POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ROZWÓJ I ZASTOSOWANIA TECHNOLOGII I SYSTEMÓW INFORMATYCZNYCH

pod redakcją:
Jana Studzińskiego
Ludosława Drelichowskiego
Olgierda Hryniewicza

ROZWÓJ I ZASTOSOWANIA TECHNOLOGII
I SYSTEMÓW INFORMATYCZNYCH

Polska Akademia Nauk • Instytut Badań Systemowych

Seria: BADANIA SYSTEMOWE
tom 28

Redaktor naukowy:

Prof. dr hab. Jakub Gutenbaum

Warszawa 2001

ROZWÓJ I ZASTOSOWANIA TECHNOLOGII I SYSTEMÓW INFORMATYCZNYCH

pod redakcją
Jana Studzińskiego, Ludosława Drelichowskiego
i Olgierda Hryniewicza

Wydano z wykorzystaniem dotacji KOMITETU BADAŃ NAUKOWYCH

Książka zawiera wybór artykułów poświęconych omówieniu aktualnego stanu badań w kraju w zakresie rozwoju technologii, modeli i systemów informatycznych oraz ich zastosowań w różnych dziedzinach gospodarki narodowej. Wyodrębnioną grupę stanowią artykuły aplikacyjne omawiające wyniki projektów badawczych i celowych KBN.

Recenzenci artykułów:

Dr hab. inż. Ryszard Budziński, prof. US
Prof. dr hab. inż. Janusz Kacprzyk
Dr hab. Adam Kopiński, prof. AE we Wrocławiu
Doc dr hab. inż. Marek Libura
Prof. dr hab. inż. Andrzej Straszak

ISBN 83-85847-59-6
ISSN 0208-8028

Rozdział 3

Metody i algorytmy obliczeniowe w systemach informatycznych

ZASTOSOWANIE KRYPTOGRAFICZNYCH FUNKCJI SKRÓTU W EKSPLORACJI DANYCH W HURTOWNIACH DANYCH

Izabela Janicka-Lipska, Mohannad Najjar, Janusz Stokłosa

Politechnika Poznańska,
Katedra Automatyki, Robotyki i Informatyki

W artykule przedyskutowano wykorzystanie funkcji skrótu do wyznaczania kluczy kryptograficznych szyfrów blokowych zapewniających wydobywanie danych z bazy przechowującej dane w postaci zaszyfrowanej oraz do uwierzytelniania wiadomości.

1. Bezpieczeństwo w hurtowniach danych

W przypadku hurtowni danych mamy do czynienia z technologią integrującą rozproszone, autonomiczne i heterogeniczne dane w celu ich użycia do zarządzania operacyjnego i strategicznego. Jest to proces uzyskiwania informacji ze źródeł, integrowania ich, przekształcania, agregowania danych w celu ich przechowania w bazie danych hurtowni (Stokłosa 1999; Stokłosa, Bilski, Pankowski 2001).

Bezpieczeństwo danych należy mieć na uwadze zarówno wówczas, gdy dane z różnych baz danych są integrowane w hurtowni, jak i wówczas, gdy informacje pozyskuje się z hurtowni. Zarządzanie bezpieczeństwem w środowisku hurtowni danych obejmuje: identyfikację danych, ich klasyfikację, wyznaczanie (oszacowanie) wartości danych, identyfikację zagrożeń bezpieczeństwa danych, identyfikację środków ochrony i ich kosztu, wybór efektywnych środków ochrony, ocenę efektywności wprowadzonych zabezpieczeń (Warigon 1997).

Dane w hurtowni podlegają konfliktom bezpieczeństwa. Z jednej strony dane z hurtowni powinny być dostępne dla wszystkich zainteresowanych informacją. Z drugiej strony, interes firmy wymaga, by pewne wartościowe dane nie były przekazane nieupoważnionym użytkownikom. Wymogi bezpieczeństwa dla danych w hurtowni są analogiczne, jak w przypadku innych systemów informatycznych: poufność danych, ich integralność, kontrola dostępu do danych poprzez uwierzytelnianie użytkowników (procesów działających w ich imieniu).

Integralność danych w hurtowni zależy od integralności danych pochodzących z systemów źródłowych. Jednakże praca w środowisku hurtowni danych zapewniająca wysoką jakość jest skomplikowana ze względu na wymagania przetwarzania, odległość od źródeł i różne oczekiwania użytkowników hurtowni danych.

Kontrola dostępu do hurtowni danych jest szczególnie ważna skoro dane w hurtowni zawierają w sobie dane z wielu systemów i są wykorzystywane do podejmowania decyzji ponad barierami organizacyjnymi. Kontrolę dostępu do danych w hurtowni powinno się rozważać na kilku poziomach, w przypadku użytkowników, którzy: (i) będą mieć dostęp do procesów pozyskujących dane operacyjne, (ii) mają dostęp do procesów przekształcających dane operacyjne do formatu odpowiedniego do umieszczenia ich w hurtowni danych, (iii) mają dostęp do danych w hurtowni.

Implementacja polityki kontroli dostępu do danych w hurtowni może stwarzać problemy. Jednym z nich może być niezgodność zachodząca pomiędzy schematami kontroli dostępu do operacyjnych baz danych a modelem bezpieczeństwa hurtowni danych. Użytkownicy transakcyjnej bazy danych nie muszą być tymi samymi użytkownikami co użytkownicy hurtowni danych, tak więc polityka kontroli dostępu stosowana w przypadku transakcyjnej bazy danych może być inna niż odpowiednia polityka realizowana w hurtowni danych.

Uwierzytelnianie, kontrola dostępu i integralność są tradycyjnie brane pod uwagę we wszystkich systemach. Jednakże hurtownie danych, ze względu na swoją specyfikę, są szczególnie narażone na niebezpieczeństwa wynikające z możliwości nieuprawnionego uzyskiwania danych poprzez wnioskowanie. Hurtownie danych opierają się na sumowaniu lub agregacji informacji w celu wspomagania procesu podejmowania decyzji. Narzędzia eksploracji danych korzystają z technik wnioskowania w celu uzyskania informacji niedostępnej bezpośrednio z danych. Jednakże te same techniki mogą być użyte do niezamierzonego ujawnienia informacji. Jednym z bardziej interesujących podejść jest stosowanie narzędzi uzyskiwania danych w celu wykrycia potencjalnych problemów występujących w procesie wnioskowania (Hinke, Delugach, Wolf 1997). Jest to podejście przypominające wykrywanie intruzów w sieci komputerowej.

2. Agenci mobilni a hurtownia danych

Agenci mobilni są inteligentnymi programami, które mogą migrować w sieci komputerowej. Mają możliwość migracji i uruchamiania się w sposób autonomiczny i asynchroniczny. Są autonomiczne, gdyż są niezależne od innych programów, wiedzą dokąd się udać i co czynić w danej chwili. Są asynchroniczne w tym sensie, że uruchamiają się niezależnie od projektanta lub tego, który je wyekspediował. Jako takie wykonują zadania poprzez odwiedzanie komputerów, gdzie te zadania muszą być wykonane lokalne (Chang, Covaci 1997).

Agenci mobilni działający w środowisku hurtowni danych, stosujący techniki kryptograficzne w celu ochrony danych klasyfikowanych są ważnym instrumentem pozyskiwania danych. Ich działanie stwarza jednak pewne problemy. Każdy agent powinien mieć upoważnienie dostępu do danych od administratora. Istotnym problemem jest zapewnienie poufności, integralności i uwierzytelniania danych. W tym celu używa się technik kryptograficznych. Mechanizmy kryptograficzne powinny zapewnić: uwierzytelnianie agenta i użytkownika, poufność gromadzonych i transmitowanych danych, integralność danych. W takich przypadkach oprócz zaimple-

mentowania odpowiednich algorytmów i protokołów należy rozwiązać problem generowania dla nich kluczy. Klucze kryptograficzne można wytwarzać korzystając z danych środowiskowych, takich jak nazwy plików, katalogów, URI, adresy pocztowe, adresy IP, adresy X.400 itp.

Wcźmy pod uwagę hurtownię danych utworzoną w sieci (Stokłosa 1999). Trudność zaprojektowania protokołu korzystającego z danych środowiskowych do generacji kluczy polega na tym, że informacja dostępna dla agenta jest także dostępna dla potencjalnego intruza. Problem ten można rozwiązać korzystając z (kryptograficznych) funkcji skrótu: klucze kryptograficzne wykorzystywane w szyfrach i protokołach można konstruować za pomocą danych środowiskowych. Agent dysponuje algorytmem poszukiwania danych w środowisku niezbędnych do wygenerowania klucza kryptograficznego. Niech x będzie daną środowiskową, h niech będzie funkcją skrótu, a r_1 i r_2 niech będą liczbami (pseudo)losowymi. Istnieje kilka możliwości wyznaczania klucza kryptograficznego k (Riordan, Schneier 1998; Krawczyk, Bellare, Canetti 1997) przez agenta mobilnego, który w każdym przypadku pamięta wartość a (wartości a_i):

- jeśli h(x) = a , to k := x,
- jeśli h(x) = a, to k := h(x||r1)⊕r2,
- jeśli h(h(x)) = a, to k := h(x),
- jeśli h(x) = a, to k:= h(r1||h(r2||x)),
- jeśli h(xi) = ai dla i=1,2,...,n, to k := h(x1||x2||...||xn),

gdzie || jest operacją konkatenacji, a ⊕ operacją sumowania modulo 2 odpowiednich bitów binarnej reprezentacji liczby.

Tak więc agent generujący klucze za pomocą funkcji skrótu h nie używa klucza kryptograficznego, korzysta jedynie z ustalonej funkcji skrótu h i ewentualnie dodatkowo liczb pseudolosowych r_1 oraz r_2 w celu wyznaczenia klucza. Klucz ten można użyć do szyfrowania danych za pomocą szyfrów blokowych w celu zapewnienia poufności tych danych. Można go także wykorzystać do uwierzytelnienia danych, na przykład za pomocą funkcji skrótu HMAC (Krawczyk, Bellare, Canetti 1997).

3. Funkcje skrótu

3.1. Uwagi ogólne

Funkcję h nazywamy funkcją skrótu, jeśli cechuje się następującymi właściwościami (Menezes, van Oorschot, Vanstone 1997; Prencel 1999; Stokłosa, Bilski, Pankowski 2001):

- kompresją – h przekształca wejście m o dowolnej długości liczonej w bitach na wyjście $h(m)$ o ustalonej długości n, zwanej rozmiarem skrótu,
- łatwością obliczeń – dla danej funkcji h i wejścia x wartość $h(x)$ jest łatwo obliczalna (tj. w czasie wielomianowym),
- jednokierunkowością – dla prawie wszystkich wartości jest obliczeniowo trudne znalezienie argumentu dającego jako skrót tę wartość.

Działanie funkcji skrótu h dla wiadomości $m = m_1 m_2 \ldots m_r$ podzielonej na r bloków o ustalonej długości można opisać następująco:

$$H_0 = IV,$$

$$H_i = f(m_i, H_{i-1}) \text{ dla } i = 1,2,\ldots,r,$$

$$h(m) = g(H_r),$$

gdzie IV jest wartością początkową, f jest funkcją kompresji, a g – przekształceniem wyjściowym. W wyniku działania funkcji skrótu h na wiadomości m otrzymuje się skrót $h(m)$.

Istotne są także następujące cechy funkcji skrótu.

- słaba odporność na kolizje – jest obliczeniowo trudne znaleźć drugi argument, który daje taki sam skrót jak dowolnie wyspecyfikowany argument ,
- odporność na kolizje – jest obliczeniowo trudne znaleźć dwa różne argumenty m i m' dające ten sam skrót, czyli takie że $h(m) = h(m')$.

Właściwość druga jest silniejsza i w praktyce znalezienie choćby kilku par (m, m'), takich że $h(m) = h(m')$ i $m = m'$, powoduje, iż z rezerwą podchodzi się do zastosowań kryptograficznych takiej funkcji skrótu.

Jednokierunkowa funkcja skrótu (słabo jednokierunkowa funkcja skrótu) jest to funkcja skrótu mająca dodatkowo własność słabej odporności na kolizje.

Funkcja skrótu odporna na kolizje (silnie jednokierunkowa funkcja skrótu) to funkcja skrótu mająca dodatkowo własności słabej odporności na kolizje i odporności na kolizje.

Ze względów funkcjonalnych wyróżnia się dwie klasy funkcji skrótu:

- podklasę funkcji skrótu bez klucza,
- podklasę funkcji skrótu z kluczem (nazywanych także kodami uwierzytelniającymi wiadomość – MAC, ang. *Message Authentification Code*).

- MAC jest rodziną funkcji h_k z kluczem tajnym k, które charakteryzują się :
- łatwością obliczeń – dla znanej funkcji h_k, dla danego k i danego m skrót $h_k(m)$ jest łatwo obliczalny,
- kompresją – h_k odwzorowuje wejście m o dowolnej długości w bitach na wyjście $h_k(m)$ o ustalonej długości n bitów,
- odpornością obliczeniową – mając zero lub więcej par tekst-MAC $(m_i, h_k(m_i))$ jest obliczeniowo trudno znaleźć dowolną parę $(m, h_k(m))$ dla dowolnego nowego wejścia $m \neq m_i$ (własność ta zachodzi dla danego opisu rodziny funkcji h i każej ustalonej dozwolonej wartości k).

Funkcje skrótu bez klucza można podzielić na (Stokłosa, Bilski, Pankowski 2001):

- wykorzystujące szyfry blokowe,
- zaprojektowane wyłącznie do wyznaczania skrótu,
- wykorzystujące w swoim działaniu operacje arytmetyki modularnej.

Jako przykładowe funkcje skrótu bez klucza można wymienić: MD2, MD4, MD5, SHA-1, RIPEMD-160, MASH, Snefru, N-Hash, RIPE-MD, Haval, Petra-1 (Schneier 1995; Menezes, van Oorschot, Vanstone 1997; Najjar, Stokłosa 2001; Stokłosa, Bilski, Pankowski 2001).

Szyfrowanie nie zapewnia autentyczności danych. Jeśli zdeszyfrowana wiadomość jest sensowna, to nie oznacza to, iż mimo współdzielenia klucza wyłącznie z danym użytkownikiem, pochodzi ona od niego. Nie zawsze intruz musi znać tajny klucz, aby zmienić wiadomość. Aby wiadomość nie podlegała zmianom w sposób niewykrywalny, należy stosować funkcje skrótu.

Integralność danych jest własnością danych świadczącą o tym, że nie zostały one zmienione w sposób nieautoryzowany, od czasu utworzenia, w wyniku przechowywana lub transmisji. Aby zapewnić integralność przesyłanych danych nadawca oblicza wartość skrótu $h(m)$, dołącza go do danych i szyfruje tak powiększoną wiadomość za pomocą szyfru E_k z kluczem k, w wyniku czego uzyskuje szyfrogram $c = E_k(m||h(m))$. Jednak aby zapobiec atakowi na wiadomość z wybranym tekstem jawnym, zaleca się stosowanie skrótu MAC z kluczem k' $(k \neq k')$ uzyskując w ten sposób wiadomość $c' = E_k(m||h_{k'}(m))$. Wadą jest konieczność zarządzania dwoma kluczami i wykluczenie zależności pomiędzy E_k i $h_{k'}$. Szyfrowanie skrótu w wyrażeniu $c' = E_k(m||h_{k'}(m))$ zapobiega wyczerpującemu atakowi na klucz algorytmu MAC.

Uwierzytelnianie wiadomości polega na potwierdzeniu autentyczności pochodzenia danych ze względu na określone źródło wiadomości oraz potwierdzeniu integralności danych. Funkcje skrótu MAC, obok podpisów cyfrowych i znaczników wiadomości są jedną z metod uwierzytelniania danych.

3.2. Metody ataku na funkcje skrótu

Atak niezależny od algorytmu jest atakiem, który może być zastosowany do dowolnej funkcji skrótu, traktowanej jako czarną skrzynkę o znanej długości skrótu n bitów (i długości klucza dla MAC) oraz czasie wykonywania jednej operacji.

a. Kopiowany jest plik zawierający wartości funkcji skrótu h (często dostępny w systemach do odczytu dla każdego).
b. Poszukiwane są takie same wartości funkcji skrótu odpowiadające rzeczywistym hasłom i wartościom obliczonym dla słownika. Znalezienie takiej pary umożliwia zidentyfikowanie hasła jednego użytkownika.

Obroną przed tym atakiem jest przechowywanie dla każdego użytkownika rekordu składającego się z losowo wybranego ciągu o ustalonej długości oraz wartości funkcji skrótu dla tekstu będącego złożeniem hasła oraz tego ciągu.

Atak urodzinowy (Menezes, van Oorschot, Vanstone 1997; Schneier 1995) oparty jest na znanym w statystyce paradoksie urodzinowym. Atak Yuvala był jednym z pierwszych algorytmów, w którym zastosowano paradoks urodzinowy. Można go zastosować do wszystkich funkcji skrótu bez klucza w czasie $O(2^{r/2})$, gdzie r jest rozmiarem skrótu wyrażonym w bitach. Atak jest nieskuteczny dla funkcji dających skrót odpowiednio długi (np. 128, 192, 256 bitów) (Lenstra, Verheul 1999; Stokłosa 2001).

Ataki łańcuchowe wykorzystują iteracyjny charakter funkcji skrótu. Atakowana jest funkcja kompresji f, której punktem stałym jest para H_{i-1}, m_i, taka że $f(H_{i-1},m_i) = H_{i-1}$. Jeśli wstawi się dowolną liczbę identycznych bloków m_i, skrót wiadomości pozostaje niezmienny. Należy liczyć się z udanym atakiem, jeśli można znaleźć punkty stałe i doprowadzić do tego, aby zmienna łańcuchowa przyjmowała specyficzną wartość oraz jeśli dla dowolnej zmiennej łańcuchowej H_{i-1} można znaleźć bloki m_i dające punkty stałe.

Kryptoanalizę różnicową można przeprowadzić także dla funkcji skrótu z kluczem, poprzez badanie różnic wejściowych funkcji i odpowiadających im różnic wyjściowych (Biham, Shamir 1993).

4. Skuteczność funkcji skrótu wykorzystywanych w eksploracji danych

Skutecznośc pracy agenta mobilnego generującego klucze kryptograficzne (do celów zapewnienia poufności lub uwierzytelniania danych) za pomocą funkcji skrótu zależy od (i) odporności tych funkcji na różne (opisane w rozdz. 3.2) typy ataków oraz (ii) szybkości przetwarzania danych za pomocą wybranej funkcji skrótu.

Co do odporności funkcji skrótu na ataki kryptograficzne, to niektóre z nich okazały się podatne (np. MD2, MD4) (Menezes, van Oorschot, Vanstone 1997), inne są odporne na opisane formy ataku (Stokłosa, Bilski, Pankowski 2001).

Co do szybkości przetwarzania, to wykonano badania korzystając z (Reddmann 1999; Najjar, Stokłosa 2001); implementacje wykonano w Delphi 5. Testy przeprowadzono w środowisku Windows 2000 (Pentium III 800 MHz, 256 Mbytes RAM, HDD:IBM30G 7200 obrotów). Opracowane wyniki zawarto w tabeli 1.

Tab. 1. Szybkość przetwarzania wybranych funkcji skrótu

Lp.	Funkcja skrótu	Rozmiar skrótu [bity]	Średnia szybkość przetwarzania [Mbity/s]
1	MD4	128	58,12
2	MD5	128	44,63
3	SHA-1	160	25,70
4	Haval	192	22,91
5	Tiger	192	20,64
6	RIPEMD-160	160	20,30
7	Petra-1	192	15,64
8	Sapphire II	192	14,13
9	Square	128	7,46
10	Snefru	256	2,54

5. Wnioski

Z przeprowadzonych analiz i badań wynika, że wykorzystanie funkcji skrótu i zmiennych środowiskowych do generowania kluczy kryptograficznych może być skutecznym krokiem zmierzającym do zapewnienia poufności, integralności i uwierzytelnianie danych. Mając wygenerowany klucz kryptograficzny można go użyć do szyfrowania danych w celu zapewnienia ich poufności, a także do ich uwierzytelniania za pomocą funkcji skrótu z kluczem. Ponadto funkcja skrótu z kluczem może służyć, w przypadkach bardziej wymagających, jako mechanizm zapewnienia niezaprzeczalności wysłanych, przekazanych do transportu lub dostraczonych do odbiorcy danych (PN-ISO/IEC 1999).

Literatura

Biham E., Shamir A. (1993) *Differential Cryptoanalysis and the Data Encryption Standard*. Springer, Berlin.

Chang D. T., Covaci S. (1997) *The OMG mobile agent facility – a submission*. Rothermel K., Popescu-Zeletin R. (eds.), *Mobile Agents*. LNCS 1219, Springer, Berlin, 98–110.

Hinke T. H., Delugach H. S., Wolf R. (1997) *A framework for inference-directed data mining*. Samarati P., Sandhu R. S. (eds.), *Database Security: Status and Prospects* X. Chapman & Hall, London, 229–239.

Krawczyk H., Bellare M. Canetti R. (1997) *HMAC: keyed-hashing for message authentication*. RFC 2104.

Lenstra A. K., Verheul E. R. (1999) *Selecting cryptographic key sizes in commercial applications*, PricewaterhouseCoopers Cryptographic Centre of Excellence (CCE) Quarterly Journal, 3–9.

Menezes A. J., van Oorschot P. C., Vanstone S. A. (1997) *Handbook of Applied Cryptography*. CRC Press, Boca Raton, FL.

Najjar M., Stokłosa J. (2001) *Petra-1 cryptographic hash function*. Advanced Computer Systems (praca przyjęta na konferencję ACS 2001), Szczecin.

PN-ISO/IEC 13888-2 (1999) *Technika informatyczna – Techniki zabezpieczeń. Niezaprzeczalność – Mechanizmy wykorzystujące techniki symetryczne.*

Preneel B. (1999) *The state of the cryptographic hash functions*. Damgård I. (ed.), *Lectures on Data Security. Modern Cryptology in Theory and Practice*. LNCS 1561, Springer, Berlin, 158–182.

Reddmann H.(1999) *Delphi Encryption Compendium*, Part 1.

Riordan J., Schneier B. (1998) *Environmental key generation towards clueless agents*. Vigna G. (ed.), *Mobile Agents and Security*. Springer, Berlin, 15–24.

Schneier B. (1995) *Kryptografia dla praktyków. Protokoły, algorytmy i programy źródłowe w języku C*. WNT, Warszawa.

Stokłosa J. (1999), *Eksploracja danych za pomocą kryptograficznych agentów mobilnych*. Gospodarowicz A. (red.), *Zastosowanie rozwiązań informatycznych w bankowości*. Wydawnictwo Akademii Ekonomicznej, Wrocław, 332–340.

Stokłosa J. (2001) *Kryptografia w sieci Internet*. Zasępa T. (red.), *Internet – fenomen społeczeństwa informacyjnego*. Edycja Świętego Pawła, Częstochowa, 489–504.

Stokłosa J., Bilski T., Pankowski T. (2001) *Bezpieczeństwo danych w systemach informatycznych*. PWN, Warszawa-Poznań .

Warigon S. (1997) *Data warehouse control and security*. Association of College and University Auditors LEDGER, Vol. 41, No. 2, 3–7.

ISSN 0208-8028
ISBN 83-85847-59-6

W celu uzyskania bliższych informacji i zakupu dodatkowych egzemplarzy prosimy o kontakt z Instytutem Badań Systemowych PAN
ul. Newelska 6, 01-447 Warszawa
tel. 837-35-78 w. 241 e-mail: bibliote@ibspan.waw.pl