Một số kết quả nghiên cứu về mã khối hạng nhẹ

Nguyễn Bùi Cương

***Tóm tắt*— Bài báo này, giới thiệu một số kết quả nghiên cứu về việc xây dựng thuật toán mã khối hạng nhẹ. Trên cơ sở thảo luận và phân tích các đặc điểm cũng như nguyên lý thiết kế mã khối hạng nhẹ, chúng tôi phát triển công cụ đánh giá độ an toàn cho một thuật toán mã khối, xây dựng hộp thế 4 bit an toàn, phát triển tầng tuyến tính cho mã khối hạng nhẹ và giới thiệu thuật toán mã khối hạng nhẹ tựa PRESENT.**

***Abstract*— This article introduces some results of studies on the construction of lightweight block ciphers algorithm. Based on the discussion and analysis of some features and design principles of lightweight block ciphers, we develop evaluation tool for a safety block ciphers, 4-bit S-box, linear stage and propose an implementation of lightweight block cipher algorithm based-on PRESENT.**

Từ khóa— mật mã hạng nhẹ; mã khối; MEDP; MILP; S-hộp kiểu SERPENT.

I. MỞ ĐẦU

Sự phát triển của khoa học kỹ thuật đã dẫn đến xuất hiện nhiều thiết bị có năng lực tính toán lớn như máy tính cá nhân có bộ vi xử lý 64 bit, tốc độ 3-4 GHz, 2 GB RAM…. Nhưng, nhu cầu sử dụng các thiết bị có kích cỡ nhỏ, khả năng tính toán thấp phục vụ các công việc và giải quyết bài toán chuyên dụng, đơn giản, điển hình như các thẻ thông minh (smartcard), vi điều khiển (microcontroller) ngày càng tăng. Trong khi đó, các mã khối truyền thống hiện có khó có thể sử dụng đa năng cho mọi kiểu thiết bị (bộ vi xử lý), do sự phức tạp, sử dụng nhiều tài nguyên, năng lượng. Một mã pháp an toàn truyền thống cũng khó có thể cài đặt hiệu quả trên các thiết bị có năng lực và tài nguyên hạn chế (như các bộ vi điều khiển 4 bit, 8 bit, có kích cỡ RAM nhỏ, tần số thấp). Vì vậy, nhu cầu cần có các hệ mật mã (mã khóa công khai, mã khối, mã dòng, hàm băm...) riêng, áp dụng cho các thiết bị/hệ thống bị hạn chế (thông tin cần phải bảo vệ không quá mật) đã và đang được đặt ra trong những năm qua.

Trong phần tổng quan chung của tiêu chuẩn ISO/IEC 29192-1 đã đưa ra khái niệm: mật mã hạng nhẹ là mật mã phù hợp với các cài đặt trong những môi trường bị hạn chế [1]. Những hạn chế đó dựa trên các đánh giá về diện tích chip (*chip area*), năng lượng tiêu thụ *(energy consumption),* kích cỡ mã nguồn chương trình *(program code size)* kích cỡ RAM, băng thông *(communication bandwidth)* và thời gian thực thi *(execution time).* Trong những trường hợp này, sử dụng các thuật toán mã khối hạng nhẹ là phù hợp và cần được quan tâm nghiên cứu.

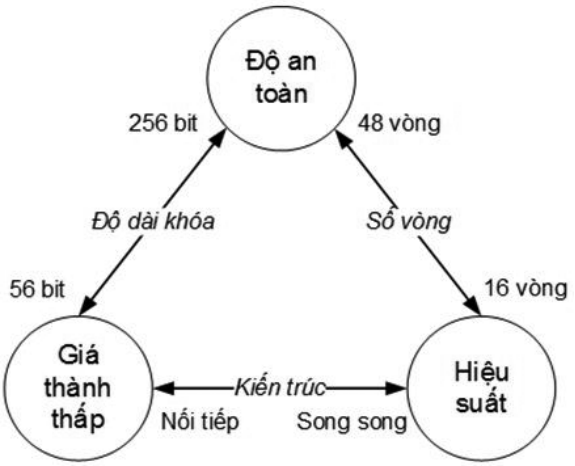
Trong bài báo này, trên cơ sở thảo luận về nguyên lý chung cho thiết kế một thuật toán mật mã hạng nhẹ nói chung; xem xét trường hợp cụ thể là chuẩn mã khối hạng nhẹ ISO/IEC 29129-2 PRESENT, chúng tôi phân tích các đặc điểm về cấu trúc và thành phần của các thuật toán mã khối hạng nhẹ. Sau đó, trình bày một số kết quả đạt được trong quá trình xây dựng thuật toán mã khối hạng nhẹ bao gồm phát triển công cụ đánh giá độ an toàn, phát triển tầng tuyến tính cho mã khối hạng nhẹ và giới thiệu thuật toán mã khối hạng nhẹ tựa PRESENT.

Bố cục của bài báo gồm bốn mục, sau mục mở đầu, Mục II trình bày nguyên lý thiết kế thuật toán mã khối hạng nhẹ, Mục III trình bày một số kết quả đạt được và Mục cuối là phần kết luận và hướng phát triển.

II. NGUYÊN LÝ THIẾT KẾ THUẬT TOÁN MÃ KHỐI HẠNG NHẸ

A. Nguyên lý thiết kế thuật toán mã khối hạng nhẹ

Trong thiết kế và đánh giá một hệ mã hạng nhẹ, chúng ta cần phải xem xét hai yêu cầu quan trọng. Thứ nhất là về độ an toàn mục tiêu trong xây dựng các hệ mật mã hạng nhẹ (light-weight) theo nghĩa: Thiết kế một hệ mật không quá yếu (không với mục đích thay thế các thuật toán mã truyền thống khác), nhưng phải đủ an toàn (tất nhiên, không thể kháng lại được các đối phương có đủ mọi điều kiện). Thứ hai là về hiệu quả trong cài đặt thường được đánh giá qua các độ đo tài nguyên được sử dụng bởi thuật toán, yêu cầu này phản ánh chi phí cài đặt cũng như hiệu suất. Người thiết kế mật mã hạng nhẹ phải thỏa hiệp giữa độ an toàn, chi phí cài đặt và hiệu suất. Một yêu cầu quan trọng đối với các thiết bị này là có khả năng tính toán trên đường truyền (on-the-fly). Tức là cần có một hệ mật không phải tốt nhất, mà phải thỏa hiệp giữa giá thành, hiệu suất và độ an toàn. Tuy nhiên, rất khó để có thể tối ưu hóa cả ba khía cạnh trên.



Hình 1. Sự thỏa hiệp trong thiết kế mật mã hạng nhẹ

Với các mã khối, độ dài khóa là sự thỏa hiệp giữa độ an toàn và giá thành, trong đó, số vòng là sự cân bằng giữa hiệu suất và độ an toàn, như biểu diễn trong Hình 1. Thông thường, ta có thể tối ưu hóa được hai tiêu chí bất kỳ trong ba tiêu chí trên, nhưng việc tối ưu hóa cả ba mục tiêu là rất khó. Bên cạnh đó, thực hiện cài đặt bằng phần cứng có hiệu suất cao cũng cần tính tới giải pháp tránh các tấn công kênh kề. Điều này thường dẫn tới các yêu cầu cao về diện tích, đồng nghĩa với giá thành cao. Mặt khác, ta cũng có thể thiết kế các mã pháp an toàn và có cài đặt phần cứng thấp nhưng hiệu suất sẽ rất thấp.

Có ba cách tiếp cận để đưa ra một nguyên thủy mật mã cho các ứng dụng hạng nhẹ (như các thẻ RFID) như sau:

- Tối ưu hóa các cài đặt cho các thuật toán được tin cậy và đã được chuẩn hóa (như AES, ECC...).
- Thay đổi một phần trong một mã pháp đã được tin cậy và đã được nghiên cứu.
- Thiết kế các mã pháp mới với mục tiêu chi phí cài đặt phần cứng thấp.

Để đạt được các yêu cầu tốt nhất về cài đặt phần cứng, chúng ta nên theo cách tiếp cận thứ ba, đó là thiết kế một mã pháp nhẹ phù hợp cho các yêu cầu cụ thể của phần cứng.

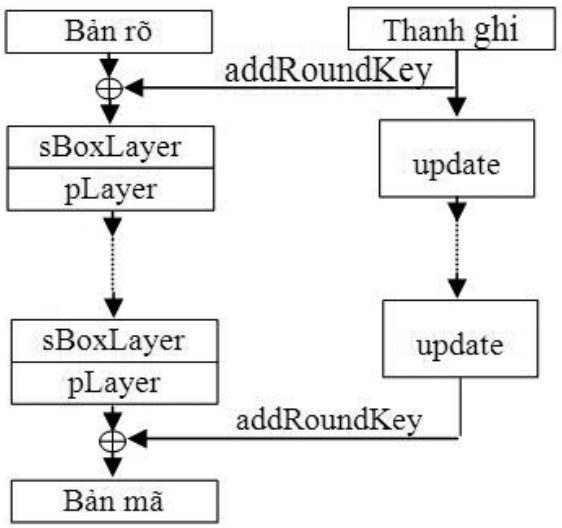
Hiện nay, trên thế giới có rất nhiều thuật toán mật mã mới được đề xuất nhằm hướng tới những yêu cầu cụ thể, như: mã khối DESL và DESXL (một mã pháp được sửa đổi từ DES), mã khối HIGHT, mã khối mCrypton, SEA, TEA, ICEBERG, PRINCE… đặc biệt là một số chuẩn mã khối như chuẩn mã khối hạng nhẹ ISO/IEC 29192-2 PRESENT, CLEFIA SIMON và SPECK. Trong đó, mã khối PRESENT được đánh giá rất cao trên phương diện cài đặt cứng hóa. Thiết kế của PRESENT là một cách tiếp cận cho mật mã hạng nhẹ, trong đó chúng ta chỉ sửa đổi một phần hoặc cài đặt hiệu quả hơn từ một mã pháp có sẵn, thay vì việc thiết kế một mã pháp hạng nhẹ và tối ưu hóa cho phần cứng.

B. Chuẩn thuật toán mã khối hạng nhẹ PRESENT (ISO/IEC 29192-2)

Mã pháp PRESENT với kích thước khối là 64 bit và độ dài khóa có thể sử dụng là 80 hoặc 128 bit. PRESENT có cấu trúc SPN với số vòng 31, mỗi vòng thực hiện phép cộng XOR để đưa vào khóa vòng, tầng phi tuyến sử dụng một S-hộp 4 bit duy nhất được áp dụng 16 lần song song trong mỗi vòng, tầng tuyến tính sử dụng hoán vị bit đơn giản. Cuối cùng là một phép cộng khóa K_{32} cho việc làm trắng sau (post-whitening). Phép toán giải mã sử dụng các biến đổi ngược của các thành phần mật mã trên. Lược đồ khóa của PRESENT có hai phiên bản khóa là 80 bit và 128 bit có thể được thực hiện dưới dạng tính toán trên đường truyền trên một thanh ghi dịch nhằm tiết kiệm tài nguyên sử dụng.

Việc lựa chọn các S-hộp 4×4 thay vì 8×8 cũng vì lý do điều khiển phần cứng, bởi vì các S-hộp 4 bit yêu cầu ít hơn một phần tư diện tích so với các S-hộp 8 bit (25 GE so với 120 GE trong cài đặt cứng hóa). Tuy nhiên, các S-hộp 4 bit phải được lựa chọn hết sức cẩn thận để có thể thu được một mức độ an toàn phù hợp (vì về mật mã chúng yếu hơn rất nhiều so với các S-hộp 8 bit). Ở tầng hoán vị là một chuyển vị rất đơn giản và phổ thông mà không ảnh hưởng đến phần cứng, nó được thực hiện bởi cách nối dây và không yêu cầu thêm transistor nào. Tầng hoán vị đảm bảo rằng 4 bit đầu ra của S-hộp sẽ được phân tán ra 4 S-hộp khác nhau ở vòng tiếp theo để đảm bảo hiệu ứng thác đổ trong mã pháp. Điều này cũng là bắt buộc để chống lại các tấn công thám mã tuyến tính và lượng sai. Tính hiệu quả trong cài đặt của PRESENT cũng đã được đánh giá trong [2]. Ở đó, Poschman đã trình bày chi tiết về các kết quả cài đặt khác nhau cho ASIC (dạng nối tiếp - dạng dựa theo từng vòng, song song hóa và đồng vi xử lý), FPGA và một số nền (platform) phần mềm khác nhau (4, 8, 16 và 32 bit). Trong đó đáng chú ý là các kết quả cài đặt cho vi xử lý 4 bit và ASIC dạng nối tiếp. Kết quả cài đặt của Poschmann [2] cho PRESENT-80 ở dạng nối tiếp yêu cầu 1.075 GE và PRESENT-128 lớn hơn từ 2 đến 3 lần.

Nhận xét: Từ nguyên lý thiết kế chung của thuật toán mã khối hạng nhẹ và nghiên cứu tìm hiểu kỹ thuật toán PRESENT, chúng tôi thấy rằng,



Hình 2. Mô tả thuật toán phép mã hóa trong thuật toán PRESENT

đối với các thuật toán mã khối hạng nhẹ, mã pháp thường có những đặc điểm sau: *64 bit là lựa chọn chung cho kích thước khối*. Do các mã pháp này được thiết kế cho nhu cầu có mức độ an toàn trung bình, nên *độ dài khóa dao động từ 64-128 bit* nhằm đạt độ an toàn 2^{64} đến 2^{80}. Hàm vòng được thiết kế đơn giản sao cho không tốn nhiều không gian. Lựa chọn chung đối với tầng xáo trộn (đối với các chiến lược thiết kế) là các *S-hộp có kích thước* 4×4.

Hoán vị các bit dễ dàng đạt được trong phần cứng, do đó, lựa chọn cho tầng tuyến tính là một phép hoán vị bit (ví dụ như mã pháp PRESENT). Các mã pháp thiết kế mới như KLEIN và LED đã đề xuất một dạng khác cho tầng tuyến tính. Nó tương tự như AES, tức là nhận trạng thái với một ma trận phân tách có khoảng cách cực đại MDS. Ưu điểm của việc sử dụng các ma trận này là giúp cho các nhà thiết kế chứng minh độ an toàn chặt chẽ hơn, cùng với tính chất khuếch tán rất tốt.

III. MỘT SỐ KẾT QUẢ NGHIÊN CỨU

Kết quả nghiên cứu thuật toán mã hóa hạng nhẹ của các tác giả đã đạt được một số kết quả như sau:

A. Phát triển công cụ đánh giá độ an toàn cho một thuật toán mã khối

Chúng tôi phát triển công cụ đánh giá độ an toàn thực hành đối với thám mã lượng sai và tuyến tính, nhằm đưa ra được độ an toàn phù hợp. Có hai công cụ đã được quan tâm đến là: thuật toán KMT1 của Keliher để đánh giá độ an toàn chống lại thám mã lượng sai và tuyến tính dựa trên các độ đo thực hành MEDP (Maximum Expected Differential Probability) và thuật toán MILP (Mixed-integer linear programming) của Mouha nhằm xác định số S-hộp lượng sai (tuyến tính) hoạt động nhỏ nhất của một mã pháp SPN tổng quát.

Với PRESENT, thuật toán này có cấu trúc SPN giống như AES. Tuy nhiên, cần chú ý rằng tầng tuyến tính của PRESENT là một hoán vị ở mức bit, nên khi áp dụng các các thuật toán này ta gặp nhiều khó khăn. Cụ thể với thuật toán KMT1, với T = 3 vòng, độ phức tạp cỡ 2^{48} phép tính để tính MEDP 3 vòng. Với độ phức tạp này, không thể tìm toàn bộ các giá trị của $(\gamma,\hat{\gamma})$ (theo ước tính cần một năm với máy tính có bộ vi xử lý 2,8 GHz). Hơn nữa, việc tìm toàn bộ các giá trị này chỉ có ý nghĩa khi ta tính toán tiếp với 4 vòng của PRESENT. Còn với mục tiêu tìm giá trị lớn nhất cho 3 vòng, chúng tôi sẽ hạn chế bằng tìm kiếm với các cặp giá trị $(\gamma,\hat{\gamma})$ có trọng số nhỏ hơn 4. Kết quả cho thấy, MEDP cho 3 vòng là 1.750×2^{-8}.

BẢNG 1. KẾT QUẢ CỦA BÀI TOÁN MILP CHO PRESENT TRONG MÔ HÌNH KHÓA ĐƠN

Vòng	Số biến	Số ràng buộc	Số S-hộp hoạt động
1	96 + 64	257	1
2	128 + 128	513	2
3	160 + 192	769	4
4	192 + 256	1025	6
5	224 + 320	1281	10
6	256 + 384	1537	12
7	288 + 448	1739	14
8	320 + 512	2049	16
9	352 + 576	2305	18
10	384 + 640	2561	20
11	416 + 704	2817	22
12	448 + 768	3073	24
13	480 + 832	3329	26
14	512 + 896	3585	28
15	544 + 960	3841	30
16	576 + 1024	4097	32
17	608 + 1088	4353	34
18	640 + 1152	4609	36
19	672 + 1216	4865	38
20	704 + 1280	5121	40
21	736 + 1344	5377	42
22	768 + 1408	5633	44
23	800 + 1472	5889	46
24	832 + 1536	6145	48
25	864 + 1600	6401	50
26	896 + 1664	6657	52
27	928 + 1728	6913	54
28	960 + 1792	7169	56
29	992 + 1856	7425	58
30	1024 + 1920	7681	60
31	1056 + 1984	7937	62

Phương pháp của Mouha đang được quan tâm nghiên cứu trên thế giới, nhất là trong trường hợp sử dụng khóa quan hệ (khi các mã pháp có dạng SPN với tầng tuyến tính là hoán vị bit) khi đó, các ràng buộc của bài toán MILP sẽ phức tạp hơn. Ý tưởng của phương pháp này là chuyển từ bài toán tối ưu xác định giá trị nhỏ nhất của số lượng các S-hộp tích cực thành bài toán quy hoạch tuyến tính nguyên bộ phận MILP, sau đó sử dụng những công cụ giải sẵn có để tìm nghiệm của bài toán MILP. Điều đó cũng có nghĩa là chúng ta nhận được giá trị nhỏ nhất của số lượng hộp thế hoạt động trong mã pháp, từ đó suy ra cận dưới của độ phức tạp thám mã lên mã pháp đang xem xét. Như vậy, điểm mấu chốt nhất trong phương pháp này chính là việc mô tả các ràng buộc khi xem xét các phép biến đổi được thực hiện lần lượt trong thuật toán. Các ràng buộc tăng lên sẽ làm cho việc mô tả gần với các giá trị thực mà các sai khác có thể sinh ra trong thực tế, tuy nhiên, số lượng các biến phụ và ràng buộc mới sẽ tăng lên, làm cho việc giải bài toán khó khăn hơn. Ngược lại, ràng buộc lỏng lẻo sẽ khiến cho miền giá trị của bài toán MILP sẽ rất lớn so với các giá trị thực mà các sai khác có thể nhận.

Đối với mã pháp hướng bit PRESENT, kết quả đánh giá trong mô hình khóa đơn của PRESENT đã được đưa ra cụ thể (Bảng 1).

Tuy nhiên, đối với mô hình khóa quan hệ, số lượng các biến và ràng buộc tăng lên nhanh chóng đòi hỏi phải cải tiến kỹ thuật xử lý. Một số đề xuất gần đây đã được đưa ra để giảm thiểu số lượng các ràng buộc, như kỹ thuật sử dụng bao lồi chứa tất cả các lượng sai có thể của S-hộp, sử dụng những ràng buộc sau khi một số xử lý như cắt bỏ hợp lệ….

BẢNG 2. KẾT QUẢ CỦA BÀI TOÁN MILP CHO PRESENT-80 TRONG MÔ HÌNH KHÓA QUAN HỆ KHI CHƯA CẢI TIẾN

Vòng	Số biến	Số ràng buộc	Số hộp hoạt động
1	97+277	530	0
2	130+474	1058	0
3	163+671	1586	1
4	196+868	2114	2
5	229+1065	2642	3
6	262+1262	3170	4
7	295+1459	3698	6
8	328+1656	4226	8
9	361+1853	4754	9
10	394+2050	5282	12
11	427+2247	5810	13
12	460+2444	6338	15
13	493+2641	8192	-
...	…	…	…

***Một số kết quả thực nghiệm*:** Các kết quả tính toán của bài toán MILP được xây dựng cho thuật toán AES và PRESENT đều được nhóm nghiên cứu thực hiện chạy trên máy chủ có năng lực tính toán là XEON 3,1GHz, RAM 8GB, hệ điều hành Linux 64 bit, sử dụng bộ giải miễn phí CBC Solver trên phần mềm tính toán mã nguồn mở SAGE. Thời gian để tính AES là rất nhanh, mô hình phức tạp nhất mà nhóm đề tài xét đến là trường hợp khóa quan hệ cho thuật toán AES 14 vòng với độ dài khóa 256 bit, quá trình tính toán chưa tới 30 phút. Tuy nhiên, để tính được bài toán MILP cho thuật toán PRESENT, quá trình tính toán nhận được diễn ra rất lâu, trong khi chỉ xét cho mô hình khóa đơn. Để tính cho 8 vòng đối với thuật toán, phải mất hơn 3 ngày. Trong khi đó, với năng lực tính toán nhanh của một PC với Intel(R) Core(TM) Quad CPU (2,83GHz, 3,25 RAM) với bộ giải có bản quyền thì chỉ mất một vài phút thực hiện.

B. Xây dựng hộp thế 4 bit an toàn cho mã khối hạng nhẹ

Bên cạnh việc đánh giá những tiêu chí truyền thống như kháng lại thám mã lượng sai, tuyến tính và đại số, chúng tôi đã xem xét một dạng tấn công kênh kề khác là tấn công phân tích năng lượng lượng sai DPA. Tấn công này là một mối đe dọa tiềm năng thực sự đối với mật mã hạng nhẹ. Nhiều giải pháp được thực hiện để ngăn chặn tiếp cận được tới các thiết bị để thực hiện tấn công khi các thiết bị phổ dụng được triển khai trong môi trường không an toàn. Từ kết quả đánh giá cho các đại lượng nghiên cứu được, chúng tôi sẽ khảo sát và chọn ra các S-hộp tốt nhất dựa trên các tiêu chí đã đưa ra. Trên cơ sở kế thừa các kết quả nghiên cứu về các S-hộp dạng Serpent đã đạt được trước đây, chúng tôi đã nghiên cứu một tiêu chuẩn mới về bậc trong suốt của S-hộp, là đại lượng đặc trưng cho S-hộp mà đánh giá khả năng kháng lại tấn công DPA của mã pháp theo mô hình lý thuyết của tấn công DPA lên một mã khối lặp có cấu trúc SPN, được trình bày chi tiết trong [4].

Từ việc phân tích mô hình lý thuyết của tấn công DPA lên các mã khối lặp dạng SPN, bậc trong suốt của S-hộp được phát biểu như sau:

Định nghĩa 1. [4] *Lấy n và m là hai số nguyên dương và F là một (n, m)-hàm. Bậc trong suốt của F, kí hiệu là:*

$$\mathcal{T}_F = \max_{\beta\in\mathbb{F}_2^m}\left(\begin{array}{l}\left|m-2H(\beta)\right| \\ -\dfrac{1}{2^{2n}-2^n}\displaystyle\sum_{a\in\mathbb{F}_2^{n^*}}\left|\sum_{\substack{v\in\mathbb{F}_2^n \\ H(v)=1}}(-1)^{v.\beta}W_{D_aF}(0,v)\right|\end{array}\right) \quad (1)$$

trong đó H(β) là trọng số Hamming của véc tơ β, W_f là biến đổi Fourier của hàm dấu của (n,m)-hàm f (hay còn gọi là biến đổi Walsh của hàm f), còn D_aF là đạo hàm của hàm F theo véc tơ $a\in\mathbb{F}_2^n$.

Đối với mã khối lặp có cấu trúc SPN, Prouff đã chỉ ra rằng, bậc trong suốt của S-hộp càng nhỏ thì khả năng kháng lại tấn công DPA của mã khối sử dụng S-hộp này càng tốt. Tuy nhiên, bậc trong suốt lại không tỷ lệ thuận với độ phi tuyến của S-hộp. Để khảo sát bậc trong suốt của các S-hộp kiểu Serpent, chúng tôi mở rộng Định lý 1 trong [3] bằng việc xét thêm dấu bằng trong ràng buộc đối với các hàm Bool thành phần như sau:

Xét biểu thức:

$$C(\beta) = \left|m-2H(\beta)\right| - \frac{1}{2^{2n}-2^n}\sum_{a\in\mathbb{F}_2^{n^*}}\left|\sum_{\substack{v\in\mathbb{F}_2^n \\ H(v)=1}}(-1)^{v.\beta}W_{D_aF}(0,v)\right|$$

Khi đó, ta có mệnh đề sau:

Mệnh đề 1. [6] *Đối với S-hộp có phổ tự tương quan của các hàm Bool thành phần f_i thỏa mãn điều kiện*

$$\forall a\in\left(\mathbb{F}_2^n\right)^*, \left|\sum_{i=1}^{n}\Delta_{f_i}(a)\right| < 2^{n+1} \quad (2)$$

thì khi đó giá trị lớn nhất của biểu thức biểu diễn bậc trong suốt C(β) sẽ đạt được tại β=0 (hoặc β=2^n-1), tức là $\mathcal{T}_F = C(0) = C\left(2^n-1\right)$.

Từ kết quả này, nhóm nghiên cứu đã chứng minh được kết quả lý thuyết về tính bất biến của bậc trong suốt đối với quan hệ tương đương hoán vị (xem định nghĩa [6]) như sau:

Mệnh đề 2. [6] *Lấy S-hộp S_1 thỏa mãn điều kiện (2) và S_2 là S-hộp tương đương hoán vị với S_1. Khi đó,*

1. *S_2 cũng thỏa mãn điều kiện (2).*
2. *S_2 có bậc trong suốt bằng với bậc trong suốt của S_1, tức là $\mathcal{T}_{S_1} = \mathcal{T}_{S_2}$.*

Kết hợp các kết quả lý thuyết với việc kiểm tra tập phổ của 20 lớp S-hộp Serpent, chúng tôi nhận được kết quả quan trọng trong [6] như sau:

Kết quả 1. [6] *Trong mỗi lớp S-hộp kiểu Serpent theo quan hệ tương đương hoán vị, bậc trong suốt của các S-hộp này là bất biến.*

Bảng 3 đề xuất các hộp thế trong lớp R_{16}, R_{17}, R_{19} của các S-hộp 4 bit kiểu Serpent. Các hộp *S* này không những là các S-hộp tốt nhất kháng lại thám mã lượng sai và tuyến tính, mà còn có giá trị tối ưu đối với khả năng kháng lại DPA.

BẢNG 3. GIÁ TRỊ BẬC TRONG SUỐT ĐỐI VỚI CÁC LỚP S-HỘP KIỂU SERPENT

Lớp	S-hộp đại diện	$\mathcal{T}$
R_0	0, 3, 5, 6, 7, 10, 11, 12, 13, 4, 14, 9, 8, 1, 2, 15	3.53
R_1	0, 3, 5, 8, 6, 9, 10, 7, 11, 12, 14, 2, 1, 15, 13, 1	3.40
R_2	0, 3, 5, 8, 6, 9, 11, 2, 13, 4, 14, 1, 10, 15, 7, 12	3.33
R_3	0, 3, 5, 8, 6, 10, 15, 4, 14, 13, 9, 2, 1, 7, 12, 11	3.53
R_4	0, 3, 5, 8, 6, 12, 11, 7, 9, 14, 10, 13, 15, 2, 1, 4	3.53
R_5	0, 3, 5, 8, 6, 12, 11, 7, 10, 4, 9, 14, 15, 1, 2, 13	3.60
R_6	0, 3, 5, 8, 6, 12, 11, 7, 10, 13, 9, 14, 15, 1, 2, 4	3.53
R_7	0, 3, 5, 8, 6, 12, 11, 7, 13, 10, 14, 4, 1, 15, 2, 9	3.33
R_8	0, 3, 5, 8, 6, 12, 15, 1, 10, 4, 9, 14, 13, 11, 2, 7	3.40
R_9	0, 3, 5, 8, 6, 12, 15, 2, 14, 9, 11, 7, 13, 10, 4, 1	3.47
R_{10}	0, 3, 5, 8, 6, 13, 15, 1, 9, 12, 2, 11, 10, 7, 4, 14	3.40
R_{11}	0, 3, 5, 8, 6, 13, 15, 2, 7, 4, 14, 11, 10, 1, 9, 12	3.33
R_{12}	0, 3, 5, 8, 6, 13, 15, 2, 12, 9, 10, 4, 11, 14, 1, 7	3.47
R_{13}	0, 3, 5, 8, 6, 15, 10, 1, 7, , 14, 4, 11, 12, 13, 2	3.53
R_{14}	0, 3, 5, 8, 7, 4, 9, 14, 15, 6, 2, 11, 10, 13, 12, 1	3.47
R_{15}	0, 3, 5, 8, 7, 9, 11, 14, 10, 13, 15, 4, 12, 2, 6, 1	3.33
R_{16}	0, 3, 5, 8, 9, 12, 14, 7, 10, 13, 15, 4, 6, 11, 1, 2	3.27
R_{17}	0, 3, 5, 8, 10, 13, 9, 4, 15, 6, 2, 1, 12, 11, 7, 14	3.27
R_{18}	0, 3, 5, 8, 11, 12, 6, 15, 14, 9, 2, 7, 4, 10, 13, 1	3.33
R_{19}	0, 3, 5, 10, 7, 12, 11, 6, 13, 4, 2, 9, 14, 1, 8, 15	3.27

C. Tầng tuyến tính cho mã khối hạng nhẹ

Thông thường, tầng tuyến tính trong mã khối có cấu trúc SPN thường được đánh giá qua *số nhánh, tiêu chuẩn về điểm bất động, khả năng cài đặt.* Trong đó, tiêu chuẩn điểm bất động được phân tích dựa trên Luận văn tiến sĩ "*Phân tích các quan hệ tuyến tính trong mã khối*", năm 2010, của Muhammad Reza Z'aba (tại trường Công nghệ Queenslan - Australia) và đã xây dựng được bộ công cụ tính toán điểm bất động cho các mã pháp dạng tựa PRESENT và AES có cấu trúc khối 64 bit.

Hiện nay, trên thế giới phương pháp xây dựng tầng tuyến tính của mã khối dựa trên ma trận đồng hành đang được quan tâm. Với các tiêu chuẩn đánh giá tầng tuyến tính nghiên cứu được (cụ thể là số

điểm bất động, khả năng cài đặt), chúng tôi đã khảo sát tất cả các ma trận đồng hành có dạng *Serial(z_1, z_2, z_3, z_4)* mà lũy thừa bậc 4 của nó là những ma trận MDS trên trường $\mathbb{F}_{2^4}$ với đa thức bất khả quy x^4+x+1. Kết quả nhận được 3.660 ma trận thỏa mãn và được phân lớp theo các tiêu chuẩn đánh giá [5].

BẢNG 4. PHÂN LỚP 3.600 MA TRẬN *SERIAL(z_1,z_2,z_3,z_4)* MÀ LŨY THỪA BẬC 4 CỦA NÓ LÀ NHỮNG MÀ TRẬN MDS TRÊN TRƯỜNG $\mathbb{F}_2^4$ VỚI ĐA THỨC KHẢ QUY x^4+x+1.

Lớp	Lớp con	Tổng XOR	ClockCycle	Số điểm bất động của tầng tuyến tính được xây dựng (khác 0)	Số lượng theo lớp con	Số lượng
1	1	31	3	0	2	4
	2	31	4	0	2	
2	1	32	3	16	2	5
	2	32	4	0	1	
	3	32	4	16	2	
3	1	33	3	0	2	4
	2	33	4	0	2	
4	1	34	3	0	3	21
	2	34	4	0	16	
	3	34	4	16	2	
5	1	35	3	0	2	2
6	1	36	3	0	2	32
	2	36	4	0	26	
	3	36	4	16	4	
7	1	37	4	0	16	16
…	…	…	…	…	…	….
33	1	63	4	0	8	8
34	1	64	4	0	2	2
35	1	65	4	0	2	2
36	1	66	4	0	4	4

D. Đề xuất thuật toán mã khối hạng nhẹ tựa PRESENT

Kế thừa các kết quả lý thuyết và từ những thành phần mật mã tốt nhất, chúng tôi đề xuất hai thuật toán mã khối có cấu trúc SPN là N1 và N2. Thuật toán N1 chỉ tạo ra sự khác biệt so với thuật toán PRESENT bằng cách thay đổi hộp thế (mô tả trong hình 1). Trong khi đó, thuật toán N2 theo một thiết kế khác được sử dụng nhiều trong một số mã khối hạng nhẹ được đề xuất gần đây, khi tầng tuyến tính giống như AES, bao gồm bước *ShiftRows* và *MixColumnSerials*. Độ an toàn trước tấn công thám mã cũng như đánh giá dựa trên các tiêu chuẩn ngẫu nhiên cho đầu ra của thuật toán đề xuất đã được kiểm chứng kỹ lưỡng, dựa vào những công cụ đã phát triển trong Mục II. Ngoài ra, tác giả đã thực hiện cài đặt phần mềm và mô phỏng phần cứng của hai thuật toán đề xuất (cả chế độ mã và giải mã) trên ngôn ngữ Verilog cho IC FPGA Spartan 6 XC6SLX75 (đóng gói 484 chân, tốc độ -3) của Xilinx. Công cụ phục vụ thiết kế là Xilinx ISE phiên bản 14.3 WebPack, sử dụng Isim để mô phỏng chức năng của thiết kế. Một số kết quả mô phỏng được trình bày trong Bảng 5.

BẢNG 5. KẾT QUẢ CÀI ĐẶT MÔ PHỎNG PHẦN CỨNG FPGA CỦA HAI THUẬT TOÁN ĐỀ XUẤT

TT	Key/ Data	Enc/ Dec	LUTs	FFs	REGs *Slices*	Tần số tối đa (MHz)	Số chu kỳ	Thông lượng (Mbps)
N1	80/64	Enc	283	149	149	282	32	582
		Dec	283	149	149	282	32	582
	128/64	Enc	335	197	197	282	32	582
		Dec	335	197	197	282	32	582
2	64/64	Enc	*467*	*240*	*297*	*315*	*10*	*2.016*
		Dec	*646*	*325*	*394*	*280*	*10*	*1.792*
	80/64	Enc	*485*	*256*	*314*	*308*	*12*	*1.641*
		Dec	*663*	*339*	*410*	*282*	*12*	*1.503*
	96/64	Enc	*500*	*271*	*324*	*301*	*14*	*1.375*
		Dec	*855*	*356*	*428*	*278*	*14*	*1.270*
AES	128/128	Enc	1059	375	520	176	10	2.252
		Dec	1324	439	657	156	10	1.996

IV. KẾT LUẬN

Trước nhu cầu phát triển các thuật toán mã khối hạng nhẹ, các nhà nghiên cứu liên tục cho ra đời các thuật toán mới dựa trên cấu trúc và nguyên lý đa dạng. Có nhiều kỹ thuật đánh giá an toàn về mặt lý thuyết cũng như cài đặt rất tinh tế. Chúng tôi đã tiến hành nghiên cứu theo hướng này và đã đạt được một số kết quả nhất định. Bên cạnh việc nắm vững các nguyên lý thiết kế và tham số an toàn cho thuật toán mã khối hạng nhẹ nói chung và của PRESENT nói riêng, chúng tôi đã xây dựng được các thành phần mật mã như hộp thế, tầng tuyến tính, lược đồ khóa… cho các thuật toán mã khối hạng nhẹ. Tuy nhiên, còn nhiều vấn đề cần giải quyết như là tìm hiểu nguyên lý thiết kế cũng như các thành phần mật mã cho các mã khối hạng nhẹ có cấu trúc Feistel hay cấu trúc chỉ sử dụng phép cộng modulo, phép dịch vòng và phép XOR, được gọi là cấu trúc ARX. Hơn nữa, cần tiếp tục tối ưu hóa cài đặt cứng hóa thuật toán đề xuất nhằm ứng dụng hiệu quả trong các thiết bị có tài nguyên hạn chế.

TÀI LIỆU THAM KHẢO

[1]. International standard ISO/IEC 29192, "Information Technology - Security Techniques - Lightweight cryptography".

[2]. Axel York Poschmann, "Lightweight cryptography: cryptographic engineering for a pervasive world", in Ph. D. Thesis. 2009. Citeseer.

[3]. Bodhisatwa Mazumdar, D Mukhopadhay, và Indranil Sengupta (2013), "Constrained search for a class of good bijective S-boxes with improved DPA resistivity".

[4]. Emmanuel Prouff, "DPA attacks and S-boxes", in Fast Software Encryption. 2005. Springer.

[5]. Bạch Nhật Hồng, Trần Duy Lai, Nguyễn Bùi Cương, "Đề xuất các ma trận đồng hành xây dựng tầng tuyến tính trong mã khối hạng nhẹ", Tạp chí Nghiên cứu Khoa học Công nghệ Quân sự, đặc san CNTT 4/2014.

[6]. Cuong Nguyen, Lai Tran, and Khoa Nguyen, "On the resistance of Serpent-type 4 bit S-boxes against differential power attacks", Communications and Electronics (ICCE), 2014 IEEE Fifth International Conference on. IEEE, 2014.

SƠ LƯỢC VỀ TÁC GIẢ

ThS. Nguyễn Bùi Cương

Đơn vị công tác: Viện Khoa học - Công nghệ Mật mã, Ban Cơ yếu Chính phủ, Hà Nội.

E-mail: nguyenbuicuong@gmail.com

Tốt nghiệp ngành Toán học, Đại học Sư phạm - Đại học Quốc gia Hà Nội năm 2004. Tốt nghiệp Thạc sĩ Toán học - Đại học Khoa học Tự nhiên - Đại học Quốc gia Hà Nội năm 2008.

Hướng nghiên cứu hiện nay: Khoa học mật mã, mã hóa đối xứng.