaną liczbą jednostek obciążenia na zbadanie jednostki ruchu, można dla każdego rozważanego profilu wyznaczyć na etapie jego budowania. Wyznaczenie średniej liczby operacji na sprawdzenie jednego ciągu pozwala na porównywanie różnych profilów na etapie budowania profilu.

Na potrzeby wyznaczania pracochłonności badania profilu, profil powinien być traktowany jako zabiór uporządkowany $Q = \langle s_0 \wedge Q_0, s_1 \wedge Q_1, ..., s_i \wedge Q_i, ..., s_k \wedge Q_k \rangle$.

Jeśli znane są histogramy rozkładu poszczególnych sygnatur w ruchu sieciowym *R*, to można określić średnie obciążenie sprawdzaniem, czy ciąg *r* należy do profilu. Średnia liczba *L* operacji elementarnych dla sprawdzenia, czy ciąg *r* należy do profilu *Q*(*R*), opisana jest wzorem:

$$L\left(Q\left(R\right)\right) = Lc\left(s_0\right) + \sum_{i=1}^{k}\left( Lc\left(s_i\right) * \left(1 - \sum_{j=1}^{i} p\left(s_{j-1}\right)\right)\right), \tag{6.4}$$

gdzie: $L\left(Q(R)\right)$ — pracochłonność profilu *Q* zbioru *R*;
$p(s)$ — prawdopodobieństwo wystąpienia sygnatury $s = \langle m, w \rangle$ (6.3).
$Lc(s)$ — pracochłonność badania sygnatury $s_i$ w podzbiorze *R*:

$$Lc\left(s_i\right) = L\left(s_i\right) + L\left(Q\left(R\left(s_i\right)\right)\right) * p\left(s_i\right);$$

$L(s)$ — pracochłonność operacji sprawdzenia (3.2).

Przy założeniu, że operacja sprawdzania jest elementarna, wartość $L(s)$ wynosi 1:

$$\begin{aligned} L\left(Q\left(R\right)\right) &= 1 + L\left(Q\left(R\left(s_0\right)\right)\right) * p\left(s_0\right) + \\ &+ \sum_{i=1}^{k}\left(1 + L\left(Q\left(R\left(s_0\right)\right)\right) * p\left(s_0\right)\right) * \left(1 - \sum_{j=0}^{i} p\left(s_{j-1}\right)\right). \end{aligned} \tag{6.5}$$

Różne uporządkowania opisu profilu będą zatem wyznaczały różne pracochłonności rozstrzygania o tym, czy ciągi należą do profilu, czy nie. Pracochłonność elementarnej operacji sprawdzenia sygnatury może być rozważana jako równa 1, można także wykorzystać jako tę wartość długość maski (liczbę jedynek w masce) albo pewne wyrażenie uwzględniające zdolności obliczeniowe mechanizmów wykonujących te operacje.

## 7. Podsumowanie

Wyznaczanie histogramów występowania poszczególnych podzbiorów (a w praktyce podciągów) ciągów binarnych *R* jest podstawą do wyznaczania

profilu tego ruchu. Zaproponowana w [10] metoda polega na budowaniu profilów w kolejnych krokach, przy czym w każdym kroku wybierany jest ciąg podstawowy, którego wartości zostaną uznane za sygnatury. Wybór dokonywany jest na podstawie histogramów wartości. Przede wszystkim wybierane są podciągi przyjmujące małą (lecz większą od 1) liczbę różnych wartości o możliwie równomiernym rozkładzie. Zwykle jednak takich kandydatów jest kilku i wybór nie jest oczywisty. Wprowadzenie miary obciążenia rozpoznawaniem anomalii daje nowy środek oceny w wyborze profilów. W tym przypadku można zbudować zbiór profilów, wyznaczając je drogą alternatywnych wyborów podzbiorów sygnatur (spośród podzbiorów o nieliczno-wartościowych, zrównoważonych histogramach). Pracochłonność $L$ (6.4) można wówczas uznać za wartość pozwalającą wybrać najlepszego kandydata.

Dowolnie wygenerowany profil może zostać poddany optymalizacji ze względu na porządek sygnatur. Zbiorem rozwiązań dopuszczalnych jest w takim przypadku zbiór wszystkich różnych uporządkowań profilu. Poszukiwanym rozwiązaniem jest porządek profilu minimalizujący pracochłonność — średnią liczbę operacji na sprawdzenie, czy pojedynczy ciąg należy do profilu.

Opisane środki budowania profilu nadal pozostają w sferze rozwiązań heurystycznych. Należy pamiętać, że nadal pozostają bez odpowiedzi istotne pytania:

— Czy najlepsze rozwiązanie jest w ogóle osiągalne drogą kolejnych wyborów sygnatur w rekurencyjnej procedurze wyznaczania profilu?
— Jak zależy skuteczność profilu od próby ruchu sieciowego (długości, miejsca, czasu i warunków rejestracji)?

Nie zmienia to faktu, że omówiony w niniejszym tekście algorytm pozwala na prostą implementację oprogramowania. Należy tu przypomnieć, że wprowadzenie 64-bitowych procesorów i specyficznej gospodarki pamięcią powoduje, że wymagania na pamięć przestają być krytyczne dla implementacji. Dla przykładu tablice robocze programu rozmieszczone w pamięci operacyjnej, pozwalające poszukiwać sygnatur uwzględniając wszystkie podciągi (podzbiory przyległych pozycji binarnych) o długości 16, 32 lub 64 bity w rozszerzonych pakietach IP ($n$ = 12 160 bitów) zajmują odpowiednio nie więcej niż 12, 24 i 48 MB. To niewiele przy typowych dla współczesnego sprzętu 4 GB pamięci operacyjnej. Oczywiście dalsze wydłużenie rozważanych ciągów odpowiednio zwiększa zapotrzebowanie na tablice robocze. Jednak szybkość działania algorytmu w fazie budowania profilu osiąga się głównie dzięki wczytywaniu do pamięci operacyjnej wielkich plików rejestracji ruchu sieciowego i uniknięciu odwołań do powolnych pamięci dyskowych.

Ponieważ proces budowania profilów będzie realizowany *off-line*, ograniczenia czasowe nie są istotne i nie stanowi problemu wielokrotne powtórzenie operacji budowania profilów i wyznaczanie ich średnich wartości obciążeń dla wybrania profilu najlepszego pod względem czasowym.

*Artykuł wpłynął do redakcji 25.03.2011 r. Zweryfikowaną wersję po recenzji otrzymano we wrześniu 2011 r.*

LITERATURA

[1] M. Blajerski, *Uniwersalny system wykrywania anomalii w ruchu sieciowym*, Praca magisterska, Wydział Cybernetyki WAT, Warszawa, 2009.

[2] CISCO: *Installing and Using Cisco Intrusion Prevention System Device Manager 6.1*, Text Part Number: OL-15169-01.

[3] *Common Attack Pattern Enumeration and Classification, CAPEC List.* http://capec.mitre.org/data/index.html The MITRE Corporation.

[4] N. A. Durgin, Zhang Pengchu, *Profile-Based Adaptive Anomaly Detection for Network Security*, SANDIA REPORT, SAND2005-7293, November 2005.

[5] S. Finlay, B. Motlagh, *Network Anomaly Detection*, Univ. of Central Florida, Dept. of Engineering Technology, Materiały kursu CET 3752 Fall 2007. http://www.ent.ucf.edu/undergraduate/ist/cet3752/Fall07_papers/TeleEssay.pdf

[6] W. Hołubowicz, R. Renk, E. Adam, *Opracowanie specyfikacji systemu wykrywania anomalii w sieci koalicyjnej oraz rekomendacji dotyczących dalszej realizacji systemu*, Sprawozdanie z realizacji zadania badawczego 13105, WIŁ, Warszawa, 2009.

[7] A. Kind, M. Stoecklin Ph., X. Dimitropoulos, *Histogram-Based Traffic Anomaly Detection*, IEEE Transactions on Network Service Management, 6, 2, June 2009.

[8] T. J. Kruk, J. Wrzesień, *Korelacja w wykrywaniu anomalii*, SECURE 2003, Materiały Konferencyjne, Warszawa, listopad 2003.

[9] A. E. Patkowski i in., *Opracowanie reguł IDS dla obrony przed atakiem wewnętrznym*, Raport końcowy z pracy badawczej 533, ITA WAT, Warszawa, 2003.

[10] A. E. Patkowski, *Mechanizmy wykrywania anomalii jako element systemu bezpieczeństwa*, Biuletyn Instytutu Teleinformatyki i Automatyki, Wydział Cybernetyki WAT, 26, Warszawa, 2009, 83-109.

[11] *The Internet Engineering Task Force (IETF)*: Repozytorium Request for Comments, http://www.ietf.org/rfc.html.

[12] A. S. Thorat, A. K. Khandelwal, B. Bruhadeshwar, K. Kishore, *Payload Content based Network Anomaly Detection*, IEEE International Conference on Applications of Digital Information and Web Technologies (ICADIWT), Report No: IIIT/TR/2008/57.

A. E. PATKOWSKI

**Optimization of a profile-based traffic anomaly detection algorithm**

**Abstract.** A detection method of network traffic anomalies is the subject of the paper. It is a pure-histogram method featured by immediate reaction that allows us to block recognized network attacks instantly. Optimization of traffic network profiles as regards the processing speed of implementation is presented.

**Keywords:** computer science, computer networks, network attacks, anomalies, algorithms, optimization