14. Lenstra H. W. Analysis and Comparison of Some Integer Factoring Algorithms, in Computational Methods in Number Theory [Text] / H. W. Lenstra, Jr. and R. Tijdeman, eds.// Math. Centre Tract 154 - 1946 - pp. 89-139.
15. Proos, J. Shor's discrete logarithm quantum algorithm for elliptic curves [Text] / Proos J., Zalka C. // QIC. – 2003. – Vol.4. – pp. 317-344.
16. Hoffstein, J. NTRU: A ring-based public key cryptosysytem [Text] / J. J. Pipher and J. Silverman // ANTS III. – 1998 – Vol.1423 – pp. 267-288.
17. Silverman, J. A Meet-The Middle Attack on an NTRU Private Key [Text] / J. Silverman, J. Odlyzko // NTRU Cryptosysytems. - Technical Report, NTRU Report - 2003 - 004, Version 2. – 7 p.
18. Ludwig, C. A faster lattice reduction method using quantum search [Text] / C. A. Ludwig // Algo Comput, - 2003 – Vol.2906. – pp. 199-208.
19. Wang, X. A quantum algorithm for searching a target solution of fixed weight [Text] / Wang, X. W. , S. Bao and X. Q. Fu// Chinese Sci Bull. - 2010.- Vol.55(29). – pp.484-488.
20. Xiong, Z. An Improved MITM Attack Against NTRU [Text] / Z. Xiong Wang J. , Wang Y. , Zhang T. , Chen L. // International Journal of Security and Its Applications. – 2012. - Vol. 6, No. 2. – pp. 269-274.
21. Wang, H. An efficient quantum meet-in-the-middle attack against NTRU-2005 [Text] / Wang Hong, MA Zhi, MA ChuanGui // Chinese Science Bulletin. – 2013. - Vol. 58, No.28-29. – pp.3514-3518.

Обґрунтовується вибір циклових функцій у схемі доказовою стійкого ключового універсального гешування, пропонується модель і метод формування кодів контролю цілісності та автентичності даних на основі модулярних перетворень, алгоритм зниження обчислювальної складності реалізації схем гешування з використанням циклових функцій. Розроблений вдосконалений алгоритм UMAC забезпечує необхідні показники колізійних властивостей універсального гешування, доказовий рівень стійкості і високі показники швидкодії

Ключові слова: коди контролю цілісності та автентичності даних, модулярні перетворення, універсальні класи геш-функцій

Обосновывается выбор цикловых функций в схеме доказуемо стойкого ключевого универсального хеширования, предлагается модель и метод формирования кодов контроля целостности и аутентичности данных на основе модулярных преобразований, алгоритм снижения вычислительной сложности реализации схем хеширования с использованием цикловых функций. Разработанный усовершенствованный алгоритм UMAC обеспечивает требуемые показатели коллизионных свойств универсального хеширования, доказуемый уровень стойкости и высокие показатели быстродействия

Ключевые слова: коды контроля целостности и аутентичности данных, модулярные преобразования, универсальные классы хеш-функций

УДК 681.3.06 (0.43)

УСОВЕРШЕНСТВОВАННЫЙ АЛГОРИТМ UMAC НА ОСНОВЕ МОДУЛЯРНЫХ ПРЕОБРАЗОВАНИЙ

С. П. Евсеев
Кандидат технических наук, доцент*
E-mail: Evseev_Serg@inbox.ru.
О. Г. Король
Преподаватель*
E-mail: korol_o@mail.ru
В. В. Огурцов
Кандидат экономических наук, доцент*
*Кафедра информационных систем
Харьковский национальный экономический университет им. С. Кузнеца
пр. Ленина 9-а, Харьков, Украина, 61166
E-mail: Vitalii.Ohurtsov@hneu.net или vetalreal@ukr.net

1. Введение

Проведенные исследования показали, что использование модулярных преобразований позволяет реализовать доказуемо стойкое хеширование информации, удовлетворяющее коллизионным свойствам универсальных хеш-функций. Доказуемо безопасный уровень стойкости обосновывается сведением

задачи нахождения прообраза и/или задачи восстановления секретных ключевых данных к решению одной из известных теоретико-сложностных задач [1 – 4].

В то же время универсальное хеширование с использованием модулярных преобразований обладает существенным недостатком – высокой вычислительной сложностью формирования хеш-кодов.

Фактически, для каждого информационного блока необходимо выполнить операцию модульного возведения в степень, что при соответствующих порядках модуля преобразований существенно повышает время хеширования информационной последовательности. Перспективным направлением в этом смысле является разработка многослойных схем универсального хеширования с использованием модулярных преобразований на последнем, заключительном этапе формирования хеш-кода. Это, как показано ниже, с одной стороны, обеспечивает высокие коллизионные свойства результирующей схемы формирования кодов контроля целостности и аутентичности данных, с другой стороны – обеспечивает высокие показатели быстродействия и доказуемый уровень безопасности используемых преобразований.

2. Цели и задачи исследования

Целью статьи является исследование свойств модулярных преобразований и построенных на их основе методов бесключевого хеширования информации (MASH-1 и MASH-2), а также методов ключевого хеширования, построенных на основе алгоритмов MASH-1 и MASH-2 при смене вектора инициализации в качестве секретных ключевых данных, позволяющих разработать и теоретически обосновать новые схемы ключевого хеширования, обеспечивающие как высокие коллизионные свойства (с сохранением свойств универсального хеширования), так и высокие показатели безопасности.

В статье исследованы различные виды цикловых функций в схеме итеративного хеширования, построенные с использованием модулярных преобразований, задача инвертирования которых эквивалентна решению одной из известных теоретико-сложностных задач.

Установлено, что применение цикловых функций на модулярных преобразований позволит строить универсальные и строго универсальные классы хеширующих функций, которые, с одной стороны, позволят обеспечить высокие показатели безопасности и применимость модели доказуемой стойкости, с другой стороны, при выполнении определенных ограничений на параметры модулярных преобразований обеспечить высокие коллизионные свойства. На основе полученных результатов проведенных исследований обоснован выбор цикловых функций в схеме доказуемо стойкого ключевого универсального хеширования, а также предлагается модель и метод каскадного формирования кодов контроля целостности и аутентичности данных (MAC) с использованием модулярных преобразований. В основе предлагаемой модели лежит усовершенствованная многослойная схема универсального хеширования UMAC с использованием на последнем, завершающем этапе модулярных преобразований.

3. Исследование свойств модулярных преобразований и методов хеширования информации на их основе

Модулярные преобразования широко используются при построении криптографических алгоритмов преобразования информации, в том числе при построении ассиметричных средств защиты информации и протоколов распространения ключевых данных [5 – 10], для формирования псевдослучайных последовательностей [6 – 8], методов хеширования и других механизмов защиты информации [6 – 8].

Проведенный анализ [4, 6 – 9] показывает, что модулярные преобразования применяются на сегодняшний день при построении бесключевых хеш-функций. Так в четвертой части международного стандарта ISO/IEC 10118-4 определены две бесключевые функции хеширования MASH-1 и MASH-2, которые используют модулярную арифметику, а именно модульное возведение в степень для построения хеш-кода [9].

Само название функций MASH-1 и MASH-2 происходит от скрашенного Modular Arithmetic Secure Hash (безопасное хеширование на основе модулярной арифметики), подчеркивающего применение модулярных преобразований при формировании хеш-образа.

В основе построения хеш-функций MASH-1 и MASH-2 лежит использование итеративной цикловой функции, которая определяется через модулярное возведение в степень (в простейшем случае через модулярное возведение в квадрат).

В данном случае используются RSA-подобные модули N, длина которых обеспечивает необходимую стойкость. Число N должно быть трудно разложимым на множители, на чем и основывается стойкость алгоритма.

Размер модуля N определяет длину блоков обрабатываемого сообщения, а также размер хеш-кода (например, 1025-битний модуль обеспечивает формирование 1024-битного хеш-кода).

В определенных международным стандартом ISO/IEC 10118-4 хеш-функциях MASH-1 и MASH-2 использованы следующие цикловые функции:

$$f(x_i, H_{i-1}) = \left(\left(\left((x_i \oplus H_{i-1}) \vee A\right)^2 \bmod N\right) \perp n\right) \oplus H_{i-1} \qquad (1)$$

и

$$f(x_i, H_{i-1}) = \left(\left(\left((x_i \oplus H_{i-1}) \vee A\right)^{2^8+1} \bmod N\right) \perp n\right) \oplus H_{i-1}, \qquad (2)$$

соответственно, где: $\vee$ – операция побитного логического ИЛИ; $\oplus$ – суммирование по модулю 2 (XOR); $\perp n$ – сохранение младших n-разрядов m-разрядного результата.

В табл. 1 приведены результаты сравнительного анализа показателей эффективности некоторых бесключевых функций хеширования, в том числе и хеш-функции на модулярной арифметике MASH-1 и MASH-2 [7].

Таблица 1

Результаты сравнительного анализа некоторых бесключевых функций хеширования

Хеш-функция	Длина хеш-кода	Применяемые преобразования	Скорость обработки данных	Модель безопасности (по NESSIE)
SHA-2	256, 384, 512	логические и арифметические	108..10^9 бит/с	Практическая секретность (Practical Security)
Whirlpool	512	В конечных полях Галуа	107..10^8 бит/с	Практическая секретность (Practical Security)
ГОСТ 34311-95	256	Блочное симметричное шифрование	107..10^8 бит/с	Практическая секретность (Practical Security)
RIPEMD-160	160	Логические и арифметические	108..10^9 бит/с	Практическая секретность (Practical Security)
MASH-1	*	Модулярное возведение в квадрат	105..10^6 бит/с	Доказуемая безопасность** ("Provable" Security)
MASH-2	*	Модулярное возведение в степень 28+1 = 257	104..10^5 бит/с	Доказуемая безопасность** ("Provable" Security)

* Определяется размерностью модуля преобразований
** Если параметры модульного возведения в степень соответствуют ограничениям на RSA-подобные системы

Проведенный анализ показал, что основным недостатком функций хеширования MASH-1 и MASH-2 является низкая скорость формирования хеш-кода. Фактически она определяется скоростью RSA-подобного шифрования, которое на 2 – 3 порядка ниже скорости шифрования современными блочно-симметричными шифрами.

Тем не менее, по причине наличия возможности использования, существующих программных и аппаратных средств модулярной арифметики, применяемых в несимметричных RSA-подобных криптосистемах, а также по причине возможности обеспечения доказуемого уровня безопасности (по классификации моделей безопасности NESSIE) рассматриваемые бесключевые хеш-функции MASH-1 и MASH-2 были стандартизированы [4, 7, 9].

Следует, однако, отметить, что алгоритмы хеширования MASH-1 и MASH-2 не в полной мере соответствуют ограничениям на параметры модульного возведения в степень, которые установлены для RSA-систем (а соответственно и обеспечиваемой модели доказуемой безопасности). Действительно, по спецификации криптографической RSA-системы, обеспечивающей доказуемую безопасность (по модели безопасности NESSIE) значение модульной экспоненты e должно быть выбрано из условия

$$\gcd(e,\varphi(N)) = 1, \qquad (3)$$

где gcd(x,y) – наибольший общий делитель чисел x и y.

Значение экспоненты e не должно содержать общих делителей с числом (значением функции Эйлера) $\varphi(N)$:

$$\varphi(N) = (p - 1)(q - 1),\ N = pq.$$

По спецификации алгоритмов MASH-1 и MASH-2 это условие может не выполняться. Таким образом, модель доказуемой безопасности (по классификации моделей безопасности NESSIE) может быть применена к алгоритмам MASH-1 и MASH-2 только условно. Полного соответствия задачи нахождения прообраза или секретного ключа в схеме хеширования и теоретико-сложностной задачи факторизации (или задачи RSA) не наблюдается.

Рассмотрим цикловые функции MASH-1 и MASH-2 на предмет построения ключевых универсальных хеширующих функций, и вариант хеширования, когда начальное состояние (вектор инициализации) задается некоторым ключевым правилом, т.е. выберем H_0 = Key.

В этом случае имеем некоторый класс хеш-функций, зависящих от параметра Key.

Для проведения экспериментальных исследований выбраны следующие параметры: p = 17, q = 19, N = 323. Исследования состояли в проверке условий универсального хеширования при полном переборе всех значений векторов инициализации (Key = 0,...,2^m-1, m = 8) по выборке из генеральной совокупности значений информационных блоков в соответствии с разработанной методикой статистических исследований коллизионных свойств, описанной в работе [3]. Полученные результаты сведены в табл. 2.

Таблица 2

Результаты исследований коллизионных свойств ключевого хеширования, построенных на основе алгоритмов MASH-1 и MASH-2 при смене значений вектора инициализации секретным ключом

Показатели	на основе алгоритма MASH-1	на основе алгоритма MASH-2
$\tilde{m}(n_1)$	41,42	0
$\tilde{D}(n_1)$	42,74	0
$P_д = P(\lvert\tilde{m}(n_1) - m(n_1)\rvert < 5)$	0,98	≈ 1
$\tilde{m}(n_2)$	3,99	1
$\tilde{D}(n_2)$	0,01	0
$P_д = P(\lvert\tilde{m}(n_2) - m(n_2)\rvert < 0,025)$	0,99	≈ 1
$\tilde{m}(n_3)$	0,26	0,31
$\tilde{D}(n_3)$	0,21	0,22
$P_д = P(\lvert\tilde{m}(n_3) - m(n_3)\rvert < 0,1)$	0,97	0,97

Исследования проводились над выборкой объема N = 100, для формирования каждого элемента выборки рассчитывался максимум по множеству из M = 100 кортежей элементов. Таким образом, общий объем формируемых наборов составил NM = 104. Для каждого проведенных N = 100 экспериментов оценивались математические ожидания $m(n_1)$, $m(n_2)$ и $m(n_3)$, дисперсии $D(n_1)$, $D(n_2)$ и $D(n_3)$, а также для фиксированной точности ε рассчитывались соответствующие доверительные вероятности $P(|\tilde{m}(n_i) - m(n_i)| < \varepsilon)$, $i = 1, 2, 3$.

Таким образом, проведенные исследования показали, что применение преобразований с использованием модулярной арифметики позволяет строить универсальные и строго универсальные классы хеширующих функций, которые с одной стороны позволяют обеспечить высокие коллизионные свойства, с другой стороны, при выполнении определенных ограничений на значение модулярной экспоненты обеспечивают высокие показатели безопасности и применимость модели доказуемой стойкости. Основными недостатками подобных конструкций являются:

– очень высокая сложность преобразований, которая обусловлена использованием в качестве цикловой функции модулярного возведения в степень. Фактически сложность применяемых преобразований выше сложности блочного симметричного шифрования на 2 – 3 порядка, что и обуславливает соответствующее повышение времени формирования кодов аутентификации сообщений (табл. 1);

– формирование кодов аутентификации сообщений с использованием ключевого хеширования, построенного на основе алгоритма MASH-1 с изменяемыми векторами инициализации, не позволяет строить универсальные и строго универсальные классы хеш-функций (табл. 2).

Это обусловлено использованием в качестве показателя степени цикловой функции значения e = 2, что при нечетных значениях простых чисел p и q всегда нарушает условие (3);

– формирование кодов аутентификации сообщений с использованием ключевого хеширования, построенного на основе алгоритма MASH-2 с изменяемыми векторами инициализации, в некоторых случаях (при выполнении условия (3)) позволяет строить универсальные и строго универсальные классы хеш-функций (табл. 2). Однако не для всех значений начальных параметров (простых чисел p и q) это условие выполнимо. Поэтому актуальной является разработка метода ключевого универсального хеширования доказуемой стойкости на основе модулярных преобразований.

В основе предлагаемого метода ключевого универсального хеширования доказуемой стойкости лежит использование модулярных преобразований, обеспечивающих сведение задачи нахождения прообраза или секретного ключа в схеме хеширования к одной из известных теоретико-сложностных задач. Подобное обоснование стойкости по классификации моделей безопасности NESSIE принято считать доказуемой безопасностью, подчеркивая тем самым сводимость задачи криптоанализа к одной из хорошо известных вычислительно неразрешимых за заданное время теоретико-сложностных задач [6]. В табл. 3 приведены результаты исследований цикловых функций: в первой колонке указана теоретико-сложностная задача, положенная в основу ее построения, во второй колонке приведена аналитическая запись цикловой функции, в третьей колонке – оценка сложности вычисления значения цикловой функции, в четвертой – оценка вычислительной сложности ее инвертирования (оценка стойкости).

Таблица 3

Кандидаты на построение цикловой функции итеративного хеширования информации

Теоретико-сложностная задача	Кандидаты на построение цикловой функции	Оценка сложности вычисления	Оценка сложности инвертирования
Проблема целочисленной факторизации	$f(x_i, H_{i-1}) = x_i H_{i-1}$, Функция определена над большими простыми числами $x_i = p$ и $H_{i-1} = q$	$O(n^2)$, где $n = \lceil \log_2 p \rceil + \lceil \log_2 q \rceil$	$L_N(\alpha, \beta) = \exp\left(\begin{matrix} (\beta + o(1))(\log N)^{\alpha} \times \\ (\log\log N)^{1-\alpha} \end{matrix} \right)$ Для поля чисел общего вида сложность инвертирования составляет $L_N\left(\frac{1}{3}, \sqrt[3]{\frac{64}{9}}\right)$, Для поля чисел специального вида $N = a^b + c$ сложность инвертирования составляет $L_N\left(\frac{1}{3}, \sqrt[3]{\frac{32}{9}}\right)$
Проблема RSA	$f(x_i, H_{i-1}) = (x_i \oplus H_{i-1})^e \bmod(N)$ $\gcd(e, \phi(p,q)) = 1$, $N = pq$	$O(\log_2 e)$ умножений алгоритм быстрого возведения в степень	
Проблема дискретного логарифмирования	$f(x_i, H_{i-1}) = (\alpha^{x_i \oplus H_{i-1}}) \bmod(p)$ α – генератор Z_p	$O(\log_2 n)$ умножений алгоритм быстрого возведения в степень, $O(n^3)$ для $\alpha = 2$, где $n = \lceil \log_2 p \rceil$	$\min\left\{\sqrt{p}, L_N(\alpha, \beta)\right\}$, где $L_N(\alpha, \beta) = \exp\left(\begin{matrix} (\beta + o(1))(\log N)^{\alpha} \times \\ (\log\log N)^{1-\alpha} \end{matrix} \right)$. Для примитивного поля $GF(p)$ сложность инвертирования составляет $\min\left\{\sqrt{p}, L_N\left(\frac{1}{3}, \sqrt[3]{\frac{64}{9}}\right)\right\}$, Для расширенного поля $GF(2^m)$ сложность инвертирования составляет $L_N\left(\frac{1}{3}, 1{,}4\right)$
Проблема Диффи-Хеллмана	$f(x_i, H_{i-1}) = (\alpha^{x_i \oplus H_{i-1}}) \bmod(p)$ α – генератор Z_p	$O(n^3)$ для $\alpha = 2$, где $n = \lceil \log_2 p \rceil$	

Проведенные исследования показывают, что наиболее целесообразным решением следует, очевидно, считать использование цикловой функции, задача инвертирования которой сопряжена с решением теоретико-сложностной задачи извлечения квадратных корней по модулю n.

При определенных ограничениях на значения составного модуля n эта задача по вычислительной сложности инвертирования сопоставима с проблемами факторизации и дискретного логарифмирования. В тоже время прямое вычисление значения функции $a \equiv (x^2) \bmod (n)$ требует значительно меньшего числа операций.

Следует, однако, отметить, что использование квадратичной цикловой функции не приводит к построению универсального хеширования. Следующей по вычислительной сложности идет цикловая функция

$$f(x_i, H_{i-1}) = (x_i \oplus H_{i-1})^e \bmod (N), \quad (4)$$

задача инвертирования которой сопряжена с решением теоретико-сложностной задачи RSA, где

$$\gcd(e, \phi(p,q)) = 1, \; N = pq,$$

$\gcd(x,y)$ – наибольший общий делитель чисел x и y.

Таким образом, применение цикловой функции (1) на основе модулярного возведение в степень позволяет строить доказуемо безопасное универсальное хеширование только при выполнении ограничений на значение модульной экспоненты и значения модуля преобразований.

Еще одним кандидатом на цикловую функцию в итеративной схеме хеширования является функция вида:

$$f(x_i, H_{i-1}) = (\alpha^{x_i \oplus H_{i-1}}) \bmod (p), \quad (5)$$

задача инвертирования которой сопряжена с решением теоретико-сложностной задачи дискретного логарифмирования, где α – генератор кольца целых чисел Z_p, а p – большое постое целое число.

Использование такой цикловой функции обеспечивает построение доказуемо безопасного хеширования, коллизионные свойства которого удовлетворяют условиям универсальности.

Таким образом, проведенные исследования показали, что для построения универсального хеширования информации с доказуемым уровнем безопасности следует использовать цикловую функцию вида (4) или вида (5).

Разработка алгоритмов итеративного ключевого хеширования доказуемой стойкости на основе использования модулярных преобразований.

В основу алгоритмов итеративного ключевого хеширования доказуемой стойкости на основе использования модулярных преобразований положен алгоритм MASH-1, при условии смены векторов инициализации и использовании рассмотренных выше цикловых функций, удовлетворяющих определенным ограничениям на применяемые модулярные преобразования.

Схема итеративного ключевого хеширования с использованием цикловой функции (4), разработанная по аналогии с рассмотренной в разделе 2 схемой NH хеширования, представлена на рис. 1. Алгоритм вычисления значения хеш-кода на основе цикловой функции (4) отличается от алгоритма MASH-2, в основном, системными установками и определением констант.

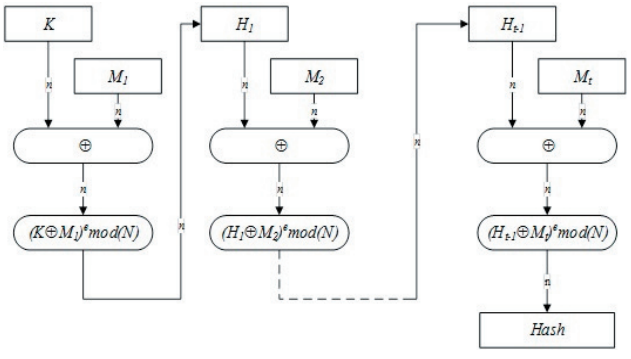

Рис. 1. Схема итеративного ключевого хеширования с использованием выражения (4)

Используя цикловую функцию (5), задача инвертирования которой базируется на решении теоретико-сложностной задачи дискретного логарифмирования, построим следующую схему хеширования (рис. 2).

Разработанные вычислительные алгоритмы отличаются от алгоритмов бесключевого хеширования MASH-1 и MASH-2, в основном, системными установками и определением констант. Кроме того, предлагаемые схемы хеширования являются ключевыми, в качестве секретных ключевых данных используются сменные вектора инициализации H_0 = Key. На применяемые модулярные преобразования в цикловой функции ключевого хеширования накладываются рассмотренные выше ограничения.

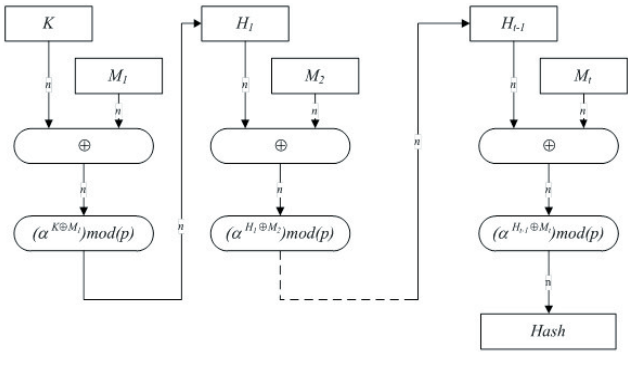

Рис. 2. Схема итеративного ключевого хеширования с использованием выражения (5)

Таким образом, предлагаемый метод универсального хеширования с использованием модулярных преобразований позволяет реализовать формирование аутентификаторов (хеш-кодов) с обеспечением требуемых показателей безопасности. Разработанные алгоритмы позволяют практически реализовать предлагаемые схемы хеширования, как в программном, так и в аппаратном виде.

Разработка предложений по реализации итеративного ключевого хеширования доказуемой стойкости с использованием модулярных преобразований.

В основе предлагаемого метода универсального хеширования лежит итеративная схема формирования хеш-кода с цикловой функцией, построенной с использованием модулярных преобразований. Для обеспечения высоких коллизионных свойств универсального хеширования предлагаемая цикловая функция должна быть реализована с использованием выражений (4) или (5) с соответствующими ограничениями на модулярные преобразования.

Проведенный анализ показывает, что наиболее затратной с вычислительной точки зрения операцией при реализации цикловых функций (4) и (5) является операция модульного возведения в степень. При непосредственном возведении в степень через цепочку операций умножений вычислительная сложность реализации таких цикловых функций растет пропорционально показателю степени, т.е. для возведения числа x в степень n в общем случае требуется выполнить $n-1$ умножений:

$$x^n = \underbrace{x \cdot x \cdot x \cdot ... \cdot x}_{n-1\ C<=>6\ 5-89}.$$

Асимптотическая оценка вычислительной сложности такой реализации операции возведения в степень есть $O(n)$ умножений.

Для снижения вычислительной сложности реализации схем хеширования с использованием цикловых функций (4) и (5) применен алгоритм быстрого возведения в степень, в основе которого лежит представление числа x^n в следующем виде:

$$\begin{aligned} x^n &= x^{((...((m_k \cdot 2+m_{k-1})2+m_{k-2})2+...)2+m_1)2+m_0} = \\ &= ((...(((x^{m_k})^2 \cdot x^{m_{k-1}})^2 ...)^2 \cdot x^{m_1})^2 \cdot x^{m_0}, \end{aligned} \quad (6)$$

где $(m_k, m_{k-1}, ..., m_0)$ – двоичное представление числа n, т.е. $m_i \in \{0,1\}$ и

$$n = m_k \cdot 2^k + m_{k-1} \cdot 2^{k-1} + ... + m_1 \cdot 2 + m_0. \quad (7)$$

Перегруппировав сомножители в представлении числа x^n, получим следующее выражение:

$$x^n = x^{m_0} \cdot (x^2)^{m_1} \cdot (x^{2^2})^{m_2} \cdot (x^{2^3})^{m_3} \cdot ... \cdot (x^{2^k})^{m_k},$$

откуда следует, что для возведения числа x в степень n требуется реализовать не более k операций возведения в квадрат и не более k операций умножений, где $k+1$ – число элементов в двоичной записи числа n, т.е. $k = (\log_2 n) - 1$. Таким образом, асимптотически вычислительную сложность вычисления x^n можно оценить как $O(\log_2 n)$.

Приведенный алгоритм позволяет существенно ускорить процедуру вычисления цикловых функций (4) и (5), лежащих в основе предлагаемого метода универсального хеширования.

В табл. 4 приведены зависимости сложности реализации операции возведения в степень через цепочку умножений и через представление (6), (7) с указанием порядка модуля преобразования, минимально необходимого для обеспечения требуемого уровня безопасности.

Данные во второй строке табл. 4 приведены из условия эквивалентности (по вычислительной сложности) операции возведения в квадрат и операции умножения.

Таблица 4

Оценки вычислительной сложности реализации операции возведения в степень различными методами

Метод возведения в степень	Порядок модуля преобразований / эквивалентная длина ключа симметричного криптоалгоритма		
	1024 / 80	3072 / 128	15360 / 256
Через цепочку произведений	10308	10924	104623
Быстрый алгоритм возведения в степень	2046	6142	30718

Анализ данных табл. 4 показывает, что реализация предложенного метода универсального хеширования через традиционный алгоритм возведения в степень вычислительно недостижима. Число умножений, которое требуется выполнить для вычисления одного значения цикловой функции даже при минимальном уровне безопасности (мощность множества ключевых данных блочного симметричного шифра равна 2^{80}) превышает возможности самых современных вычислительных систем.

Последняя строка табл. 3 является, фактически, оценкой вычислительной сложности предлагаемой схемы хеширования. Так, при минимальном уровне стойкости (мощность множества ключевых данных блочного симметричного шифра равна 2^{80}) для вычисления одного значения цикловой функции потребуется не более 2046 операций умножений. Для достаточного уровня стойкости (мощность множества ключевых данных БСШ равна 2^{128}), соответствующего национальному стандарту шифрования США FIPS-197 (AES), для вычисления значения цикловой функции потребуется выполнить не более 6142 операций умножения. Для высокого уровня стойкости (мощность множества ключевых данных БСШ равна 2^{256}), соответствующего действующему отечественному стандарту симметричного криптопреобразования ГОСТ-28147-89, для вычисления значения цикловой функции потребуется выполнить не более 30718 операций умножения.

4. Разработка модели каскадного формирования MAC с использованием модулярных преобразований и обоснование практических рекомендаций по ее использованию

В статье предлагается модель каскадного формирования кодов контроля целостности и аутентичности данных (MAC) с использованием модулярных преобразований. В основе предлагаемой модели лежит многослойная схема универсального хеширования с использованием на последнем, завершающем этапе модулярных преобразований.

Свойства многослойной (композиционной) конструкции лучше всего пояснить с помощью языка отображений [4, 5]. Пусть X, Y, U являются множествами из n, m, u элементов, $n < m < u$. H_1 есть множество функций f_1, осуществляющих отображение $X \to U$, а

H_2 – множество функций f_2 осуществляющих отображение $U \to Y$. Тогда $H = H_2 \circ H_1$ есть множество функций f, являющееся композицией $f = f_1 \circ f_2$.

Характеристики многослойной конструкции представлены результатом следующей теоремы [1 – 3].

Теорема 1. Композиция из универсального класса хеш-функций $\varepsilon_1 - U(N_1,n,u)$ и строго универсального класса хеш-функций $\varepsilon_2 - SU(N_2,u,m)$ является строго универсальным классом с параметрами

$$\varepsilon - SU(N_1N_2,n,m),$$

где $\varepsilon = \varepsilon_1 + \varepsilon_2 - \varepsilon_1\varepsilon_2$.

Таким образом, используя композицию алгоритмов формирования кодов аутентификации, эквивалентных алгоритмам вычисления универсальных и строго универсальных классов хеш-функций получим многослойную схему формирования MAC. Свойства, сформированного таким образом кода контроля целостности и аутентичности данных, будут удовлетворять свойствам строго универсального класса хеш-функций.

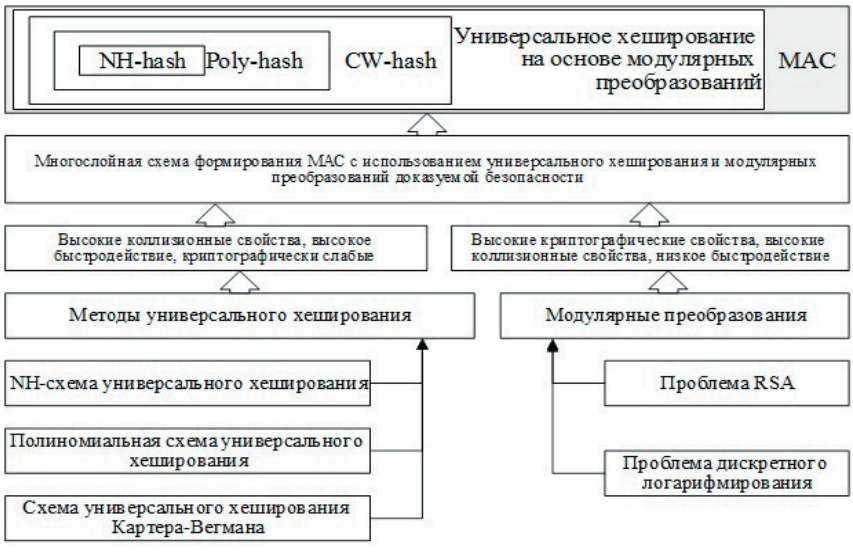

Рис. 3. Предлагаемая схема каскадного формирования кодов контроля целостности и аутентичности данных с использованием модулярных преобразований

В предлагаемом методе формирования кодов контроля целостности и аутентичности данных первые слои преобразования предлагается реализовать традиционными для алгоритма UMAC высокоскоростными, но криптографически слабыми схемами универсального хеширования, последний слой предлагается реализовать с использованием разработанной безопасной (криптографически сильной) схемы строго универсального хеширования на основе модулярных преобразований.

Формально предлагаемая схема каскадного формирования кодов контроля целостности и аутентичности данных представлена на рис. 3.

Основная часть информационных данных, обрабатывается первыми слоями универсального хеширования. Формируемый в результате такого преобразования хеш-код на последнем, заключительном этапе обрабатывается криптографически сильной функцией строго универсального хеширования на основе модулярных преобразований.

Таким образом, в основе предлагаемой схемы формирования MAC с использованием модулярных преобразований лежит использование:

– на первых слоях – высокоскоростных методов универсального хеширования (NH-хеширование, полиномиальное хеширование, хеширование Картера-Вегмана);

– на последнем слое – безопасного строго универсального хеширования на основе модулярных преобразований (с использованием цикловых функций (4) и/или (5)).

В табл. 5 приведено сравнение вычислительной сложности некоторых функций хеширования.

Данные по быстродействию для предлагаемой схемы MAC с модулярными преобразованиями приведены для минимального уровня стойкости (мощность множества ключевых данных блочного симметричного шифра равна 2^{80}) и достаточного уровня стойкости (для модулярных преобразований эквивалентная длина ключа блочного симметричного шифра равна 128 битам). Длина формируемого при этом MAC равна 80 и 128 битам, соответственно.

Таблица 5

Оценка сложности формирования MAC различными схемами

Алгоритм	Длина входных данных, байт					
	2048	4096	8192	16384	32768	65536
HMAC-MD5 (128 бит)	9	9	9	9	9	9
HMAC-RIPE-MD (160 бит)	27	27	27	27	27	27
HMAC-SHA-1 (160 бит)	25	25	25	25	25	25
HMAC-SHA-2 (512бит)	84	84	84	84	84	84
CBC MAC-Rijndael (128 бит)	26	26	26	26	26	26
CBC MAC-DES (64 бита)	62	62	62	62	62	62
Предлагаемая схема MAC с модулярными преобразованиями (80 бит)	38	22	14	10	8	7
Предлагаемая схема MAC с модулярными преобразованиями (128 бит)	294	150	78	42	24	15

Для всех функций, приведенных в табл. 5 (кроме предложенных, с использованием модулярных преобразований) удельная сложность формирования кодов контроля целостности и аутентичности данных не зависит от объема обрабатываемых данных. Для

предлагаемой модели с использованием модулярных преобразований удельная сложность с ростом длины обрабатываемых данных снижается.

Так для высокого уровня стойкости (эквивалентная длина ключа блочного симметричного шифра равна 128 битам) уже для блоков, данных из 32768 байт сопоставима с известными и применяемыми в протоколах сетевой безопасности алгоритмами формирования MAC.

Для минимального уровня стойкости (мощность множества ключевых данных блочного симметричного шифра равна 280) предлагаемая схема каскадного формирования кодов контроля целостности и аутентичности данных с использованием модулярных преобразований уже для пакетов данных из 2048 байт практически не уступает по быстродействию применяемым на сегодняшний день алгоритмам формирования MAC в протоколах сетевой безопасности, в том числе в протоколах IPSec и перспективных системах безопасности коммерческих банков Украины.

5. Выводы

Проведенные исследования показали, что модулярные преобразования традиционно использовались при построении бесключевых схем хеширования (алгоритмы MASH-1 и MASH-2), формирование кодов аутентификации сообщений с использованием ключевого хеширования, построенного на основе алгоритма MASH-1 с изменяемыми векторами инициализации, не позволяет строить универсальные и строго универсальные классы хеш-функций. Это обусловлено использованием в качестве показателя степени цикловой функции значения e = 2, что при нечетных значениях простых чисел p и q всегда нарушает условие (3), использование алгоритма MASH-2 с изменяемыми векторами инициализации, в некоторых случаях (при выполнении условия (3)) позволяет строить универсальные и строго универсальные классы хеш-функций. Однако не для всех значений начальных параметров (простых чисел p и q) это условие выполнимо. Исследования различных вариантов построения цикловых функций, использующих модулярные преобразования показали, что для построения универсального хеширования информации с доказуемым уровнем безопасности следует использовать цикловую функцию вида (4) или (5). При выполнении соответствующих ограничений итеративное формирование хеш-кодов позволяет с одной стороны обеспечить выполнение условий модели доказуемой безопасности, т.е. обеспечить высокую криптографическую стойкость, с другой стороны – обеспечить выполнение условий универсального хеширования, т.е. обеспечить высокие коллизионные свойства. Платой за достижение таких свойств хеширования является сравнительно высокая вычислительная сложность формирования хеш-кодов.

Разработанная модель и метод каскадного формирования кодов контроля целостности и аутентичности данных с использованием на последнем, заключительном этапе криптографически сильной функции строго универсального хеширования на основе модулярных преобразований позволяет обеспечить высокие коллизионные свойства строго универсального хеширования, низкую вычислительную сложность при обработке больших массивов данных и обеспечить высокие показатели безопасности на уровне современных средств криптографической защиты доказуемой стойкости.

Литература

1. Stinson, D. R. Some constructions and bounds for authentication codes [Text] / D. R. Stinson // J. Cryptology. – 1988. – № 1. – P. 37–51.
2. Stinson, D. R. The combinatorics of authentication and secrecy codes [Text] / D. R. Stinson // J. Cryptology. – 1990. – № 2. – P. 23–49.
3. Кузнецов, А. А. Исследование коллизионных свойств кодов аутентификации сообщений UMAC // А. А. Кузнецов, О. Г. Король, С. П. Евсеев. Прикладная радиоэлектроника. – Харьков: Изд-во ХНУРЭ, 2012. – Т. 11 № 2. – С. 171-183.
4. Король, О. Г. Разработка модели и метода каскадного формирования MAC с использованием модулярных преобразований // О. Г. Король, С. П. Евсеев, Л. Т. Пархуць / Захист інформації: науково-технічний журнал. – 2013. – Т. 15, № 3. – С. 186 – 196.
5. Maitra, S. Further constructions of resilient Boolean functions with very high nonlinearity [Text] / S. Maitra, E. Pasalic // Accepted in SETA. – May, 2001.
6. Кузнецов, О. О. Захист інформації в інформаційних системах / О. О. Кузнецов, С. П. Євсеєв, О .Г. Король. – Х.: Вид. ХНЕУ, 2011. – 504 с.
7. . Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption, April 19, 2004 – Version 0.15 (beta), Springer-Verlag.
8. Столлингс, В. Криптография и защита сетей: принципы и практика, 2-е изд. : пер. с англ. – М. : издательский дом «Вильям», 2001. – 672 с.
9. Король, О. Г. Исследование методов обеспечения аутентичности и целостности данных на основе односторонних хеш-функций // О. Г. Король, С. П. Евсеев. Науково-технічний журнал «Захист інформації». Спецвипуск (40). – 2008. – С. 50 – 55.
10. Ищейнов, В. Я. Модель безопасности конфиденциальной информации в информационной системе / В. Я. Ищейнов, С. М. Чудинов // Научные ведомости БелГУ.– Изд-во НИУ «БелГУ», 2012. – Выпуск 23/1. – № 13(132). – С. 205 – 210.