VAST – metoda zapewnienia wszechstronnej anonimowości dla użytkowników systemu WWW

Igor Margasiński, Krzysztof Szczypiorski
Instytut Telekomunikacji
Politechnika Warszawska
E-mail: {I.Margasinski,K.Szczypiorski}@tele.pw.edu.pl
http://security.tele.pw.edu.pl/

Streszczenie

W artykule zaprezentowano oryginalną metodę zapewnienia anonimowości dla użytkowników systemu WWW – VAST (*Versatile Anonymous SysTem for Web Users*). Obok spopularyzowanych sposobów ochrony prywatności takich jak serwery pośredniczące (*third party proxy servers*), przeanalizowano sieci wielu węzłów pośredniczących (*chaining with encryption*). Wskazano ograniczenia występujące w tych rozwiązaniach – w pierwszym przypadku – koncentrację danych o aktywności użytkownika w pojedynczym miejscu; w drugim – wysokie koszty budowy infrastruktury niezbędnej do realizacji sieci wielu węzłów oraz odczuwalny spadek wydajności przez zwiększenie opóźnień. Obydwie klasy rozwiązań są podatne na ataki polegające na analizie ruchu generowanego przez serwery usługodawcy. Zaproponowana w artykule nowa metoda – VAST – niweluje wskazane ograniczenia, jednocześnie oferując użytkownikom wszechstronną anonimowość względem wszystkich stron biorących udział w wymianie danych opartej na WWW. W artykule przedstawiono zarys implementacji metody w języku Java.

The prairie realm – vast ocean's paraphrase –
Rich in wild grasses numberless, and flowers
Unnamed save in mute Nature's inventory
No civilized barbarian trenched for gain.
"Tecumseh", Charles Mair (1838-1927)

1. Wprowadzenie

Przenosząc swoje życie w cyfrową otchłań, ludzie mają prawo zachować swoje dotychczasowe przyzwyczajenia i potrzeby. Kształtowanie otoczenia, kreowanie nowych środowisk egzystencji powinno być podporządkowane człowiekowi, nie maszynom. WWW (*World Wide Web*) będące, obok poczty elektronicznej, najpopularniejszą aplikacją w Internecie jest pozbawione ochrony prywatności. Uzyskanie prywatności w sieci może nastąpić dzięki dostarczeniu narzędzi zapewniających **anonimowość**. Jest to jedna z dróg do wykreowania cyfrowego środowiska zbliżonego do obecnej rzeczywistości.

Anonimowość można rozumieć na dwa sposoby: jako **anonimowość osoby** i jako **anonimowość przekazu**. Anonimowość osoby można rozgraniczyć na **anonimowość autora** i **anonimowość odbiorcy**. Należy też zauważyć, że zazwyczaj postrzega się **anonimowość względem pewnych wybranych stron**. Np. anonimowość autora wierszy często odnosi się jedynie do czytelników i współistnieje zazwyczaj z ujawnianiem tożsamości poety wobec wydawcy. Jeżeli anonimowość odnosi się do wszystkich stron – jest to **wszechstronna anonimowość**. Poza pojęciem anonimowości nadawcy i odbiorcy istotny jest również termin **braku możliwości powiązania** (*unlinkability*), oznaczający że pomimo iż wiemy, że nadawca i odbiorca uczestniczą w pewnej komunikacji, nie możemy stwierdzić, że komunikują się ze sobą.

W tej pracy będziemy zajmować się **anonimowością odbiorcy w stosunku do wszystkich podmiotów mogących mieć dostęp do informacji o nim**. W stosunku do strony usługodawcy nie będziemy dążyć do ukrycia tożsamości usługobiorcy, lecz do ukrycia jego aktywności WWW. Będzie to zatem brak możliwości powiązania użytkownika z serwerami docelowymi. Przedstawiamy propozycję rozwiązania problemu utraty prywatności użytkowników WWW – system VAST – ***V****ersatile* ***A****nonymous* ***S****ys****T****em for Web Users*.

Układ pracy
Praca została skomponowana w następujący sposób: w niniejszym **rozdziale 1** określono dziedzinę artykułu oraz przybliżono zasadnicze pojęcia. W **rozdziale 2** zaprezentowano podstawowe klasy spotykanych obecnie rozwiązań zapewniania anonimowości, a także ukazano ich ograniczenia. Zawarta krytyka tych systemów stanowi podłoże wskazujące przyczyny i motywacje do podjęcia naszych prac. W **rozdziale 3** przedstawiono cele oraz założenia projektu VAST, natomiast w **rozdziale 4** zaprezentowano koncepcję metody oraz ogólny opis sposobu działania. W **rozdziale 5** przedstawiono schemat i opis części składowych systemu. W **rozdziale 6** opisano uszczegółowiony sposób działania z naciskiem na opis specyficznego mechanizmu generacji ruchu nadmiarowego. **Rozdział 7** został poświęcony wydajności systemu VAST. W **rozdziale 8** zawarto analizę bezpieczeństwa tego rozwiązania, następnie w **rozdziale 9** przedstawiono kierunki rozwoju metody VAST. **Rozdział 10** jest podsumowaniem artykułu.

2. Rozwiązania pokrewne

2.1 Serwery pośredniczące

Obecnie coraz większą popularność zdobywają systemy zapewniania anonimowości oparte na idei serwera pośredniczącego. Serwer pośredniczący (*third party proxy server*) jest odległą maszyną, stanowiącą trzecią stronę pośredniczącą w pobieraniu zasobów z WWW. Umiejscowienie węzła pośredniczącego pozwala na ukrywanie wszelkich informacji o kliencie przed serwerem docelowym (np. adres protokołu IP – *Internet Protocol*). Dodatkowo możliwe jest szyfrowanie przekazu pomiędzy klientem a serwerem pośredniczącym, dzięki czemu ukrywany jest przebieg podejmowanej aktywności przed stronami mogącymi mieć dostęp do przekazywanych transakcji WWW (np. przed dostawcą usług internetowych – ISP – *Internet Service Provider*).

Serwer pośredniczący daje również szerokie możliwości kontroli nad przesyłanymi zasobami: możliwe jest, zarządzanie mechanizmem *cookies* [7] oraz blokowanie niepożądanych dodatków (np. okien reklamowych), jak również usuwanie skryptów, czy programów. Serwer pośredniczący może dokonywać dowolnych transformacji doręczanego dokumentu. Przykłady realizacji systemów serwera pośredniczącego to: *Anonymizer*, *Magusnet Proxy*, *Rewebber* oraz opracowany przez autorów niniejszego artykułu system *Lustro Weneckie* [8].

Względy, które pozwoliły zyskać dużą popularność tej klasie systemów zapewniania anonimowości to:
- wypełnienie luki w technologii systemu WWW związanej z niedostatkami w ochronie prywatności użytkowników końcowych,
- duża skuteczność w ukrywaniu danych identyfikujących użytkownika,
- łatwy dostęp do usługi, poparty brakiem dodatkowych wymagań od użytkowników (potrzebny jest jedynie dowolny sposób dostępu do Internetu oraz przeglądarka WWW wyposażona w standardowe technologie),
- łatwość w użytkowaniu,
- nieduże opóźnienia przy przeglądaniu stron WWW,
- prostota,
- stosunkowo niskie nakłady wymagane przy realizacji systemu,
- brak konieczności modyfikacji istniejących węzłów sieci i protokołów; budowa systemu w oparciu o istniejące i sprawdzone standardy.

* * *

Serwery pośredniczące posiadają także poważne wady. **Stosowane obecnie serwery pośredniczące mają dostęp do informacji o tym, które strony są przeglądane przez użytkownika. Nakłaniają do zawierzenia „na słowo", że dane takie nie są przez usługodawcę gromadzone ani wykorzystywane.** Użytkownik podejmuje zatem poważne ryzyko korzystając z takiej usługi – ryzyko związane z koncentracją danych o swojej aktywności w systemie WWW w jednym miejscu. W przypadku próby wykorzystania tej możliwości przez osoby zarządzające usługą, użytkownik narażony jest na straty poważniejsze niż przy klasycznym przeglądaniu stron WWW, ponieważ informacje gromadzone przez różne serwisy, są trudniejsze do powiązania. Co więcej, systemy takie nie zabezpieczają przed śledzeniem użytkowników usługi, poprzez analizę ruchu generowanego przez serwer pośredniczący i węzły z nim współpracujące (por. rozdz. 8.5). Nic nie stoi na przeszkodzie by postronna osoba atakująca mogła obserwować natężenie ruchu, a odwołania dokonywane przez serwer pośredniczący kojarzyła z poprzedzającymi je zgłoszeniami do niego. Podobnie jak wcześniej, wiąże się to z poważnym zagrożeniem, ponieważ daje możliwość osobom postronnym tworzenia szczegółowych profili zainteresowań [8]. Wadą serwerów pośredniczących jest również ograniczenie zbioru elementów, jakie mogą być pobierane. Obecność niektórych technologii rozszerzających standard HTML – *HyperText Markup Language* (takich jak JavaScript) w przekazywanych plikach HTML może stanowić bardzo poważne

zagrożenie funkcjonowania całego systemu [10]. Możliwe i łatwe jest przeprowadzenie za ich pomocą skutecznych ataków, całkowicie kompromitujących system serwera pośredniczącego.

2.2 Sieci wielu węzłów pośredniczących

Znając ograniczenia związane z bezpieczeństwem pojedynczego serwera pośredniczącego, naturalną drogą jest dokonanie rozproszenia lokalizacji informacji o podejmowanej aktywności. Jest to realizowane poprzez zastąpienie jednego węzła wieloma. Zakłada się, że każdy z węzłów takiej sieci posiada ograniczoną wiedzę o przekazywanych zasobach – może to być osiągnięte poprzez zastosowanie kryptografii asymetrycznej. Szyfrując wielokrotnie daną wiadomość (np. zapytanie do konkretnego serwera WWW) kluczami publicznymi kolejnych węzłów, możliwe jest przekazywanie danych pomiędzy komputerem użytkownika, a serwisem internetowym tak by żaden z węzłów nie wiedział jednocześnie o oryginalnym nadawcy i docelowym odbiorcy. Wybór trasy spośród dostępnych węzłów powinien mieć charakter losowy. Dzięki temu wędrujące w sieci pakiety wzajemnie się przeplatają, co ma prowadzić do uniemożliwienia ataku opartego na analizie ruchu (por. rozdz. 8.5). Koncepcja ta wywodzi się z teorii Davida Chauma [2] dotyczącej zapewniania anonimowości przy przesyłaniu poczty elektronicznej – MIXNET, w której węzły pośredniczące to tzw. MIXy. Przykłady realizacji systemów zapewniania anonimowości w systemie WWW w oparciu o sieć wielu węzłów pośredniczących to: *Onion Routing* ([12], [6]), *Crowds* [11], *Freedom* [5] (pierwszy system tego typu o charakterze komercyjnym).

Systemy wielu węzłów pośredniczących zakładają budowę kosztownej infrastruktury (sieci wielu stron pośredniczących działających w oparciu o ideę Davida Chauma – *chaining with encryption*), co stanowi czynnik zniechęcający potencjalnych inwestorów. Zastosowanie sieci stron pośredniczących, stosujących względem siebie wielokrotne szyfrowanie jest doskonałym sposobem na zapewnienie anonimowości przy przesyłaniu poczty elektronicznej. Poważne opóźnienia, które nie są uciążliwe w przesyłaniu listów elektronicznych, stanowią jednak dużą barierę dla osób przeglądających strony WWW. Aby podwyższyć szybkość działania systemu trzeba stosować bardzo silne maszyny jako węzły pośredniczące. Są one jednak kosztowne. Autorzy projektu *Crowds* dokonali analizy opóźnień wprowadzanych przez kolejne węzły pośredniczące ich systemu. Posłużymy się tymi danymi do zobrazowania jak wydłuża się czas oczekiwania na odbiór strony WWW w zależności od liczby węzłów pośredniczących.

Liczba węzłów pośredniczących	Rozmiar strony WWW [kB]					
	0	1	2	3	4	5
2 + 1	20,8%	35,0%	23,7%	13,7%	17,8%	26,3%
2 + 2	42,1%	43,4%	32,3%	22,8%	18,6%	28,2%
2 + 3	73,1%	72,1%	60,7%	40,0%	36,6%	45,0%

Tabela 2-1 Opóźnienia przy wprowadzeniu kolejnych węzłów w stosunku do sieci o 2 węzłach systemu Crowds

Wyniki (tabela 2-1) pokazują, że zapewnienie anonimowości, w systemach opartych na sieci węzłów pośredniczących, jest opłacone znacznym wydłużeniem czasu oczekiwania przez użytkownika na pobranie stron WWW. Co istotne, system *Crowds* zawiera poważne uproszczenia w stosunku do sieci MIXNET, mające na celu podwyższenie prędkości działania. Mimo tych kosztownych – ze względów bezpieczeństwa – ustępstw wydajność nie jest zadawalająca. Większość nowych projektów również wymaga od użytkowników instalacji dodatkowego oprogramowania, co obniża poziom zaufania do usługi. Pobierana aplikacja może stanowić tzw. konia trojańskiego. Użytkownik nie ma możliwości sprawdzenia, co naprawdę jest wysyłane do serwera usługi. Wymóg ten również oznacza zawężenie grupy usługobiorców, ponieważ stanowi uzależnienie od konkretnej platformy sprzętowej i systemu operacyjnego.

* * *

W systemach zapewniania anonimowości WWW działających w oparciu o sieć wielu węzłów pośredniczących nie udało się w pełni wyeliminować zagrożenia spowodowanego atakami polegającymi na analizie natężenia ruchu. Anonimowość wciąż opierana jest na istnieniu zaufanej trzeciej strony – wiedza dostępna dla pojedynczego węzła w systemie serwera pośredniczącego została jedynie rozproszona na wiele węzłów. Nie ma jednak gwarancji, że serwery te nie współpracują ze sobą. **Uważamy, że usługa WWW wymaga indywidualnego podejścia**. Przy projektowaniu systemu anonimowego przeglądania stron WWW istotne są inne czynniki niż przy przesyłaniu wiadomości pocztowych. Usługa WWW, dzisiaj coraz silniej, zyskuje charakter multimedialny. Użytkownicy oczekują by

wskazane przez nich teksty i obrazy dostarczane były bez zwłoki. Ponadto realizacja takiego rozwiązania nadal wiąże się z dużymi nakładami finansowymi. Usługodawca nie ma również możliwości przedstawienia dowodu, że węzły systemu nie współpracują ze sobą. Autorzy systemów opartych na wielu węzłach pośredniczących starają się nie wspominać o tym fakcie i zazwyczaj jedynie lakonicznie wspominają, że poszczególne węzły powinny należeć do różnych właścicieli (firm). Czy jest to jednak wystarczający powód by wierzyć, że węzły nie współpracują? Spotykamy się przecież dzisiaj często ze współdziałaniem różnych, niezależnych firm, w procesie tworzenia profili internautów [8]. Dlaczego w tym przypadku miało by być inaczej? Realizacja takich systemów nie odbywa się na powszechną skalę. Możemy zatem tylko przepuszczać, że w przypadku popularyzacji tych rozwiązań i implementacji węzłów pośredniczących przez różne prywatne firmy, brak zabezpieczeń przed przekazywaniem sobie informacji zostanie wykorzystany.

3. Cele i założenia

Przy projektowaniu własnego rozwiązania, przyjęliśmy następujące wytyczne:

- zachowanie zalet osiągniętych w systemie *Lustro Weneckie* [8] – tożsamych z zaletami systemów zapewniania anonimowości opartych na idei serwera pośredniczącego (por. rozdz. 2.1),
- zapewnienie pełnej anonimowości (również względem usługodawcy, a także przeciwdziałanie atakom polegającym na analizie ruchu),
- szybkość działania – minimalizowanie różnic w stosunku do tradycyjnej nawigacji WWW),
- ogólnodostępność – brak jakichkolwiek dodatkowych wymagań od użytkownika poza dostępem do sieci Internet, w szczególności, brak aplikacji instalowanych na maszynie użytkownika,
- łatwość we wdrożeniu – stosunkowo niskie koszty.

Naszym celem było opracowanie metody wpisanej w specyfikę systemu WWW i wykorzystującej ją.

4. Koncepcja systemu VAST

W systemie VAST mamy do czynienia z jednym węzłem pośredniczącym. Aby osiągnąć anonimowość względem serwera pośredniczącego, a także by uniemożliwić śledzenie użytkowników poprzez analizę natężenia ruchu, wprowadzony został **mechanizm generacji ruchu nadmiarowego**. Pomiędzy klientem HTTP (*HyperText Transfer Protocol*) ([1],[4]) a serwerem pośredniczącym przekazywane jest więcej stron, niż w istocie przegląda użytkownik (rysunek 5-1). Wiedza o tym, które zasoby stanowią przedmiot zainteresowania internauty, jest dostępna tylko dla niego samego. Z przeglądarką użytkownika współpracuje agent (aplet Java), który w czasie, gdy użytkownik zapoznaje się z treścią stron, naśladuje nawigację użytkownika, kierując do serwera pośredniczącego wybrane przez siebie zapytania. Geneza rozwiązania wywodzi się z obserwacji typowej nawigacji WWW. Użytkownik nie pobiera stron WWW w sposób ciągły. Zapytania o kolejne strony występują w różnych odstępach czasu, pomiędzy którymi użytkownik zapoznaje się z treścią serwisu. Dodatkowo wykorzystywana jest łatwość generacji ruchu nadmiarowego jaki występuje w systemie WWW. Mamy tu do czynienia z dostępem do szerokiego zakresu zasobów internetowych opartego o indeksacje w wyszukiwarkach (*web search engines*). Możemy zatem łatwo symulować komunikacje z serwerami. Jest to kolejna specyficzna cecha systemu WWW, którą wykorzystujemy. W odróżnieniu od innych proponowanych obecnie rozwiązań, system nie wykonuje dodatkowych czynności w trakcie pobierania zasobów, ale wykorzystuje wolny czas na działania zapewniające wszechstronną anonimowość. Osoba korzystająca z systemu VAST będzie miała dostęp do kodu źródłowego apletu. Dzięki temu będzie możliwe sprawdzenie, czy nie jest to tzw. koń trojański.

5. Schemat systemu

VAST składa się z dwóch podstawowych elementów:

- **agenta działającego** w środowisku przeglądarki WWW użytkownika,
- **serwera pośredniczącego**, który stanowi węzeł pomiędzy tymże agentem, a docelowymi serwerami WWW.

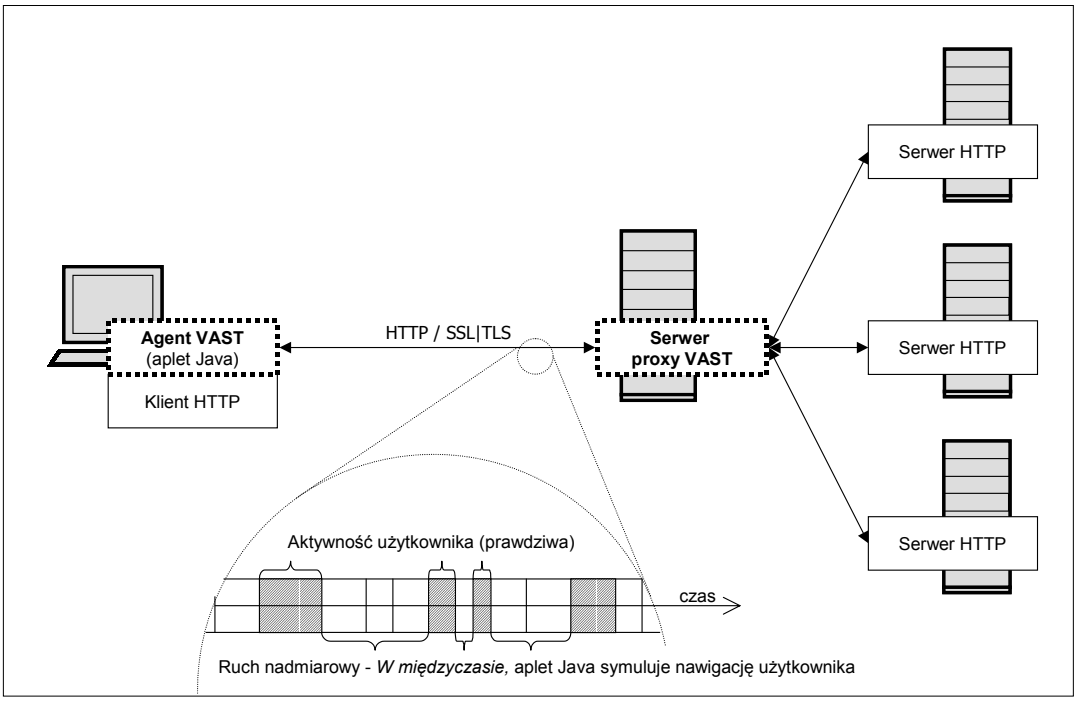
5.1 Agent

Agent to aplet napisany w języku Java, działający w środowisku przeglądarki WWW.

Podstawowe funkcje agenta systemu VAST to:

- komunikacja z serwerem pośredniczącym z wykorzystaniem bezpiecznego połączenia SSL/TLS – *Secure Socket Layer / Transport Layer Security* [3],
- naśladowanie aktywności użytkownika,

- generowanie adresów URI (*Uniform Resource Identifier*), będących tłem dla adresów, które podaje użytkownik,
- przyjmowanie od użytkownika i przekazywanie do serwera pośredniczącego parametrów pracy (w szczególności adresów URI i ustawień poziomu bezpieczeństwa) podawanych przez użytkownika,
- przekazywanie adresów właściwych i nadmiarowych do serwera pośredniczącego,
- odbieranie zasobów dostarczanych przez serwer pośredniczący,
- podział odebranych zasobów na te, które zostały wskazane przez użytkownika i na nadmiarowe,
- prezentacja w przeglądarce użytkownika właściwych stron WWW (z pominięciem nadmiarowych),
- analiza poziomu anonimowości użytkownika na podstawie stosunku pobieranych zasobów wskazanych przez niego, do ruchu nadmiarowego,
- informowanie użytkownika o aktualnym poziomie anonimowości,
- komunikacja z użytkownikiem za pośrednictwem interfejsu graficznego.



Rysunek 5-1 Schemat działania systemu VAST

5.2 Serwer pośredniczący

W projektowaniu serwera pośredniczącego VAST wykorzystano doświadczenia zdobyte przy implementacji systemu serwera pośredniczącego *Lustro Weneckie*. Serwer pośredniczący będący częścią składową systemu VAST jest bardzo zbliżony do niego – istotna różnica dotyczy braku interfejsu użytkownika. Rola komunikacji z użytkownikiem w systemie VAST została przeniesiona do agenta.

Podstawowe funkcje serwera pośredniczącego VAST to:

- ukrywanie wszelkich danych o kliencie HTTP przed serwerem docelowym – w szczególności adres protokołu IP,
- szyfrowanie wszelkich danych przekazywanych od i do agenta – w szczególności adresu URL (*Uniform Resource Locator*) zasobów przeglądanych przez użytkownika,
- opcjonalne szyfrowanie przekazu pomiędzy systemem a docelowym serwerem WWW,
- blokowanie zapisu *cookies* pochodzących od serwera docelowego,
- blokowanie skryptów i programów pochodzących od serwera docelowego [10],
- blokowanie apletów Java pochodzących od serwera docelowego [10].

6. Sposób działania

Przy połączeniu się przez użytkownika z serwerem usługi, pobierany jest specjalny aplet Java. W czasie, gdy użytkownik zapoznaje się z treścią pobranej strony WWW agent ten symuluje aktywność użytkownika. Za pomocą narzędzi wyszukujących (*search engines*) stosowanych przez użytkownika, oraz odnośników dostępnych na pobieranych stronach, agent ten wystosowuje kolejne zapytania.

6.1 Generacja ruchu nadmiarowego

Na potrzeby opisu sposobu działania systemu przyjmujemy następujące określenia:

- *Transakcja WWW* – zbiór żądań HTTP oraz odpowiadających im odpowiedzi serwera, potrzebnych do pobrania pojedynczego dokumentu HTML (plik HTML oraz elementy w nim zawarte, np. pliki graficzne).
- *Sesja tematyczna* – zbiór kolejnych transakcji WWW generowanych przy przeglądaniu przez użytkownika stron (nie należy mylić z sesją przeglądarki). Zbiór transakcji WWW nazywamy sesją wtedy, gdy możliwe jest powiązanie poszczególnych transakcji na podstawie odnośników zawartych w plikach HTML należących do tych transakcji. Przyjmujemy, że podczas przeglądania stron WWW przez użytkownika, rozpoczyna się nowa sesja, wtedy, gdy zapytanie HTTP nie może być powiązane z jednym z odnośników dostępnych na poprzednio odwiedzanych stronach. W dalszym opisie stosowana będzie skrócona nazwa – sesja.

Zakładamy, że strona atakująca mająca dostęp do treści przesyłanych danych[1] potrafi wyodrębnić z ruchu generowanego przez użytkownika poszczególne transakcje oraz sesje.

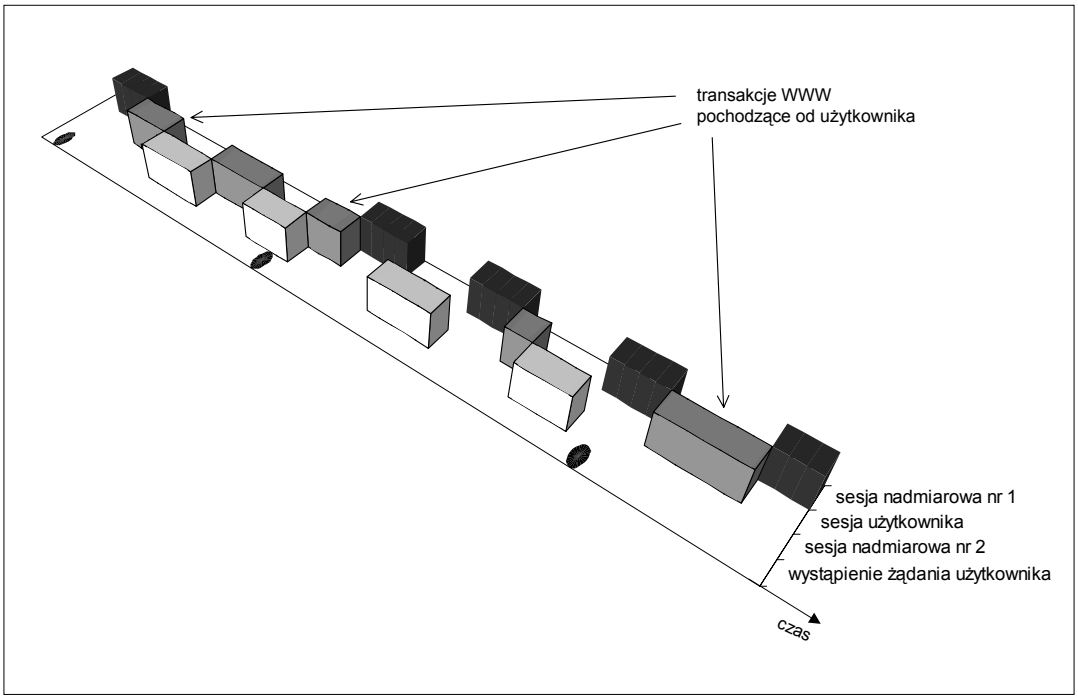
Generacja ruchu nadmiarowego polega na wprowadzaniu dodatkowych sesji. Transakcje w obrębie tych sesji dokonywane są zazwyczaj w czasie gdy użytkownik zapoznaje się z treścią pobranej już strony. Poza generacją zapytań w obrębie dodatkowych sesji, agent wystosowuje nadmiarowe zapytania w obrębie samej sesji użytkownika. Dzięki temu rozróżnienie, która sesja pochodzi od użytkownika nie jest możliwe. Swoista właściwość sesji generowanej przez człowieka – potencjalne zależności tematyczne kolejnych transakcji – jest tracona. W przypadku gdy użytkownik rozpoczyna nową sesję agent rozpoznaje to zdarzenie i sam dokonuje zamiany prowadzonych sesji nadmiarowych na nowe. Strona atakująca (znająca algorytm działania agenta – ogólnodostępny kod źródłowy), nie może przesądzić czy poszczególne zapytania pochodzą od użytkownika, czy od symulatora. Przy zastosowaniu silnych narzędzi automatycznie analizujących zapytania poszczególnych użytkowników, możliwe jest jedynie wyodrębnienie poszczególnych sesji. Usługodawca – najgroźniejsza potencjalna strona atakująca (o największych możliwościach) – analizując poszczególne zapytania może wyodrębnić jedynie poszczególne sesje. Na tej podstawie może określić że użytkownika interesuje ogólna tematyka jednej z nich. Nie wie jednak której. Nie wie również które zapytania w obrębie jednej z sesji mogą należeć do użytkownika. Użytkownik ma możliwość konfiguracji ilości sesji nadmiarowych generowanych przez system. Mając na uwadze przepustowość swojego połączenia internetowego, oraz częstość generacji zapytań, może wybrać poziom anonimowości, jaki ma zapewniać system. Mierzy się on wartością prawdopodobieństwa tego, że użytkownik jest zainteresowany tematyką wybranej sesji. Jest ono opisane następującą zależnością:

$$P \leq \frac{1}{ilość\ równoległych\ sesji\ nadmiarowych\ +\ 1}$$

Warto tu zaznaczyć, że wprowadzenie tylko jednej sesji nadmiarowej daje anonimowość określaną terminem „prawdopodobnie niewinny" (*probable innocence*) [11]. W systemie *Crowds* anonimowość taka jest osiągana tylko w pewnych okolicznościach (duża liczba użytkowników) i tylko w stosunku do niektórych stron. W stosunku do otoczenia lokalnego nie zakłada się tam żadnej ochrony. Przed rozpoczęciem korzystania z systemu, użytkownik będzie musiał dokonać konfiguracji – konieczne jest określenie listy stosowanych przez niego narzędzi wyszukujących. Agent będzie następnie wykorzystywał je do generacji ruchu nadmiarowego. Agent będzie korzystał ze **słownika** haseł/terminów pobieranego z serwera usługodawcy VAST. Należy zapewnić, by był on jak najbardziej obszerny. W przypadku, gdy użytkownik będzie wprowadzał hasło, które nie znajduje się w słowniku, będzie o tym informowany i ostrzegany, że usługodawca systemu VAST może wnioskować, że to zapytanie nie zostało sztucznie wygenerowane. Wystosowanie zapytania poprzez narzędzie wyszukujące oznacza rozpoczęcie nowej sesji. Na tej samej zasadzie rozpoczynane są sesje nadmiarowe. Przy rozpoczynaniu pracy z systemem, zapytania użytkownika nie są od razu realizowane. To która transakcja zostanie wystosowana jako pierwsza ma podłoże losowe. W przypadku następnych transakcji, zapytania użytkownika mają priorytet w wykonywaniu przez agenta. Jednak w przypadku gdy ich częstość

[1] w przypadku systemu VAST jest to tylko i wyłącznie strona serwera pośredniczącego

zaczyna przewyższać częstość transakcji w obrębie poszczególnych sesji nadmiarowych, użytkownik jest o tym informowany. Agent wystosowuje ostrzeżenie, że strona atakująca może na podstawie większej częstości występowania pewnych transakcji przepuszczać, że należą one do użytkownika. Do generacji adresów wykorzystywane są narzędzia wyszukujące stosowane przez użytkownika (lista narzędzi wyszukujących użytkownika). Reprezentacja graficzna przebiegu przykładowej komunikacji pomiędzy agentem systemu VAST a serwerem pośredniczącym przedstawia rysunek 6-1. W przykładzie zastosowano dwie sesje nadmiarowe. Prostopadłościany obrazują transakcje WWW w obrębie poszczególnych sesji. Na rysunku zostały wskazane strzałkami te transakcje, które pochodzą od użytkownika.



Rysunek 6-1 Ilustracja przykładowego przebiegu komunikacji pomiędzy agentem a serwerem pośredniczącym. Prostopadłościany reprezentują pojedyncze transakcje WWW

7. Wydajność

Zaprezentowany system został zaprojektowany, aby umożliwić wysoką wydajność – zbliżoną do tradycyjnego przeglądania stron WWW. Dodatkowe działania, których zadaniem jest zapewnienie anonimowości, nie następują w czasie wystosowywania zapytania do serwera, ani w czasie pobierania zasobów, lecz wtedy, gdy użytkownik zapoznaje się z odebraną stroną internetową. Wciąż jednak, pozostaje pytanie: jak ruch nadmiarowy w rzeczywistości opóźnia nawigację? Czasami użytkownik pobieżnie i szybko przegląda treść stron. Ile w takim razie będzie musiał czekać? VAST może blokować wszystkie dane, które pochodzą od serwerów stanowiących trzecią stronę. Oznacza to wyłączenie wszystkich multimedialnych pasków reklamowych (*banners ads*) trzecich stron, często umieszczanych na wielu stronach. Przeprowadzona przez autorów analiza statystyczna wykazała, że objętość reklam umieszczanych na popularnych stronach WWW i portalach często przekracza 50% całkowitej objętości stron, a ilość zapytań potrzebnych do pobrania stron jest często wielokrotnością zapytań wystosowywanych tylko do docelowego serwera ([9] – rozdz. 4.7). W systemie VAST zapytania do serwerów trzeciej strony zastąpione są zapytaniami nadmiarowymi. Użytkownicy godzą się z pobieraniem licznych dodatków reklamowych. Można zatem zakładać, że zastąpienie reklam danymi nadmiarowymi które pozwalają na zapewnienia skutecznej ochrony prywatności, będzie tym bardziej akceptowane.

8. Bezpieczeństwo

Celem systemu VAST jest zapewnienie **wszechstronnej** anonimowości rozumianej jako anonimowość względem wszystkich stron biorących lub mogących brać udział w transakcji HTTP. Anonimowość względem serwera docelowego oraz stron znajdujących się w lokalnym otoczeniu użytkownika, jest zapewniana poprzez zastosowanie specyficznej architektury – wprowadzeniu serwera pośredniczącego. Dzięki zastosowaniu ruchu nadmiarowego chronimy użytkownika przed atakiem polegającym na analizie ruchu oraz przed samym usługodawcą. Jedynie sam

użytkownik wie, które zapytania zostały wystosowane przez niego. Anonimowość w tym zakresie uzyskiwana jest, zatem nie przez **ukrywane, ale** przez **maskowanie**. Aby zapobiec atakom ze strony serwerów docelowych z wykorzystaniem niebezpiecznych elementów stron WWW, serwer pośredniczący filtruje przekazywane do agenta zasoby i odrzuca elementy aktywne takie jak skrypty, czy programy, mogące stanowić zagrożenie. W szczególności usuwane są aplety Java.

Anonimowość w stosunku do serwera pośredniczącego VAST osiągania jest na zasadzie **maskowania**. Oznacza to, że usługodawca VAST zna realizowane transakcje, lecz nie może stwierdzić czy są one generowane przez użytkownika. Precyzyjnym określeniem jest tu brak możliwości powiązania (*unlinkability*) użytkownika z docelowymi serwerami WWW. W stosunku do wszystkich innych stron, anonimowość osiągana jest na zasadzie **ukrywania**. Wszystkie inne strony takie jak dostawca usług internetowych, serwery WWW, strony podsłuchujące, pracodawcy itp. nie mają dostępu do przekazywanych transakcji, nie mają dostępu do innych informacji poza tym, że użytkownik przegląda strony WWW, z wykorzystaniem usługi VAST. System oparty na przedstawionej myśli, zawiera w sobie automatycznie zabezpieczenie przed wykorzystaniem go w celach przestępczych. Aktywność użytkownika odwiedzającego strony WWW o podłożu przestępczym, nie będzie skutecznie ukrywana przez ruch nadmiarowy. Zapytania generowane przez agenta systemu VAST wybierane są z odnośników dostarczanych przez popularne narzędzia wyszukujące. Strony WWW, które nie należą do tego zbioru, stanowić będą kontrast. Strony o treściach przestępczych nie są zazwyczaj indeksowane.

8.1 Anonimowość w stosunku do stron z otoczenia lokalnego użytkownika

Komunikacja pomiędzy komputerem użytkownika a serwerem docelowym zabezpieczana jest protokołem SSL/TLS. Oznacza to, że strony z otoczenia lokalnego nie mają dostępu do przesyłanych danych. Jest to anonimowość osiągana na zasadzie ukrywania. Bezpieczeństwo w tym zakresie opiera się na bezpieczeństwie samego protokołu SSL/TLS oraz algorytmów kryptograficznych w nim wykorzystywanych. Ponieważ zarówno agent jak i serwer pośredniczący są elementami implementowanymi przez usługodawcę, możliwy jest dobór odpowiednich algorytmów kryptograficznych oraz zastosowanie np. wersji 3 protokołu SSL. W odstępach pomiędzy transakcjami WWW użytkownika, transmitowany jest ruch nadmiarowy. Stanowi to bardzo skuteczną ochronę przed atakiem polegającym na analizie ruchu.

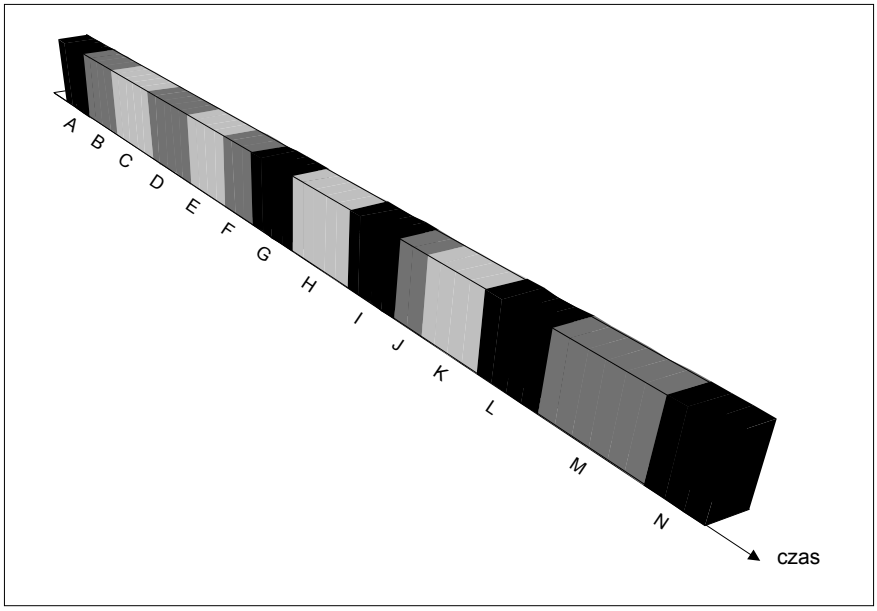
8.2 Anonimowość w stosunku do innych użytkowników sieci Internet

Tak jak zaznaczono wyżej transakcja w relacji agent – serwer pośredniczący jest bardzo skutecznie chroniona przed podsłuchem, czy możliwością powiązania zapytań z odpowiedziami serwera na podstawie czasowych zależności w natężeniu ruchu. Komunikacja serwer pośredniczący – docelowy serwer WWW nie musi być zabezpieczana. Podsłuch tych danych oznacza tylko przechwycenie informacji o aktywności serwera pośredniczącego. W przypadku, gdy z systemu korzysta wielu użytkowników jest to informacja o znikomej wartości. Jeżeli przyjmiemy skrajny przypadek, że w danym czasie tylko jeden użytkownik używa systemu VAST, oraz że strona podsłuchująca ma możliwość przejęcia wszystkich zapytań realizowanych przez serwer pośredniczący, wówczas bezpieczeństwo wobec takiej strony atakującej jest takie same, jak względem usługodawcy (patrz kolejny rozdział – 8.3).

8.3 Anonimowość w stosunku do usługodawcy VAST

Nawiązując do ilustracji przykładowej komunikacji agenta z serwerem usługodawcy (rysunek 6-1), strona atakująca może dokonać jedynie powiązania poszczególnych zapytać tak jak to przedstawia rysunek 8-1. Widzimy że strona serwera pośredniczącego może dokonać podziału na poszczególne sesje (trzy sesje odróżniane na rysunku różnymi odcieniami). Strona atakująca nie może określić, który odcień oznacza aktywność użytkownika. Liczy się również z faktem, że zapytania użytkownika to jedynie niektóre spośród bloków reprezentowanych tym samym (nie wiadomo którym) odcieniem. Możliwe jest zatem, po zastosowaniu analizy treści realizowanych transakcji, określenie że przesyłane dane z przedziałów: A, G, I, L, M to jedna sesja, B, D, F, J, M – druga, a C, E, H, K to trzecia sesja. Strona atakująca wie, że tematyka jednej z sesji jest w zakresie zainteresowania użytkownika. Nie może określić która, ani tym bardziej wyodrębnić z niej zapytania użytkownika (w tym przykładzie: B, F, M).

Przedstawiona ilustracja jest uproszczeniem graficznym. Nie należy utożsamiać regularnych tu bloków z ilością przesyłanych bitów. Pojedynczy blok reprezentuje całą transakcję WWW i składa się z kilku zapytań i odpowiedzi różnej długości.



Rysunek 8-1 Ilustracja komunikacji w relacji agent – serwer pośredniczący z punktu widzenia strony serwera pośredniczącego (porównaj z rysunkiem 6-1).

8.4 Anonimowość w stosunku do docelowego serwera WWW

Docelowy serwer WWW nie ma możliwości określenia kim jest użytkownik systemu. Dla serwera pośredniczącego dostępne są jedynie dane o serwerze pośredniczącym, który przekazuje zapytania użytkownika. Aktywne elementy umieszczane na stronach serwera WWW, które mogą powodować bezpośrednie komunikowanie się z tymże serwerem docelowym, są usuwane przez serwer pośredniczący VAST.

8.5 Ochrona przez atakami opartymi na analizie ruchu

Do tej pory nie został zrealizowany żaden system skutecznie chroniący przed wszystkim wymienionymi typami ataków (atak czasowy, analiza natężenia, atak potokowy, atak połączeniowy). Systemy oparte o idee sieci MIXów Davida Chauma, których działanie nie zakłada istnienia kompromisów na rzecz prędkości działania, chronią skutecznie jedynie przed atakiem czasowym oraz przed atakiem opartym na analizie natężenia ruchu. Przeprowadźmy dyskusję ochrony dostarczanej przez system VAST w odniesieniu do poszczególnych ataków.

Analiza czasowa (*timing attack*) polega na obserwacji czasu trwania komunikacji poprzez łączenie potencjalnych punktów końcowych i wyszukiwanie korelacji pomiędzy rozpoczęciem i/lub zakończeniem zdarzenia na każdym z możliwych punktów końcowych. System VAST w pełni chroni przed tym atakiem dzięki zastosowaniu ruchu nadmiarowego. Strona obserwująca nie może odróżnić poszczególnych zapytań, ponieważ zaraz po realizacji jednej transakcji natychmiast następuje kolejna. Nie jest zatem możliwe ustalenie czy zapytanie dotyczy określonej transakcji czy już następnej. Oczywiście w przypadku, gdy z systemu korzysta tylko jeden użytkownik (skrajny przypadek), można zakładać, że wszystkie zapytania serwera pośredniczącego pochodzą od strony tego użytkownika. Jednak nawet w tym przypadku anonimowość osoby jest zachowana. Wówczas poziom anonimowości jest taki, jak w przypadku anonimowości użytkownika w stosunku do strony serwera pośredniczącego.

Atak poprzez analizę natężenia ruchu (*message volume attack*) to obserwacja ilości transmitowanych danych (np. długości wiadomości) oraz korelacja wejścia w wyjściem. Tak jak wyżej zaznaczono system VAST wypełnia przedziały bezczynności użytkownika ruchem nadmiarowym. Wyodrębnienie poszczególnych wiadomości z zaszyfrowanego kanału agent – serwer pośredniczący jest dzięki temu praktycznie niemożliwe.

Atak potokowy (*flooding attack*), czyli wysyłanie dużej liczby wiadomości, lub wiadomości w pewien sposób charakterystycznych przez innych użytkowników systemu mające na celu wyodrębnienie wiadomość użytkownika, udaremniany jest poprzez samą postać wiadomości wysyłanych do serwera pośredniczącego. Po przeprowadzeniu

skutecznej izolacji wiadomości użytkownika, nadal nie wiadomo, które zapytania są generowane sztucznie, a które przez człowieka.

System VAST w przedstawionej postaci nie chroni skutecznie przed **atakiem połączeniowym** (*linking attack*), czyli atakiem opartym na długoterminowej obserwacji. Wykorzystywane są zmiany w ruchu związane z połączeniem z usługą (lub z jego brakiem) poszczególnych użytkowników. W koncepcji systemu VAST – dla zachowania prostoty opisu – celowo nie rozpatrzono tego rodzaju zagrożeń. Możliwe jest jednak wzbogacenie systemu VAST o mechanizm skutecznie przeciwdziałający atakowi połączeniowemu – zostało to opisane w rozdziale 9.1.

9. Rozwój systemu

9.1 Ochrona przed atakiem opartym na długoterminowej obserwacji

Przedstawiony schemat systemu prezentuje koncepcję, która przy realizacji praktycznej wymaga wprowadzenia dodatkowych mechanizmów. Przy implementacji systemu do użytku publicznego, należy wziąć pod uwagę możliwość ataku poprzez długoterminową obserwację prowadzoną przez serwer pośredniczący lub przez stronę obserwującą ruch generowany przez serwer pośredniczący. Chodzi tu o możliwość wyodrębniania powtarzających się zapytań na przestrzeni wielu sesji. Aby zapewnić skuteczną ochronę przed tego typu atakiem należy zastosować w programie agenta mechanizm zapamiętywania powtarzających się odwołań. Wtedy możliwe będzie wprowadzanie ruchu nadmiarowego, naśladującego aktywność nie tylko w obrębie sesji ale i dłuższym. Ważne jest przy tym, by informacje gromadzone przez agenta nie mogły być przechwycone przez inne strony. Agent zapisuje powtarzające się odwołania użytkownika. Program agenta nie przekazuje tych danych żadnej stronie. Odwołaniom powtarzającym się, towarzyszy ruch nadmiarowy również należący do zakresu adresów wcześniej generowanych. Dzięki temu możliwe jest tworzenie ruchu nadmiarowego, który naśladuje aktywność użytkownika na przestrzeni wielu sesji. Strona atakująca będzie mogła wtedy jedynie stwierdzić, że użytkownik korzysta z ulubionych serwisów, jednak nie będzie możliwe przesądzenie z których. Należy rozważyć tu przekształcenie apletu agenta w program, który stanowi tzw. lokalny serwer pośredniczący (*local proxy*). Dzięki temu możliwe będzie zapisywanie plików na lokalnym dysku użytkownika zawierających zapis historii. Umożliwi to również przechowywanie plików nadmiarowych pobranych w obrębie sesji użytkownika (*cache*). Użytkownik może wybrać stronę WWW pobraną w ramach transakcji nadmiarowej. W takim wypadku przyśpieszamy znacznie nawigację.

9.2 Wprowadzenie nowych usług

Pytania o przyszłość opisanego systemu powinny skupiać się również na możliwościach rozszerzenia go o obsługę innych usług obecnych w Internecie (poczta elektroniczna, grupy dyskusyjne, transfer plików i itd.). Obecne zawężenie do WWW jest celowe i wynika ze specyficznych wymagań obecnych przy przeglądaniu stron internetowych. Anonimowość WWW wymaga indywidualnego podejścia – przedstawiony system jest dedykowany do WWW zapewniając szybkość działania oraz prostotę w budowie usługi i w jej użytkowaniu.

10. Podsumowanie

W niniejszym artykule przedstawiliśmy oryginalne rozwiązanie – VAST – chroniące prywatność użytkowników końcowych systemu WWW poprzez zapewnianie wszechstronnej anonimowości. Rozwiązanie to jest rozwinięciem sprawdzonych w WWW systemów opartych na pojedynczym serwerze pośredniczącym. Stanowi rozwiązanie pełne a jednocześnie niwelujące niedostatki związane z dużymi opóźnieniami w działaniu, dostępem usługodawcy do danych użytkownika oraz problemami z wdrożeniem usługi. Nowatorska cecha systemu – wprowadzanie ruchu nadmiarowego – może być w niektórych przypadkach traktowana jako jego wada. W przypadku użytkowników, których opłaty za dostęp do sieci Internet uzależnione są od ilości przesyłanych danych, wiąże się to z dodatkowymi kosztami. Jednak warto zwrócić uwagę, że system może odrzucać elementy reklamowe pochodzące od serwerów trzecich stron. Oznacza to, że następuje zamiana plików graficznych z trzecich stron, na ruch nadmiarowy. Anonimowość w stosunku do usługodawcy VAST osiągana jest na zasadzie maskowania. Oznacza to, że usługodawca może z pewnym prawdopodobieństwem (wybranym przez użytkownika) przypuszczać, że określone zapytania pochodzą od użytkownika. Trzeba jednak zaznaczyć, że całkowite wyeliminowanie tej wady, oznaczałoby uzyskanie absolutnej anonimowości WWW osiąganej za pomocą środków technicznych, co wydaje się być praktycznie nieosiągalne.

Literatura

[1] Berners-Lee, T., Fielding, R., Frystyk, H. *Hypertext Transfer Protocol – HTTP/1.0*. RFC 1945, 1996.

[2] Chaum, D. *Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms*. Communications of the ACM, 1981.
[3] Dierks T., Allen C. *The TLS-Protocol Version 1.0*. RFC 2246, 1999.
[4] Fielding, R., Gettys, J.,Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee T. *HyperText Transfer Protocol – HTTP/1.1*. RFC 2616, 1999.
[5] Goldberg, I., Shostack, A. *Freedom Network 1.0 Architecture and Protocols*. Zero-Knowledge Systems. White Paper, 1999.
[6] Goldschlag, D. M., Reed, M. G., Syverson, P. F. *Onion Routing for Anonymous and Private Internet Connections*. Communications of the ACM, 1999.
[7] Kristol, R., Montulli, L. *HTTP State Management Mechanism*. RFC 2965, 2000.
[8] Margasiński, I. *Zapewnianie anonimowości przy przeglądaniu stron WWW*. Krajowe Sympozjum Telekomunikacji, 2002.
[9] Margasiński, I. *Wszechstronna anonimowość użytkowników końcowych w systemie WWW*, Praca dyplomowa magisterska, Politechnika Warszawska, Instytut Telekomunikacji, 2003.
[10] Martin, D., Schulman, A. *Deanonymizing Users of the SafeWeb Anonymizing Service*. Privacy Foundation, Boston University, 2002.
[11] Reiter, M.K., Rubin, A.D. *Crowds: Anonymity for Web Transactions*. ACM Transactions on Information and System Security, 1997.
[12] Syverson, P. F., Goldschlag, D. M., Reed, M. G. *Anonymous Connections and Onion Routing*. IEEE Symposium on Security and Privacy, 1997.

Artykuł recenzowany.