УДК 004.056

В. І. Селезньов

АНАЛІЗ МЕТОДІВ МАЛОРЕСУРСНОГО ГЕШУВАННЯ

Вінницький національний технічний університет

Анотація

Виконано огляд сучасних методів гешування, що застосовуються для малоресурсних пристроїв. Проаналізовано відомі методи малоресурсного гешування та виконано їх порівняння з точки зору забезпечення малоресурсності. Доведено актуальність проблеми гешування для малоресурсних пристроїв та необхідність розробки нових методів малоресурсного гешування для забезпечення оптимального співвідношення між споживаними ресурсами та захищеністю, що надається.

Ключові слова: малоресурсна криптографія, геш-функція, алгоритм, апаратна складність.

Abstract

The review of contemporary hashing methods applied to low-resource devices has been conducted. Well-known approaches to low-resource hashing have been analyzed and compared in terms of resource efficiency. The relevance of the hashing problem for low-resource devices has been demonstrated, emphasizing the need for the development of novel low-resource hashing methods to achieve an optimal balance between resource consumption and provided security.

Keywords: low-resource cryptography, hash function, algorithm, hardware complexity.

Вступ

У сучасному світі з появою все більшої кількості малоресурсних пристроїв, таких як вбудовані системи, пристрої RFID і сенсорні мережі та пристрої Інтернету речей (IoT), зростають вимоги до методів малоресурсного гешування. Традиційні алгоритми гешування, які широко використовуються, наприклад SHA-256, вимагають значних обчислювальних ресурсів та пам'яті, що призводить до зниження продуктивності та споживання енергії малоресурсних пристроїв [1, 2].

Враховуючи ці обмеження, метою даної роботи є проведення аналізу методів малоресурсного гешування з метою пошуку оптимального співвідношення між апаратною складністю та забезпеченою захищеністю. Основна увага приділяється дослідженню ефективності нових геш-функцій, спеціально призначених для малоресурсних пристроїв.

Дослідження зосереджуються на знаходженні оптимальних алгоритмів гешування, які б мінімізували обчислювальні витрати та використовували обмежені ресурси пристроїв ефективно. Для досягнення цієї мети, розглядаються різні малоресурсні геш-функції, такі як SPONGENT, S-Quark, D-Quark, Keccak та PHOTON, і проводиться їх порівняння за такими критеріями, як розмір коду, апаратна складність та швидкодія [2, 3, 4].

Основна частина

Стандартизовані криптографічні геш-функції такі як MD5 і SHA1, чи більш сучасні неефективно, а в деяких випадках неможливо використовувати для малоресусних пристроїв Інтернету речей. Таким чином NIST рекомендували нові методи гешування, такі як SPONGENT, PHOTON, Quark і Lesamnta-LW. Ці методи займають значно менше пам’яті та можуть застосовуватись на пристроях з обмеженими обчислювальними ресурсами [1, 2, 5].

SPONGENT базується на функції P-губки у якій перестановками є модифікована версія блокового шифру PRESENT. Кількість ітерацій PRESENT-подібної перестановки буває від 45 для 88 розрядного алгоритму SPONGENT-88 до 140 для 256 розрядного SPONGENT-256 [4, 6].

Алгоритм малоресурсного гешування Lesamnta-LW використовує структуру AES як своє ядро. Автори вважають, що для реалізації Lesamnta-LW потрібно лише 8240 GE і він має пропускну здатність 125 Мбіт/с (що в п’ять разів швидше, ніж SHA-256 і також дає 256-бітний геш). Для реалізації на 8-бітному процесорі, Lesamnta-LW вимагає лише 50 байт оперативної пам’яті [6].

Алгоритм Quark використовує P-губку з апаратно-орієнтованою перестановкою. Quark базується на полегшених блокових шифрах KTANTAN і KATAN, а також апаратно-орієнтованому потоковому шифру Grain. Існує три основних варіації u-Quark (геш довжиною 136 біт), d-Quark (геш довжиною 176 біт) і s-Quark (геш довжиною 256 біт). Особливість реалізації алгоритму полягає в тому, що в ньому відсутнє проміжне значення функції губки, тому не потрібно додаткових елементів для зберігання цього значення [7, 8].

Keccak - це сімейство криптографічних губкових функцій, які стали стандартом FIPS 202 (SHA-3) у 2015 році. Keccak базується на конструкції губки, в якій основною функцією є перестановка, вибрана в наборі з семи перестановок Keccak-f, позначених як Keccak-f (25, 50, 100, 200, 400, 800, 1600) з сімома різними довжинами перестановок {1, 2, 4, 8, 16, 32, 64} Keccak може забезпечити хорошу гнучкість і хорошу продуктивність як при апаратній, так і при програмній реалізації з помірним розміром коду та споживанням оперативної пам'яті, що підходить для легких програм [6].

Малоресурсний метод криптографічного гешування PHOTON базується на конструкції P-губки та використовує блоковий шифр AES. PHOTON може створювати 80-бітні, 128-бітні, 160-бітні, 224-бітні та 256-бітні геш-значення. Функція внутрішньої перестановки f подібна до AES з 12 раундами. Для забезпечення достатнього рівня безпеки з найменшою довжиною геш-значення (PHOTON-80/20/16) бітова швидкість під час поглинання дорівнює 20 Мбіт/с та 16 Мбіт/с під час стискання [8].

У таблиці 1 представлено порівняння методів малоресурсного гешування з використанням ключових метрик, що впливають на ефективність та можливість використання методів для малоресурсної криптографії [7, 8].

Таблиця 1 – Порівняння малоресурсних геш-функцій

Геш-функція	геш [біти]	розмір коду [байти]	апаратна складність [GE]	ОЗУ [байти]	ОЗУ стек	Цикли [m=8 байт]	Цикли [m=50 байт]	Цикли [m=100 байт]	Цикли [m=500 байт]
SPONGENT256/256/128	256	364	3281	96	5	1 542 923	3 856 916	6 170 900	25 454 100
SPONGENT160/160/80	160	598	2190	60	6	795294	278341	4771186	20674746
S-Quark	256	1106	2296	60	5	708 783	1 417 611	2 339 023	9 427 023
D-Quark	176	974	1702	42	5	631 871	1 516 685	2 570 035	10 996 835
Keccak[r = 40, c = 160]	160	752	5090	45	3	58 063	162 347	278 269	1 205 627
Keccak[r = 144, c = 256]	256	608	8588	92	4	90 824	181 466	317 221	1 313 291
PHOTON-160/36/36	160	764	1396	39	11	620 921	1 655 364	2 793 265	11 999 914
PHOTON-256/32/32	256	1244	2177	68	10	254 871	486 629	787 896	3 105 396

В результаті дослідження та порівняння методів малоресурсного гешування з’ясовано, що високі показники захищеності мають усі розглянуті методи, окрім U-Quark. Метод Keccak має найкращі показники швидкодії, однак має найбільшу апаратну складність, що робить недоречним його використання для малоресурсних пристроїв. Геш-функції SPONGENT мають найгіршу швидкодію, середні показники апаратної складності та найменший розмір коду. Оптимальними серед розглянутих методів гешування можна вважати матоди Quark та PHOTON, оскільки вони вимагають найменшої апаратної складності та середні показники швидкодії та використаної пам'яті.

Висновки

Розглянуто методи малоресурсного гешування, здійснено їх порівняння на основі ключових метрик, що впливають на ефективність та можливість використання методів для малоресурсної криптографії. В результаті порівняння з'ясовано, що функція PHOTON-160/36/36 має найменший розмір коду і вимагає найменше апаратної складності. Вона може бути гарним вибором для обмежених ресурсів або вбудованих систем з обмеженою потужністю, однак генерує геш лише довжиною 160 біт. Keccak[r = 40, c = 160] також має помірний розмір коду і може бути використана в різних сферах, якщо потрібна гарантована безпека, але має високу відносно апаратну складність. У SPONGENT160/160/80 та D-Quark мають середній розмір коду та помірну апаратну складність, тому може бути використана у випадках, коли вимоги до ресурсів та безпеки є помірними. Серед розглянутих методів не виявлено оптимального співвідношення між апаратною складністю та забезпеченою захищеністю, оскільки оптимальні з точки зору вимог до захищеності методи вимагають все ще значних обчислювальних ресурсів. Тому задача розробки нових методів малоресурсного гешування залишається дуже актуальною. Для подальшого дослідження та розробки нового методу малорусурсного гешування варто приділити увагу методам SPONGENT160/160/80 та D-Quark, що найбільше наближаються до оптимальних з точки зору вимог до захищеності та обчислювальних ресурсів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. William J. Buchanan, Shancang Li & Rameez Asif. "Lightweight cryptography methods." Journal of Cyber Security Technology, 1:3-4, 187-201, 2017. URL: https://www.tandfonline.com/doi/pdf/10.1080/23742917.2017.1384917?needAccess=true (дата звернення: 07.03.2023)
2. Лужецький, В. А., Баришев Ю. В. "Підходи до побудови швидких алгоритмів хешування." Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні: науково-технічний збірник. 2009. Вип. 2(19). С. 57-66.
3. Лужецький В. А., Слободян С. О., Кисюк Д. В. "Методи байт-орієнтованого хешування даних низькоресурсної криптографії." Інформаційні технології та комп'ютерне моделювання: матеріали міжнар. наук.-практ. конф., м. Івано-Франківськ, 15-20 травня 2017 р., 2017. С. 216 – 219.
4. Aleksandra Mileva, Vesna Dimitrova, Orhun Kara. "Catalog and Illustrative Examples of Lightweight Cryptographic Primitives." Security of Ubiquitous Computing Systems, 2021. pp 21–47. URL: https://link.springer.com/content/pdf/10.1007/978-3-030-10591-4.pdf (дата звернення 16.03.2023)
5. Jian Guo, Thomas Peyrin, Axel Poschmann. "The PHOTON Family of Lightweight Hash Functions." Institute for Infocomm Research, Singapore Nanyang Technological University, Singapore. URL: https://perso.uclouvain.be/fstandae/source_codes/hash_atmel/specs/photon.pdf (дата звернення 19.03.2023)
6. Andrey Bogdanov, Miroslav Knezevic, Gregor Leander, et al. "SPONGENT: A Lightweight Hash Function." Katholieke Universiteit Leuven, ESAT/COSIC and IBBT, Belgium. URL: https://www.academia.edu/4959797/SPONGENT_A_Lightweight_Hash_Function (дата звернення 18.04.2023)
7. Josep Balasch, Barış Ege, Thomas Eisenbarth, et al. "Compact Implementation and Performance Evaluation of Hash Functions in ATtiny Devices." International Conference on Smart Card Research and Advanced Applications CARDIS 2012: Smart Card Research and Advanced Applications. pp 158–172. URL: https://perso.uclouvain.be/fstandae/source_codes/hash_atmel/paper.pdf (дата звернення 25.03.2023)
8. Tobias Meuser, Larissa Schmidt, Alex Wiesmaier. "Comparing Lightweight Hash Functions – PHOTON & Quark." Technische Universität Darmstadt, Germany AGT International, 2015. URL: https://download.hrz.tu-darmstadt.de/pub/FB20/Dekanat/Publikationen/CDC/2015-07-06_TR_PhotonQuark.pdf (дата звернення 10.04.2023)

Селезньов Віталій Ігорович — аспірант групи 125-22а, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: seleznov.vitalii@gmail.com

Науковий керівник – ***Лужецький Володимир Андрійович*** — д. т. н., професор, завідувач кафедри захисту інформації, Вінницький національний технічний університет, м. Вінниця, e-mail: v.luzhetskyi@vntu.edu.ua

Seleznov Vitalii — Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, email: seleznov.vitalii@gmail.com

Scientific supervisor – ***Luzhetskiy Vladimir*** — Doctor of Technical Science, Professor, Head of Information Security Department, Vinnytsia National Technical University, Khmelnytske shosse 95, Vinnytsia, Ukraine, e-mail: v.luzhetskyi@vntu.edu.ua