Phân tích các thành phần mật mã trong hoán vị Keccak-p

Nguyễn Văn Long

***Tóm tắt*— Keccak là hàm băm đã chiến thắng trong cuộc thi SHA-3. Nghiên cứu này sẽ tập trung phân tích và chi tiết một số tính chất mật mã của các biến đổi thành phần cấu thành nên hoán vị Keccak-p trong hàm băm Keccak. Cụ thể sẽ đưa ra lập luận chi tiết cho số nhánh của biến đổi tuyến tính trong hàm vòng của hoán vị Keccak-p và xem xét sự phụ thuộc giữa các bit đầu vào và đầu ra trong hàm vòng này. Mặt khác cũng đưa ra một vài phân tích về khả năng cài đặt của Keccak dựa trên những biến đổi thành phần này.**

***Abstract*— Keccak is a winning hash function in the SHA-3 competition. This study will focus on analyzing and detailing some of the cryptographic properties of the constituent composition changes, permutating Keccak-p in the hash function Keccak. Specifically, a detailed argument will be given for the number of branches of linear transformation in the loop function of Keccak-p permutation and considering the dependency between input and output bits in this loop function. On the other hand, also gives some analysis of Keccak's installation ability based on these component changes.**

***Từ khóa*— Hàm băm Keccak; hoán vị Keccak; SHA-3.**

***Keywords*—Keccak hash function; Keccak hash function; SHA-3.**

I. GIỚI THIỆU

Hàm băm mật mã là một thành phần quan trọng trong mật mã hiện đại. Có hai nguyên lý thiết kế điển hình hiện nay cho các hàm băm là dựa trên cấu trúc lặp Merkle-Damgärd [1, 2] và cấu trúc Sponge [3]. Trong khi ở cấu trúc thứ nhất, các mã khối được sử dụng để thiết kế các hàm nén theo những cấu trúc nhất định, thì ở cấu trúc thứ 2 lại sử dụng các hoán vị lặp. Tuy nhiên, các hàm băm có được thiết kế theo nguyên lý nào đi nữa thì vẫn có thể thấy rằng nhân mật mã của chúng được xây dựng dựa trên nguyên lý lặp đi lặp lại các biến đổi tuyến tính và phi tuyến đơn giản (nguyên lý của Shannon). Theo đó, biến đổi phi tuyến cung cấp tính xáo trộn cho các bit được xử lý qua hàm vòng, còn biến đổi tuyến tính sẽ đảm đương nhiệm vụ khuếch tán rộng hơn tính xáo trộn này. Trong tài liệu [4] nói rằng: Việc sử dụng đơn lẻ hai tính chất này sẽ không mang lại hiệu quả trong các thiết kế mật mã. Chúng chỉ mang lại hiệu quả khi được kết hợp với nhau.

Keccak là hàm băm đã chiến thắng trong cuộc thi tuyển chọn hàm băm SHA-3 do NIST tổ chức. Nguyên lý thiết kế của nó cũng dựa trên nguyên tắc trên. Hàm vòng của nó có dạng [5]:

$$Round(A, i_r) = \iota(\chi\left(\pi\left(\rho\left(\theta(A)\right)\right)\right), i_r).$$

Trong đó, tầng tuyến tính của nó là kết hợp bởi một số thành phần tuyến tính như biến đổi theta (phép θ), biến đổi pi (phép π), biến đổi rho (phép ρ) và biến đổi iota (phép ι). Còn biến đổi phi tuyến được đảm bảo bởi biến đổi χ.

Trong [6], các tác giả đưa ra số nhánh của biến đổi tuyến tính θ bằng 4. Mặt khác, khi kết hợp các biến đổi tuyến tính và phi tuyến thì 1 bit đầu vào có khả năng ảnh hưởng tới 31 bit đầu ra và ngược lại. Tuy nhiên, những số liệu này không được các tác giả trình bày chi tiết trong [6].

Đóng góp của chúng tôi. Trên cơ sở phân tích biến đổi tuyến tính θ, chúng tôi chứng minh chi tiết cho đại lượng số nhánh của biến đổi này. Còn khi kết hợp với biến đổi phi tuyến, chúng tôi cũng giải thích chi tiết cho sự phụ thuộc của các biến bit vào và đầu ra trong hàm vòng của hoán vị Keccak-*p*. Ngoài ra, đối với mỗi biến đổi thành phần nói trên, chúng tôi đưa ra những

Bài báo được nhận ngày 1/12/2018. Bài báo được nhận xét bởi phản biện thứ nhất vào ngày 5/12/2018 và được chấp nhận đăng vào ngày 21/12/2018. Bài báo được nhận xét bởi phản biện thứ hai vào ngày 10/12/2018 và được chấp nhận đăng vào ngày 20/12/2018.

phân tích về khả năng cài đặt của chúng trên các môi trường phần mềm.

Trong phạm vi nghiên cứu của bài báo này chúng tôi sẽ chỉ tập trung phân tích cho hoán vị Keccak-p của hàm băm Keccak trong chuẩn SHA-3. Có nghĩa là thực hiện phân tích đối với tham số $w = 64$. Các trường hợp khác phụ thuộc vào giá trị của tham số này được thực hiện tương tự.

Bố cục phần còn lại bài báo gồm: Mục II sẽ trình bày về quy ước mảng trạng thái của hoán vị Keccak-p. Mô tả các biến đổi thành phần cùng với một vài phân tích về khả năng cài đặt của chúng sẽ được đưa ra ở Mục III. Trong Mục IV sẽ xem xét làm tường minh một số tính chất mật mã của các biến đổi thành phần này. Cuối cùng là Mục Kết luận.

II. QUY ƯỚC MẢNG TRẠNG THÁI

Trạng thái là một mảng các bit được liên tục cập nhập trong quá trình xử lý. Đối với một phép hoán vị Keccak-p, trạng thái được biểu diễn bằng một chuỗi hoặc một mảng ba chiều [5].

Trạng thái cho phép hoán vị Keccak-$p[b, n_r]$ bao gồm b bit và n_r vòng của hoán vị. Bản đặc tả thông số kỹ thuật trong bộ tiêu chuẩn SHA-3 bao gồm hai đại lượng khác liên quan đến b là $b/25$ và $\log_2(b/25)$, lần lượt ký hiệu là w và l, trong đó $w = 2^l, l \in \{0,1,2,3,4,5,6\}$.

Có thể biểu diễn trạng thái đầu vào và đầu ra của phép hoán vị là các chuỗi b bit và biểu diễn trạng thái đầu vào và đầu ra của các ánh xạ con là một mảng bit $5\times5\times w$. Nếu S là ký hiệu một chuỗi biểu diễn trạng thái, thì các bit của nó được đánh số từ 0 đến $b-1$, do đó:

$$S = S[0] \| S[1] \| ... \| S[b-2] \| S[b-1].$$

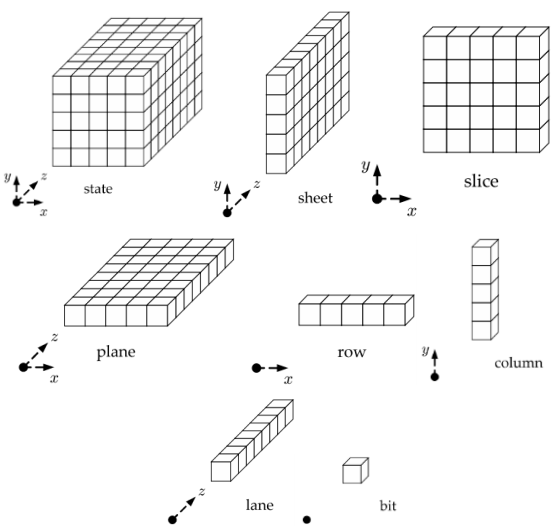
Nếu A là ký hiệu của một mảng bit $5\times5\times w$ biểu diễn trạng thái, thì chỉ số của nó là bộ ba số nguyên (x, y, z) sao cho $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$. Bit tương ứng với (x, y, z) được ký hiệu là $A[x, y, z]$. Mảng trạng thái biểu diễn cho trạng thái bằng một mảng ba chiều với chỉ số được xác định theo cách này.

A. Thành phần của mảng trạng thái

Đối với một phép hoán vị Keccak-p, một mảng $5 \times 5 \times w$ bit biểu diễn trạng thái. Các chỉ số thỏa mãn: $0 \le x \le 4$, $0 \le y \le 4$, và $0 \le z \le (w-1)$.

Mảng trạng thái cho một phép hoán vị Keccak-p và các mảng con ít chiều hơn (được minh họa trong Hình 1 dưới đây) đối với trường hợp $b = 200$, do đó $w = 8$. Các mảng con hai chiều được gọi là các *sheet, plane* và *slice,* và các mảng con một chiều được gọi là *column* (cột), *row* (hàng) và *lane* (làn), trong đó:

- *sheet*: là một mảng con gồm $b/5$ bit theo trục tọa độ x cố định.
- *plane*: là một mảng con gồm $b/5$ bit theo trục tọa độ y cố định.
- *slice*: là một mảng con gồm 25 bit theo trục tọa độ z cố định.
- *lane*: là một mảng con gồm $b/25$ bit theo các trục tọa độ x và y cố định.
- *row* (hàng): là một mảng con gồm 5 bit theo tọa độ y và z cố định.
- *column* (cột): là một mảng con gồm 5 bit với trục tọa độ x và z không đổi.



Hình 1. Thành phần của mảng trạng thái tổ chức theo nhiều chiều (w = 8)

B. Chuyển từ chuỗi sang mảng trạng thái

Cho S là ký hiệu của một chuỗi b bit biểu diễn cho trạng thái của phép hoán vị Keccak $-p[b, n_r]$. Mảng trạng thái tương ứng ký hiệu là A được định nghĩa như sau:

Đối với mọi bộ ba (x, y, z) sao cho $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$, ta có

$$A[x, y, z] = S[w(5y + x) + z].$$

C. Chuyển từ mảng trạng thái sang chuỗi

Cho A là ký hiệu của một mảng trạng thái. Biểu diễn chuỗi tương ứng ký hiệu là S có thể được cấu trúc từ các lane và plane của A như sau:

Đối với mỗi cặp số nguyên (i, j) sao cho $0 \le i < 5$ và $0 \le j < 5$, xác định chuỗi $lane[i, j]$:

$$lane[i, j] = A[i, j, 0] \| A[i, j, 1] \| A[i, j, 2] \| \ldots \| A[i, j, w-2] \| A[i, j, w-1].$$

Đối với mỗi số nguyên j, $0 \le j < 5$ định nghĩa $plane(j)$ bởi

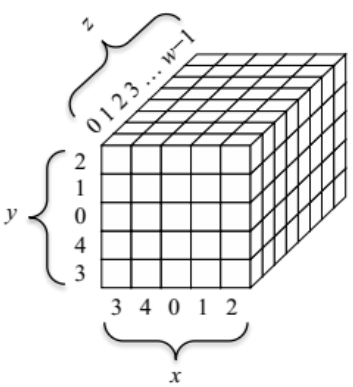
$$plane[j] = lane[0, j] \| lane[1, j] \| lane[2, j] \| \| lane[3, j] \| lane[4, j].$$

Do vậy,

$$S = plane[0] \| plane[1] \| plane[2] \| \| plane[3] \| plane[4].$$

D. Quy ước nhãn mảng trạng thái

Trong sơ đồ trạng thái đi kèm với các thông số kỹ thuật của ánh xạ bước, *lane* tương ứng với tọa độ $(x, y) = (0, 0)$ nằm ở trung tâm của *slice*.



Hình 2. Tọa độ theo các trục *x*, *y* và *z* cho sơ đồ ánh xạ bước

Nhãn đầy đủ của các tọa độ (x, y) và z được chỉ ra trong Hình 2.

E. Quy ước lấy tọa độ trên lane phụ thuộc vào giá trị dịch bit

Cho bit $A[x, y, z]$ thuộc $lane[x, y]$. Khi thực hiện phép dịch vòng sang phải đi a bit trên $lane[x, y]$, có nghĩa là thực hiện tính $lane[x, y] \ggg a$, thì tọa độ của bit $A'[x, y, z]$ đã cho là $A[x, y, (z + a) \bmod w]$. Có nghĩa rằng nếu bit $A[x, y, z]$ thuộc *slice* có tọa độ z, thì khi thực hiện $lane[x, y] \ggg a$, bit này sẽ thuộc *slice* có tọa độ $(z + a) \bmod w$.

III. CÁC BIẾN ĐỔI THÀNH PHẦN CỦA HÓA VỊ KECCAK-p

Hoán vị Keccak-p được xây dựng trên cơ sở hàm vòng $Round(A, i_r) = \iota(\chi(\pi(\rho(\theta(A)))), i_r)$. như đã được giới thiệu trong Mục Giới thiệu. Sau đây chúng tôi sẽ xem xét hoạt động của mỗi biến đổi thành phần này và một số phân tích của chúng tôi lên khả năng cài đặt của chúng.

A. Biến đổi theta θ

Thuật toán 1 sau đây mô tả hoạt động của phép biến đổi θ.

Thuật toán 1: $\theta(A)$

Input: Mảng trạng thái *A*

Output: Mảng trạng thái *A'*

Các bước biến đổi như sau:

1. Với tất cả các cặp (*x*, *z*) với $0 \le x < 5$ và $0 \le z < w$

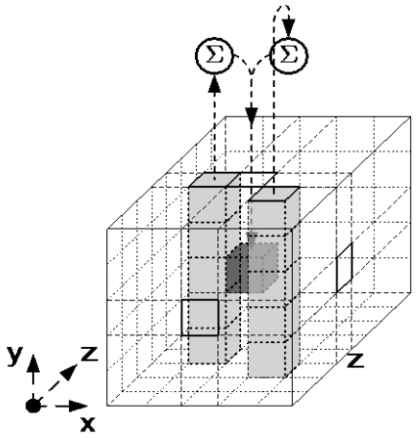
$$C[x, z] = A[x, 0, z] \oplus A[x, 1, z] \oplus A[x, 2, z] \oplus A[x, 3, z] \oplus A[x, 4, z]$$

2. Với tất cả các cặp (*x*, *z*) với $0 \le x < 5$ và $0 \le z < w$

$$D[x, z] = C[(x - 1) \bmod 5, z] \oplus C[(x + 1) \bmod 5, (z - 1) \bmod w]$$

3. Với tất cả các bộ ba (*x*, *y*, *z*) với $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$

$$A'[x, y, z] = A[x, y, z] \oplus D[x, z]$$



Hình 3. Minh họa phép biến đổi θ áp dụng cho từng bit

Như đã thấy trong thuật 1, biến đổi θ sử dụng các phép toán trên bit. Điều này là một lợi thế trong cài đặt cứng hóa. Tuy nhiên biến đổi này cũng có thể cài đặt hiệu quả bằng phần mềm trên môi trường các thanh ghi khác nhau tùy theo giá trị của tham số w trong bảng 1. Do đó, trong mục này chúng tôi sẽ phân tích khả năng cài đặt của biến đổi θ dựa theo quan điểm phần mềm.

Chúng ta thấy rằng, phép tính các giá trị C ở bước 1 của thuật toán 1 không phụ thuộc vào tọa độ y của bit trạng thái. Đây là phép toán cộng các bit ở một cột. Khi ghép tất cả các bit trong mỗi *lane* theo tọa độ z và cộng tất cả các *lane* trong một *sheet* ta sẽ nhận được các véc tơ C^* có độ dài w bit:

$$C^*[x] = \bigoplus_{y=0}^{4} lane(x,y) = lane(x,0) \oplus lane(x,1) \oplus lane(x,2) \oplus lane(x,3) \oplus lane(x,4),$$

với $|C^*[x]| = w$, $|lane(x,y)| = w$, $0 \le x, y \le 4$.

Từ biểu thức tính các bit $D[x,z]$

$$D[x,z] = C[(x-1)\ mod\ 5, z] \oplus C[(x+1)\ mod\ 5, (z-1)\ mod\ w].$$

Ta có thể tính véc tơ D^* như sau:

$$D^*[x] = C^*[(x+4)\ mod\ 5] \oplus (C^*[(x+1)\ mod\ 5] \lll 1),$$

trong đó $|D^*[x]| = w, 0 \le x \le 4$.

Do vậy 25 *lane* của trạng thái có thể được tính bởi:

$$lane(x,y) = lane(x,y) \oplus D^*[x],$$

trong đó, $0 \le x, y \le 4$.

Rõ ràng quá trình trên cho phép các thao tác xử lý qua phép θ trực tiếp trên cả *lane*. Ví dụ với trường hợp độ dài b = 800, hoặc 1600 bit (tương ứng với $w = b/25$ = 32 hoặc 64), ta có thể cài đặt phép θ trên các môi trường với thanh ghi 32 hoặc 64 bit.

B. Biến đổi π

Hình 4 và thuật toán 2 dưới đây đặc tả biến đổi π:

Thuật toán 2: $\pi(A)$

Input: mảng trạng thái *A*

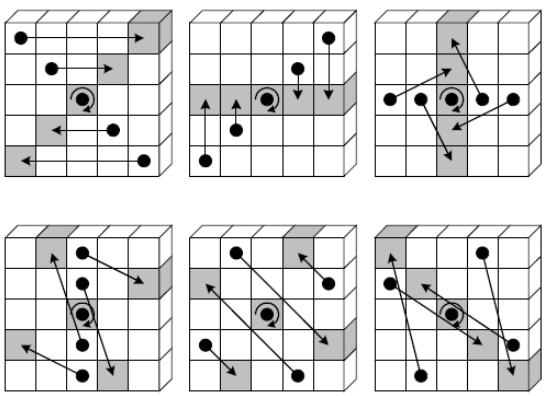
Output: mảng trạng thái *A'*

Các bước biến đổi:

1. Với tất cả các bộ 3 (*x*, *y*, *z*) thỏa mãn điều kiện $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$, ta đặt:

   A'[*x*, *y*, *z*]= A[(*x* + 3*y*) mod 5, *x*, *z*]

2. Return *A'*.



Hình 4. Minh họa phép biến đổi π áp dụng cho một *slice* đơn

Biến đổi π thực chất là phép hoán vị các bit trên một *slice* của khối trạng thái. Việc hoán vị này là giống nhau cho toàn bộ w *slice* trong mảng trạng thái. Như vậy có thể ghép tất cả các *slice* này và thực hiện hoán vị các *lane* trong khối trạng thái. Theo thuật toán 2, $lane[x,y]$ chính là giá trị $lane[(x+3y)mod\ 5, x]$. Do

vậy, việc cài đặt phần cứng hoặc phần mềm đối với biến đổi này có thể được thực hiện một cách đơn giản.

C. Biến đổi ρ

Thuật toán 3 dưới đây minh họa hoạt động của biến đổi ρ:

Thuật toán 3: $\rho(A)$

Input: Mảng trạng thái A

Output: Mảng trạng thái A'

Các bước biến đổi như sau:

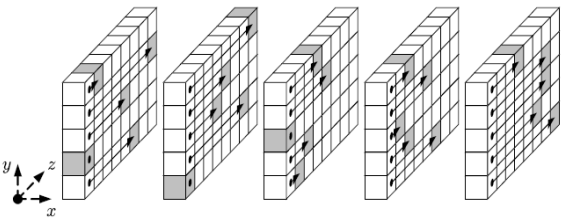
1. Với tất cả z với $0 \le z < w$, ta đặt $A'[0,0,z]=A[0,0,z]$
2. Đặt $(x, y) = (0, 1)$
3. Cho t chạy từ 0 tới 23:
    a. Với tất cả z thỏa mãn $0 \le z < w$ ta đặt $A'[x, y, z]= A[x, y, (z-(t+1)(t+2)/2) \bmod w]$.
    b. Đặt $[x, y]=[y, (2x + 3y) \bmod 5]$.
4. Return A'.

Tác động của phép biến đổi ρ là để xoay các bit của từng *lane* theo 1 chiều dài gọi là *offset*, với việc phụ thuộc vào các tọa độ cố định của x và y trong *lane*. Tương đương với từng bit trong *lane*, tọa độ z được sửa đổi bằng cách cộng modulo các *offset* theo kích thước *lane*.

Bảng 1. Các *offset* của ρ

	$x = 3$	$x = 4$	$x = 0$	$x = 1$	$x = 2$
$y = 2$	153	231	3	10	171
$y = 1$	55	276	36	300	6
$y = 0$	28	91	0	1	190
$y = 4$	120	78	210	66	253
$y = 3$	21	136	105	45	15

Minh họa phép biến đổi ρ với $w = 8$ được biểu diễn ở Hình 5. Các nhãn chuyển đổi cho các tọa độ cố định x, y ở hình 4 được biểu diễn tương tự như như trong hình 5, tương đương với các hàng và các cột trong bảng . Ví dụ *lane*[0,0] được miêu tả ở giữa của *sheet* giữa, còn *lane*[2,3] được miêu tả ở dưới cùng của *sheet* ngoài cùng bên phải.



Hình 5. Minh họa phép biến đổi ρ với b=200

Biến đổi ρ thực chất là phép dịch các bit một cách độc lập ở từng *sheet* theo từng *lane*. Giá trị dịch bit phụ thuộc vào tọa độ x và y. Do vậy có thể cài đặt đơn giản trong môi trường phần cứng hoặc trên phần mềm đối với phép biến đổi ρ.

D. Biến đổi χ

Thuật toán 4 dưới đây minh họa hoạt động của biến đổi χ:

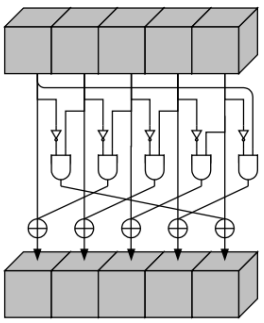
Thuật toán 4: $\chi(A)$

Input: Mảng trạng thái A

Output: Mảng trạng thái A'

Những bước biến đổi:

1. Với tất cả những bộ 3 (x, y, z) thỏa mãn những điều kiện $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$ tính $A'[x, y, z]= A[x, y, z] \oplus ((A[(x+1) \bmod 5, y, z] \oplus 1) \cdot A[(x+2) \bmod 5, y, z])$.
2. Return A'



Hình 6. Minh họa phép biến đổi χ
áp dụng cho từng *row* riêng lẻ

Trên thực tế, các nhà thiết kế lựa chọn χ có biểu thức đại số đơn giản để thuận tiện cho các cài đặt cứng hóa. Tuy nhiên, có thể ghép các bit trên cùng 1 *lane* để thực hiện. Theo đó:

$$lane'[x,y] = lane[x,y] \oplus (lane[(x+1) \bmod 5, y] \oplus (1)^w) \cdot lane[(x+2) \bmod 5, y],$$

trong đó $(1)^w = \underbrace{1||\dots||1}_{w \text{ lần}}$. Với biểu diễn này, biến đổi χ có thể thực hiện trên *lane* và rất thuận tiện trong cài đặt phần mềm.

E. Biến đổi ι

Biến đổi ι chỉ tác động lên *lane* gốc, nghĩa là *lane* có tọa độ $x = y = 0$. Bản chất của nó là cộng vào *lane* gốc các hằng số phụ thuộc vào chỉ số vòng của hoán vị Keccak-*p*. Do vậy, biến đổi này có thể dễ dàng cài đặt trong phần cứng và phần mềm.

Phép ánh xạ ι được tham số hóa bởi chỉ số vòng i_r, những giá trị này được xác định trong bước 2 của thuật toán tính hoán vị Keccak–*p*[*b*, n_r] ở phần sau. Trong phạm vi phép biến đổi ι ở thuật toán 6 bên dưới, tham số này xác định $l+1$ bit của giá trị *lane* được gọi là hằng số vòng, và được định nghĩa là RC. Mỗi bit của $l+1$ bit được tạo ra bởi một hàm mà hàm này dựa trên một thanh ghi dịch tuyến tính có phản hồi. Hàm này ký hiệu là *rc* và được định nghĩa ở thuật toán 5.

Thuật toán 5: rc(t)

Input: số nguyên *t*

Output: bit *rc*(*t*)

Các bước của thuật toán

1. Nếu *t* mod 255 =0 , return 1
2. Đặt *R* = 10000000
3. Cho *i* chạy từ 1 tới *t* mod 255, đặt:
   - 3.1. *R*= 0 ‖ *R*
   - 3.2. *R*[0]= *R*[0] ⊕ *R*[8]
   - 3.3. *R*[4]= *R*[4] ⊕ *R*[8]
   - 3.4. *R*[5]= R[5] ⊕ *R*[8]
   - 3.5. *R*[6]= *R*[6] ⊕ *R*[8]
   - 3.6. *R* = Trunc$_8$[*R*]
4. Return *R*[0]

Thuật toán 6: ι(A,i_r)

Input: Mảng trạng thái *A*

Chỉ số vòng i_r

Output: Mảng trạng thái *A'*

Các bước của thuật toán:

1. Với tất cả các bộ 3 (*x, y, z*) thỏa mãn điều kiện $0 \le x < 5, 0 \le y < 5$ và $0 \le z < w$, ta đặt:

   A'[*x, y, z*] = *A*[*x, y, z*]
2. Đặt *RC* = 0^w
3. Cho *j* chạy từ 0 tới ℓ , ta đặt

   $RC[2^j - 1] = rc([j+7i_r])$
4. Với tất cả *z* thỏa mãn $0 \le z < w$, ta đặt

   $A'[0,0,z] = A'[0,0,z] \oplus RC[z]$
5. Return *A'*.

Tác động của phép biến đổi ι là để biến đổi một vài bit của l*ane*[0, 0] phụ thuộc vào chỉ số vòng i_r. Còn lại 24 *lane* khác đều không bị ảnh hưởng bởi phép biến đổi ι.

Ánh xạ ι bao gồm việc thêm các hằng số vòng và hướng tới phá vỡ tính đối xứng. Các bit của các hằng số vòng là khác nhau từ vòng này đến vòng kia và được lấy là đầu ra của LFSR có độ dài lớn nhất. Các hằng số này chỉ được thêm trong một *lane* của trạng thái. Do đó sự phá vỡ này sẽ được lan truyền thông qua θ và χ đối với tất cả các *lane* của trạng thái sau một đơn.

IV. TÍNH CHẤT MẬT MÃ CÁC BIẾN ĐỔI THÀNH PHẦN TRONG HOÁN VỊ KECCAK-p

Trong mục này chúng tôi xem xét hai tính chất mật mã, gồm số nhánh của biến đổi tuyến tính, và sự ảnh hưởng của các bit đầu vào (hoặc đầu ra) lên các bit đầu ra (hoặc đầu vào) của hàm vòng.

Đối với biến đổi tuyến tính, chúng ta chỉ quan tâm đến sự khuếch tán θ, bởi vì các biến đổi π và ρ không làm thay đổi số lượng bit tích cực mà chỉ thay đổi vị trí của các bit này trong mảng trạng thái. Còn biến đổi ι thực chất là phép cộng với hằng số đối với các bit trong *lane*[0, 0]. Do vậy, nó không tác động lên số lượng bit tích cực trong hàm vòng.

Đối với việc xem xét sự hành hưởng của các bit đầu vào (hoặc đầu ra) lên các bit đầu ra (hoặc đầu vào) của hàm vòng, chúng tôi sẽ thực hiện biểu diễn một bit đầu ra phụ thuộc vào các bit đầu vào.

A. Số nhánh của biến đổi θ

Ánh xạ θ là tuyến tính và đảm nhiệm vai trò khuếch trong hoán vị Keccak-p. Tác động của nó có thể được mô tả như sau: Cộng XOR mỗi bit $a[x][y][z]$ trong trạng thái với giá trị chẵn/lẻ (tổng XOR các bit) của hai cột $a[x-1][\cdot][z]$ và $a[x+1][\cdot][z-1]$. Nếu không có biến đổi θ, hoán vị Keccak-f sẽ không có tính khuếch tán. Đối với các trạng thái mà ở đó tổng bit trong tất cả các cột của nó là số chẵn, thì θ là đồng nhất. Như vậy, những trạng thái mà có trọng số Hamming nhỏ nhất là bằng 2, có nghĩa là có một cột có 2 bit tích cực, các cột khác đều chứa các bit bằng 0. Khi đó số nhánh của biến đổi θ chỉ là 4. Trong [6], các tác giả lập luận và đưa ra số nhánh như vậy. Tuy nhiên, để khẳng định điều này ta cần xem xét để chứng tỏ trong những trường hợp khác, số nhánh không thể nhỏ hơn 4. Mệnh đề dưới đây sẽ chi tiết hơn về vấn đề này.

Mệnh đề 1: *Số nhánh của biến đổi θ trong hoán vị Keccak-p bằng* 4.

Chứng minh: Gọi A là mảng trạng thái đầu vào, còn A' là mảng trạng thái đầu ra qua biến đổi θ. Khi đó, số nhánh theo bit của biến đổi θ được xác định bởi công thức

$$B_\theta = wt(A) + wt(A') = wt(A) + wt(\theta(A)).$$

Xét các trường hợp sau:

Trường hợp 1: $wt(A) = 1$. Có nghĩa rằng trạng thái A chỉ có một bit có giá trị bằng 1. Giả sử bit đó có tọa độ là (x_i, y_j, z_k): $A[x_i, y_j, z_k] = 1$.

Khi đó,

$$\begin{cases} C[x = x_i, z = z_k] = 1 \\ C[x \neq x_i, z \neq z_k] = 0 \end{cases}$$

Từ biểu thức của $D[x,z]$, có

$$\begin{cases} D[x = (x_i + 1) mod\ 5\ , z = z_k] = \\ C[x_i, z_k] \oplus C[(x+2) mod\ 5, (z_k - 1) mod\ w] \\ = 1 \oplus 0 = 1 \\ D[x = (x_i - 1) mod\ 5\ , z = (z_k + 1) mod\ w] = \\ C[(x-2) mod\ 5, (z_k + 1) mod\ w] \\ \oplus C[x_i, z_k] = 0 \oplus 1 = 1 \end{cases}$$

Còn trong các trường hợp còn lại của tọa độ x và z, thì $D[x,z] = 0$. Do vậy, các bit của trạng thái A' bằng 1, gồm:

- $A'[x_i, y_j, z_k] = A[x_i, y_j, z_k] \oplus D[x_i, z_k] = 1 \oplus 0 = 1$,
- $A'[(x_i + 1)\ mod\ 5, y, z_k] = A[(x_i + 1)\ mod\ 5, y, z_k] \oplus D[(x_i + 1)\ mod\ 5, z_k] = 0 \oplus 1 = 1$, với $0 \leq y < 5$, và
- $A'[(x_i - 1) mod\ 5, y, (z_k + 1) mod\ w] = A[(x_i - 1) mod\ 5, y, (z_k + 1)\ mod\ w] \oplus D[(x_i - 1) mod\ 5, (z_k + 1) mod\ w] = 0 \oplus 1 = 1$, với $0 \leq y < 5$.

Từ đây có $wt(A') = 1 + 2 \times |y| = 1 + 2 \times 5 = 11$ và $B_\theta = wt(A) + wt(A') = 1 + 11 = 12 > 4$.

Trường hợp 2: $wt(A) = 2$. Xét các khả năng sau:

- Nếu hai bit có giá trị bằng 1 trong trạng thái A cùng nằm trên hai cột. Khi đó tất cả các giá trị $C[x,z]$ đều bằng 0, với $0 \leq x < 5, 0 \leq z < w$. Điều này dẫn tới tất cả các giá trị $D[x,z]$ cũng đều bằng 0 với mọi (x,z).Vì

$$A'[x,y,z] = A[x,y,z] \oplus D[x,z] = A[x,y,z],$$

nên $wt(A') = wt(A) = 2$.

Do vậy $B_\theta = 2 + 2 = 4$.

- Nếu hai bit có giá trị bằng 1 trong trạng thái A nằm ở 2 cột khác nhau. Khi đó lập luận tương tự như trong trường hợp 1, có $B_\theta > 4$.

***Trường hợp* 3**: $wt(A) = 3$.

- Nếu ba bit có giá trị bằng 1 trong A đều thuộc một cột. Khi đó ta sẽ tính được $wt(A') = 11$ tương tự như trong trường hợp 1. Do vậy, $B_\theta = 3 + 11 = 14 > 4$.

- Nếu ba bit có giá trị bằng 1 trong A không thuộc cùng một cột. Khi đó hoặc chúng sẽ thuộc ba cột khác nhau, hoặc thuộc hai cột khác nhau. Lập luận tương tự ta cũng sẽ có $B_\theta > 4$.

Ở các trường hợp còn lại, khi mà $wt(A) \geq 4$, ta sẽ luôn luôn có $B_\theta = wt(A) + wt(A') > 4$. Do vậy số nhánh của biến đổi tuyến tính θ là bằng 4.

B. Sự phụ thuộc các bit đầu vào và đầu ra của hàm vòng trong hoán vị Keccak-p

Việc xem xét sự lan truyền giữa các bit đầu vào/ra, hay nói cách khác sự phụ thuộc lẫn nhau của các bit đầu vào và đầu ra là một tính chất quan trọng trong thiết kế các nguyên thủy mật mã. Trong [6], các tác giả nói rằng, khi kết hợp tầng tuyến tính với biến đổi χ trong hàm vòng của hoán vị Keccak-p, thì mỗi bit tại đầu vào của hàm vòng có khả năng ảnh hưởng tới 31 bit tại đầu ra và mỗi bit tại đầu ra của hàm vòng phụ thuộc vào 31 bit đầu vào của nó. Tuy nhiên, khi xây dựng chương trình thực hiện hàm vòng của hoán vị Keccak-p, chúng tôi đã tìm ra rất nhiều trạng thái, mà khi thay đổi 1 bit đầu vào hoặc đầu ra sẽ làm thay đổi 32 hoặc 33 bit đầu ra hoặc đầu vào tương ứng. Mặt khác khi biểu diễn sự phụ thuộc các bit đầu ra bởi các bit đầu vào chúng tôi cũng nhận được các đánh giá tương tự. Mệnh đề sau đây sẽ chi tiết vấn đề này. Ở đây chúng tôi chỉ chứng minh các kết quả cho trường hợp hoán vị Keccak-p trong chuẩn hàm băm SHA3, có nghĩa rằng lựa chọn giá trị w = 64 và $b = 1600$.

Mệnh đề 2. *Đối với biến đổi vòng trong hoán vị Keccak-p của hàm băm SHA-*3 *có:*

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

Chứng minh. Trong chứng minh này chúng tôi sẽ xem xét sự ảnh hưởng của các bit đầu vào lên 1 bit đầu ra bằng cách biểu diễn biểu thức mỗi *lane* đầu ra qua các *lane* đầu vào. Từ đó cho phép nhận được các đánh giá về sự phụ thuộc của các bit đầu ra vào các bit đầu vào. Xét *lane* có tọa độ (x, y) bất kỳ, $0 \leq x, y \leq 4$. Và thực hiện biểu diễn nó qua các ánh xạ $\chi^{-1}, \pi^{-1}, \rho^{-1}$ và θ^{-1} trong biến đổi vòng của hoán vị Keccak-p.

$$
\begin{aligned}
&lane[x,y] \xrightarrow{\chi^{-1}} lane[x,y] \\
&\oplus\, lane[(x+2) mod\ 5, y] \oplus \\
&\oplus\, lane[(x+1) mod\ 5, y] \\
&\cdot\, lane[(x+2) mod\ 5, y] \\
&\xrightarrow{\pi^{-1}} lane[(x+3y) mod\ 5, x] \\
&\oplus\, lane[(x+3y+2) mod\ 5, (x+2) mod\ 5] \\
&\oplus \\
&\oplus\, lane[(x+3y+1) mod\ 5, (x+1) mod\ 5] \cdot \\
&\cdot\, lane[(x+3y+2) mod\ 5, (x+2) mod\ 5] \\
&\xrightarrow{\rho^{-1}} \underbrace{(lane[(x+3y) mod\ 5, x] \ggg a)}_{A} \\
&\oplus \underbrace{(lane[(x+3y+2) mod\ 5, (x+2) mod\ 5] \ggg b)}_{B} \\
&\oplus \underbrace{(lane[(x+3y+1) mod\ 5, (x+1) mod\ 5] \ggg c)}_{C} \cdot \\
&\cdot\, (lane[(x+3y+2) mod\ 5, (x+2) mod\ 5] \\
&\qquad \ggg b) \\
&= A \oplus B \oplus C \cdot B,
\end{aligned}
$$

trong đó a, b, c là các giá trị *offset* được quy định bởi biến đổi ρ. Trong trường hợp $w = 64$ có $a \neq b \neq c$.

Qua biến đổi θ^{-1}, ta có,

Đối với biểu thức A:

$$
\begin{aligned}
A &\xrightarrow{\theta^{-1}} (lane[(x+3y) mod\ 5, x] \ggg a) \\
&\oplus (D^*[(x+3y) mod\ 5] \ggg a) = \\
&= (lane[(x+3y) mod\ 5, x] \ggg a)
\end{aligned}
$$

$$\oplus\ (C^*[(x+3y+4)mod\ 5 \ggg a) \oplus$$
$$\oplus\ (C^*[(x+3y+1)mod\ 5 \lll 1) \ggg a) =$$
$$= (lane[(x+3y)mod\ 5, x] \ggg a)$$
$$\oplus\ (C^*[(x+3y+4)mod\ 5 \ggg a) \oplus$$
$$\oplus\ \big(C^*[(x+3y+1)mod\ 5) \ggg (a-1)\big) =$$
$$= (lane[(x+3y)mod\ 5, x] \ggg a) \oplus$$
$$\sum_{i=0}^{4} (lane[(x+3y+4)mod\ 5, i] \ggg a) \oplus$$
$$\sum_{i=0}^{4} \big((lane[(x+3y+1)mod\ 5, i]) \ggg (a-1)\big).$$

Đối với biểu thức *B*:

$$B \xrightarrow{\theta^{-1}}$$
$$(lane[(x+3y+2)mod\ 5, (x+2)mod\ 5] \ggg b) \oplus$$
$$\sum_{i=0}^{4} (lane[(x+3y+1)mod\ 5, i] \ggg b) \oplus$$
$$\sum_{i=0}^{4} \big((lane[(x+3y+3)mod\ 5, i]) \ggg (b-1)\big).$$

Đối với biểu thức *C*:

$$C \xrightarrow{\theta^{-1}}$$
$$(lane[(x+3y+1)mod\ 5, (x+1)mod\ 5] \ggg c) \oplus$$
$$\sum_{i=0}^{4} (lane[(x+3y)mod\ 5, i] \ggg c) \oplus$$
$$\sum_{i=0}^{4} \big((lane[(x+3y+2)mod\ 5, i]) \ggg (c-1)\big).$$

Ta thấy rằng phép dịch trong mỗi *lane* ở mỗi biểu thức *A*, *B* hoặc *C* cho ta tọa độ *z* được dịch đi, hay nói cách khác, phép dịch thể hiện xem các tọa độ x, y của trạng thái $A[x, y, z]$ nằm ở *slice* nào.

Các giá trị dịch trong mỗi *lane* xác định bởi:

$$a = offset[(x+3y)mod\ 5, x]$$
$$b = offset[(x+3y+2)mod\ 5, (x+2)mod\ 5]$$
$$c = offset[(x+3y+1)mod\ 5, (x+1)mod\ 5]$$

Từ biểu thức của *A*, *B* hoặc *C* thấy rằng mỗi biểu thức tương ứng phụ thuộc vào 11 *lane*. Mặt khác, theo bảng *offset* của biến đổi ρ có các trường hợp sau:

- *Trường hợp 1.* $(x, y) = (0,0)$: Trong trường hợp này có $b = 43$ và $c = 44$ là thỏa mãn điều kiện $b = c - 1$.
- *Trường hợp 2.* $(x, y) = (1,0)$: Trong trường hợp này có $a = 44$ và $b = 43$ là thỏa mãn điều kiện $a - 1 = b$.
- *Trường hợp 3.* $(x, y) \neq (0,0)$ và $(x, y) \neq (1,0)$: Trong trường hợp này $a \neq c$, $a - 1 \neq b$ và $b \neq c - 1$.

Xét các trường hợp trên:

***Trường hợp* 1:** Với $(x, y) = (0,0)$, có

$$B = (lane[(0+3\cdot 0+2)mod\ 5, (0+2)mod\ 5] \ggg 43) \oplus$$
$$\sum_{i=0}^{4} (lane[(0+3\cdot 0+1)mod\ 5, i] \ggg 43) \oplus$$
$$\sum_{i=0}^{4} \big((lane[(0+3\cdot 0+3)mod\ 5, i]) \ggg (43-1)\big) =$$
$$(\boldsymbol{lane}[\mathbf{2,2}] \ggg \mathbf{43}) \oplus \sum_{i=0}^{4} (lane[1, i] \ggg 43) \oplus \sum_{i=0}^{4} (lane[3, i] \ggg 42).$$

và

$$C = (lane[(0+3\cdot 0+1)mod\ 5, (0+1)mod\ 5] \ggg 44) \oplus$$
$$\sum_{i=0}^{4} (lane[(0+3\cdot 0)mod\ 5, i] \ggg 44) \oplus$$
$$\sum_{i=0}^{4} \big((lane[(0+3\cdot 0+2)mod\ 5, i]) \ggg (44-1)\big) =$$
$$(lane[1,1] \ggg 44) \oplus \sum_{i=0}^{4} (lane[0, i] \ggg 44) \oplus \sum_{i=0}^{4} (lane[2, i] \ggg 43) =$$

$$(lane[1,1] \ggg 44) \oplus \sum_{i=0}^{4} (lane[0,i] \ggg 44) \oplus (lane[2,0] \ggg 43) \oplus \oplus (lane[2,0=1] \ggg 43) \oplus (\boldsymbol{lane}[\mathbf{2},\mathbf{2}] \ggg \mathbf{43}) \oplus (lane[2,3] \ggg 43) \oplus (lane[2,4] \ggg 43).$$

Như vậy, biểu thức của B và C có 1 *lane* chung (màu đậm trong biểu thức ở trên). Do vậy, biểu thức $A \oplus B \oplus C \cdot B$ sẽ có 11 + 11 + 10 = 32 *lane* tham gia. Kết quả là sẽ có 64 bit ở đầu ra có tọa độ $(x, y, z) = (0,0, z), 0 \le z \le 63$ phụ thuộc vào 32 bit đầu vào (các bit này thuộc $lane[0,0]$).

***Trường hợp* 2:** Với $(x, y) = (1,0)$, ta thực hiện phân tích tương tự như trong trường hợp 1. Khi đó, trong biểu thức của *A* và *B* sẽ có 1 *lane* chung là $\boldsymbol{lane}[\mathbf{2},\mathbf{1}] \ggg \mathbf{43}$. Do vậy, cũng sẽ có 64 bit đầu ra trong $lane[1,0]$ phụ thuộc vào 32 bit đầu vào.

***Trường hợp* 3**. $(x, y) \neq (0,0)$ và $(x, y) \neq (1,0)$. Khi đó ta có các kết quả sau:

- Các tọa độ *z* trong $(lane[(x + 3y) mod\ 5, x] \ggg a)$ của *A* và trong mỗi *lane* trong tổng $\sum_{i=0}^{4} (lane[(x + 3y) mod\ 5, i] \ggg c)$ của *C* là nằm trên các *slice* khác nhau (ở đây xét $i = x$), vì phép dịch cùng 1 *lane* đi hai vị trí khác nhau.

- Các tọa độ *z* trong mỗi *lane* trong tổng $\sum_{i=0}^{4}\big((lane[(x + 3y + 1) mod\ 5, i]) \ggg (a - 1)\big)$ của *A* và trong mỗi *lane* tương ứng trong tổng $\sum_{i=0}^{4} (lane[(x + 3y + 1) mod\ 5, i] \ggg b)$ của *B* cũng nằm trên các *slice* khác nhau, vì phép dịch cùng 1 *lane* đi hai vị trí khác nhau.

- Các tọa độ *z* trong mỗi *lane* $(lane[(x + 3y + 2) mod\ 5, (x + 2) mod\ 5] \ggg b)$ của *B* và trong mỗi *lane* trong tổng $\sum_{i=0}^{4}\big((lane[(x + 3y + 2) mod\ 5, i]) \ggg (c - 1)\big)$ của *C* cũng nằm trên các *slice* khác nhau (với $x \neq 0$ và $y \neq 0$) (ở đây xét $i = (x + 2) mod\ 5$), vì phép dịch cùng 1 *lane* đi hai vị trí khác nhau.

Hay nói cách khác, biểu thức *A*, *B* và *C* không chứa các *lane* chung. Từ đây ta có kết quả rằng $1600 - 2 \times 64 = 1472$ bit đầu ra phụ thuộc vào 33 biến đầu vào. Các bit này nằm trên 23 *lane* có tọa độ $(x, y) \neq (0,0)$ và $(x, y) \neq (1,0)$.

V. KẾT LUẬN

Trong bài báo này, chúng tôi tập trung nghiên cứu phân tích một số tính chất mật mã của các biến đổi thành phần trong hoán vị Keccak-*p* của chuẩn hàm băm SHA-3. Đối với biến đổi thành phần ban đầu chúng tôi mô tả hoạt động, sau đó đưa ra nhận xét về khả năng cài đặt hiệu quả trên phần mềm của chúng. Riêng với đại lượng số nhánh của biến đổi θ chúng tôi đã làm tường minh kết quả về số nhánh bằng 4 của nó. Ngoài ra, khi kết hợp tầng tuyến tính với biến đổi phi tuyến χ, chúng tôi cũng chính xác hóa lại về số lượng các bit đầu ra phụ thuộc vào các bit đầu vào đã được đưa ra trong [6]. Cụ thể đã tìm ra 128 bit đầu ra ở $lane[0,0]$ và $lane[1,0]$ phụ thuộc vào 32 bit đầu vào, 1472 bit ở những *lane* còn lại phụ thuộc vào 33 bit đầu vào của hàm vòng.

Từ đây, có thể đưa ra một hướng nghiên cứu để tăng số bit phụ thuộc giữa đầu vào và đầu ra của hàm vòng, đó là sử dụng các S-hộp có bậc đại số lớn hơn so với S-hộp trong ánh xạ χ của Keccak-*p*. Tuy nhiên điều này có thể ảnh hưởng đến khả năng cài đặt tổng thể của thuật toán, do vậy cần phải tiếp tục nghiên cứu khi xem xét các đề xuất cụ thể của S-hộp 5 bit.

TÀI LIỆU THAM KHẢO

1. Damgård, I.B. "*A design principle for hash functions*. in *Advances in Cryptology—CRYPTO'89 Proceedings"*. Springer, 1989.
2. Merkle, R.C. "One way hash functions and DES. in Advances in Cryptology"—CRYPTO'89 Proceedings, Springer, 1989.
3. Guido, B., et al., "Cryptographic sponge functions". 2011.
4. Зензин, О. and М. "Иванов, Стардарт криптографической защиты-AES". Конечные поля, КУДРИЦ-ОБРАЗ М, 2002.
5. NIST, SHA-3 Stadard: "Permutation-Based Hash And Extendable Output Functions". 8/2015.
6. Bertoni, G., et al., "The Keccak reference, version 3.0", URL:http://keccak.noekeon.org/Keccakreference-3.0.pdf. Citations in this document. 4, 2011.

SƠ LƯỢC VỀ TÁC GIẢ

TS. Nguyễn Văn Long

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

Email: longnv@bcy.gov.vn

Quá trình đào tạo: Nhận bằng Kỹ sư chuyên nghành An toàn thông tin các hệ thống viễn thông tại Học viện FSO - Liên bang Nga năm 2008. 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 năm 2015.

Hướng nghiên cứu hiện nay: Nghiên cứu, cài đặt, thiết kế các thuật toán mã khối.