Về một phương pháp xây dựng hàm băm không khóa trên vành đa thức chẵn

A Method of Buiding Unkeyed Hash Function on Even Polynomial Ring

Nguyễn Toàn Thắng, Ngô Đức Thiện*[*], Nguyễn Trung Hiếu

[1]*Học viện Công nghệ Bưu chính Viễn thông – Số 122, Hoàng Quốc Việt, Cầu Giấy, Hà Nội*

Tóm tắt

Các hàm băm mật mã được sử dụng trong nhiều ứng dụng của mật mã hiện đại như: chứng thực, kiểm tra tính toàn vẹn của thông điệp, chữ ký số… Các hàm băm thường được xây dựng theo các thuật toán riêng biệt, hoặc dựa trên các hệ mật mã khối kết hợp với các sơ đồ băm cố định. Bài báo này đề xuất một hệ mật mã khóa đối xứng 256 bit vào /ra. Sơ đồ mã hóa thực hiện theo mạng Feistel bốn nhánh không cân bằng, hàm mã hóa và các khóa con của các vòng mã hóa dựa trên các cấp số nhân cyclic của vành đa thức chẵn $\mathbb{Z}_2[x]/x^n+1$ *với* $n=2^k$. *Trên cơ sở hệ mật này, bài báo đề xuất áp dụng vào xây dựng một hàm băm không khóa 512 bit. Cuối cùng là một vài đánh giá về độ khuếch tán của hệ mật và hàm băm đề xuất.*

Từ khóa: Vành đa thức, cấp số nhân cyclic, mật mã khối, mạng Feistel, hàm băm.

Abstract

Cryptographic hash functions are used in many applications of modern cryptography, such as authentication, integrity checking of data messages, digital signatures... This paper proposes a symmetry-key block crypto system 256 bits input /output. The encryption scheme is unbalanced four branches Feistel network; the encryption function and subkeys for the encryption loops are based on cyclic geometric progressions over even polynomial ring $\mathbb{Z}_2[x]/x^n+1$ *with* $n=2^k$. *In the next part of the paper, the article recommended applies this crypto system to a hash function with 512 bits output length. Finally, some diffusion calculations of the proposed crypto system and hash function are also presented.*

Keywords: Polynomial ring, Cyclic geometric progressionc, Block symmetry crypto system, Feistel network, hash function.

1. Giới thiệu

Hàm băm mật mã đóng một vai trò quan trọng nhằm thực hiện các yêu cầu cơ bản của mật mã học hiện đại như chứng thực; kiểm tra tính nguyên vẹn của thông điệp; chữ ký số, kiểm tra mật khẩu… Các sơ đồ hàm băm thường được xây dựng theo một trong hai cách: (1) theo các thuật toán riêng biệt (các hàm băm họ MD, SHA,..) (2) hoặc dựa trên các sơ đồ cố định kết hợp với một hệ mật mã khối.

Căn cứ theo mục đích sử dụng, các hàm băm được chia thành hai kiểu: hàm băm có khóa (MAC - Message Authentication Code) và hàm băm không khóa (MDC - Modification Detection Code). Với mục đích nghiên cứu và phát triển các hàm băm không khóa, bài báo này đề xuất một sơ đồ hàm băm mở rộng (MDC-2) thực hiện theo sơ đồ Miyaguchi - Preneel, trong đó các khối mật mã là một hệ mật 256 bit được xây dựng dựa trên các cấp số nhân cyclic (CGP- Cyclic Geometric Progression) của vành đa thức chẵn có dạng $\mathbb{Z}_2[x]/x^n+1$, với $n=2^6=64$.

Nội dung bài báo được chia làm bốn phần. Phần 2 trình bày đề xuất một hệ mật mã khối xây dựng từ vành đa thức chăn. Trong phần 3, áp dụng hệ mật đề xuất vào thực hiện một hàm băm 512 bit. Cuối cùng, phần 4 là một số đánh giá về hàm băm mới và kết luận của bài báo.

2. Hệ mật xây dựng từ CGP của vành đa thức chẵn

2.1. Nhóm nhân cyclic và cấp số nhân cyclic trên vành đa thức

Định nghĩa 1: Tập các đa thức $f(x)$ trong vành đa thức $\mathbb{Z}_2[x]/x^n+1$ với một phép toán nhân đa thức tạo nên một nhóm nhân G [1].

$$\langle f(x);*\rangle = G \quad (1)$$

Nếu: $g(x), f(x) \in G$ thì $g(x)*f(x)=d(x)\in G$.

[*] Địa chỉ liên hệ: Tel.: (+84) 912.928.928
Email: thiennd@ptit.edu.vn

Trong nhóm nhân G luôn tồn tại phần tử đơn vị $e(x)$ với $f(x)*e(x)=f(x)$.

Định nghĩa 2: Nhóm nhân cyclic (CMG – Cyclic Multiplicative Group) trên vành đa thức $\mathbb{Z}_2[x]/x^n+1$ là tập hợp các đa thức có dạng sau:

$$CMG=\left\{g^i(x) \bmod x^n+1,\ i=1,2,3,...\right\} \quad (2)$$

với $g(x)\in\mathbb{Z}_2[x]/x^n+1\setminus\{0\}$ là phần tử sinh.

Cấp của $g(x)$ ký hiệu $ord\,g(x))$ là số phần tử của CMG.

$$ord\,g(x)=|CMG| \quad (3)$$

Xét vành đa thức $\mathbb{Z}_2[x]/x^n+1$ với $n=2^k$, ta có các bổ đề sau:

Bổ đề 1: Tập các đa thức có trọng số lẻ tạo thành nhóm nhân G theo modulo x^n+1 [2], [3].

Bổ đề 2: Mọi phần tử trong G có cấp là 2^k hoặc có cấp là ước của 2^k [2], [3].

Bổ đề 3: Đa thức $a(x)$ là phần tử cấp n khi nó có chứa một số lẻ các đơn thức có mũ lẻ có cấp n và một số chẵn các đơn thức có mũ chẵn có cấp là ước của n. Số các đa thức cấp n bằng 2^{n-2} [2].

Theo bổ đề 3, có tất cả 2^{n-2} CMG cấp n, kể cả nhóm nhân đơn vị I.

Định nghĩa 3: Cấp số nhân cyclic (CGP) trên vành đa thức có dạng sau:

$$CGP=\left\{a(x).g^i(x) \bmod x^n+1;\ i=1,2,3,...\right\} \quad (4)$$

với $a(x)$ là phần tử đầu; $g(x)$ là phần tử sinh.

Nói cách khác, nếu ta nhân các phần tử của một CMG với một phần tử bất kỳ trong nhóm nhân G ta sẽ thu được một CGP có công bội là phần tử sinh của CMG và phần tử đầu chính là đa thức đem nhân.

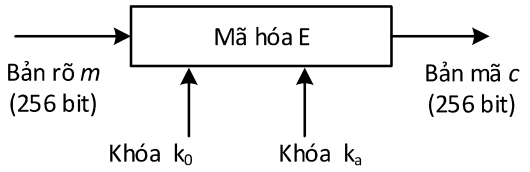
Bổ đề 4: Trong vành đa thức $\mathbb{Z}_2[x]/x^n+1$ với $n=2^k$, số các CGP cấp n xây dựng được trong G được xác định như sau [3]:

$$N=2^{n-1}2^{n-2}=2^{2n-3} \quad (5)$$

Dễ dàng nhận thấy: Theo bổ đề 3 có tất cả 2^{n-2} CMG có cấp n và theo bổ đề 1 các đa thức lẻ tạo thành nhóm nhân G, số phần tử lẻ của vành là 2^{n-1} và do đó ta có công thức (5).

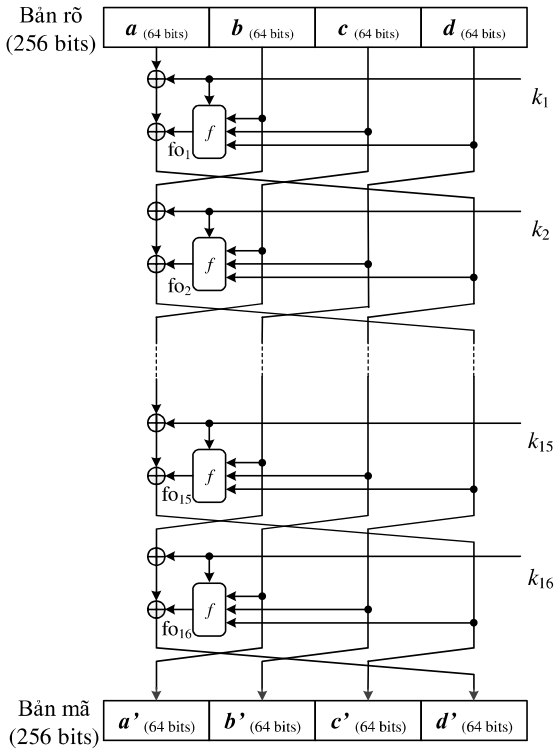
2.2. Mô tả hệ mật đề xuất

Sơ đồ khối phần mã hóa của hệ mật như hình 1. Trong sơ đồ này, hệ mật sẽ thực hiện mã hóa các khối bản rõ 256 bit và cho ra bản mã cũng có độ dài 256 bit. Hệ mật gồm hai khóa bí mật là k_0 và k_a (đều có độ dài 64 bit) là hai đa thức tạo CGP làm khóa con cho hệ mật, được mô tả theo công thức (12).



Hình 1. Sơ đồ khối của hệ mật

Sơ đồ mã hóa của hệ mật dựa trên mạng Feistel bốn nhánh không cân bằng [4], [5] như mô tả trong hình 2.



Hình 2. Sơ đồ mã hóa của hệ mật

Quá trình mã hóa trải qua 16 vòng, mỗi vòng mã hóa thực hiện theo thuật toán sau:

$$\begin{cases} a_i = b_{i-1} \\ b_i = c_{i-1} \\ c_i = d_{i-1} \\ d_i = fo_{i-1} \\ fo_i = a_i \oplus k_i \oplus f(b_i, c_i, d_i, k_i) \end{cases} \quad (6)$$

Quá trình giải mã thực hiện theo chiều ngược lại, thứ từ các khóa con sẽ từ $k_{16} \to k_1$ và cũng trải qua 16 vòng, mỗi vòng mã hóa thực hiện theo thuật toán sau:

$$\begin{cases} b_i = a_{i-1} \\ c_i = b_{i-1} \\ d_i = c_{i-1} \\ a_i = d_{i-1} \oplus f(b_i, c_i, d_i, k_i) \oplus k_i \end{cases} \quad (7)$$

Trong sơ đồ hình 1, hàm mã hóa f có bốn đầu vào, ba đầu vào dữ liệu (b_i, c_i và d_i) và một đầu vào khóa k_i. Độ dài của tất cả bốn đầu vào và đầu ra đều là 64 bit. Việc mã hóa được thực hiện bởi các CGP trên vành đa thức $\mathbb{Z}_2[x]/x^n+1$ với $n = 2^6 = 64$.

Cấu trúc của CGP sử dụng trong hàm f có dạng:

$$A_i = \left\{ k_i(x).g^j(x) \bmod x^{64} + 1;\ j = 1,2,...,64 \right\} \quad (8)$$

Trong đó: $k_i(x)$ là đa thức đầu và cũng là các khóa con của các vòng mã hóa, $g(x)$ là một đa thức sinh nào đó có cấp 64. Theo bổ đề 4 có tất cả $2^{64-2} = 2^{62}$ đa thức cấp 64, để thuận tiện ta có thể chọn $g(x) = x$ và khi đó ta có:

$$A_i = \left\{ k_i(x).x^j \bmod x^{64} + 1;\ j = 1,2,3,...,64 \right\} \quad (9)$$

Ta có thể mô tả A_i theo dạng ma trận sau [3]:

$$A_i = \begin{bmatrix} k_i(x).x^0 \\ k_i(x).x^1 \\ \vdots \\ k_i(x).x^{63} \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & \cdots & a_{63} \\ a_1 & a_2 & \ldots & a_{62} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{bmatrix}_{64\times 64} \quad (10)$$

Trong đó, $a_t \in \{0,1\}$.

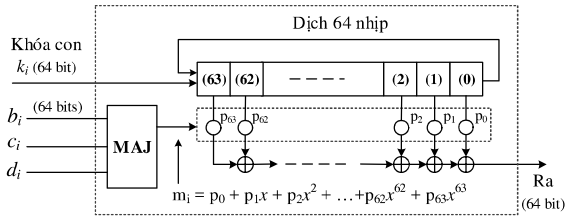
Chú ý: các hàng của ma trận A tương ứng với các đa thức của CGP, các phần tử a_t trong một hàng là hệ số của đơn thức x^t tương ứng trong đa thức. Ví dụ, hàng 1 của A là đa thức $k_i(x).x^0 = 1 + x^2 + x^3$ thì các hệ số a_t của hàng 1 sẽ là $a_t = 1$ với $t = 0,2,3$ và $a_t = 0$ với các giá trị t còn lại [4].

Tại các vòng mã hóa i, hàm f thực hiện các phép tính sau:

$$\begin{cases} m_i = \text{MAJ}(b_i, c_i, d_i) = b_i c_i + c_i d_i + d_i b_i \\ out_i = m_i A_i^T \end{cases} \quad (11)$$

Với A_i^T là ma trận chuyển vị của A_i. Ma trận A_i được tính như (10).

Sơ đồ mạch thực hiện hàm f như trong hình 3.



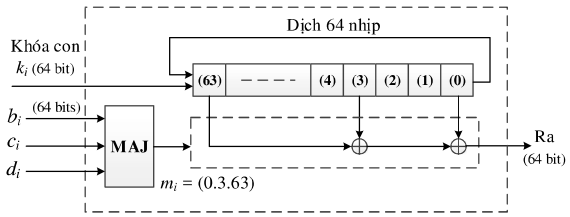
Hình 3. Mạch mã hóa hàm f

Hoạt động của mạch mã hóa hình 3 như sau: Các bit của khóa con k_i (tương ứng các hệ số của đa thức $k_i(x)$) sẽ được nạp vào thành ghi 64 bit; ba đầu vào b_i, c_i, d_i sau khi đưa qua mạch MAJ [5] sẽ cho ra một chuỗi bit, tương ứng với đa thức m_i nào đó. Các hệ số p_j (hay các bit) của đa thức m_i sẽ tương ứng với các khâu p_j ($j = 0,1,...,63$) của mạch mã hóa. Hoạt động của các khâu p_i phụ thuộc vào các hệ số p_j của đa thức m_i như sau:

+ p_j nối tắt nếu hệ số $p_j = "1"$,
+ p_j hở mạch nếu hệ số $p_j = "0"$.

Các bit trong thanh ghi sẽ được dịch vòng và sau 64 nhịp dịch ta sẽ có mã đầu ra có độ dài 64 bit.

Hình 4 là một ví dụ mạch điện mã hóa khi $m_i = 1 + x^3 + x^{63} \Leftrightarrow (0.3.63)$ (biểu diễn dạng số mũ của đa thức), với m_i được tính theo hàm MAJ như biểu thức (11). Tương ứng ta có: $p_0, p_3, p_{63} = "1"$, còn $p_j = "0"$ với các j khác.



Hình 4. Mạch mã hóa với $m_i = 1 + x^3 + x^{63}$

Tất cả các khóa con k_i cũng được tạo từ các CGP trên vành $\mathbb{Z}_2[x]/x^{64}+1$. Do k_i là phần tử đầu trong CGP A_i nên nó phải có trọng số lẻ (theo bổ đề 1). Cấu trúc của các khóa con k_i như sau:

$$k_i = \left\{ k_0 k_a^i \bmod x^{64} + 1,\ i = 1,2,3,...,16 \right\} \quad (12)$$

trong đó, k_0 là đa thức đầu, k_a là đa thức sinh. Để k_i có trọng số lẻ thì cả k_0 và k_a cùng phải có trọng số lẻ.

Theo bổ đề 4, số CGP có cấp 64 tính sẽ là 2^{125}. Tuy nhiên, do sơ đồ mã hóa chỉ có 16 vòng nên CGP tạo k_i cũng chỉ cần tối thiểu 16 phần tử, do đó trên thực tế số CGP có thể chọn sẽ nhiều hơn.

Trong sơ đồ mã hóa hình 1, có một khác biệt nhỏ so với sơ đồ Feistel 4 nhánh không cân bằng nguyên thủy. Đó là, ở bước mã hóa thứ i dữ liệu ở khối a_i được cộng với khóa con k_i, điều này để tránh trường hợp khi bản rõ đầu vào đều là bit "0" hoặc bit "1" thì bãn mã đầu ra cũng sẽ toàn bit "0" hoặc bit "1" tương ứng.

2.3. Đánh giá độ khuếch tán của hệ mật

Tiến hành tính toán độ khuếch tán của hệ mật khi thay đổi dữ liệu bản rõ. Đa thức sinh k_a và đa thức đầu k_0 để tạo khóa của hệ mật được giữ cố định và chọn như sau:

$$k_a = 1 + x^3 + x^4,\ k_0 = 1 + x + x^2 \qquad (13)$$

Bản rõ bao gồm 256 bit và được chọn và biểu diễn theo dạng hexa như sau:

$$M_0 = \begin{matrix} 00112233445566778899AABBCCDDEEFF \\ 00112233445566778899AABBCCDDEEFF \end{matrix} \qquad (14)$$

Bảng 1. Độ khuếch tán giữa các cặp bản mã của một vài bước tính toán khi thay đổi một bit bản rõ

j	Bản rõ tại bước j (M_j)	Bản mã tại bước j (C_j)	$d_H(C_0, C_j)$
0	0011223344556677 8899AABBCCDDEEFF 0011223344556677 8899AABBCCDDEEFF	71877CBA2367BA32 4AFA1EF5B894D0CA 02C3D5B87A1EF3D6 A6FF644BF1FF4010	0 (Chưa đổi)
1	**1**011223344556677 8899AABBCCDDEEFF 0011223344556677 8899AABBCCDDEEFF	F8596D78B7F01E1C 1BFD93C76E61A525 259462525D4F1020 F8EB0B0E4FE85395	135
...	...	...	...
64	001122334455667**F** 8899AABBCCDDEEFF 0011223344556677 8899AABBCCDDEEFF	0731F31001CE3563 EAAF1881A7FA55AC 1B6BF9E899E71AD5 0E4083A37AC7B4FA	128
...	...	...	...
128	0011223344556677 8899AABBCCDDEEF**7** 0011223344556677 8899AABBCCDDEEFF	8CAFFE01729B9EFE 0D4051BDA77C611A 5001A91E2E8D1358 6A2AAC26FB120CD5	125
...	...	...	...
256	0011223344556677 8899AABBCCDDEEFF 0011223344556677 8899AABBCCDDEEF**7**	8B584F7F841CC9B4 B69915BBE2A0DB80 40800B7E51447C2A C9359355F1D93E6B	140

Mỗi bước mô phỏng ta chỉ thay đổi một bit bản rõ [6], tiến hành thay đổi lần lượt từ bit số 1 đến bit thứ 256 rồi tính khoảng cách Hamming trung bình (cũng chính là độ khuếch tán) ta được:

$$d_{H(tb)} = \frac{1}{256}\sum_{j=1}^{256} d_H(C_0, C_j) = 127{,}19\,(bit) \qquad (15)$$

Kết quả tính toán $d_H(C_0, C_j)$ của một vài bước thay đổi bản rõ như trong bảng 1.

Chú ý: trong các bảng 1, 3 các ký tự hexa in đậm là ký tự chứa bit bị thay đổi.

Thực hiện 10 lần tính toán tương tự như trên với tham số khóa k_0, k_a vẫn chọn như (13), bản rõ được tạo ngẫu nhiên cho mỗi lần tính toán. Mỗi lần tính ta cũng thay đổi lần lượt từng bit từ 1 đến 256 của mỗi bản rõ ban đầu, kết quả có được như trong bảng 2.

Bảng 2. Kết quả của 10 lần tính độ khuếch tán khi thay đổi bản rõ

t	Bản rõ ban đầu M_t	Độ khuếch tán trung bình $d_{H(tb)_t}$
1	BF621A5DCBA779FF81E36CD649F5CFB6 770AEB6C1866F6E3C8A18EFEDB561068	127.28
2	AF508BBEE0CEAD0AE00230E0216D48B9 5BC3C8A3BF4A6AD7FB0D31585EE98F9F	127.92
3	43FAB9F62753DF4D03556DD0DAA879A9 0E46D6859B63EC7F3C20F4B6EF175B4C	127.91
4	90349D10DE0244BE8B5B96F10E6559E1 B1BE52564EE50307C13E11BB11A0BA2A	127.77
5	4C49D70D5657DC30F945367D82F4D442 C905C14B8BEC84EC002346F285A35B39	127.34
6	7C72111C7F84B941AAEC8D09D518B2D1 F7F186D65B6964D47F7801ADA40FAA05	128.15
7	9F52132E8FE8666168E6B97C22DFEA47 55C25F4DCBA9CFA193E18967B3C082DE	127.58
8	AD817EE501A151F612F8DDF82E8736E3 BA9FBE874EBDACFC6B6F8590F524D500	128.23
9	5B6BA487AE8A189F2F84E83ACB1791A0 08AF0D9E4F756F8188F8111A27D1D0C5	128.43
10	F6954DD4D314B4E474AC3AE492973457 95053977F3E8868C279794983DD18998	127.81

Độ khuếch tán trung bình của 10 lần tính toán:

$$d_{H(tb)} = \frac{1}{10}\sum_{t=1}^{10} d_{H(tb)_t} = 127{,}84 \qquad (16)$$

Tiến hành tính toán độ khuếch tán khi thay đổi khóa ban đầu k_0, bản rõ được giữ cố định như (13).

Đa thức sinh tạo khóa vẫn chọn $k_a = 1 + x^3 + x^4$, đa thức đầu k_0 được chọn như sau:

$$k_{0(hex)} = 0123456789ABCDE\mathbf{7}$$

Bit 64 (bit kiểm tra)

$$k_{0(bin)} = 0000.1000.0100.\ldots\ldots.1011.0111.1110$$

$$k_{0(hex)} = \mathbf{0\ .\ 1\ .\ 2\ \ldots\ldots\ D\ .\ E\ .\ 7}$$

Vì khóa k_0 phải có trọng số lẻ nên ta dùng bit thứ 64 làm bit kiểm tra. Nếu tổng 63 bit đầu tiên là lẻ hoặc chẵn thì bit 64 tương ứng là "0" hoặc "1".

Với khóa k_0 chọn như trên thì trọng số của khóa là $W(k_0) = 31$.

Tiến hành thay đổi lần lượt từng bit của khóa ban đầu k_0 từ bit 1 đến bit 63 [6], ta tính được độ khuếch tán giữa các cặp bản mã $d_H(C_0, C_j)$ của từng lần thay đổi, sau đó tính giá trị trung bình như sau:

$$d_{H(tb)} = \frac{1}{63}\sum_{j=1}^{63} d_H(C_0, C_j) = 127,02(bit) \qquad (17)$$

Kết quả tính toán $d_H(C_0, C_j)$ của một vài bước thay đổi khóa như trong bảng 3.

Bảng 3. Độ khuếch tán giữa các cặp bản mã của một vài bước tính toán khi thay đổi một bit khóa k_0

j	Khoá ban đầu $k_{0.j}$	Bản mã C_j	$d_H(C_0, C_j)$
0	0123456789ABCDE7	157A2A4A3FA24F94 80D0E9BFF1B8B3F1 D391B10032B69014 33F36C843E3E0A95	0
1	**1**123456789ABCDEF	A0DED61415AB0544 68D8C11032A3577E 35523A44D67815CF 50554A4ECCAE1178	125
...	...	...	...
16	012**B**456789ABCDEF	2C1FB5DD6876DE6A 0DF72FDA0DCE20EB DE9EBEEF342C0B58 B0EA9F1D974D41D9	136
...	...	...	...
32	0123456**F**89ABCDEF	BADB77A3E0AD5C5B 2D3613241F99504D 1A3E798C7DB6EB34 9F68AFD9FBCB19AA	143
...	...	...	...
63	0123456789ABCDE**B**	99EDD2569C7E1BAD AC6BDAB5E8B13D57 62466CA2A5696117 E5207E26DD6A345D	129

3. Ứng dụng hệ mật vào một hàm băm không khóa

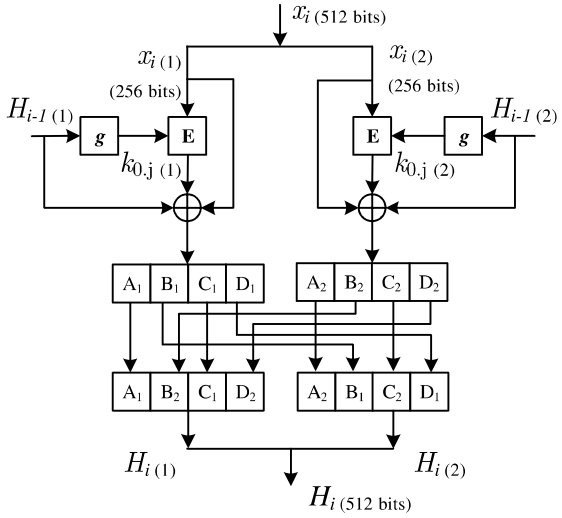
3.1. Mô tả hàm băm

Trong phần này, chúng tôi áp dụng hệ mật trên vào một hàm băm không khóa MDC-2, có độ dài mã băm 512 bit. Sơ đồ thực hiện hàm băm như mô tả trong hình 5.

Sơ đồ hàm băm gồm hai nhánh, mỗi nhánh dựa trên sơ đồ băm Miyaguchi – Preneel [7]. Khối mã hóa E là hệ mật như đề xuất ở mục 2. Mã băm đầu ra của mỗi nhánh băm sẽ được tráo đổi xen kẽ 64 bit (các khối A, B, C, D của hai nhánh đều có kích thước 64 bit).

Các khối g thực hiện việc trích trọn từ các mã băm của vòng trước (H_{i-1}) để tạo các khóa $k_{0.j}$ cho vòng hiện tại. Cụ thể, từ 256 bit mã băm đầu ra của từng nhánh, khối g sẽ trích lấy 63 bit từ bit thứ nhất và cách nhau 4 vị trí, tức là các bit 1, 5, 9, 13,... Bit thứ 64 dùng để kiểm tra tính chẵn lẻ của $k_{0.j}$.

Để băm khối đầu tiên, ta phải chọn một mã băm khởi tạo H_0 (256 bit) cho cả hai nhánh băm. Sau khi đi qua khối g sẽ cho ra đa thức đầu $k_{0.1}$ cho vòng mã hóa đầu tiên, mã băm H_0 có thể chọn tùy ý.



Hình 5. Sơ đồ hàm băm không khóa MDC-2

3.2. Đánh giá độ khuếch tán của hàm băm

Tiến hành tính toán độ khuếch tán của hàm băm khi thay đổi dữ liệu băm với các thông số như sau:

CGP tạo ma trận mã hóa của E như biểu thức (9).

Đa thức sinh tạo khóa của hệ mật k_a vẫn chọn như (13), khóa này được chọn giống nhau cho 2 nhánh và được giữ cố định khi tính toán độ khuếch tán.

Mã băm khởi tạo H_0 (256 bit) cho cả hai nhánh băm được chọn như sau:

```
H0 =  00001111222233334444555566667777
      88889999AAAABBBBCCCCDDDDEEEEFFFF
```  (18)

Dữ liệu băm đầu vào x có độ dài 5120 bit được tạo ngẫu nhiên (khi băm sẽ được chia thành 10 khối, mỗi khối x_i 512 bit).

Bảng 4 là một vài kết quả khuếch tán mã băm khi thực hiện một trăm lần thay đổi ngẫu nhiên một bit bất kỳ trong 5120 bit dữ liệu băm đầu vào. Tất cả các lần băm đều dùng chung mã băm khởi tạo ban đầu H_0 như trong (18). Khoảng cách Hamming trung bình của 100 lần băm tính được như sau:

$$d_{H(tb)} = \frac{1}{100}\sum_{i=1}^{100} d_H(C_0, C_i) = 243,48\,\text{bit} \qquad (19)$$

Bảng 4. Độ khuếch tán của một vài mã băm khi thay đổi dữ liệu đầu vào

TT	Vị trí bit dữ liệu băm thay đổi	Mã băm C_i (128 số hexa ↔ 512 bit)	$d_H(C_0,C_i)$
0	(Chưa thay đổi)	5904935A3FA5737D04FD0C37D2535F25 477807BF98F0F6AA751D0B9A81E9C0CB 986CB4AB61C450A8A1F77395FDAB7B49 4EAD87D518215183E65076050A011250	0
1	3777	37B9A70975EA72287C4846BB128EBE25 4C6B3B22441FEA9AC6059883652A4BA6 7B5189536FB7D345C0D8DA623C90AB00 CD161FE4D60CE39589316681C89A9A04	247
2	3663	AB347A544B3A0D8F8C0F324571817FAE 42CB0DB15922CBC43512A5A401EA655A 5987013635AFE7D2A23473EBC230D151 EAB78E9DF92A6AD8168EEE6016C8F97E	247
...	...	...	...
50	1002	0195C0427E13E466E7903732F9222147 1FD84F0970D08A8994E3456BB1B3CF43 1D3FDA7B6270745BF63D25C9890D85F7 D24FFB7946174C5972AD70CF09FC02A5	253
...	...	...	...
99	541	213DF5EF6494F68D98EAA69B9CB8F6A5 D35872AC5230F57B391D0470D7AAA95D EF0FDA5D6153697960CA3FE7887AB42C E5BB3256E6EB698B009750BCA6E7DB55	246
100	1559	02233F2A5DEAD4903CC2BD7DB116CC3C DA472A73FDA01F3E2A1A2F604F9FFAE2 61F57D0E667BD6F53100B70E1CB7AE3B AB62B02B69DD2C9EDD107E7B9697F91E	275

Tiến hành tính toán như trên 10 lần, mỗi lần với dữ liệu đầu vào khác nhau (chọn ngẫu nhiên), mã băm khởi tạo H_0 chọn như (18), ta có kết quả như bảng 5.

Bảng 5. Kết quả 10 lần tính khuếch tán của hàm băm

TT	$d_{H(tb)_t}$ (bit)	TT	$d_{H(tb)_t}$ (bit)
1	242,09	6	243,61
2	243,44	7	239,44
3	241,48	8	244,60
4	248,34	9	251,15
5	241,72	10	244,30

Khoảng cách Hamming trung bình của 10 lần:

$$d_{H(tb)} = \frac{1}{10}\sum_{t=1}^{10} d_{H(tb)_t} = 244,02\,(bit) \qquad (20)$$

4. Kết luận

Hệ mật mã khối khóa đối xứng 256 bit như đề xuất sử dụng cấp số nhân cyclic trên vành đa thức $Z_2[x]/x^{64}+1$ làm hàm mã hóa và các khóa con trong các vòng mã hóa. Hệ mật gồm 2 khóa bí mật, tạo nên tổng số CGP cũng chính là không gian khóa là 2^{125} khóa, đảm bảo an toàn trước tấn công vét cạn. Sơ đồ mã hóa của hệ mật xây dựng theo mạng Feistel bốn nhánh không cân bằng, cấu trúc hàm mã hóa *f* đơn giản và tốc độ mã hóa nhanh. Độ khuếch tán khi thay đổi dữ liệu bản rõ và thay đổi khóa khá tốt, đạt xấp xỉ một nửa độ dài từ mã (theo các biểu thức (15), (16) và (17)).

Với mục đích xây dựng các hàm băm có độ dài lớn để hạn chế phép tấn công ngày sinh nhật, bài báo đã áp dụng hệ mật đề xuất vào một hàm băm MDC-2 với độ dài mã băm 512 bit, các kết quả tính độ khuếch tán của hàm băm cũng cho thấy khá tốt (theo biểu thức (19), (20)). Để có các đánh giá hoàn chỉnh về hàm băm này, cần có các khảo sát đánh giá thêm về các tính chất khác của hàm băm, như tính va chạm, khó tìm nghịch ảnh…

Tài liệu tham khảo

[1] Nguyen Binh, "Cyclic and Local Cyclic Codes over Polynomial Ring", Journal of Science and Technology, Vietnam, Vol. 50 (2012), pp. 735-749, ISSN 0866 708X.

[2] Nguyen Binh, Le Dinh Thich - The oders of polynomials and algorithms for defining order of polynomial over polynomial ring, VICA-5, Hanoi, Vietnam, 2002.

[3] Ho Quang Buu, Ngo Duc Thien, Tran Duc Su, "Constructing a Crypto system based on Cyclic Geometric Progressions in Polynomial Rings", Jounal of Science and Technology, Vietnam Academy of Science and Technology, Vol. 50 - 2A, (2012) pp. 109-119, ISSN 0866 708X.

[4] Ngo Duc Thien, Dang Hoai Bac, "A method of building a crypto system based on unbalanced Feistel network and its application in hash functions", Tạp chí Nghiên cứu khoa học và công nghệ quân sự, ISSN 1859-1043, số 34, 2014, pp.41-48.

[5] Pascal JUNOD, "Statistical Cryptanalysis of Block Ciphers", Thèse N^0 3179, Insitute de systèmes de communication, Ècole Polytechnique Fédérale de Lausanne, 2005.

[6] Jean-Yves Chouinard - ELG 5373, "Secure Communications and Data Encryption, School of Information Technology and Engineering", University of Ottawa, April 2002.

[7] A. Menezes, P. van Oorschot, and S. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1996.