Đề xuất S-hộp có tính chất mật mã tốt cho hoán vị của hàm băm Keccak

Nguyễn Văn Long, Lê Duy Đức

***Tóm tắt*—Keccak là hàm băm giành được chiến thắng trong cuộc thi SHA-3 của Viện Tiêu chuẩn và Công nghệ Mỹ (NIST) tổ chức. Có nhiều tấn công thám mã khai thác bậc đại số thấp trong hoán vị của hàm băm này. Chính những kết quả này mà nhóm tác giả thiết kế Keccak đã tăng số vòng từ 18 lên 24 trong hoán vị của nó. Trên cơ sở đó, bài báo tập trung phân tích tính chất đại số của hoán vị Keccak-*f* trong hàm băm này, sau đó đề xuất một thành phần S-hộp mới có tính chất mật mã tốt để sử dụng trong hoán vị của hàm băm Keccak.**

***Abstract*—Keccak is the winner of the SHA-3 competition of National Institute of Standards and Technology (NIST). There are many cryptographic attacks that exploit the low algebraic degree in permutation of this hash function. Due to these results, the Keccak design team increased the number of rounds from 18 to 24 in its permutation. On that basis, the paper focuses on analyzing the algebraic properties of the Keccak-*f* permutation in this hash function, then proposes a new S-box with good cryptographic properties used in Keccak's permutation.**

Từ khóa—Keccak; S-hộp; bậc đại số; SHA-3; tấn công phân biệt.

Keywords—Keccak; S-box; algebraic degree; SHA3; distinguishing attack.

I. GIỚI THIỆU

Cuộc thi tuyển chọn hàm băm SHA-3 do NIST tổ chức bắt đầu từ tháng 11/2007, kết thúc vào tháng 10/2012. Cuộc thi diễn ra trong 3 vòng với sự tham gia của 64 hàm băm dự tuyển. Sau khi kết thúc cuộc thi, Keccak là hàm băm chiến thắng và được lựa chọn để xây dựng chuẩn hàm băm mới SHA-3 của NIST. Chuẩn được công bố năm 2015 với tên gọi FIPS 202 [1].

Bài báo được nhận ngày 30/6/2020. Bài báo được nhận xét bởi phản biện thứ nhất ngày 03/8/2020 và được chấp nhận đăng ngày 03/8/2020. Bài báo được nhận xét bởi phản biện thứ hai ngày 11/7/2020 và được chấp nhận đăng ngày 29/8/2020.

Ngay từ khi được đề xuất, Keccak đã nhận được sự quan tâm của cộng đồng mật mã quốc tế. Một trong những lý do được quan tâm là cấu trúc thiết kế của hàm băm này dựa trên kiến trúc Sponge, đạt được độ an toàn chứng minh được một cách rõ ràng. Hơn nữa, các thành phần mật mã bên trong Keccak tạo nhiều lợi thế trong cài đặt trên nhiều nền tảng khác nhau. Đến nay, đã có hàng trăm công trình nghiên cứu về các tính chất cũng như thám mã lên hàm băm này, hầu như tất cả các nghiên cứu được nhóm thiết kế công bố và cập nhật thường xuyên trên website chính thức của hàm băm Keccak *(https://keccak.team/keccak.html)*.

Trong số các hướng nghiên cứu lên Keccak, nhóm tác giả đặc biệt quan tâm đến các kết quả đánh giá tính chất của hoán vị Keccak-*f* của nhóm tác giả C. Boura và cộng sự [2]-[5]. Công trình nghiên cứu của nhóm tác giả này khai thác tính chất tổng bằng không (zezo-sum property) trên cơ sở đạo hàm bậc cao, từ đó cho phép đánh giá tính chất phân biệt qua các vòng của hoán vị. Chính kết quả của nhóm nghiên cứu này mà các nhà thiết kế hàm băm Keccak đã quyết định tăng số vòng của hoán vị lên 24 thay vì 18 như đề xuất ban đầu.

Nghiên cứu đầu tiên theo hướng khai thác tính chất tổng bằng không là của nhóm J. Aumasson và W. Meier trong CHES 2009 [6]. Dựa vào việc đánh giá bậc đại số qua các vòng của hoán vị Keccak-*f*, các tác giả đã xây dựng bộ phân biệt lên 16 vòng của hoán vị này. Năm 2010, C. Boura và A. Canteaut đã công bố công trình nghiên cứu trong hội nghị ISIT 2010 [3]. Nghiên cứu này trình bày về tính chất tổng bằng không lên toàn bộ 18 vòng hoán vị Keccak-*f* trong phiên bản đầu tiên của hàm băm Keccak. Chính kết quả này mà nhóm thiết kế Keccak thay đổi số vòng của hoán vị lên 24. Cũng trong năm 2010 tại hội nghị SAC,

C. Boura và A. Canteaut đã mở rộng kết quả nghiên cứu trước đó và áp dụng để xây dựng bộ phân biệt lên 20 vòng cho phiên bản hàm băm Keccak mới. Kết quả cho phép xây dựng bộ phân biệt tổng bằng không có kích thước 2^{1586} lên 20 vòng của hoán vị Keccak-*f* [2]. Năm 2011, tại hội nghị FSE, C. Boura, A. Canteaut và C. De Cannière thực hiện nghiên cứu các tính chất vi sai bậc cao của Keccak [4]. Từ đó cho phép xây dựng bộ phân biệt tổng bằng không lên toàn bộ 24 vòng của hàm băm Keccak. Một nghiên cứu khác theo hướng này thuộc về nhóm tác giả M. Duan và X. Lai [7], được công bố năm 2012 khi cải tiến cận đánh giá của nhóm C. Boura và cộng sự để nhận được độ phức tạp nhỏ hơn.

Một hướng nghiên cứu khác khai thác tính chất tuyến tính hóa không đầy đủ của S-hộp (Non-Full S-box Linearization) để thực hiện tấn công lên Keccak. Hướng nghiên cứu này được Ling Song và cộng sự khai thác trong [8] và K. Qiao cùng cộng sự khai thác trong [9] để đánh giá độ an toàn lên số vòng rút gọn của Keccak. Ý tưởng chính trong những nghiên cứu này là thành lập các phương trình tuyến tính trên các tập con đầu vào của S-hộp của Keccak, sau đó khai thác để đưa ra các ước lượng an toàn cho số vòng rút gọn của Keccak.

Có thể thấy rằng, các phương trình biểu diễn S-hộp của Keccak có bậc đại số thấp chính là lý do các dạng tấn công mà tác giả liệt kê ở trên có thể khai thác. Với những phân tích như vậy, nhóm tác giả hướng đến đối tượng nghiên cứu trong báo cáo này là các S-hộp trong hoán vị Keccak-*f*, sự ảnh hưởng của nó lên độ an toàn và đề xuất S-hộp mới với mục đích tăng độ an toàn lên hoán vị Keccak-*f*. Với S-hộp đề xuất này, khi thay thế S-hộp trong hoán vị Keccak-*f* sẽ nhận được một hàm băm mới có cấu trúc Sponge (hàm băm Keccak sửa đổi).

Trên cơ sở như vậy, bố cục của bài báo được tổ chức như sau: Phần II mô tả về hoán vị Keccak-*f*; ở Phần III là một số phân tích về tính chất của hoán vị này; Phần IV trình bày về đề xuất thay thế S-hộp gốc trong Keccak bởi một S-hộp mới; Phần V sẽ trình bày về một số phân tích an toàn của hàm băm Keccak sửa đổi khi dùng S-hộp đề xuất; tiếp theo đánh giá khả năng thực thi khi sử dụng S-hộp đề xuất trong phần VI. Cuối cùng là phần kết luận.

II. MÔ TẢ HOÁN VỊ KECCAK-*F*

Họ hoán vị trong hàm băm Keccak được ký hiệu là *Keccak-f*$[b]$, với b là độ rộng (width) của hoán vị. Họ hoán vị này gồm các giá trị trong tập $\{25, 50, 100, 200, 400, 800, 1600\}$. Hoán vị hoạt động trong n_r vòng. Phụ thuộc vào giá trị độ rộng b, số vòng được xác định bởi $n_r = 12 + 2l$, ở đây $2^l = \frac{b}{25}$. Đối với *Keccak-f*$[1600]$, $n_r = 24$. Hàm vòng ký hiệu là $Round$, 24 vòng hoạt động của hoán vị trong Keccak được mô tả như sau:

$Keccak - f[b](X)$
  $for\ i\ in\ 0\ to\ n_r - 1$
    $X = Round[b](X, RC[i])$
  $return\ X$

Một vòng của hoán vị *Keccak-f* gồm một chuỗi các ánh xạ khả nghịch hoạt động trên trạng thái X mà được tổ chức bởi 5×5 *lane* (thuật ngữ *lane* có thể tham khảo trong [1]). Theo đó, mỗi phần tử của mảng X tương đương với 1 *lane* và mỗi *lane* có độ dài $w \in \{1, 2, 4, 8, 16, 32, 64\}$ bit. Một *lane* có tọa độ (x, y) trong mảng X được ký hiệu là $X[x, y]$. Trong dạng biểu diễn trạng thái theo *lane*, hàm vòng được mô tả như sau:

$Round[b](X, RC)$

$\theta\ step$:

$$C[x] = X[x,0] \oplus X[x,1] \oplus X[x,2] \oplus X[x,3] \oplus X[x,4], \quad 0 \le x \le 4$$

$$D[x] = C[x-1] \oplus ROT(C[x+1], 1), \quad 0 \le x \le 4$$

$$X[x,y] = X[x,y] \oplus D[x], 0 \le x, y \le 4$$

$\rho\ and\ \pi\ steps$:

$$Y[y, 2x+3y] = ROT(X[x,y], r[x,y]), \quad 0 \le x, y \le 4$$

χ *step*:

$$X[x,y] = Y[x,y] \oplus \begin{pmatrix} (NOT\ Y[x+1,y]) \\ AND\ Y[x+2,y] \end{pmatrix}, \quad 0 \le x,y \le 4$$

ι *step*:

$$X[0,0] = X[0,0] \oplus RC.$$

Trong đó, phép "+" được thực hiện theo modulo 5, X là trạng thái của hoán vị, Y, C, D là các biến trung gian, $\oplus$ là phép cộng theo modulo 2, NOT là phép phủ định, AND là phép nhân theo bit và ROT là phép dịch vòng trái của các *lane* đi r bit. Chi tiết về giá trị của $r[x,y]$, và $RC[i]$ có thể tham khảo trong [14].

Trong khuôn khổ bài báo này, nhóm tác giả tập trung lên hoán vị Keccak-*f*[1600], có nghĩa rằng mỗi *lane* trong nó có độ dài là một từ 64 bit ($w = 64$). Đây chính là hoán vị được sử dụng để xây dựng hàm băm trong chuẩn SHA-3.

III. Một số phân tích cho hoán vị Keccak-*f*

Nội dung phần này tập trung trình bày về S-hộp trong ánh xạ phi tuyến của Keccak, ước lượng bậc đại số qua các vòng của hoán vị Keccak-*f* và phân tích tính chất tuyến tính hóa không đầy đủ của hoán vị này.

A. S-hộp trong hoán vị Keccak-f

Hoán vị Keccak-*f* hoạt động trong 24 vòng với các biến đổi tuyến tính θ, π, ρ, ι và phi tuyến χ. Thành phần sử dụng trong biến đổi phi tuyến này là các S-hộp 5×5 bit. Biểu diễn hàm bool của nó ở dạng chuẩn tắc đại số ANF như sau:

$$y_0 = x_0 \oplus x_2 \oplus x_1x_2$$
$$y_1 = x_1 \oplus x_3 \oplus x_2x_3$$
$$y_2 = x_2 \oplus x_4 \oplus x_3x_4$$
$$y_3 = x_3 \oplus x_0 \oplus x_4x_0$$
$$y_4 = x_4 \oplus x_1 \oplus x_0x_1$$

Nhận thấy rằng, bậc đại số của S-hộp trên chỉ bằng 2 là không cao đối với một S-hộp 5×5 bit. Chính giá trị này được khai thác trong nhiều phân tích tấn công lên Keccak.

Đối với S-hộp của Keccak, nhóm tác giả đưa ra kết quả sự phụ thuộc các bit trong một vòng của hoán vị Keccak-*f* của hàm băm SHA-3 như sau:

Mệnh đề 1 [15]. *Đối với biến đổi vòng trong hoán vị Keccak-f của hàm băm SHA-3 có:*

- 128 *bit đầu ra phụ thuộc vào* 32 *bit đầu vào;*
- 1472 *bit đầu ra phụ thuộc vào* 33 *bit đầu vào.*

B. Bậc đại số của hoán vị Keccak-f

Để đánh giá bậc đại số của hoán vị trong nhiều nguyên thủy mật mã, C. Boura và cộng sự đã phát biểu và chứng minh định lý sau.

Định lý 2 (Theorem 3 [5]). *Cho F là hoán vị từ $\mathbb{F}_2^n$ vào $\mathbb{F}_2^n$ tương ứng với phép nối của s hoán vị $S_1, \dots, S_s$ nhỏ hơn trên $\mathbb{F}_2^{n_0}$. Gọi δ_k là bậc lớn nhất của tích k hàm tọa độ nào đó của những S-hộp này. Khi đó với hàm G bất kỳ từ $\mathbb{F}_2^n$ vào $\mathbb{F}_2^m$, ta có:*

$$\deg(G \circ F) \le n - \frac{n - \deg(F)}{\gamma},$$

trong đó

$$\gamma = \max_{1 \le k \le n_0 - 1} \frac{n_0 - k}{n_0 - \max_{1 \le j \le s} \delta_k(S_j)}$$

Đặc biệt, ta có:

$$\gamma \le \max_{1 \le j \le s} \max\left(\frac{n_0 - 1}{n_0 - \deg(S_j)}, \frac{n_0}{2} - 1, \deg(S_j^{-1})\right).$$

Từ biểu diễn ANF của S-hộp trong ánh xạ χ của Keccak và khi xét tất cả các tổ hợp có thể của những hàm bool này chúng ta có:

k	1	2	3	4	5
$\delta_k(S_j)$	2	4	4	4	5

Tương tự đối với S-hộp nghịch đảo trong ánh xạ χ^{-1} có:

k	1	2	3	4	5
$\delta_k(S_j^{-1})$	3	3	4	4	5

Từ bảng bậc đại số của $\delta_k(\chi)$, $\delta_k(\chi^{-1})$ ở trên và theo Định lý 2, tính được:

$$\gamma(\chi) = \max\left(\frac{4}{3}, \frac{3}{1}, \frac{2}{1}, \frac{1}{1}\right) = 3.$$

$$\gamma(\chi^{-1}) \leq \max\left(\frac{5-1}{5-3}, \frac{3}{2}, 2\right) = 2.$$

Như vậy, biểu thức sau áp dụng cho cả hoán vị Keccak-*f* và nghịch đảo của nó:

$$\deg(R^r) \leq 1600 - \frac{1600 - \deg(R^{r-1})}{3}$$

$$\deg(invR^r) \leq 1600 - \frac{1600 - \deg(invR^{r-1})}{2}$$

Từ đây, có thể ước lượng về bậc đại số qua các vòng của hoán vị Keccak-*f* như sau (phần in đậm tính theo công thức $\deg(R^r)$ và $\deg(invR^r)$ ở trên):

BẢNG 1. BẬC ĐẠI SỐ QUA CÁC VÒNG CỦA HOÁN VỊ KECCAK-F

r	$\deg(R^r)$	$\deg(invR^r)$
1	2	3
2	4	9
3	8	27
4	16	81
5	32	243
6	64	729
7	128	**1164**
8	256	**1382**
9	512	**1491**
10	1024	**1545**
11	**1408**	**1572**
12	**1536**	**1586**
13	**1578**	**1583**
14	**1592**	**1596**
15	**1597**	**1598**
16	**1599**	**1599**

Kết quả này đã được công bố năm 2011 tại Hội nghị FSE bởi C. Boura, A. Canteaut và C. De Cannière. Từ đây, nhóm tác giả thực hiện xây dựng bộ phân biệt lên toàn bộ 24 vòng của hoán vị Keccak-*f* [4].

Từ phân tích ở trên, ta thấy rằng bậc đại số thấp của S-hộp trong Keccak là ảnh hưởng đến độ an toàn của hàm băm. Việc tăng bậc đại số sẽ làm tăng độ phức tạp trước tấn công phân biệt. Tuy nhiên sẽ kéo theo sự phức tạp trong cài đặt, và một đề xuất cụ thể cần được xem xét theo một phương diện tổng thể.

C. Tính tuyến tính hóa không đầy đủ của S-hộp

Khái niệm tính tuyến tính hóa không đầy đủ của S-hộp (Non-Full S-box Lineariation) được đưa ra bởi Ling Song và cộng sự trong [8]. Tuy nhiên, trước đó tính chất này cũng đã được Dinur và cộng sự sử dụng trong [10] và K. Qiao cùng cộng sự trong [9].

Bản chất của việc áp dụng này xuất phát từ một nhận xét quan trọng là trạng thái trong của hàm băm Keccak có kích thước lớn hơn rất nhiều so với kích thước giá trị băm, cho phép kẻ tấn công có số lượng lớn bậc tự do (nhiều lựa chọn cho tham số để thành lập các hệ phương trình tuyến tính cho số vòng rút gọn). Một vài tập con thuộc các không gian khả dĩ với những tính chất đặc biệt có thể được lựa chọn để tăng tốc quá trình tấn công. Trong trường hợp của Keccak, các tác giả trong [9] chọn các tập con có tính chất tuyến tính tương ứng với S-hộp, có nghĩa là biểu thức của S-hộp có thể viết lại thành các biến đổi tuyến tính khi đầu vào được giới hạn bởi tập con như vậy. Xem xét định nghĩa sau:

Định nghĩa 3 (**Definition 1 [9]**) *Những không gian con affine có thể tuyến tính là những không gian con các đầu vào affine, mà trên những không gian con này, S-hộp là tương đương với một biến đổi tuyến tính. Nếu V là một không gian con có thể tuyến tính của biến đổi $S(.)$ của S-hộp, khi đó $\forall x \in V, S(x) = A.x + b$, trong đó A là ma trận và b là hằng số.*

Ví dụ, khi đầu vào của S-hộp trong Keccak-*f* được giới hạn trong tập {00000,00001,00100,00101} hoặc {00,01,04,05} trong hệ Hex, thì đầu ra tương ứng

của S-hộp này là: {00000,01001,00101,01100} và nó có thể biểu diễn bởi:

$$y = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot x$$

Nhận xét 1 (**Observation 1 [9]**):

- *Tồn tại 80 không gian con 2 chiều affine có thể tuyến tính đối với S-hộp của Keccak-f.*
- *Không tồn tại không gian con có thể tuyến tính với số chiều lớn hơn hoặc bằng 3.*

Trong [9], các tác giả tìm được 80 không gian con có số chiều bằng 2 như trong nhận xét trên. Tuy nhiên khi kiểm tra lại, nhóm tác giả thấy nhiều bộ không thỏa mãn. Ví dụ bộ $\{1, 2, 9, A\} = \{00001, 00010, 01001, 01010\}$. Thấy rằng:

000**0**1
000**1**0
01**0**0**1**
01**0**1**0**

Trong bộ trên, bit x_2 và x_4 luôn bằng 0. Do vậy, từ phương trình biểu diễn ANF các bit đầu ra của S-hộp trong Keccak có:

$$y_0 = x_0 + (x_1 + 1) \cdot x_2 = x_0$$
$$y_1 = x_1 + (x_2 + 1) \cdot x_3 = x_1 + x_3$$
$$y_2 = x_2 + (x_3 + 1) \cdot x_4 = x_2$$
$$y_3 = x_3 + (x_4 + 1) \cdot x_0 = x_3 + x_0$$
$$y_4 = x_4 + (x_0 + 1) \cdot x_1 = x_1 + x_0 \cdot x_1$$

Thấy rằng chỉ có 4 trong số 5 phương trình là tuyến tính, nên không thể biểu diễn về dạng:

$$y = M \times x,$$

trong đó, M là ma trận nhị phân 5×5, $x = (x_0, x_1, x_2, x_3, x_4)^T$ và $x \in \{1, 2, 9, A\}$.

Nhóm tác giả đã thực hiện tính lại theo điều kiện của Định nghĩa 3, kết quả chỉ tìm được 40 bộ như trong Bảng 2 dưới đây:

BẢNG 2. 40 KHÔNG GIAN CON AFFINE CÓ THỂ TUYẾN TÍNH ĐỐI VỚI S-HỘP CỦA KECCAK

{ 0, 1, 4, 5}, { 0, 1, 8, 9}, { 0, 2, 8, A}, { 0, 2,10,12}, { 0, 4,10,14}, { 1, 3, 9, B}, { 1, 3,11,13}, { 1, 5,11,15}, { 2, 3, 6, 7}, { 2, 3, A, B}, { 2, 6,12,16}, { 3, 7,13,17}, { 4, 5, C, D}, { 4, 6, C, E}, { 4, 6,14,16}, { 5, 7, D, F}, { 5, 7,15,17}, { 6, 7, E, F}, { 8, 9, C, D}, { 8, A,18,1A}, { 8, C,18,1C}, { 9, B,19,1B}, { 9, D,19,1D}, { A, B, E, F}, { A, E,1A,1E}, {B,F,1B,1F}, {C,E,1C,1E}, {D,,1D,1F}, {10,11,14,15}, {10,11,18,19}, {10,12,18,1A}, {11,13,19,1B}, {12,13,16,17}, {12,13,1A,1B}, {14,15,1C,1D}, {14,16,1C,1E}, {15,17,1D,1F}, {16,17,1E,1F}, {18,19,1C,1D}, {1A,1B,1E,1F}

Vì không gian con affine được sử dụng cùng với các vệt vi sai, nên chúng ta quan tâm đến các không gian con affine có thể tuyến tính được với sai khác đầu vào và đầu ra cố định. Và chúng liên quan đến phân bố vi sai theo bảng DDT của S-hộp.

Từ đây, các tác giả trong [9] đưa ra nhận xét 2 như sau:

Nhận xét 2 (Observation 2 [9]). *Với sai khác đầu vào 5 bit δ_{in} và sai khác đầu ra 5 bit δ_{out} thỏa mãn $DDT(\delta_{in}, \delta_{out}) \neq 0$, ký hiệu tập $V = \{x: S(x) \oplus S(x \oplus \delta_{in}) = \delta_{out}$ và $S(V) = \{S(x): x \in V\}$. Ta có:*

- *Nếu $DDT(\delta_{in}, \delta_{out}) = 2$ hoặc 4, thì V là không gian con affine có thể tuyến tính.*
- *Nếu $DDT(\delta_{in}, \delta_{out}) = 8$, có 6 tập con 2 chiều $W_i \subset V, i = 0, 1, \dots, 5$ thỏa mãn W_i là những không gian con affine có thể tuyến tính.*

Trường hợp $DDT(\delta_{in}, \delta_{out}) = 2$ có thể dễ dàng kiểm tra được V là không gian con affine có thể tuyến tính. Tuy nhiên, khi $DDT(\delta_{in}, \delta_{out}) = 4$ thì không phải tất cả V là không gian con affine có thể tuyến tính. Thật vậy, bảng DDT S-hộp trong Keccak cho thấy rằng có 120 tập V thỏa mãn $DDT(\delta_{in}, \delta_{out}) = 4$. Trong khi đó, các tác giả Qiao và cộng sự trong [9] chỉ ra có 80 tập gồm 4 phần tử, còn nhóm tác giả chỉ ra chỉ có 40 tập như trong Bảng 2.

Bằng thực nghiệm, nhóm tác giả thấy rằng, cũng chỉ có 40 tập trong Bảng 2 là thỏa mãn Nhận xét 2 mà thôi.

Trong mỗi trường hợp $DDT(\delta_{in}, \delta_{out}) = 8$, chỉ có 4 tập thỏa mãn mà không phải 6 như trong Nhận xét 2.

Ở một hướng khác, Ling Song và cộng sự trong [8] không cần phải sử dụng sự tuyến tính đầy đủ trong các tập, mà chỉ cần sử dụng một số thành phần tuyến tính trong tập để thực hiện tấn công tìm va chạm lên 5 vòng. Tấn công là có thể thực hành được và dựa trên nhận xét sau:

Nhận xét 3 (Observation 2 [8]). *Gọi δ_{in} và δ_{out} là sai khác đầu vào và đầu ra của S-hộp 5 bit trong Keccak-f mà thỏa mãn $DDT(\delta_{in}, \delta_{out}) = 8$. Khi đó, 4 trong số 5 đầu ra của S-hộp là tuyến tính nếu đầu vào được chọn trong tập $V = \{x: S(x) + S(x + \delta_{in}) = \delta_{out}\}$.*

Ví dụ $DDT(01,01) = 8$, tập V tính được là $V = \{10,11,14,15,18,19,1C,1D\}$. V được biểu diễn về dạng nhị phân như sau:

$$10: \mathbf{1}00\mathbf{0}0$$
$$11: \mathbf{1}00\mathbf{0}1$$
$$14: \mathbf{1}01\mathbf{0}0$$
$$15: \mathbf{1}01\mathbf{0}1$$
$$18: \mathbf{1}10\mathbf{0}0$$
$$19: \mathbf{1}10\mathbf{0}1$$
$$1C: \mathbf{1}11\mathbf{0}0$$
$$1D: \mathbf{1}11\mathbf{0}1$$

Khi xét trên tập này ta luôn có: $x_1 = 0$, còn $x_4 = 1$. Như vậy, đầu ra của S-hộp trên tập này có thể biểu diễn dưới dạng:

$$y_0 = x_0 + x_2$$
$$y_1 = (x_2 + 1)x_3$$
$$y_2 = x_2 + x_3 + 1$$
$$y_3 = x_3$$
$$y_4 = 1$$

Rõ ràng 4 trong số 5 bit đầu ra là tuyến tính.

Khai thác các tính chất như vậy, năm 2017, nhóm L. Song và cộng sự trong [8], [11] đã đưa ra phân tích an toàn trước tấn công tìm và chạm lên 5 vòng của Keccak[r = 1142, c = 448] với độ phức tạp 2^{50}; năm 2019 nhóm J. Guo và cộng sự đưa ra tấn công thực hành và xây dựng được va chạm thực sự lên 5 vòng của SHA3-224, SHA3-256 và 5 vòng của SHAKE128 [12]; năm 2019, M. S. Rajasree công bố công trình phân tích lý thuyết việc tìm tiền ảnh lên 4 vòng đối với một số phiên bản của SHA-3 [13].

Các kết quả trên cho thấy sự ảnh hưởng tính chất của S-hộp lên hoán vị Keccak-*f*. Do đó, việc lựa chọn S-hộp làm sao để không có tính chất như trong các Nhận xét 1, 2, 3 sẽ góp phần tăng độ an toàn của nguyên thủy mật mã sử dụng nó.

IV. ĐỀ XUẤT S-HỘP SỬ DỤNG TRONG HOÁN VỊ KECCAK-*F*

Chúng tôi đề xuất sử dụng S-hộp 5×5 bit mà biểu diễn hàm bool của nó ở dạng ANF là:

$$y_0 = 1 \oplus x_1 \oplus x_0x_2 \oplus x_1x_2 \oplus x_3x_4 \oplus x_1x_2x_3x_4$$
$$y_1 = 1 \oplus x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_4x_0 \oplus x_2x_3x_4x_0$$
$$y_2 = 1 \oplus x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_0x_1 \oplus x_3x_4x_0x_1$$
$$y_3 = 1 \oplus x_4 \oplus x_3x_0 \oplus x_4x_0 \oplus x_1x_2 \oplus x_4x_0x_1x_2$$
$$y_4 = 1 \oplus x_0 \oplus x_4x_1 \oplus x_0x_1 \oplus x_2x_3 \oplus x_0x_1x_2x_3 \quad (1)$$

thay thế cho S-hộp sử dụng trong ánh xạ χ của hoán vị Keccak-*f*. Tính chất mật mã của nó so với S-hộp gốc trong Keccak-*f* như trong Bảng 3 sau:

BẢNG 3. TÍNH CHẤT MẬT MÃ CỦA S-HỘP TRONG KECCAK VÀ ĐỀ XUẤT

Tính chất	S-hộp trong Keccak	S-hộp đề xuất
Độ phi tuyến	8	10
Bậc đại số	2	4
Giá trị AC	32	24
Số điểm bất động	2	0
Giá trị vi sai cực đại	8	4
Giá trị xấp xỉ tuyến tính cực đại	8	6
Tính cân bằng	Có	Có

Từ Bảng 3 thấy rằng, S-hộp đề xuất có các tính chất mật mã tốt hơn so với S-hộp trong Keccak.

S-hộp đề xuất có bậc đại số cao hơn. Tính chất này để đảm bảo sự ảnh hưởng các bit đầu vào/đầu ra lên các bit đầu ra/đầu vào là tốt hơn. Đặc biệt, các tính chất đại số của hoán vị Keccak-*f* mới sẽ tốt hơn. (cụ thể sẽ phân tích ở những phần sau). Tính chất vi sai của S-hộp này cũng cao hơn (bằng 4 so với 8 như của Keccak) sẽ đảm bảo một xác suất của vệt vi sai nhỏ hơn, được khai thác trong các tấn công dạng vi sai (tấn công rebound) lên Keccak.

Cách tiếp cận trong xây dựng S-hộp trên là như sau:

- Chọn lần lượt các hàm bool có bậc đại số cao nhất có thể mà có dạng biểu diễn ANF đơn giản để đảm bảo tính cài đặt.
- Xây dựng S-hộp sử dụng hàm bool trên theo quy tắc "dịch vòng": chỉ số các biến trong hàm bool sau bằng chỉ số các biến trong hàm bool trước cộng với 1 theo modulo 5. Ví dụ với hàm bool thứ nhất:

$$y_0 = 1 \oplus x_0 \oplus x_0x_3 \oplus x_2x_4,$$

thì hàm bool thứ 2 sẽ là:

$$y_1 = 1 \oplus x_1 \oplus x_1x_4 \oplus x_3x_0,$$

còn hàm bool thứ 3 sẽ là:

$$y_2 = 1 \oplus x_2 \oplus x_2x_0 \oplus x_4x_2.$$

- Tính các tính chất mật mã của S-hộp nhận được để lựa chọn các S-hộp tốt nhất.

Trong số các S-hộp như vậy, nhóm tác giả tìm được 20 S-hộp có dạng "dịch vòng" như trên mà có bậc đại số bằng 4, giá trị vi sai cực đại bằng 4, cân bằng, độ phi tuyến bằng 10.

Những S-hộp này có cấu trúc giống hệt nhau và có dạng đối xứng theo quy tắc "dịch vòng" như cấu trúc S-hộp trong Keccak. Tuy nhiên chúng lại có 02 điểm bất động là S[0] = 0 và S[31] = 31. Để loại bỏ điều này, nhóm tác giả sử dụng dạng phủ định của nó. Từ đây nhận được S-hộp đề xuất có dạng biểu diễn ANF như trên.

Dạng phủ định này cũng sẽ đơn giản hóa trong quá trình phân tích cài đặt. Thật vậy, với biểu diễn ANF như trên, các hàm bool trên có thể đưa về dạng:

$$y_0 = x_1 \oplus x_0x_2 \oplus (1 \oplus x_1x_2)(1 \oplus x_3x_4) = x_1 \oplus x_0x_2 \oplus \overline{x_1x_2} \cdot \overline{x_3x_4}$$

$$y_1 = x_2 \oplus x_1x_3 \oplus (1 \oplus x_2x_3)(1 \oplus x_4x_0) = x_2 \oplus x_1x_3 \oplus \overline{x_2x_3} \cdot \overline{x_4x_0}$$

$$y_2 = x_3 \oplus x_2x_4 \oplus (1 \oplus x_3x_4)(1 \oplus x_0x_1) = x_3 \oplus x_2x_4 \oplus \overline{x_3x_4} \cdot \overline{x_0x_1}$$

$$y_3 = x_4 \oplus x_3x_0 \oplus (1 \oplus x_4x_0)(1 \oplus x_1x_2) = x_4 \oplus x_3x_0 \oplus \overline{x_4x_0} \cdot \overline{x_1x_2}$$

$$y_4 = x_0 \oplus x_4x_1 \oplus (1 \oplus x_0x_1)(1 \oplus x_2x_3) = x_0 \oplus x_4x_1 \oplus \overline{x_0x_1} \cdot \overline{x_2x_3}$$

Nếu đặt các biến phụ

$$t_0 = \overline{x_0x_1}, t_1 = \overline{x_1x_2}, t_2 = \overline{x_2x_3}, t_3 = \overline{x_3x_4}, t_4 = \overline{x_4x_0},$$

ta có bảng so sánh cài đặt của S-hộp đề xuất và Keccak như sau:

BẢNG 4. SỐ PHÉP TOÁN LOGIC CÀI ĐẶT CHO S-HỘP CỦA KECCAK VÀ ĐỀ XUẤT

	Số biến phụ	Số phép AND	Số phép XOR	Số phép NOT
S-hộp trong Keccak	0	5	5	5
S-hộp đề xuất	5	15	15	5

Trên đây là một số phân tích về cơ sở đề xuất S-hộp để thay thế cho S-hộp trong hoán vị Keccak-*f*.

V. ĐÁNH GIÁ AN TOÀN KHI SỬ DỤNG S-HỘP ĐỀ XUẤT TRONG HOÁN VỊ KECCAK-*F*

Với S-hộp đề xuất ở Phần IV, chúng ta có thể thay thế nó vào S-hộp trong hàm băm Keccak để nhận được một sửa đổi mới của hàm băm có cấu trúc Sponge. Tuy nhiên, khi thay thế như vậy sẽ dẫn tới một số câu hỏi đặt ra là liệu có ảnh hưởng đến độ an toàn của hàm băm nhận được hay không? Trong phần này, nhóm tác giả sẽ tập trung phân tích một số khía cạnh như: sự phụ thuộc các bit đầu vào và đầu ra của hàm vòng trong hoán vị, bậc đại số của hoán vị, tính tuyến tính hóa không đầy đủ của S-hộp và một số bình luận về độ an toàn của cấu trúc Sponge tổng thể.

A. Sự phụ thuộc các bit đầu vào/đầu ra trong hàm vòng

Tương tự như Mệnh đề 1, nhóm tác giả phát biểu và chứng minh Mệnh đề 2 với S-hộp đề xuất như sau:

Mệnh đề 2. *Đối với biến đổi vòng trong hoán vị Keccak-f mà sử dụng S-hộp đề xuất có:*

- 320 *bit đầu ra, mỗi bit phụ thuộc vào* 54 *bit đầu vào;*
- 1280 *bit đầu ra, mỗi bit phụ thuộc vào* 55 *bit đầu vào.*

Chứng minh

Ví dụ đối với hàm y_0, có:

$$y_0 = x_1 \oplus x_0x_2 \oplus (1 \oplus x_1x_2)(1 \oplus x_3x_4) \quad (2)$$

Xét *lane* có tọa độ (x, y) bất kỳ, $0 \le x, y \le 4$. Để biểu thức nhỏ gọn hơn, ký hiệu *lane* bởi ký tự L. Thực hiện biểu diễn hàm bool trên qua các ánh xạ χ, π, ρ và θ trong biến đổi vòng của hoán vị Keccak-p, có:

$$\begin{aligned} L[x,y] \overset{\chi}{\leftarrow}\ & L[x+1,y] \oplus L[x,y] \cdot L[x+2,y] \\ & \oplus (1 + L[x+1,y] \\ & \cdot L[x+2,y]) \\ & \cdot (1 + L[x+3,y] \cdot L[x+4,y]) \end{aligned}$$

$$\begin{aligned} \overset{\pi}{\leftarrow}\ & L[x+3y+1, x+1] \oplus L[x+3y, x] \\ & \cdot L[x+3y+2, x+2] \\ & \oplus (1 + L[x+3y+1, x+1] \\ & \cdot L[x+3y+2, x+2]) \\ & \cdot \begin{pmatrix} 1 + L[x+3y+3, x+3] \\ L[x+3y+4, x+4] \end{pmatrix} \end{aligned}$$

$$\begin{aligned} \overset{\rho}{\leftarrow}\ & \underbrace{(L[x+3y+1, x+1] \ggg a)}_{A} \oplus \\ & \underbrace{(L[x+3y, x] \ggg b)}_{B} \cdot \\ & \underbrace{(L[x+3y+2, x+2] \ggg c)}_{C} \oplus \Big(1 \oplus \\ & \underbrace{(L[x+3y+1, x+1] \ggg a)}_{A} \cdot \\ & \underbrace{(L[x+3y+2, x+2] \ggg c)}_{C}\Big) \cdot \Big(1 \oplus \\ & \underbrace{(L[x+3y+3, x+3] \ggg d)}_{D} \cdot \\ & \underbrace{(L[x+3y+4, x+4] \ggg e)}_{E}\Big), \end{aligned} \quad (3)$$

trong đó, $a, b, c, d,$ và e là các giá trị *offset* được quy định bởi biến đổi ρ. Giá trị *offset* có thể tham khảo trong [1]:

$$\begin{aligned} a &= offset[x+3y+1, x+1] \\ b &= offset[x+3y, x] \\ c &= offset[x+3y+2, x+2] \end{aligned}$$

$$d = offset[x + 3y + 3, x + 3]$$
$$e = offset[x + 3y + 4, x + 4]$$

Trong trường hợp $w = 64$, có $a \neq b \neq c \neq e \neq d$.

Như vậy,

$$L[x, y] = A \oplus BC \oplus (1 \oplus AC)(1 \oplus DE).$$

Xét biểu thức A qua ánh xạ θ, có:

$$A = (L[x + 3y + 1, x + 1] \ggg a) \Rightarrow$$

$$A \overset{\theta}{\leftarrow} (L[x + 3y + 1, x + 1] \ggg a) \oplus (D^*[x + 3y + 1] \ggg a)$$

$$= (L[x + 3y + 1, x + 1] \ggg a) \oplus (C^*[x + 3y] \ggg a) \oplus \big((C^*[x + 3y + 2] \lll 1) \ggg a\big)$$

$$= (L[x + 3y + 1, x + 1] \ggg a) \oplus (C^*[x + 3y] \ggg a) \oplus \big(C^*[x + 3y + 2] \ggg (a - 1)\big)$$

$$= (L[x + 3y + 1, x + 1] \ggg a) \oplus \sum_{i=0}^{4}(L[x + 3y, i] \ggg a) \oplus \sum_{i=0}^{4}\big(L[x + 3y + 2, i] \ggg (a - 1)\big) \quad (4)$$

Đối với biểu thức B:

$$B = (L[x + 3y, x] \ggg b) \Rightarrow$$

$$B \overset{\theta}{\leftarrow} (L[x + 3y, x] \ggg b) \oplus \sum_{i=0}^{4} (L[x + 3y + 4, i] \ggg b) \oplus \sum_{i=0}^{4}\big(L[x + 3y + 1, i] \ggg (b - 1)\big). \quad (5)$$

Đối với biểu thức C:

$$C = (L[x + 3y + 2, x + 2] \ggg c) \Rightarrow$$

$$C \overset{\theta}{\leftarrow} (L[x + 3y + 2, x + 2] \ggg c) \oplus \sum_{i=0}^{4}(L[x + 3y + 1, i] \ggg c) \oplus \sum_{i=0}^{4}\big(L[x + 3y + 3, i] \ggg (c - 1)\big). \quad (6)$$

Đối với biểu thức D:

$$D = (L[x + 3y + 3, x + 3] \ggg d) \Rightarrow$$

$$D \overset{\theta}{\leftarrow} (L[x + 3y + 3, x + 3] \ggg d) \oplus \sum_{i=0}^{4}(L[x + 3y + 2, i] \ggg d) \oplus \sum_{i=0}^{4}\big(L[x + 3y + 4, i] \ggg (d - 1)\big). \quad (7)$$

Đối với biểu thức E:

$$E = (L[x + 3y + 4, x + 4] \ggg e) \Rightarrow$$

$$E \overset{\theta}{\leftarrow} (L[x + 3y + 4, x + 4] \ggg e) \oplus \sum_{i=0}^{4}(L[x + 3y + 3, i] \ggg e) \oplus \sum_{i=0}^{4}\big(L[x + 3y, i] \ggg (e - 1)\big). \quad (8)$$

Trong các biểu thức (4), (5), (6), (7) và (8) ở trên, mỗi biểu thức sẽ phụ thuộc vào 11 biến theo mỗi tọa độ của x và y. Tiếp theo, tùy thuộc vào giá trị của bảng *offset* trong biến đổi ρ, chúng ta cần tìm các biến chung ở mỗi biểu thức A, B, C, D và E để có thể biết chính xác mỗi bit đầu ra phụ thuộc vào bao nhiêu bit đầu vào.

Từ bảng *offset* của biến đổi ρ có:

1. **(x, y) = (0, 0)**: ***a = 44***, b = 0, ***c = 43***, d = 31, e = 14
2. (x, y) = (0, 1): a = 20, b = 28, c = 3, d = 45, e = 61
3. (x, y) = (0, 2): a = 6, b = 1, c = 25, d = 8, e = 18
4. (x, y) = (0, 3): a = 36, b = 27, c = 10, d = 15, e = 56
5. (x, y) = (0, 4): a = 55, b = 62, c = 39, d = 41, e = 2
6. **(x, y) = (1, 0)**: ***a = 43***, ***b = 44***, c = 31, d = 14, e = 0
7. (x, y) = (1, 1): a = 3, b = 20, c = 45, d = 61, e = 28
8. (x, y) = (1, 2): a = 25, b = 6, c = 8, d = 18, e = 1

9. (x, y) = (1, 3): a = 10, b = 36, c = 15, d = 56, e = 27
10. (x, y) = (1, 4): a = 39, b = 55, c = 41, d = 2, e = 62
11. **(x, y) = (2, 0)**: a = 31, ***b = 43***, c = 14, d = 0, ***e = 44***
12. (x, y) = (2, 1): a = 45, b = 3, c = 61, d = 28, e = 20
13. (x, y) = (2, 2): a = 8, b = 25, c = 18, d = 1, e = 6
14. (x, y) = (2, 3): a = 15, b = 10, c = 56, d = 27, e = 36
15. (x, y) = (2, 4): a = 41, b = 39, c = 2, d = 62, e = 55
16. **(x, y) = (3, 0)**: a = 14, b = 31, c = 0, ***d = 44***, ***e = 43***
17. (x, y) = (3, 1): a = 61, b = 45, c = 28, d = 20, e = 3
18. (x, y) = (3, 2): a = 18, b = 8, c = 1, d = 6, e = 25
19. (x, y) = (3, 3): a = 56, b = 15, c = 27, d = 36, e = 10
20. (x, y) = (3, 4): a = 2, b = 41, c = 62, d = 55, e = 39
21. **(x, y) = (4, 0)**: a = 0, b = 14, ***c = 44***, ***d = 43***, e = 31
22. (x, y) = (4, 1): a = 28, b = 61, c = 20, d = 3, e = 45
23. (x, y) = (4, 2): a = 1, b = 18, c = 6, d = 25, e = 8
24. (x, y) = (4, 3): a = 27, b = 56, c = 36, d = 10, e = 15
25. (x, y) = (4, 4): a = 62, b = 2, c = 55, d = 39, e = 41

Trong mỗi *lane* ở các biểu thức trên, đại lượng a, b, c, d và e sẽ quy định xem các bit có nằm cùng 1 *slice* hay không (*slice* là một mặt bit có kích thước 5×5 bit, chi tiết minh họa thuật ngữ *slice* có thể tham khảo trong [1]. Chúng ta chỉ quan tâm đến các giá trị của a, b, c, d và e mà chúng liên tiếp nhau. Ví dụ, trong trường hợp $(x, y) = (0, 0)$, có $a = 44$ và $c = 43$. Vì trong các biểu thức của A, B, C, D và E chứa các thành phần $a, a-1, b, b-1, c, c-1, d, d-1, e$ và $e-1$ (thỏa mãn như phần in đậm ở phân tích theo bảng *offset)*. Nếu không có các giá trị liên tiếp như vậy, có nghĩa là các bit nằm ở các *slice* khác nhau, nghĩa là trong biểu thức của A, B, C, D và E sẽ không chứa các *lane* chung hoặc chứa các *lane* chung nhưng các bit tương ứng nằm ở các *slice* khác nhau. Từ đây, chúng ta sẽ có kết luận về số bit phụ thuộc.

Xét các trường hợp cụ thể sau:

Trường hợp $(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{0}, \boldsymbol{0})$***, có***:

$$A \xrightarrow{\theta^{-1}} (L[1,1] \ggg 44) \oplus \sum_{i=0}^{4} (L[0,i] \ggg 44) \oplus \sum_{i=0}^{4} (\boldsymbol{L}[\boldsymbol{2}, \boldsymbol{i}] \ggg \boldsymbol{43})$$

$$C \xrightarrow{\theta^{-1}} (\boldsymbol{L}[\boldsymbol{2}, \boldsymbol{2}] \ggg \boldsymbol{43}) \oplus \sum_{i=0}^{4} (L[1,i] \ggg 43) \oplus \sum_{i=0}^{4} (L[3,i] \ggg 42)$$

Thấy rằng khi $i = 2$, hai biểu thức trên có 1 *lane* chung là $L[2,2] \ggg 43$. Do vậy, trong trường hợp này 64 bit thuộc $L[0,0]$ sẽ phụ thuộc vào 11 + 11 + 11 + 11 + 11 – 1 = 54 bit đầu vào.

Trường hợp $(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{1}, \boldsymbol{0})$***, có***:

$$A \xrightarrow{\theta^{-1}} (\boldsymbol{L}[\boldsymbol{2}, \boldsymbol{2}] \ggg \boldsymbol{43}) \oplus \sum_{i=0}^{4} (L[1,i] \ggg 43) \oplus \sum_{i=0}^{4} (L[3,i] \ggg 42)$$

$$B \xrightarrow{\theta^{-1}} (L[1,1] \ggg 44) \oplus \sum_{i=0}^{4} (L[0,i] \ggg 44) \oplus \sum_{i=0}^{4} (\boldsymbol{L[2,i]} \ggg \mathbf{43})$$

Thấy rằng, khi $i = 2$, hai biểu thức trên có 1 *lane* chung là $L[2,2] \ggg 43$. Do vậy, trong trường hợp này, 64 bit thuộc *lane* $L[1,0]$ sẽ phụ thuộc vào $11 + 11 + 11 + 11 + 11 - 1 = 54$ bit đầu vào.

Tương tự, trong các trường hợp $(x, y) = (2, 0), (3, 0)$ và $(4, 0)$ thì mỗi 64 bit tương ứng thuộc mỗi *lane* $L[2,0], L[3,0]$ và $L[4,0]$ phụ thuộc vào 54 bit đầu vào. Như vậy, sẽ có $5 \times 64 = 320$ bit đầu ra phụ thuộc vào 54 bit đầu vào. Còn lại $1600 - 320 = 1280$ bit đầu ra phụ thuộc vào 55 bit đầu vào.■

Như vậy, với S-hộp đề xuất, hàm vòng tương ứng nhận được có số bit phụ thuộc lớn hơn nhiều so với trong Keccak. Điều này có được là do biểu thức đại số của các hàm bool trong S-hộp đề xuất là phức tạp hơn. Do vậy dạng biểu diễn phương trình đại số qua các vòng của hoán vị sử dụng S-hộp này sẽ có bậc đại số cao hơn trong Keccak.

B. Bậc đại số của hoán vị Keccak-f sử dụng S-hộp đề xuất

Áp dụng các kết quả nghiên cứu về bộ phân biệt tổng bằng 0 cho hoán vị Keccak-*f* trong [2], chúng ta có bảng ước lượng sau:

BẢNG 5. BẬC ĐẠI SỐ CHO SỐ VÒNG RÚT GỌN TRONG KECCAK-*F* VÀ KECCAK-*F* SỬ DỤNG S-HỘP ĐỀ XUẤT

r	Hoán vị gốc		Hoán vị sử dụng S-hộp đề xuất	
	$\deg(R^r)$	$\deg(invR^r)$	$\deg(R^r)$	$\deg(invR^r)$
1	2	3	4	4
2	4	9	16	16
3	8	27	64	64
4	16	81	256	256
5	32	243	1024	1024
6	64	729	1456	1456
7	128	1164	1564	1564
8	256	1382	1591	1591
9	512	1491	1598	1598
10	1024	1545	**1599**	**1599**
11	1408	1572	**1599**	**1599**
12	1536	1586	**1599**	**1599**
13	1578	1593	**1599**	**1599**
14	1592	1596	**1599**	**1599**
15	1597	1598	**1599**	**1599**
16	**1599**	**1599**	**1599**	**1599**

Từ Bảng 5 thấy rằng, phải đến vòng thứ 16 thì bậc đại số của dạng biểu diễn phương trình đại số đối với hoán vị trong Keccak mới đạt cực đại. Còn khi thay bằng S-hộp đề xuất sẽ là 10. Như vậy, khi bậc đại số của S-hộp cao hơn sẽ ảnh hưởng trực tiếp đến các ước lượng trong Bảng 5. Hơn nữa, điều này cũng được giải thích phần nào thông qua đánh giá số bit phụ thuộc ở Mệnh đề 2, 3. Do vậy, hoán vị Keccak-*f* với S-hộp đề xuất là có tính chất đại số tốt hơn của Keccak-*f* nguyên thủy. Với tính chất đại số như vậy, hàm băm Keccak sử dụng S-hộp đề xuất sẽ có khả năng kháng lại tốt hơn trước các tấn công phân biệt dựa trên tổng bằng 0 so với hàm băm Keccak nguyên thủy.

C. Tính tuyến tính hóa không đầy đủ của S-hộp đề xuất

Ở các mục trên, nhóm tác giả đã phân tích tính chất tuyến tính hóa không đầy đủ đối với S-hộp trong Keccak-*f*. Có nghĩa rằng, tồn tại các tập con mà ở đó một số phương trình biểu diễn S-hộp trong Keccak-*f* có dạng tuyến tính (ví dụ các tập trong Bảng 2). Với tính chất này, các tác giả trong [8]-[10] thực hiện việc lập hệ phương trình cho số vòng nhỏ của Keccak. Từ đó đưa ra tấn công. Áp dụng tương tự đối với S-hộp đề xuất trong bài báo thấy rằng, các tính chất như trong [8] là không còn đúng nữa. Do vậy, việc sử dụng các phương trình tuyến tính để mở rộng số vòng trong tấn công tìm va chạm theo các cách tiếp cận trong [8]-[10] là không áp dụng

được cho phiên bản hàm băm Keccak mà sử dụng S-hộp đề xuất này.

D. Sự ảnh hưởng đến cấu trúc Sponge của hàm băm Keccak

Các nhà thiết kế mật mã thông thường sẽ lựa chọn một cấu trúc tổng thể, sau đó có những đánh giá về cấu trúc này trước khi thiết kế cho các thành phần ở bên trong nó. Ví dụ như mã khối có các cấu trúc thông dụng: mạng SPN, mạng Feistel,... Hàm băm có: cấu trúc Merkle-Damgard, cấu trúc Sponge,... Trước khi đánh giá lên những cấu trúc này, chúng ta thường lý tưởng hóa thành phần bên trong nó như là các mã khối lý tưởng, hoán vị ngẫu nhiên, biến đổi hay hàm ngẫu nhiên,... Kết quả, chúng ta sẽ có được những tấn công tổng quát. Nói cách khác, tấn công tổng quát là những tấn công mà không khai thác bất kỳ một thuộc tính mật mã bên trong một nguyên thủy mật mã, mà chỉ sử dụng cấu trúc tổng thể trong thiết kế của nó.

Đối tượng mà chúng tôi muốn hướng ở phần này đến là cấu trúc Sponge trong thiết kế hàm băm Keccak. Các tấn công tổng quát lên nó có thể kể đến là: tìm va chạm bên trong, tìm chu kỳ đầu ra, tìm đường dẫn đến trạng thái trong, khôi phục trạng thái,... [14]. Ở những phân tích này, các tác giả đã đưa ra những ước lượng độ an toàn cụ thể cho hai trường hợp sử dụng biến đổi ngẫu nhiên và hoán vị ngẫu nhiên trong cấu trúc Sponge. Ở một khía cạnh khác, với các biến đổi ngẫu nhiên và hoán vị ngẫu nhiên, nhóm tác giả trong [14] đã có những đánh giá lợi thế phân biệt của cấu trúc Sponge với bộ tiên tri ngẫu nhiên (Theorem 7, Theorem 9 [14]). Có thể nói rằng, những phân tích trong các tài liệu nói trên đã không sử dụng một thuộc tính nào của hàm được sử dụng trong cấu trúc Sponge. Chính vì vậy, việc thay đổi và đề xuất S-hộp mới không làm thay đổi cấu trúc Sponge trong hàm băm Keccak. Nó không ảnh hưởng đến độ an toàn chứng minh được của cấu trúc Sponge trong hàm băm này.

VI. ĐÁNH GIÁ KHẢ NĂNG THỰC THI KHI SỬ DỤNG S-HỘP ĐỀ XUẤT TRONG HOÁN VỊ KECCAK-*F*

Nhóm tác giả đã thực hiện xây dựng chương trình cài đặt trên ngôn ngữ C cho thuật toán SpongeHash. Cài đặt ở đây không áp dụng một chỉ lệnh SIMD hay Assembler nào. Cài đặt và biên dịch sử dụng Visual Studio 12 trên một nhân máy Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz 2.40GHz, Windows 10, phiên bản x64 bit. Bảng thống kê dưới đây thể hiện tốc độ thực thi của các thuật toán, trong đó (AT) – ký hiệu cài đặt an toàn. Cài đặt an toàn được thực hiện theo kỹ thuật mặt nạ hai chia sẻ trong [16].

BẢNG 6. TỐC ĐỘ THỰC THI CỦA SPONGEHASH VÀ KECCAK

Thuật toán		**Tốc độ Mb/s**	
Tên	**Phiên bản (bit)**	**Cài đặt bình thường**	**Cài đặt an toàn**
Keccak	256	1101,93	781,25
	512	599,70	422,83
SpongeHash	256	925,93	702,99
	512	520,83	384,99
SHA-2	256	970	
	512	640	

BẢNG 7. SO SÁNH TỐC ĐỘ CÀI ĐẶT CỦA SPONGEHASH VÀ KECCAK

Hàm băm		**Keccak**			
	Phiên bản (bit)	512	256	512 (AT)	256 (AT)
SpongeHash	512	↓13,1%			
	256		↓15,9%		
	512 (AT)			↓10,0%	
	256 (AT)				↓8,9%

BẢNG 8. TỐC ĐỘ THỰC THI CÀI ĐẶT AN TOÀN SO VỚI CÀI ĐẶT THÔNG THƯỜNG CỦA SPONGEHASH VÀ KECCAK

Hàm băm		**Keccak**		**SpongeHash**	
	Phiên bản (bit)	512	256	512	256
Keccak	512 (AT)	↓29,4%			
	256 (AT)		↓29,1%		
SpongeHash	512 (AT)			↓26,1%	
	256 (AT)				↓24,1%

Kết quả thống kê thấy rằng, sử dụng S-hộp đề xuất cho tốc độ thực thi có thể so sánh được so với phiên bản nguyên thủy, mặt khác độ an toàn lại được cải thiện.

VII. KẾT LUẬN

Trong bài báo, nhóm tác giả đã khảo sát độ an toàn hàm băm Keccak dựa trên phân tích tính chất của S-hộp được sử dụng trong nó. Kết quả chỉ ra rằng, các tham số mật mã của S-hộp trong hoán vị Keccak-*f* đóng vai trò quan trọng ảnh hưởng lên độ an toàn của nó. Trên cơ sở các phân tích này, chúng tôi đề xuất lựa chọn một S-hộp có các tính chất mật mã tốt hơn, thay thế S-hộp này vào hoán vị Keccak-*f* và thực hiện phân tích sự ảnh hưởng của chúng như đối với hoán vị Keccak-f ban đầu. Kết quả phân tích chứng tỏ S-hộp đề xuất mang lại độ an toàn cao hơn. Với S-hộp đề xuất, bài báo cũng so sánh khả năng cài đặt so với S-hộp gốc (Bảng 4). Một điều cũng dễ hiểu rằng, với cấu trúc đại số phức tạp hơn thì S-hộp đề xuất sẽ có tính chất cài đặt phức tạp hơn so với S-hộp ban đầu. Tuy nhiên, phụ thuộc vào người sử dụng và từng bối cảnh cụ thể, với một tốc độ băm chấp nhận được, trong khi độ an toàn được nâng cao hơn chắc chắn đây là một lựa chọn không tồi trong bối cảnh phát triển của khoa học thám mật mã trong thám mã.

Bài báo cũng chưa đề cập đến độ an toàn của đề xuất mới trước thám mã vi sai. Tuy nhiên, S-hộp đề xuất có xác suất vi sai tốt hơn S-hộp nguyên thủy trong Keccak, nó sẽ đảm bảo được rằng hàm băm Keccak sử dụng S-hộp đề xuất có khả năng kháng lại thám mã vi sai và biến thể không kém hàm băm nguyên thủy. Nhóm tác giả sẽ tập trung phân tích những đánh giá theo hướng này ở những nghiên cứu tiếp theo.

Một vấn đề mở cũng đặt ra ở đây liên quan đến các tấn công của nhóm Boura và cộng sự [2]-[5], cụ thể với các phân tích của nhóm này mà số vòng của Keccak phiên bản hiện thời đã được tăng lên 24 so với 18 như ở phiên bản đầu tiên. Vì vậy, khi sử dụng S-hộp đề xuất với các tính chất đại số tốt hơn, liệu có thể giảm số vòng được không? Khi đó tốc độ có thể cân bằng được với phiên bản nguyên thủy.

TÀI LIỆU THAM KHẢO

[1] NIST, SHA-3 Stadard: Permutation-Based Hash And Extendable Output Functions. 8/2015.

[2] Boura, C. and A. Canteaut. Zero-sum distinguishers for iterated permutations and application to Keccak-f and Hamsi-256. in International Workshop on Selected Areas in Cryptography. 2010. Springer.

[3] Boura, C. and A. Canteaut. A zero-sum property for the Keccak-f permutation with 18 rounds. in 2010 IEEE International Symposium on Information Theory. 2010. IEEE.

[4] Boura, C., A. Canteaut, and C. De Canniere. Higher-order differential properties of Keccak and Luffa. in International Workshop on Fast Software Encryption. 2011. Springer.

[5] Boura, C. and A. Canteaut. On the algebraic degree of some SHA-3 candidates. in Proceedings of the Third SHA-3 Candidate Conference, Washington DC. 2012.

[6] Aumasson, J.-P. and W. Meier, Zero-sum distinguishers for reduced Keccak-f and for the core functions of Luffa and Hamsi. rump session of Cryptographic Hardware and Embedded Systems-CHES, 2009. 2009: p. 67.

[7] Duan, M. and X. Lai, Improved zero-sum distinguisher for full round Keccak-f permutation. Chinese Science Bulletin, 2012. 57(6): p. 694-697.

[8] Song, L., G. Liao, and J. Guo. Non-full sbox linearization: applications to collision attacks on round-reduced Keccak. in Annual International Cryptology Conference. 2017. Springer.

[9] Qiao, K., et al. New collision attacks on round-reduced Keccak. in Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2017. Springer.

[10] Dinur, I., O. Dunkelman, and A. Shamir. New attacks on Keccak-224 and Keccak-256. in International Workshop on Fast Software Encryption. 2012. Springer.

[11] Li, M. and L. Cheng. Distinguishing property for full round keccak-f permutation. in Conference on Complex, Intelligent, and Software Intensive Systems. 2017. Springer.

[12] Guo, J., et al., Practical collision attacks against round-reduced SHA-3. Journal of Cryptology, 2020. 33(1): p. 228-270.

[13] Rajasree, M.S. Cryptanalysis of Round-Reduced KECCAK Using Non-linear Structures. in International Conference on Cryptology in India. 2019. Springer.

[14] Bertoni, G., et al. Sponge functions. in ECRYPT hash workshop. 2007. Citeseer.

[15] Nguyễn Văn Long. "Phân tích các thành phần mật mã trong hoán vị Keccak-p". Nghiên cứu Khoa học và Công nghệ trong lĩnh vực An toàn thông tin, ISSN 2615-9570, No 08. Vol 02. 2018.

[16] Bertoni, G., et al., Keccak implementation overview. URL: http://keccak. neokeon. org/Keccak-implementation-3.2. pdf, 2012.

SƠ LƯỢC VỀ TÁC GIẢ

TS. Nguyễn Văn Long

Đơn vị công tác: Phân viện NCKHMM, Viện KH-CN mật mã, Ban Cơ yếu Chính phủ.

Email: longnv@bcy.gov.vn.

Quá trình đào tạo: Năm 2008 tốt nghiệp Học viện FSO – Liên bang Nga chuyên ngành "An toàn thông tin các hệ thống viễn thông". Năm 2015 bảo vệ thành công luận án tiến sĩ tại học viện FSO Liên bang Nga theo chuyên ngành "Các phương pháp bảo vệ thông tin".

Hướng nghiên cứu hiện nay: Nghiên cứu, thiết kế các thuật toán mã đối xứng an toàn, hiệu quả trong cài đặt.

ThS. Lê Duy Đức

Đơn vị công tác: Khoa Kỹ thuật cơ sở, Học viện Phòng không - Không quân.

Email: leduchnnt@gmail.com.

Quá trình đào tạo: Năm 2006 tốt nghiệp Học viện Kỹ thuật quân sự; Năm 2014 tốt nghiệp Thạc sĩ tại Học viện Kỹ thuật quân sự.

Hướng nghiên cứu hiện nay: vô tuyến điện tử.