УДК 004.056 : 004.424.47

В.А. Лужецький[1], Ю.В. Баришев[1], О.В. Оводенко[2]
[1] Вінницький національний технічний університет, м. Вінниця
кафедра захисту інформації
E-mail: yuriy.baryshev@gmail.com
[2] Донецький національний технічний університет, м. Донецьк
кафедра радіотехніки та захисту інформації
E-mail: ovoda@i.ua

МЕТОДИ ФОРМУВАННЯ ВЕКТОРІВ КЕРУВАННЯ ДЛЯ КЕРОВАНОГО БАГАТОКАНАЛЬНОГО ХЕШУВАННЯ ДАНИХ

Анотація
Лужецький В.А., Баришев Ю.В., Оводенко О.В. Методи формування векторів керування для керованого багатоканального хешування даних. *Проаналізовано відомі методи формування векторів керування. Запропоновано функції ущільнення та узагальнену конструкцію багатоканального керованого хешування, для яких розроблено методи формування векторів керування.*
Ключові слова: *кероване хешування, багатоканальність, гнучкість.*

Вступ. Для збільшення стійкості хеш-функцій до загальних атак [1], які використовують властивості математичних моделей ітерацій, необхідно здійснити не лише перегляд криптографічних примітивів, які використовуються у функції ущільнення, але й підходів до хешування. Підходи, які використовують проміжні хеш-значення значно більшої довжини (мінімум в два рази), дозволяють збільшити стійкість [2], однак збільшення розрядності проміжних хеш-значень без зміни розміру блоку даних породжує зменшення швидкості хешування. Саме тому реалізація концепцій хешування, які використовують інші параметри хешування для збільшення швидкості хешування зі збереженням стійкості, є актуальною. Одним з таких напрямків розвитку хеш-функцій є кероване хешування [3-5].

Кероване хешування дозволяє розпаралелити процес обчислення проміжних хеш-значень, що дозволить збільшити швидкість хешування на багатоядерних обчислювальних платформах [3, 6]. Особливої значущості це набуває, якщо враховувати необхідність використання хешування для додатків, які працюють в режимі реального часу [6]. Крім того, багатоканальність хешування дозволить легко модифікувати відомі методи хешування для формування вихідних хеш-значень довільної довжини.

Реалізація керованого хешування пов'язана з необхідністю розв'язку низки задач, зокрема задачі формування векторів керування.

Метою даного дослідження є підвищення швидкості хешування за рахунок реалізації концепції керованого хешування шляхом розробки методів формування векторів керування.

Для досягнення мети необхідно розв'язати такі задачі:

- визначити конструкції керованого хешування та функції ущільнення, що їх реалізують;
- проаналізувати відомі методи формування векторів керування щодо можливості їх застосування для даних конструкцій та функції ущільнення;
- розробити методи формування векторів керування для реалізації керованого хешування.

Узагальнена конструкція керованого хешування та функції ущільнення, що її реалізують. Конструкції керованого хешування відрізняються від відомих конструкцій наявністю додаткового параметра – вектора керування. Оскільки, як зазначалося вище, сучасне хешування повинно передбачати розпаралелення обчислень, воно повинно бути багатоканальним. При цьому зав'язування каналів у керованому хешування можливе шляхом використання проміжних хеш-значень і як аргументів функції ущільнення, і як аргументів функції формування вектора керування. Нехай q – загальна кількість каналів, k – кількість каналів, які зав'язуються з даним за допомогою функції ущільнення, а ϕ – кількість каналів, які зав'язуються з даним за допомогою функції формування вектора керування, тоді узагальнена конструкція керованого хешування буде такою:

$$\begin{cases} h_i^{(1)} = f_{v_i^{(1)}}\left(h_{i-1}^{(1)}, h_{i-1}^{(2)}, \ldots, h_{i-1}^{(k)}, m_i\right) \\ h_i^{(2)} = f_{v_i^{(2)}}\left(h_{i-1}^{(2)}, h_{i-1}^{(3)}, \ldots, h_{i-1}^{(k+1)}, m_i\right) \\ \ldots \\ h_i^{(q)} = f_{v_i^{(q)}}\left(h_{i-1}^{(q)}, h_{i-1}^{(1)}, \ldots, h_{i-1}^{(k-1)}, m_i\right) \\ v_i^{(1)} = g\left(h_{i-1}^{(q)}, h_{i-1}^{(q-1)}, \ldots, h_{i-1}^{(q-\phi+1)}\right) \\ v_i^{(2)} = g\left(h_{i-1}^{(1)}, h_{i-1}^{(q)}, h_{i-1}^{(q-1)}, \ldots, h_{i-1}^{(q-\phi+2)}\right) \\ \ldots \\ v_i^{(q)} = g\left(h_{i-1}^{(q-1)}, h_{i-1}^{(q-2)}, \ldots, h_{i-1}^{(q-\phi)}\right) \end{cases}, \quad (1)$$

де $h_i^{(j)}$ – проміжне хеш-значення, отримане у j-му каналі ($j = \overline{1,q}$) на i-й ітерації ($i = \overline{1,l}$); m_i – i-й блок даних; $f_{v_i^{(j)}}(\cdot)$ – функція ущільнення, що забезпечує сталу довжину вихідного значення; $v_i^{(j)}$ – вектор керування, який визначає параметри перетворення функції ущільнення $f_{v_i^{(j)}}(\cdot)$ у j-му каналі на i-й ітерації ($i = \overline{1,l}$); $g(\cdot)$ – функція формування вектора керування.

Для конструкції (1) доцільно, щоб виконувалось співвідношення між параметрами $k < (q - \phi)$, оскільки інакше деякі проміжні хеш-значення будуть використовуватись і як джерела формування вектора керування, який визначає параметри перетворень у функції ущільнення, і як аргументи над якими виконуються ці керовані операції. Це може спричинити залежність між значенням аргументу функції ущільнення та параметрами керованої операції, яка над ним виконується, що полегшить криптоаналіз таких функцій. За допомогою зміни кількості каналів $(k + \phi)$, що пов'язують з даним, пропонується змінювати параметр стійкість/швидкість для того, щоб він найкращим чином відповідав вимогам, які ставить перед хешуванням конкретна задача. Так, зменшуючи $(k + \phi)$, зменшуємо кількість аргументів у функції формування вектора керування та функції ущільнення, що зменшує кількість операції, які над ними виконуються, тим самим збільшуючи швидкість. І навпаки – збільшуючи $(k + \phi)$, збільшуємо стійкість зав'язування каналів.

Вихідне значення хеш-функцій конструкції (1) формуватиметься як конкатенація вихідних хеш-значень кожного каналу $h_l = h_l^{(1)} \| h_l^{(2)} \| \ldots \| h_l^{(q)}$. Очевидно, що для досягнення однакового впливу вихідних хеш-значень з кожного каналу на вихідне хеш-значення, отримане внаслідок процесу хешування, необхідно, щоб усі проміжні хеш-значення $h_i^{(j)}$ мали однакову довжину, що дорівнює n/q бітів, де n – довжина вихідного хеш-значення h_l.

Як функції ущільнення пропонується обрати дещо модифіковані логічні функції, які використовуються у стандарті хешування SHA-2 [7], наприклад таку:

$$\begin{aligned} h_i^{(j)} = &\left(m_i >>> u_i^{(j)(m1)} \wedge h_{i-1}^{(j)} >>> u_i^{(j)(h1)}\right)\oplus \\ \oplus &\left(\sim m_i >>> u_i^{(j)(m1)} \wedge h_i^{(j+1)} >>> u_i^{(j)(h2)}\right)\oplus \\ \oplus &\left(m_i >>> u_i^{(j)(m2)} \vee h_{i-1}^{(j+2)} >>> u_i^{(j)(h3)}\right)\oplus \\ \oplus &\left(\sim m_i >>> u_i^{(j)(m2)} \vee h_i^{(j+3)} >>> u_i^{(j)(h4)}\right) \end{aligned}, \quad (2)$$

де $u_i^{(j)(x\kappa)}$ – кількість бітів на яку зсувається змінна x у к -й позиції у функції ущільнення на i-й ітерації в j-му каналі хешування.

Як керований параметр перетворень у функції ущільнення пропонується обрати кількість бітів $u_i^{(j)(x\kappa)}$, на яку відбувається зсув аргументів, оскільки таке керування легко реалізувати як апаратно, так і програмно. У даному випадку вектор керування представлятиме собою конкатенацію параметрів $u_i^{(j)(x\kappa)}$, зокрема для функції ущільнення (2) вектор керування буде таким: $v_i^{(j)} = u_i^{(j)(m1)} \| u_i^{(j)(h1)} \| u_i^{(j)(h2)} \| u_i^{(j)(m2)} \| u_i^{(j)(h3)} \| u_i^{(j)(h4)}$. Хоча підхід до побудови функції ущільнення (2) дозволяє реалізувати функцію ущільнення для будь-яких значень параметру k узагальненої конструкції багатоканального керованого хешування (1).

Аналіз відомих методів формування векторів керування. У роботі [4] пропонується використовувати для керування дещо модифіковані вхідні дані. Однак саме це дозволить зловмиснику впливати на процес хешування, якщо він матиме змогу нав'язувати свої блоки даних. Саме тому даний підхід не може використовуватись у даному дослідженні.

Автором роботи [5] запропоновано використовувати низку "зовнішніх" характеристик даних для організації керованості, наприклад ім'я файлу, його адресу тощо. Використання таких характеристик передбачає жорстке прив'язування даних до носіїв, на яких вони зберігаються, операційних систем, способів кодування символів тощо. Саме тому, зміна деяких з цих параметрів без зміни самих даних спричинить незбіжність хеш-значень, що істотно ускладнює практичне використання такого хешування.

У роботі [3] як джерела формування векторів керування пропонується, зокрема, використовувати ключ, що розгортається. Такий підхід дозволяє отримувати на кожній ітерації вектор керування необхідної довжини. Однак цей підхід має недолік – недоцільність його застосування при безключовому хешуванні, оскільки саме відкритість ключа дозволить зловмиснику визначити (і передбачити) параметри перетворень, які відбуватимуться над кожним блоком даних, що зведе нанівець сенс реалізації керованості перетворень.

Крім ключа, у роботі [3] пропонується формувати вектор керування на основі блоків даних, а також результату проміжного раундового перетворення всіх вхідних даних. Недоліком цих методів є те, що вони розроблялися для керованих підстановочно-престановочних мереж та залежали від довжини проміжних хеш-значень, а відтак їх застосування для функцій ущільнення (2), для яких ця залежність не бажана, проблематично. Саме тому останній підхід, запропонований у [3], доцільно використати у даних дослідженнях, однак повинні бути розроблені нові методи формування вектора керування для функцій ущільнення виду (2).

Методи формування векторів керування. Для ускладнення зловмиснику криптоаналізу хеш-функції необхідно, щоб кожен з аргументів функції формування векторів керування $g(\cdot)$ мав однаковий вплив на вихідний результат. З точки зору простоти реалізацій найкращим варіантом цієї функцій є той, що використовує операцію конкатенації проміжних хеш-значень для отримання вектора керування:

$$v_i^{(j)} = h_{i-1}^{(j+x_1)} \| h_{i-1}^{(j+x_2)} \| \ldots \| h_{i-1}^{(j+x_\phi)}, \quad (3)$$

де $x_1, x_2, \ldots, x_\phi$ – деякі константи, що залежать від конструкції хешування, що забезпечують мінімальне значення затримки у поширенні опосередкованого впливу каналів один на одного (для конструкції (1): $x_1 = -1$, $x_2 = -2$, …, $x_\phi = -\phi$).

Щоб використовувати функцію формування вектора керування (3) необхідно виконання такої умови:

$$n_v = \frac{\phi \cdot n}{q}, \tag{4}$$

де n_v – бітова довжина вектора керування.

Умова (4) є вельми жорсткою й істотно обмежує користувачів хеш-функції у виборі параметрів n, q, ϕ, а через них опосередковано і на вибір значення параметра k. Саме тому для "пом'якшення" цієї умови пропонується отримувати вектор керування шляхом конкатенації та побітового додавання проміжних хеш-значень. Нехай для деяких параметрів $a, b \in \mathbf{N}$ виконується така умова:

$$\begin{cases} \dfrac{n_v \cdot q}{n} = a \\ \text{НСД}(a, \phi) = a, \\ \dfrac{\phi}{a} = b \end{cases} \tag{5}$$

де $\text{НСД}(a, \phi)$ – найменший спільний дільник чисел a, ϕ.

З урахуванням умови (5) функція формування вектора керування $g(\cdot)$ буде мати вигляд:

$$\begin{aligned} v_i^{(j)} = &\left(h_{i-1}^{(j+x_1)} \parallel h_{i-1}^{(j+x_2)} \parallel \ldots \parallel h_{i-1}^{(j+x_a)}\right) \oplus \left(h_{i-1}^{(j+x_{a+1})} \parallel h_{i-1}^{(j+x_{a+2})} \parallel \ldots \parallel h_{i-1}^{(j+x_{2a})}\right) \oplus \ldots \oplus \\ &\oplus \left(h_{i-1}^{(j+x_{b-a+1})} \parallel h_{i-1}^{(j+x_{b-a+2})} \parallel \ldots \parallel h_{i-1}^{(j+x_\phi)}\right) \end{aligned}. \tag{6}$$

Умова (5) накладає не такі жорсткі обмеження, як умова (4), однак її впровадження значно обмежує кількість можливих практичних реалізацій конструкції (1), тому її пропонується послабити таким чином:

$$\begin{cases} \dfrac{n_v \cdot q}{n} = a \\ \dfrac{\phi - (\phi \bmod a)}{a} = b \end{cases}. \tag{7}$$

Відповідно до умови (7) у функції формування вектора керування (6) останній доданок буде меншої довжини, ніж n_v, тобто формула (6) набуде такого вигляду:

$$\begin{aligned} v_i^{(j)} = &\left(h_{i-1}^{(j+x_1)} \parallel h_{i-1}^{(j+x_2)} \parallel \ldots \parallel h_{i-1}^{(j+x_a)}\right) \oplus \left(h_{i-1}^{(j+x_{a+1})} \parallel h_{i-1}^{(j+x_{a+2})} \parallel \ldots \parallel h_{i-1}^{(j+x_{2a})}\right) \oplus \ldots \oplus \\ &\oplus \left(h_{i-1}^{(j+x_{b+1})} \parallel h_{i-1}^{(j+x_{b+2})} \parallel \ldots \parallel h_{i-1}^{(j+x_\phi)}\right) \end{aligned}. \tag{8}$$

Підемо далі у напрямку послаблення умов. Припустимо, що довжина вектора керування n_v не ділиться націло на довжину проміжного канального хеш-значення n/q. В такому випадку замість умови (7) отримуємо:

$$\begin{cases} n_v = \alpha \cdot \dfrac{n}{q} + \beta \\ \alpha < \phi \\ \beta < \dfrac{n}{q} \end{cases}, \tag{9}$$

де $\alpha, \beta \in \mathbf{N}$ – деякі константи ($\beta < (n/q)$).

При виконанні умови (9) функцію формування вектора керування пропонується реалізовувати, розділяючи проміжні хеш-значення з деяких каналів на частини, зокрема з $(\alpha + 1)$-го каналу. Даний підхід формалізується таким чином:

$$v_i^{(j)} = \left(h_{i-1}^{(j+x_1)} \| h_{i-1}^{(j+x_2)} \| \ldots \| h_{i-1}^{(j+x_\alpha)} \| \overbrace{h_{i-1}^{(j+x_\alpha+1)}}^{\beta} \right) \oplus \left(\underbrace{h_{i-1}^{(j+x_\alpha+1)}}_{n/q-\beta} \| h_{i-1}^{(j+x_{\alpha+2})} \| \ldots \right) \oplus \ldots, \quad (10)$$

де $\overbrace{h_{i-1}^{(j+x_\alpha+1)}}^{\beta}$ – старші β бітів змінної $h_{i-1}^{(j+x_\alpha+1)}$; $\underbrace{h_{i-1}^{(j+x_\alpha+1)}}_{n/q-\beta}$ – молодші $(n/q - \beta)$ бітів змінної $h_{i-1}^{(j+x_\alpha+1)}$.

На рисунку 1 наведено схемну інтерпретацію формули (10).

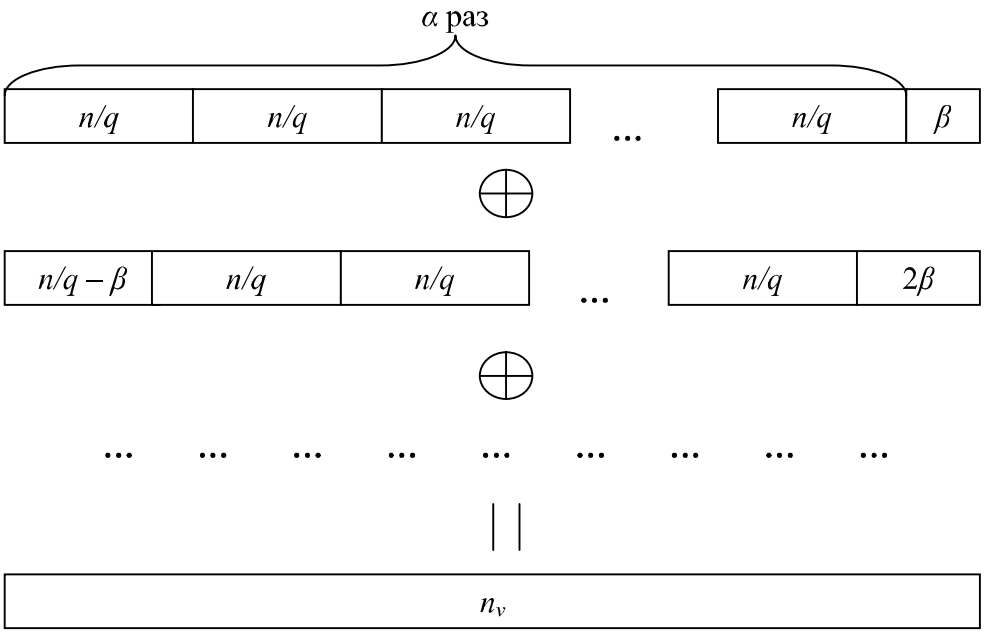

Рисунок 1 – Схема формування вектора керування

Схема, зображена на рис. 1, є частковим випадком (10), коли умова (9) перетворюється у таку:

$$\begin{cases} n_n = \alpha \cdot \dfrac{n}{q} + \beta \\ \alpha < \phi / 2 \\ 2\beta < \dfrac{n}{q} \end{cases}. \quad (11)$$

Використання методу, описаному у вигляді формул (9) та (10), дозволяє формувати вектори керування, довжина яких не залежить від довжини проміжних хеш-значень, кількості каналів, зав'язаних з даним за допомогою вектора керування.

Висновки. Необхідність створення нових концепцій хешування, які не використовуватимуть однакові параметри перетворень на кожній ітерації хешування обумовлюється появою загальних атак. Саме тому пропонується розвивати концепцію керованого хешування, яка дозволяє усунути ці недоліки сучасного хешування. Аналіз відомих методів формування вектора керування показав, що вони не пристосовані до багатоканального хешування та не дозволяють адаптуватися до змін параметрів хешування, що важливо при створенні методів

хешування, які дозволятимуть легко варіювати критерій швидкість/стійкість з метою підбору найкращого варіанту хешування для конкретної практичної задачі (до таких методів хешування належить зокрема і метод хешування, розглянутий в даній статті). Саме тому, запропоновано методи формування вектора керування, які накладають різні за жорсткістю умови на параметри конструкції хешування. Причому менші обмеження на значення параметрів породжують більшу складність реалізації функції формування вектора керування. Зокрема, запропоновано метод формування вектора керування, який не накладає жодних умов на параметри конструкції хешування (крім тих, що випливають природно з самої конструкції). Поєднання запропонованих методів формування векторів керування з функціями ущільнення (зокрема виду (2)) дозволяє реалізувати кероване багатоканальне хешування, що буде гнучким до зміни параметрів своєї конструкції.

Література

1. Gauravaram P. Cryptographic Hash Functions: Cryptanalysis, Design and Applications. Thesis submitted in accordance with the regulations for Degree of Doctor of Philosophy / Praveen Gauravaram. – 2009. – 298 с. – Режим доступу до дисертаційної роботи: http://eprints.qut.edu.au/16372/1/Praveen_Gauravaram_Thesis.pdf.

2. Lucks S. Design Principles for Iterated Hash Functions / Stefan Lucks // Cryptology ePrint Archive. – 2004. – 22 с. Режим доступу до ресурсу : http://eprint.iacr.org/2004/253.pdf

3. Молдовян Н.А. Криптография: от примитивов к синтезу алгоритмов / Н.А. Молдовян, А.А. Молдовян, М.А. Еремеев. – С-Пб.: БХВ-Петербург, 2004. – 448 с.

4. Zijie Xu. Dynamic SHA. / Zijie Xu. // Cryptology ePrint Archive. – 2007. – 34 с. – Режим доступу до ресурсу: http://eprint.iacr.org/2007/476.pdf

5. Rules-Driven Hash Building / Oscar Thorbjornsson: патент США на корисну модель № 0103715 A1 МПК H 04 L 9/28, G 06 F 12/14, G 06 F 17/30; заявник та патентовласник Itnternational Business Machines Corporation. – 11/875,117; заявл. 19.10.07; опубл. 23.04.09.

6. Корченко А.Г. Построение систем защиты информации на нечетких множествах. Теория и практические решения / А. Г. Корченко. – К.: "МК-Пресс", 2006. – 320 с.

7. Secure Hash Standard: Federal Information Processing Publication Standard Publication 180-3. – Gaithersburg, 2008. – 27 с. – Режим доступу до стандарту: http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf.

Надійшла до редакції:
21.03.2011

Рекомендовано до друку:
д-р техн. наук, проф. Чичикало Н.І.

Abstract

Luzhetsky V., Baryshev Y., Ovodenko O. Methods of driving vector forming for the driven multipiped data hashing. *The known methods of driving vector forming are analyzed. The reduction function and the driven multipiped hash construction are proposed, for which methods of driving vector forming were developed.*

Key words: *driven hashing, multipipe, flexibility.*

Аннотация

Лужецкий В.А., Барышев Ю.В., Оводенко А.В. Методы формирования векторов управления для управляемого многоканального хеширования данных. *Проанализированы известные методы формирования векторов управления. Предложены функция сжатия и обобщенная конструкция многоканального управляемого хеширования, для которых разработаны методы формирования векторов управления.*

Ключевые слова: *управляемое хеширование, многоканальность, гибкость.*