TCVN TIÊU CHUẨN QUỐC GIA

TCVN 11495-1:2016
ISO/IEC 9797-1:2011

CÔNG NGHỆ THÔNG TIN - CÁC KỸ THUẬT AN TOÀN - MÃ XÁC THỰC THÔNG ĐIỆP (MAC) - PHẦN 1: CƠ CHẾ SỬ DỤNG MÃ KHỐI

Information technology - Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher

HÀ NỘI - 2016

Mục lục **Trang**

Lời nói đầu

TCVN 11495-1:2016 hoàn toàn tương đương với ISO/IEC 9797-1:2011.

TCVN 11495-1:2016 do Tiểu ban kỹ thuật tiêu chuẩn quốc gia TCVN/JTC1/SC 27 *Kỹ thuật an ninh* biên soạn, Tổng cục Tiêu chuẩn Đo lường Chất lượng đề nghị Bộ Khoa học và Công nghệ công bố.

Bộ tiêu chuẩn TCVN 11495 (ISO/IEC 9797) *Công nghệ thông tin – Các kỹ thuật an toàn – Mã xác thực thông điệp (MAC)* gồm các tiêu chuẩn sau:

- Phần 1: Cơ chế sử dụng mã khối;
- Phần 2: Cơ chế sử dụng hàm băm chuyên dụng;
- Phần 3: Cơ chế sử dụng hàm băm phổ biến;

Lời giới thiệu

Trong môi trường công nghệ thông tin, thường yêu cầu người ta có thể xác thực rằng dữ liệu điện tử đã không bị thay đổi theo một cách trái phép và người ta có thể cung cấp đảm bảo rằng thông điệp đã được khởi tạo bởi một thực thể mà nắm giữ khóa bí mật. Thuật toán Mã Xác thực Thông điệp (MAC – Message Authentication Code) là một cơ chế toàn vẹn dữ liệu thường được sử dụng mà có thể thỏa mãn những yêu cầu này.

Tiêu chuẩn này chỉ ra sáu thuật toán MAC, chúng dựa trên một mã khối *n*-bit. Các thuật toán này tính ra một chuỗi ngắn thông qua một hàm của một khóa bí mật và một thông điệp có độ dài thay đổi.

Độ mạnh của cơ chế toàn vẹn dữ liệu và cơ chế xác thực thông điệp phụ thuộc vào độ dài (tính theo bit) k^* và tính bí mật của khóa, vào độ dài khối (theo bit) *n* và độ mạnh của mã khối, vào độ dài (theo bit) *m* của MAC và vào cơ chế cụ thể.

Cơ chế thứ nhất được chỉ ra trong tiêu chuẩn này thường được biết như CBC-MAC (CBC là viết tắt của Cipher Block Chaining – Móc xích Khối Mã).

Năm cơ chế còn lại là các biến thể của CBC-MAC. Các thuật toán MAC 2, 3, 5 và 6 áp dụng một biến đổi đặc biệt tại cuối của quá trình xử lý. Thuật toán MAC 6 là biến thể được tối ưu hóa của Thuật toán MAC 2. Thuật toán MAC 5 sử dụng số lần mã tối thiểu. Thuật toán MAC 5 chỉ yêu cầu thiết lập khóa mã khối đơn nhưng nó cần một khóa nội bộ dài hơn. Thuật toán MAC 4 áp dụng một biến đổi đặc biệt tại lúc bắt đầu và lúc kết thúc quá trình xử lý; thuật toán này được khuyến cáo để sử dụng trong các ứng dụng mà yêu cầu độ dài khóa của thuật toán MAC bằng hai lần độ dài mã khối.

Công nghệ thông tin - Các kỹ thuật an toàn - Mã xác thực thông điệp (MAC) - Phần 1: Cơ chế sử dụng mã khối

Information technology - Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher

1 Phạm vi áp dụng

Tiêu chuẩn này quy định sáu thuật toán MAC có sử dụng một khóa bí mật và một mã khối *n*-bit để tính ra một MAC *m*-bit.

Tiêu chuẩn này có thể áp dụng cho các dịch vụ an toàn của các kiến trúc, quy trình hay ứng dụng an toàn thông tin bất kỳ.

Phạm vi tiêu chuẩn này không bao gồm các cơ chế quản lý khóa.

Tiêu chuẩn này quy định các định danh đối tượng mà được sử dụng để định danh từng cơ chế phù hợp với ISO/IEC 8825-1. Các ví dụ số và một phân tích độ an toàn được đề cập cho mỗi thuật toán trong sáu thuật toán cụ thể đã cung cấp, và các mối quan hệ của tiêu chuẩn này với các tiêu chuẩn khác trước đó được giải thích.

2 Tài liệu viện dẫn

Các tài liệu viện dẫn sau đây rất cần thiết cho việc áp dụng tiêu chuẩn này. Đối với các tài liệu ghi năm công bố thì áp dụng phiên bản được nêu. Đối với các tài liệu không ghi năm công bố thì áp dụng phiên bản mới nhất, bao gồm cả các sửa đổi, bổ sung (nếu có).

TCVN 11367-3 (ISO/IEC 18033-3), *Công nghệ thông tin – Các kỹ thuật an toàn – Thuật toán mã hóa – Phần 3: Mã khối (Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers)*.

3 Thuật ngữ và định nghĩa

Tiêu chuẩn này áp dụng các thuật ngữ và định nghĩa sau đây.

3.1

khối (block)

chuỗi bit có chiều dài *n*.

3.2

khóa mã khối (block cipher key)

khóa dùng để kiểm soát hoạt động của mã khối.

3.3

bản mã (ciphertext)

dữ liệu đã được biến đổi để che giấu nội dung thông tin.

[ISO/IEC 9798-1:2010]

3.4

toàn vẹn dữ liệu (data integrity)

tính chất của dữ liệu đã không bị thay đổi hay bị làm hư hại theo một cách trái phép.

[ISO 7498-2]

3.5

giải mã (decryption)

việc đảo ngược một phép mã hóa tương ứng.

[ISO/IEC 9798-1:2010]

3.6

mã hóa (encryption)

phép toán có thể đảo ngược bởi một thuật toán mật mã chuyển đổi dữ liệu thành bản mã để che giấu được nội dung thông tin của dữ liệu.

[ISO/IEC 9798-1:2010]

3.7

khóa (key)

chuỗi các ký hiệu dùng để điều khiển hoạt động của một phép biến đổi mật mã.

CHÚ THÍCH Các ví dụ về phép mã hóa, giải mã, tính hàm kiểm tra mật mã, sinh chữ ký hoặc xác thực chữ ký.

[ISO/IEC 9798-1:2010]

3.8

khóa thuật toán MAC (MAC algorithm key)

khóa dùng để điều khiển hoạt động của một thuật toán MAC.

3.9

Mã Xác thực Thông điệp (Message Authentication Code)

MAC

chuỗi các bit là đầu ra của một thuật toán MAC.

CHÚ THÍCH Một mã MAC đôi khi được gọi là một giá trị kiểm tra mật mã (xem ví dụ trong ISO 7498-2 [1]).

3.10

thuật toán Mã Xác thực Thông điệp (Message Authentication Code algorithm)

thuật toán MAC (MAC algorithm)

thuật toán để tính ra một hàm ánh xạ các chuỗi bit và một khóa bí mật thành các chuỗi bit có chiều dài cố định, thỏa mãn hai tính chất sau:

- đối với khóa bất kỳ và chuỗi đầu vào bất kỳ, hàm có thể tính được một cách hiệu quả;
- đối với khóa cố định bất kỳ, khi không cho biết thông tin về khóa trước về mặt tính toán là không thể tính ra giá trị của hàm trên bất kỳ chuỗi đầu vào mới nào, thậm chí khi đã biết thông tin về tập chuỗi đầu vào và các giá trị hàm tương ứng, trong đó giá trị của chuỗi đầu vào thứ i có thể được chọn sau khi quan sát $i-1$ giá trị hàm đầu tiên (đối với số nguyên $i > 1$).

CHÚ THÍCH 1 Thuật toán MAC đôi khi được gọi là hàm kiểm tra mật mã (xem ví dụ trong ISO/IEC 7498-2 [1]).

CHÚ THÍCH 2 Khả năng tính toán phụ thuộc vào các yêu cầu và môi trường an toàn do người dùng quy định.

3.11

mã khối *n*-bit (*n*-bit block cipher)

mã khối có tính chất là các khối bản rõ và các khối bản mã đều có độ dài là n bit.

[ISO/IEC 10116]

3.12

biến đổi đầu ra (output transformation)

hàm được áp dụng tại điểm cuối thuật toán MAC, trước phép toán cắt ngắn.

3.13

bản rõ (plaintext)

thông tin chưa được mã hóa.

CHÚ THÍCH Chấp nhận từ ISO/IEC 9798-1:2010.

4 Ký hiệu và giải thích

Tiêu chuẩn này sử dụng các ký hiệu và giải thích sau:

CT_i	Biểu diễn nhị phân n-bit của số nguyên i.
D	Chuỗi dữ liệu làm đầu vào cho thuật toán MAC.
D_j	Khối được rút ra từ chuỗi dữ liệu D sau khi thực hiện quy trình đệm và phân tách.
$d_K(C)$	Giải mã của bản mã C với mã khối e sử dụng khóa K.
$e_K(P)$	Mã hóa của bản rõ P với mã khối e sử dụng khóa K.
F	Phép lặp cuối cùng.
g	Phép biến đổi đầu ra mà ánh xạ khối H_q thành khối G.
G	Khối kết quả của phép biến đổi đầu ra.
GF(2^n)	Trường hữu hạn có chính xác 2^n phần tử.
$H_0, H_1, ..., H_q$	Các khối được sử dụng trong thuật toán MAC để lưu giữ các kết quả trung gian.
k	Độ dài (theo bit) của khóa của mã khối.
k^*	Độ dài (theo bit) của khóa thuật toán MAC.
K, K', K''	Các khóa bí mật của mã khối có độ dài (theo bit) k.
K_1, K_2	Các khóa che giấu bí mật có độ dài (theo bit) n.
L	Khối chỉ độ dài, được sử dụng trong Phương pháp Đệm 3, bằng với biểu diễn nhị phân của độ dài của thông điệp đầu vào, được đệm về bên trái thành một khối n-bit.
L_D	Độ dài (theo bit) của chuỗi dữ liệu D.
m	Độ dài (theo bit) của MAC.
multx(T)	Phép toán trên một chuỗi n-bit T được định nghĩa như $T*x$, trong đó T được xử lý như một phần tử trong trường hữu hạn GF(2^n), và được nhân bởi phần tử tương ứng với đơn thức x trong GF(2^n). Nó có thể được tính như sau, trong đó T_{n-1} ký hiệu bit bên trái nhất của T, << ký hiệu phép toán dịch sang trái 1 bit $\text{multx}(T) = \begin{cases} T \ll 1 & khi\ T_{n-1} = 0 \\ (T \ll 1) \oplus \tilde{p}_n & khi\ T_{n-1} = 1 \end{cases}$
n	Độ dài khối (theo bit) của mã khối.
$p_n(x)$	Các đa thức bất khả quy có bậc n trên GF(2), tức là, các đa thức không có các ước không tầm thường.
$\tilde{p}_n$	Chuỗi bit có độ dài n, bao gồm n hệ số phải nhất (tương ứng với x^{n-1}, x^{n-2}, ..., x, $x^0 = 1$) của đa thức bất khả quy $p_n(x)$. Đối với $n = 128$, $p_n(x) = x^{128} + x^7 + x^2 + x + 1$ thì $\tilde{p}_{128} = 0^{120}10000111$. Đối với $n = 64$, $p_n(x) = x^{64} + x^4 + x^3 + x + 1$ thì $\tilde{p}_{64} = 0^{59}11011$.
q	Số các khối trong chuỗi dữ liệu D sau quá trình đệm và phân tách
S	Chuỗi bí mật có độ dài (theo bit) n.
S_1, S_2	Các chuỗi bí mật có độ dài (theo bit) $t \cdot n$.
t	Số nguyên nhỏ nhất lớn hơn hoặc bằng với k/n.
j~X	Chuỗi nhận được từ chuỗi X bằng cách lấy j bit bên trái nhất của X.
$X \oplus Y$	Phép XOR của các chuỗi bit X và Y.

$X\|\|Y$	Phép ghép nối chuỗi bit X và Y (theo thứ tự này).
0^n	Chuỗi bao gồm n bit 0.
:=	Ký hiệu định nghĩa phép toán "đặt bằng với" được sử dụng trong các đặc tả thủ tục của các thuật toán MAC, trong đó nó chỉ ra rằng giá trị của chuỗi ở bên trái của ký hiệu sẽ được làm bằng với giá trị của biểu thức ở bên phải của ký hiệu
*	Phép nhân của trường hữu hạn. Trong biểu diễn đa thức, mỗi phần tử của GF(2^n) được biểu diễn bởi một đa thức nhị phân có bậc nhỏ hơn n. Chính xác hơn, chuỗi bit $A = a_{n-1}...a_2a_1a_0$ được ánh xạ vào đa thức nhị phân $a(x) = a_{n-1}x^{n-1} + ... + a_2x^2 + a_1x + a_0$. Phép nhân trong trường hữu hạn GF(2^n), được ký hiệu bởi $A*B$, tương ứng với phép nhân của 2 đa thức $a(x)b(x)$ lấy modulo theo một đa thức bất khả quy nhị phân $p_n(x)$ có bậc n; tức là, $A*B$ là đa thức có bậc nhiều nhất n-1 nhận được bằng cách nhân $a(x)$ và $b(x)$, chia kết quả bởi $p_n(x)$, và sau đó lấy phần dư. Ở đây $p_n(x)$ được chọn là đa thức đầu tiên theo thứ tự từ điển trong số các đa thức bất khả quy có bậc n mà có số các hệ số khác 0 nhỏ nhất. Đối với $n = 128$, $p_n(x) = x^{128} + x^7 + x^2 + x + 1$.
$X << 1$	Chuỗi nhận được từ chuỗi X bằng cách dịch trái 1 bit; nếu độ dài của X là n bit thì $X<<1$ là chuỗi có chứa n bit bênphải nhất của $X\|\|0$.

5 Các yêu cầu

Người dùng muốn áp dụng một thuật toán MAC trong tiêu chuẩn này phải lựa chọn:

- một mã khối e, hoặc một mã khối trong các quy định tại TCVN 11367-3 (ISO/IEC 18033-3) hoặc mã khối DEA (quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]). DEA chỉ có thể được sử dụng cùng với các Thuật toán MAC 3 và 4;
- một phương pháp đệm từ trong số các phương pháp được quy định tại Điều 6.3;
- một thuật toán MAC từ trong số các thuật toán được quy định tại Điều 7;
- độ dài (theo bit) m của MAC;
- một phương pháp dẫn xuất khóa chung nếu Thuật toán MAC 4 được sử dụng; một phương pháp dẫn xuất khóa chung có thể cũng được yêu cầu cho các Thuật toán MAC 2 và 6.

Thỏa thuận trên các lựa chọn này giữa những người dùng là cần thiết vì mục đích hoạt động của cơ chế toàn vẹn dữ liệu.

Độ dài m của MAC phải là một số nguyên dương nhỏ hơn hoặc bằng với độ dài khối n.

Nếu Phương pháp Đệm 3 được sử dụng, độ dài theo bit của chuỗi dữ liệu D sẽ phải nhỏ hơn 2^n.

Nếu Thuật toán MAC 4 được sử dụng, số các khối trong phiên bản đã được đệm của chuỗi dữ liệu phải lớn hơn hoặc bằng 2, tức là, $q \geq 2$.

Việc tuyển chọn một mã khối cụ thể e, phương pháp đệm, thuật toán MAC, giá trị cho m, và phương pháp dẫn xuất khóa (nếu có) là nằm ngoài phạm vi của tiêu chuẩn này.

CHÚ THÍCH 1 Các lựa chọn này ảnh hưởng tới mức an toàn của thuật toán MAC. Thông tin chi tiết được đề cập tại Phụ lục C.

Cùng một khóa sẽ được sử dụng để tính và kiểm tra MAC. Nếu chuỗi dữ liệu cũng được mã hóa, thì khóa được sử dụng để tính MAC cần phải khác với khóa được sử dụng để mã hóa.

CHÚ THÍCH 2 Được xem là một thực hành mật mã tốt khi có các khóa độc lập cho bảo mật và cho toàn vẹn dữ liệu.

Độ an toàn của các thuật toán MAC trong tiêu chuẩn này phụ thuộc chủ yếu vào các thủ tục và các thực hành được tuân theo để quản lý các khóa. Thông tin về quản lý khóa có thể được tìm thấy trong ISO 8732 [3], TCVN 7817 (ISO/IEC 11770) [8] và ISO 11568 [9].

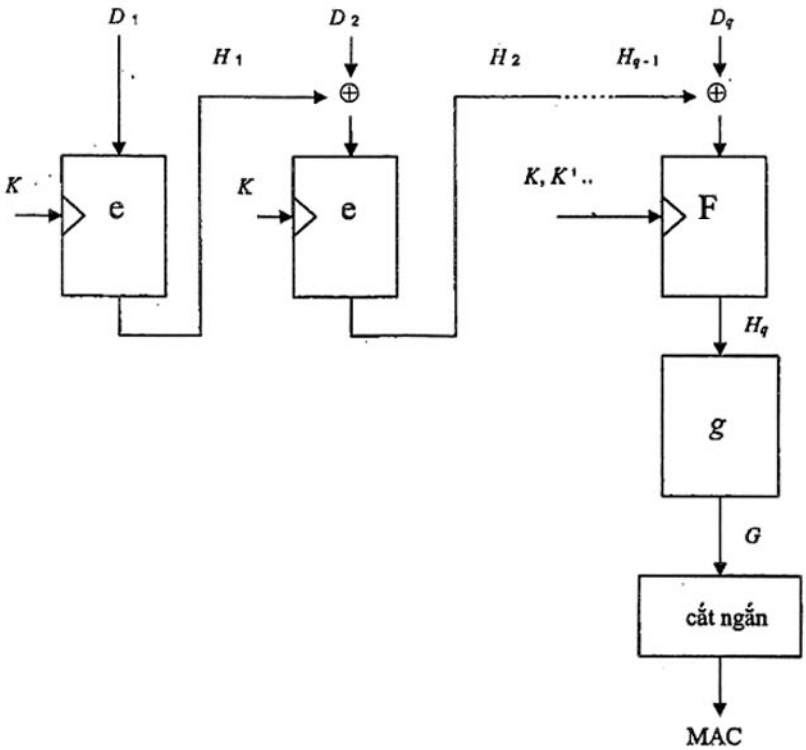
Việc bộc lộ các giá trị trung gian trong lúc tính các thuật toán MAC có thể cho phép giả mạo và/hoặc các tấn công khôi phục khóa (xem Phụ lục C).

6 Mô hình cho các thuật toán MAC

6.1 Tổng quát

Việc áp dụng thuật toán MAC yêu cầu 7 bước sau: dẫn xuất khóa (tùy chọn), thêm đệm, phân tách, áp dụng lặp của mã khối, lặp lần cuối, biến đổi đầu ra và cắt ngắn. Các bước từ 4 đến 7 được minh họa trong Hình 1.

CHÚ THÍCH Trong Thuật toán MAC 4, giai đoạn thứ nhất của việc lặp (Bước 4) khác với các giai đoạn khác.



Hình 1 – Việc áp dụng các bước 4, 5, 6 và 7 của thuật toán MAC.

6.2 Bước 1 (dẫn xuất khóa)

6.2.1 Tổng quan

Thuật toán MAC 5 sử dụng một thuật toán dẫn xuất khóa, nó dẫn xuất ra 2 khóa che giấu từ một khóa mã khối. Các Thuật toán MAC 2, 4 và 6 có thể cần một thuật toán dẫn xuất khóa, mà dẫn xuất hai khóa mã khối từ một khóa mã khối.

Tiêu chuẩn này quy định 2 thuật toán dẫn xuất khóa.

6.2.2 Phương thức Dẫn xuất Khóa 1

Phương pháp dẫn xuất khóa này tính ra 2 khóa mã khối K' và K'', mỗi khóa có độ dài k (theo bit), từ một khóa mã khối K.

Phương pháp dẫn xuất khóa này sử dụng Phương pháp Con đếm (Counter Method – CTR) đã được định nghĩa trong ISO/IEC 10116 [7]. Nó bao gồm các phép toán sau.

- Định nghĩa số nguyên t như là số nguyên nhỏ nhất lớn hơn hoặc bằng với k/n;
- Định nghĩa con đếm CT_i, $1 \le i \le 2t$ như chuỗi chứa biểu diễn nhị phân của số nguyên i được đệm về bên trái bởi một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một khối n-bit;
- Tính chuỗi S_1 có độ dài tn (theo bit) bằng với $e_K(CT_1)||e_K(CT_2)||$ .... $||e_K(CT_t)$ và đặt $K' := k \sim S_1$;
- Tính chuỗi S_2 có độ dài tn (theo bit) bằng với $e_K(CT_{t+1})||e_K(CT_{t+2})||$ .... $||e_K(CT_{2t})$ và đặt $K'' := k \sim S_2$.

6.2.3 Phương thức Dẫn xuất Khóa 2

Phương pháp dẫn xuất khóa này tính ra hai khóa che giấu K_1 và K_2 có độ dài n theo bit từ một khóa mã khối. Nó bao gồm các phép tính sau:

- Trước hết chuỗi bí mật S có độ dài n bit được tính như sau: $S := e_K(0^n)$;
- Tiếp theo, khóa che giấu K_1 nhận được từ: $K_1 :=$ multx(S);
- Cuối cùng khóa che giấu K_2 được dẫn xuất từ K_1: $K_2 :=$ multx(K_1).

6.3 Bước 2 (đệm)

6.3.1 Tổng quan

Bước này bao hàm việc gắn tiền tố và/hoặc gắn hậu tố cho chuỗi dữ liệu D bằng các bit 'đệm' bổ sung sao cho phiên bản đã được đệm của chuỗi dữ liệu sẽ luôn có độ dài là một bội nguyên dương của n. Các bit đệm đã được thêm vào chuỗi dữ liệu gốc (tùy thuộc vào phương pháp đệm đã chọn) chỉ được sử dụng để tính MAC. Do đó, các bit đệm này (nếu có) không cần phải được lưu giữ hoặc được truyền đi cùng với dữ liệu. Người kiểm tra phải biết dù các bit đệm có được lưu giữ hoặc truyền đi hay không, và phương pháp đệm nào đang được sử dụng.

Tiêu chuẩn này quy định 4 phương pháp đệm. Các phương pháp đệm 1, 2 và 3 có thể được chọn cho các Thuật toán MAC 1, 2, 3, 4 và 6 được quy định trong tiêu chuẩn này. Phương pháp đệm 4 chỉ được sử dụng cùng với Thuật toán MAC 5.

6.3.2 Phương pháp Đệm 1

Chuỗi dữ liệu D là đầu vào của thuật toán MAC phải được đệm bên phải bằng một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một chuỗi dữ liệu mà độ dài của nó (theo bit) là một bội nguyên dương của n.

CHÚ THÍCH 1 Các thuật toán MAC mà sử dụng Phương pháp Đệm 1 có thể phải chịu các tấn công giả mạo tầm thường. Xem Phụ lục C để biết thêm các chi tiết.

CHÚ THÍCH 2 Nếu chuỗi dữ liệu là trống, Phương pháp Đệm 1 chỉ ra rằng nó được đệm về bên phải bởi n bit '0'.

6.3.3 Phương pháp Đệm 2

Chuỗi dữ liệu D là đầu vào của thuật toán MAC phải được đệm bên phải bởi một bit '1' duy nhất. Chuỗi kết quả sau đó phải được đệm bên phải bằng một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một chuỗi dữ liệu mà độ dài của nó (theo bit) là một bội nguyên dương của n.

CHÚ THÍCH Nếu chuỗi dữ liệu là trống, Phương pháp Đệm 2 quy định rằng nó được đệm bên phải bằng một bit '1' duy nhất, kèm theo sau là n-1 bit '0'.

6.3.4 Phương pháp Đệm 3

Chuỗi dữ liệu D là đầu vào của thuật toán MAC phải được đệm bên phải bằng một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một chuỗi dữ liệu mà độ dài của nó (theo bit) là một bội nguyên dương của n. Chuỗi kết quả sau đó được đệm bên trái bởi một khối L. Khối L bao gồm biểu diễn nhị phân của độ dài L_D (theo bit) của chuỗi dữ liệu chưa được đệm D, được đệm bên trái bằng một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một khối n-bit. Bit bên phải nhất của khối L tương ứng với bit có nghĩa nhỏ nhất của biểu diễn nhị phân của L_D.

CHÚ THÍCH 1 Phương pháp Đệm 3 không thích hợp để sử dụng trong các tình huống trong đó độ dài của chuỗi dữ liệu không thể biết trước trước khi bắt đầu tính MAC.

CHÚ THÍCH 2 Nếu chuỗi dữ liệu là rỗng, Phương pháp Đệm 3 quy định rằng nó được đệm bên phải bằng n bit '0' và được đệm bên trái bởi một khối L bao gồm n bit '0'.

6.3.5 Phương pháp Đệm 4

Nếu chuỗi dữ liệu D là đầu vào của thuật toán MAC có độ dài (theo bit) là một bội nguyên dương của n, thì không áp dụng việc đệm. Ngược lại, chuỗi dữ liệu D phải được đệm bên phải bằng một bit '1' duy nhất. Chuỗi kết quả sau đó sẽ được đệm bên phải bằng một số ít nhất các bit '0' như cần thiết (có thể không có) để nhận được một chuỗi dữ liệu mà độ dài của nó (theo bit) là một bội nguyên dương của n.

CHÚ THÍCH Nếu chuỗi dữ liệu là rỗng, Phương pháp Đệm 4 quy định rằng nó được đệm bên phải bằng một bit '1' duy nhất, kèm theo sau là n-1 bit '0'.

6.4 Bước 3 (phân chia)

Phiên bản đã được đệm của chuỗi dữ liệu D được phân chia thành q khối n-bit $D_1, D_2, \ldots, D_q$. Ở đây D_1 biểu diễn n bit đầu tiên của phiên bản đã được đệm của D_1, D_2 biểu diễn n bit tiếp theo, và cứ tiếp tục như thế.

6.5 Bước 4 (lặp)

Các khối H_1, H_2, .., H_{q-1} được tính bằng cách áp dụng lặp mã khối cùng với khóa mã khối K vào phép XOR theo từng bit của khối dữ liệu D_i và kết quả trước đó H_{i-1}:

$H_0 := 0$;

For i from 1 to q - 1:

$H_i := e_K(D_i \oplus H_{i-1})$;

Nếu q bằng 1, Bước 4 phải được bỏ qua.

CHÚ THÍCH Phép toán này tương ứng với chế độ CBC (Móc xích Khối mã- Cipher Block Chaining) cùng với biến bắt đầu được cố định bằng 0^n, như đã được định nghĩa trong ISO/IEC 10116 [7].

6.6 Bước 5 (lặp lần cuối)

6.6.1 Tổng quan

Phép lặp lần cuối F được áp dụng vào khối cuối cùng D_q của chuỗi dữ liệu đã được đệm để nhận được khối H_q.

Mỗi một thuật toán trong số 6 thuật toán MAC được quy định trong tiêu chuẩn này sử dụng một trong ba kiểu phép lặp lần cuối.

6.6.2 Phép lặp Lần cuối kiểu 1

Phép biến đổi này dùng cùng khóa mã khối K như trong phép lặp. Khối H_q được tính bằng cách áp dụng mã khối cùng với khóa K như sau:

$H_q := e_K(D_q \oplus H_{q-1})$.

6.6.3 Phép lặp Lần cuối kiểu 2

Phép biến đổi này sử dụng khóa mã khối K', nó là khác với khóa mã khối K đã được sử dụng trong phép lặp. Khối H_q được tính bằng cách áp dụng mã khối cùng với khóa K' như sau:

$H_q := e_{K'}(D_q \oplus H_{q-1})$.

6.6.4 Phép lặp Lần cuối kiểu 3

Phép biến đổi này sử dụng cùng khóa mã khối K như trong phép lặp và hai khóa che giấu K_1 và K_2 có độ dài n. Khối H_q được tính bằng cách XOR đầu vào với khóa K_1 hoặc K_2 phụ thuộc vào phép toán đệm, sau đó là phép mã hóa bằng khóa mã khối K.

Theo Phương pháp Đệm 4, nếu chuỗi dữ liệu là đầu vào của thuật toán MAC có độ dài (theo bit) là một bội nguyên dương của n thì:

$H_q := e_K(D_q \oplus H_{q-1} \oplus K_1)$;

ngược lại thì

$H_q := e_K(D_q \oplus H_{q-1} \oplus K_2)$.

6.7 Bước 6 (biến đổi đầu ra)

6.7.1 Tổng quan

Phép biến đổi đầu ra g được áp dụng vào giá trị H_q đã nhận như là kết quả của Bước 5.

Tiêu chuẩn này chỉ ra 3 phép biến đổi đầu ra.

6.7.2 Biến đổi Đầu ra 1

Phép biến đổi đầu ra này là hàm đồng nhất, tức là

$G := H_q$.

6.7.3 Biến đổi Đầu ra 2

Phép biến đổi đầu ra này bao gồm việc áp dụng mã khối với khóa mã khối K' vào H_q, tức là

$G := e_{K'}(H_q)$.

6.7.4 Biến đổi Đầu ra 3

Phép biến đổi đầu ra này bao gồm việc áp dụng mã khối (trong chế độ giải mã) cùng với khóa K' vào H_q, được theo sau bởi áp dụng mã khối với khóa K vào kết quả của phép toán này, tức là

$G := e_K(d_{K'}(H_q))$.

6.8 Bước 7 (cắt ngắn)

MAC có m bit sẽ nhận được bằng cách lấy m bit bên trái nhất của khối G, tức là

MAC := $m \sim G$.

7 Các thuật toán MAC

7.1 Tổng quan

Tiêu chuẩn này quy định 6 thuật toán MAC. Phép lần lặp cuối và phép biến đổi đầu ra được quy định trong từng trường hợp.

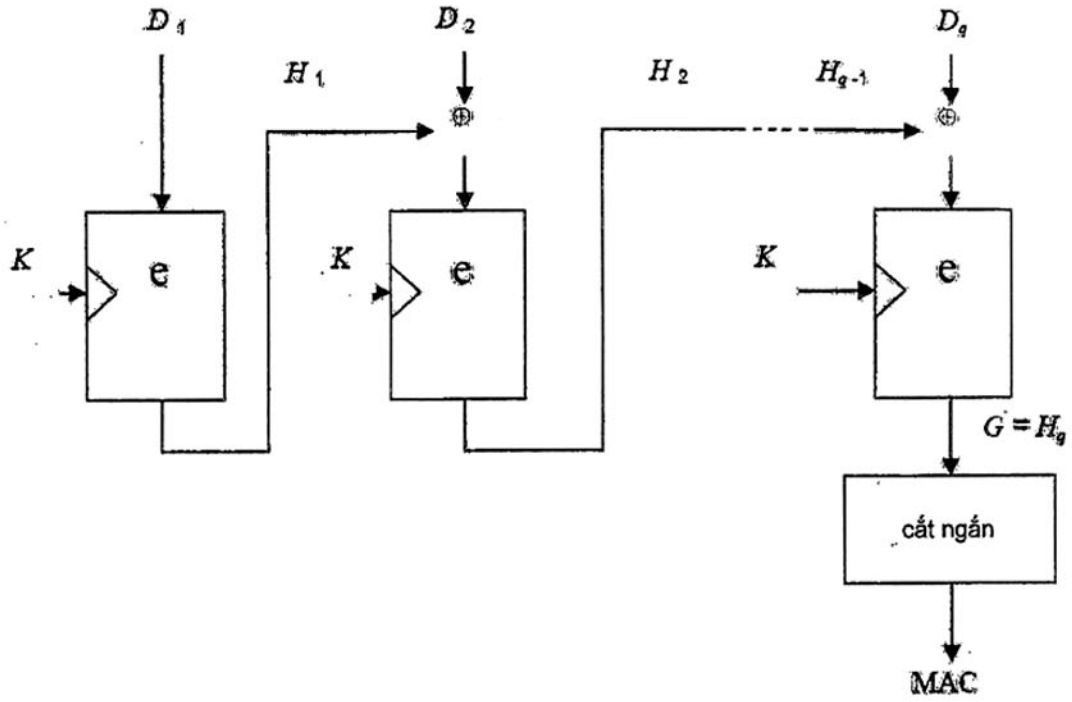
7.2 Thuật toán MAC 1

Thuật toán MAC 1 sử dụng Lặp Lần cuối 1 và Biến đổi Đầu ra 1. Khóa của thuật toán MAC bao gồm khóa mã khối K. Thuật toán MAC 1 được minh họa trong Hình 2.

Thuật toán MAC 1 có thể được sử dụng cùng với Phương pháp Đệm 1, 2 hoặc 3 đã quy định trong Điều 6.3.

CHÚ THÍCH 1 Lựa chọn của phương pháp đệm ảnh hưởng tới độ an toàn của thuật toán MAC. Xem Phụ lục C để biết thêm các chi tiết.

CHÚ THÍCH 2 Thuật toán MAC 1 là đối tượng của các tấn công giả mạo XOR (xem Phụ lục C). Như một kết quả thuật toán này chỉ nên được sử dụng khi các tấn công như vậy là không thể, ví dụ trong trường hợp độ dài các thông điệp là cố định.



Hình 2 – Thuật toán MAC 1

7.3 Thuật toán MAC 2

Thuật toán MAC 2 sử dụng Lặp Lần cuối 1 và Biến đổi Đầu ra 2. Khóa của thuật toán MAC bao gồm 2 khóa mã khối *K* và *K'*. Các giá trị của *K* và *K'* có thể được dẫn xuất ra từ một khóa chủ chung (một khóa mã khối) theo một cách sao cho *K* và *K'* là khác nhau với xác suất rất cao.

CHÚ THÍCH 1 Thuật toán MAC 2 thường được biết là EMAC [24].

CHÚ THÍCH 2 Một ví dụ về cách để dẫn xuất *K* và *K'* từ một khóa chủ chung là Phương pháp Dẫn xuất Khóa 1.

CHÚ THÍCH 3 Nếu *K* và *K'* là bằng nhau, có thể bị tấn công giả mạo XOR đơn giản. Xem Phụ lục C để biết thêm chi tiết.

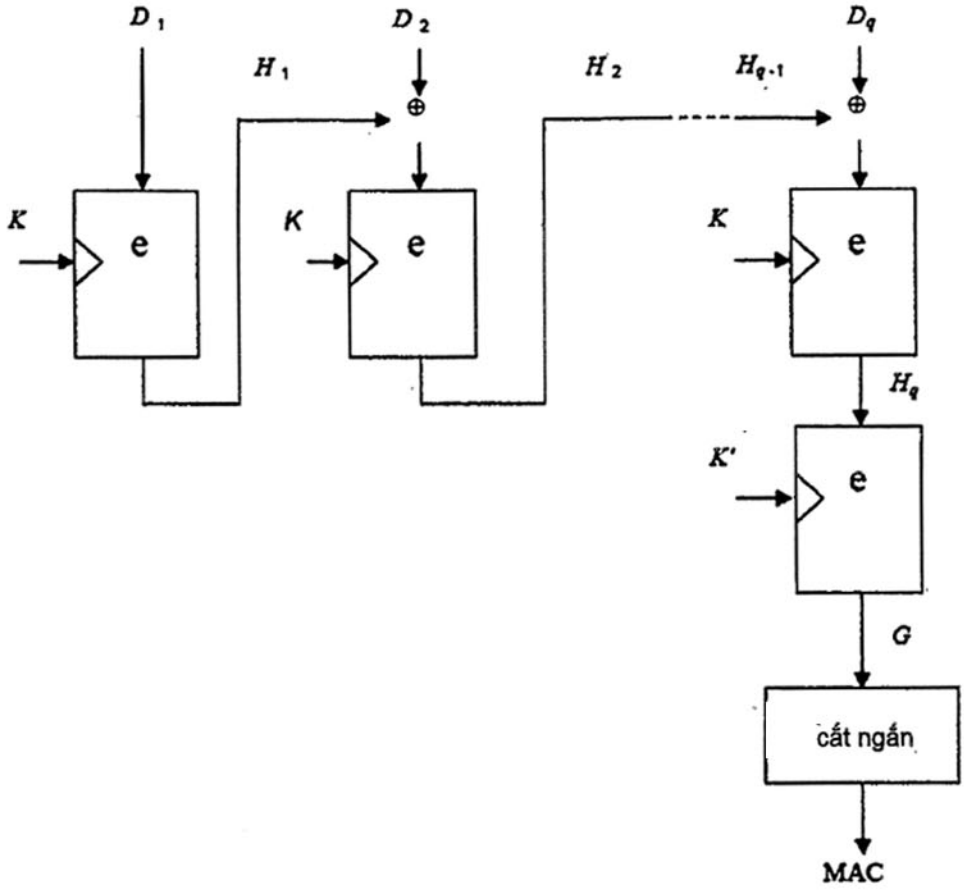
CHÚ THÍCH 4 Nếu *K* và *K'* là độc lập, mức an toàn chống lại các tấn công khôi phục khóa là nhỏ hơn so với đã được đề xuất bởi kích thước khóa của thuật toán MAC. Xem Phụ lục C để biết thêm chi tiết.

Thuật toán MAC 2 được minh họa trong Hình 3.

Thuật toán MAC 2 có thể được sử dụng cùng Phương pháp Đệm 1, 2 hoặc 3 như đã quy định trong Điều 6.3.

CHÚ THÍCH 5 Lựa chọn của phương pháp đệm có ảnh hưởng tới độ an toàn của thuật toán MAC. Xem Phụ lục C để biết thêm chi tiết.

CHÚ THÍCH 6 Nếu Thuật toán MAC 2 được sử dụng kết hợp với một thuật toán mà tính định danh khóa (công khai) như $S = e_K(0^n)$, chẳng hạn như X9.24 [13], thì Thuật toán MAC 2 là đối tượng của các tấn công giả mạo XOR (xem Phụ lục C). Trong trường hợp này thuật toán chỉ nên được sử dụng khi các tấn công như vậy là không thể, ví dụ bởi vì các độ dài thông điệp là cố định.



Hình 3 – Thuật toán MAC 2

7.4 Thuật toán MAC 3

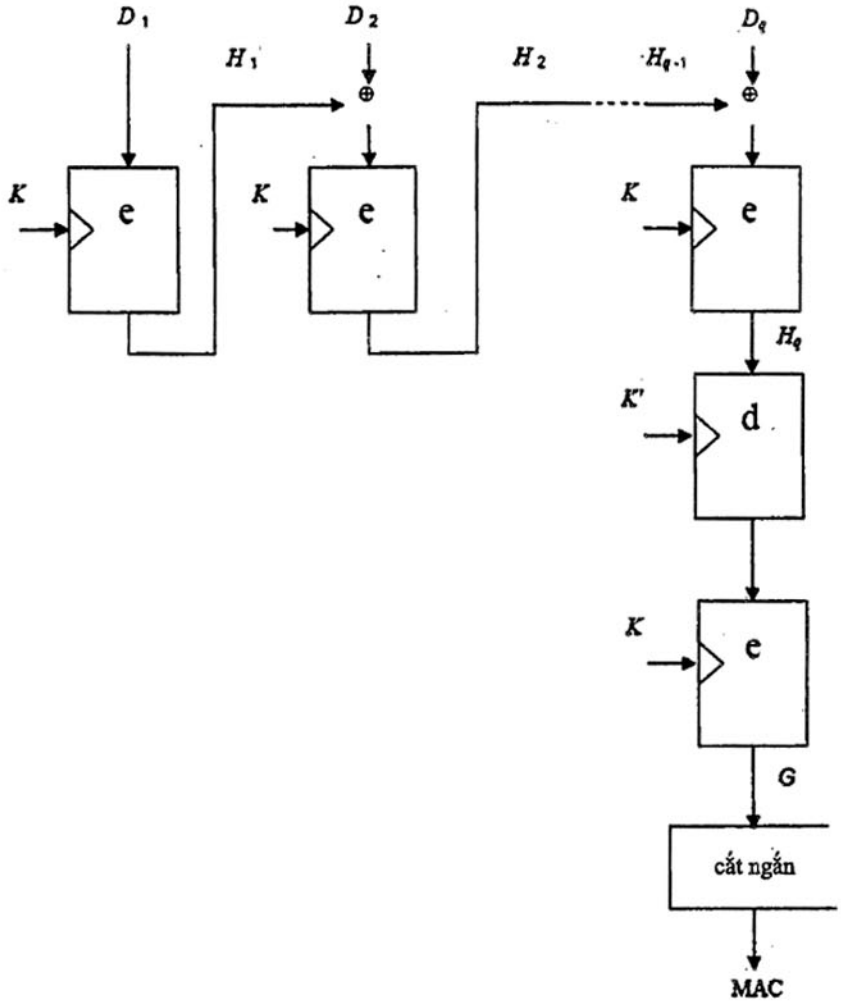
Thuật toán MAC 3 sử dụng Lặp Lần cuối 1 và Biến đổi Đầu ra 3. Khóa của thuật toán MAC bao gồm 2 khóa mã khối *K* và *K'*. Các giá trị của *K* và *K'* cần phải được chọn một cách độc lập. Nếu *K* = *K'*, Thuật toán MAC 3 quay trở về Thuật toán MAC 1. Thuật toán MAC 3 được minh họa trong Hình 4.

CHÚ THÍCH 1 Thuật toán MAC 3 thường được biết đến như MAC cho lĩnh vực bán lẻ của ANSI [12]. Khi Thuật toán MAC 3 được sử dụng cùng với DEA (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]), độ dài khóa của mã khối là 56 bit, trong khi độ dài khóa là 112 bit.

Thuật toán MAC 3 có thể được sử dụng cùng với Phương pháp Đệm 1, 2 hoặc 3 như đã quy định trong Điều 6.3.

CHÚ THÍCH 2 Việc lựa chọn của phương pháp đệm có ảnh hưởng tới độ an toàn của thuật toán MAC. Xem Phụ lục C để biết thêm chi tiết.

CHÚ THÍCH 3 Nếu Thuật toán MAC 3 được sử dụng kết hợp với thuật toán mà tính ra định danh khóa (công khai) bằng $S = e_K(0^n)$ chẳng hạn như X9.24 [13] thì Thuật toán MAC 3 là đối tượng của các tấn công giả mạo XOR (xem Phụ lục C). Trong trường hợp này thuật toán chỉ nên được sử dụng khi các tấn công như vậy là không thể, ví dụ trong trường hợp độ dài các thông điệp là cố định.



Hình 4 – Thuật toán MAC 3

7.5 Thuật toán MAC 4

Thuật toán MAC 4 sử dụng Lặp Lần cuối 1 và Biến đổi Đầu ra 2. Ngoài ra, Thuật toán MAC 4 thay đổi quá trình xử lý cho khối đầu tiên.

CHÚ THÍCH 1 Khi được sử dụng cùng với DEA (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]), Thuật toán MAC 4 được biết đến như MacDES.[21]. Trong trường hợp này, độ dài khóa của mã khối là 56 bit, trong khi độ dài khóa là 112 bit.

Khóa của thuật toán MAC bao gồm 2 khóa mã khối K và K', và cần được chọn một cách độc lập. Khóa mã khối thứ ba K'' phải được dẫn xuất từ K'. Các giá trị của K, K' và K'' phải khác nhau. Các khóa mã khối K và K'' được sử dụng trong quá trình xử lý cho khối thứ nhất, và các khóa mã khối K và K' được sử dụng cùng với phép Biến đổi Đầu ra 2.

CHÚ THÍCH 2 Một ví dụ của cách dẫn xuất ra K'' từ K là lấy đảo bit các chuỗi con luân phiên có 4 bit của K' bắt đầu từ 4 bit đầu tiên. Ví dụ khác để dẫn xuất cả K' và K'' từ một khóa chủ chung như đã quy định trong Phương pháp Dẫn xuất Khóa 1.

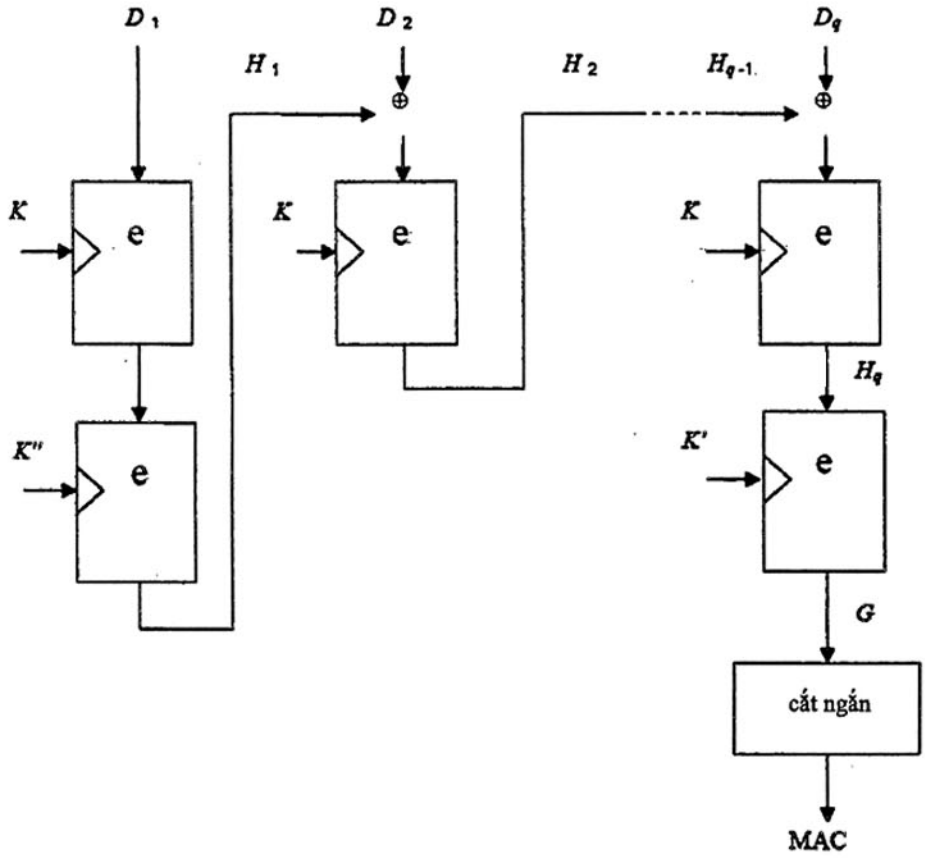
Khối đầu tiên không được xử lý bằng phép lặp thông thường (với một lần mã) mà thay vào sử dụng phương trình sau (với hai lần mã):

$H_1 := e_K(e_K(D_1))$.

Thuật toán MAC 4 được minh họa trong Hình 5.

Thuật toán MAC 4 có thể được sử dụng cùng với Phương pháp Đệm 1, 2 hoặc 3 đã quy định trong Điều 6.3.

CHÚ THÍCH 3 Lựa chọn của phương pháp đệm có ảnh hưởng tới độ an toàn của thuật toán MAC. Xem Phụ lục C để biết thêm chi tiết.



Hình 5 – Thuật toán MAC 4

7.6 Thuật toán MAC 5

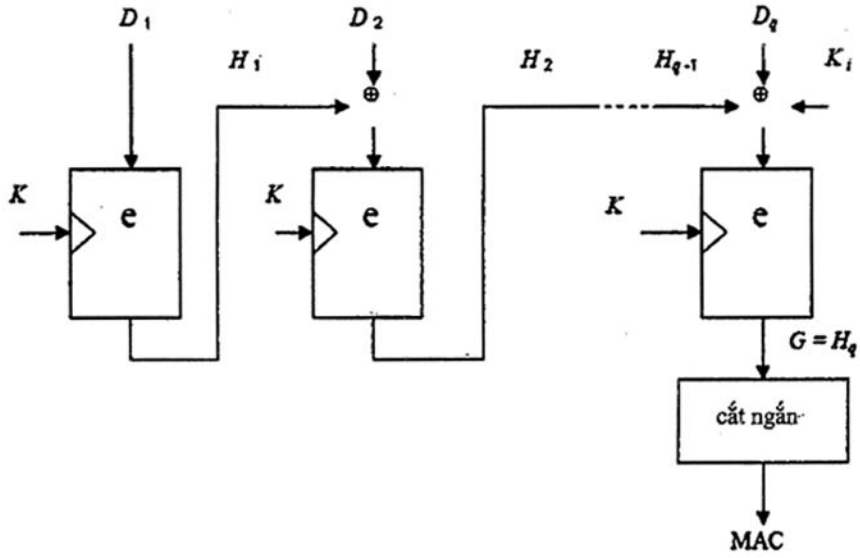
Thuật toán MAC 5 sử dụng Phương pháp Dẫn xuất Khóa 2, phép Lặp Lần cuối 3 và phép Biến đổi Đầu ra 1. Thuật toán MAC 5 chỉ được sử dụng cùng Phương pháp Đệm 4. Các khóa che giấu K_1 và K_2 mà được sử dụng trong phép Lặp Lần cuối 3 là được dẫn xuất từ khóa thuật toán MAC K dùng Phương pháp Dẫn xuất Khóa 2.

CHÚ THÍCH 1 Thuật toán MAC 5 thường được biết đến là OMAC1 [19] hoặc CMAC [14].

Khóa của thuật toán MAC bao gồm một khóa mã khối K đơn lẻ.

Thuật toán MAC 5 được minh họa trong Hình 6, trong đó $K_l = K_1$ hoặc K_2.

CHÚ THÍCH 2 Nếu Thuật toán MAC 5 được sử dụng kết hợp với thuật toán mà tính ra định danh khóa (công khai) là $S = e_K(0^n)$ chẳng hạn như X9.24 [13] thì Thuật toán MAC 5 là đối tượng của các tấn công giả mạo XOR (xem Phụ lục C). Trong trường hợp này thuật toán chỉ nên được sử dụng khi các tấn công như vậy là không thể, ví dụ trong trường hợp độ dài các thông điệp là cố định.



Hình 6 – Thuật toán MAC 5

7.7 Thuật toán MAC 6

Thuật toán MAC 6 sử dụng Lặp Lần cuối 2 và Biến đổi Đầu ra 1. Khóa của thuật toán MAC bao gồm 2 khóa của mã khối K và K'. Các giá trị của K và K' có thể được rút ra từ một khóa chủ chung (một khóa mã khối) theo một cách sao cho K và K' là khác nhau với xác suất rất cao.

CHÚ THÍCH 1 Thuật toán MAC 6 thường được biết đến là LMAC.

CHÚ THÍCH 2 Một ví dụ về cách để dẫn xuất K và K' từ một khóa chủ chung là Phương pháp Dẫn xuất Khóa 1.

CHÚ THÍCH 3 Nếu K và K' bằng nhau, tấn công giả mạo XOR được áp dụng. Xem thông tin chi tiết tại Phụ lục C.

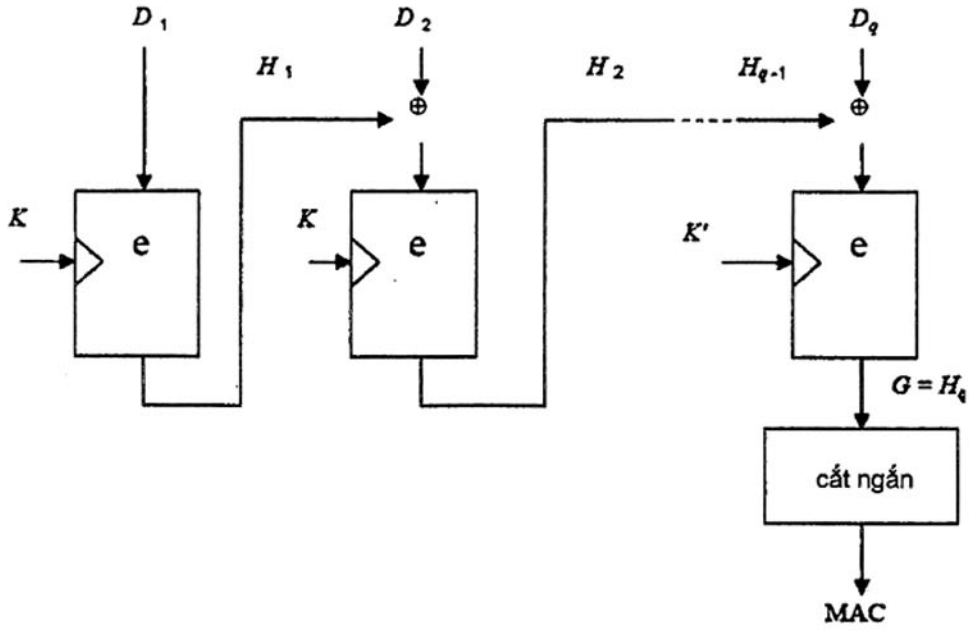
CHÚ THÍCH 4 Nếu K và K' là độc lập, mức an toàn chống lại các tấn công khôi phục khóa là nhỏ hơn so với đã được đề xuất bởi kích thước khóa thuật toán MAC. Xem thông tin chi tiết tại Phụ lục C.

Thuật toán MAC 6 được minh họa trong Hình 7.

Thuật toán MAC 6 có thể được sử dụng cùng với Phương pháp Đệm 1, 2 hoặc 3 đã được quy định trong Điều 6.3.

CHÚ THÍCH 5 Việc lựa chọn phương pháp đệm có ảnh hưởng tới độ an toàn của thuật toán MAC. Xem thông tin chi tiết tại Phụ lục C.

CHÚ THÍCH 6 Nếu Thuật toán MAC 6 được sử dụng kết hợp với thuật toán mà tính ra định danh khóa (công khai) là $S = e_K(0^n)$ chẳng hạn như X9.24 [13] thì Thuật toán MAC 6 là đối tượng của các tấn công giả mạo XOR (xem Phụ lục C). Trong trường hợp này thuật toán chỉ nên được sử dụng khi các tấn công như vậy là không thể, ví dụ trong trường hợp độ dài các thông điệp là cố định.



Hình 7 – Thuật toán MAC 6.

Phụ lục A
(quy định)
Các định danh đối tượng

```
MessageAuthenticationCodesPart1 {
      iso(1) standard(0) message-authentication-codes(9797) part(1)
            asn1-module(0) algorithm-object-identifiers(0)}
DEFINITIONS
EXPLICIT TAGS ::=
BEGIN
IMPORTS
      ALGORITHM, BlockAlgorithms
            FROM EncryptionAlgorithms-3 {iso(1) standard(0)
                  encryption-algorithms(18033) part(3)
                  asn1-module(0) algorithm-object-identifiers(0)};
OID ::= OBJECT IDENTIFIER
BASE-OID ::= OID
-- gán OID
-- ==============
is9797-1 OID ::= {iso standard message-authentication-codes(9797) part1(1)}
id-kdm BASE-OID ::= {is9797-1 keyDerivationMethod(1)}
id-pm BASE-OID ::= {is9797-1 padMethod(2)}
id-ma BASE-OID ::= {is9797-1 macAlgo(3)}
-- giải thích quy định:
-- kết nối của các OID có liên quan và OID cơ sở id-kdm chỉ ra một định
-- danh đối tượng đầy đủ cho mỗi một trong số phương pháp dẫn xuất khóa
-- đã được chỉ ra để có thể sử dụng trong các tài liệu khác
id-kdm-1 RELATIVE-OID ::= {1}
id-kdm-2 RELATIVE-OID ::= {2}
-- giải thích quy định:
-- kết nối của các OID có liên quan và OID cơ sở id-pm chỉ ra một định
-- danh đối tượng đầy đủ cho mỗi một trong số phương pháp dẫn xuất khóa
-- đã được chỉ ra để có thể sử dụng trong các tài liệu khác
id-pad-1 RELATIVE-OID ::= {1}
id-pad-2 RELATIVE-OID ::= {2}
id-pad-3 RELATIVE-OID ::= {3}
id-pad-4 RELATIVE-OID ::= {4}
id-mac-1 OID ::= {id-ma 1}
```

```
id-mac-2 OID ::= {id-ma 2}
id-mac-3 OID ::= {id-ma 3}
id-mac-4 OID ::= {id-ma 4}
id-mac-5 OID ::= {id-ma 5}
id-mac-6 OID ::= {id-ma 6}
-- kiểu định danh thuật toán MAC và tập các thuật toán MAC đã nhận diện
-- ========================================================================
MessageAuthenticationCode ::= AlgorithmIdentifier {{ MacAlgorithms }}
MacAlgorithms ALGORITHM ::= {
{ OID id-mac-1 PARMS MacParameters-1 } |
{ OID id-mac-2 PARMS MacParameters-2 } |
{ OID id-mac-3 PARMS MacParameters-3 } |
{ OID id-mac-4 PARMS MacParameters-4 } |
{ OID id-mac-5 PARMS MacParameters-5 } |
{ OID id-mac-6 PARMS MacParameters-6 } ,
... - các thuật toán bổ sung mong chờ --
}
-- các định nghĩa kiểu tham số MAC
-- ==============================
-- cấu trúc này có thể chỉ ra một phương pháp dẫn xuất khóa được định
-- nghĩa bởi ứng dụng
KdAlgo ::= CHOICE {
      specifiedKdAlgo RELATIVE-OID,
      generalKdAlgo         OID
}
-- một cấu trúc tham số được sử dụng ở 5 trong số sáu thuật toán MAC
-- các tham số tùy chọn hoặc không được sử dụng (ví dụ thuật toán MAC
-- 1 không sử dụng một phương pháp dẫn xuất khóa) hoặc có thể được thỏa
-- thuận bởi các cách khác
MacParameters ::= SEQUENCE {
      bcAlgo BlockCipher OPTIONAL,
      padAlgo [0] RELATIVE-OID ({1}|{2}|{3}) OPTIONAL,
      kdAlgo [1] KdAlgo OPTIONAL,
      m INTEGER (1..MAX)
}
MacParameters-1 ::= MacParameters
MacParameters-2 ::= MacParameters
MacParameters-3 ::= MacParameters
MacParameters-4 ::= MacParameters
```

```
-- đối với thuật toán MAC 5 thì phương pháp đệm và phương pháp dẫn xuất
-- khóa là cố định
MacParameters-5 ::= SEQUENCE {
      bcAlgo BlockCipher OPTIONAL,
      m INTEGER (1..MAX)
}
MacParameters-6 ::= MacParameters
-- các định nghĩa phụ
-- ====================
-- định nghĩa của định danh thuật toán mã khối
BlockCipher ::= AlgorithmIdentifier {{BlockAlgorithms}}
AlgorithmIdentifier {ALGORITHM:IOSet} ::= SEQUENCE {
      algorithm ALGORITHM.&id({IOSet}),
      parameters ALGORITHM.&Type({IOSet}{@algorithm}) OPTIONAL
}
END -- MessageAuthenticationCodes --
```

Phụ lục B
(tham khảo)
Các ví dụ

B.1 Tổng quan

Phụ lục này trình bày các ví dụ của việc sinh ra MAC.

Đối với các Thuật toán MAC 1-4, các bản rõ là các mã ASCII 7-bit (không có bit tính chẵn lẻ) cho chuỗi dữ liệu 1: "Now□is□the□time□for□all□" và chuỗi dữ liệu 2: "Now□is□the□time□for□it", trong đó "□" ký hiệu một dấu cách. Mã hóa ASCII là tương đương với mã hóa dùng ISO 646. Tất cả các giá trị MAC và các giá trị khóa được viết ở ký tự hexa.

Đối với chuỗi dữ liệu 1, các kết quả của việc áp dụng các phương pháp đệm 1-3 là như sau:

– Phương pháp Đệm 1: q = 3

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 61 6C 6C 20

– Phương pháp Đệm 2: q = 4

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 61 6C 6C 20
D_4	80 00 00 00 00 00 00 00

– Phương pháp Đệm 3: q = 4

D_1	00 00 00 00 00 00 00 C0
D_2	4E 6F 77 20 69 73 20 74
D_3	68 65 20 74 69 6D 65 20
D_4	66 6F 72 20 61 6C 6C 20

Đối với chuỗi dữ liệu 2, các kết quả của việc áp dụng các phương pháp đệm 1 đến 3 là như sau:

– Phương pháp Đệm 1: q = 3

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 69 74 00 00

– Phương pháp Đệm 2: q = 3

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 69 74 80 00

– Phương pháp Đệm : q = 4

D_1	00 00 00 00 00 00 00 B0
D_2	4E 6F 77 20 69 73 20 74
D_3	68 65 20 74 69 6D 65 20
D_4	66 6F 72 20 69 74 00 00

B.2 Thuật toán MAC 1

Các ví dụ được đưa ra sử dụng DEA như mã khối (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]). Giá trị khóa được sử dụng là K = 0123456789ABCDEF (hexa). Độ dài m theo bit của MAC là bằng 32.

– Chuỗi dữ liệu 1 với Phương pháp Đệm 1

khóa (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 EC A9 E9 4A
$G = H_3$	70 A3 06 40 CC 76 DD 8B

MAC = 70 A3 06 40

– Chuỗi dữ liệu 1 với Phương pháp Đệm 2

khóa (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 EC A9 E9 4A
H_3	70 A3 06 40 CC 76 DD 8B
$D_4 \oplus H_3$	F0 A3 06 40 CC 76 DD 8B
$G = H_4$	10 E1 F0 F1 08 34 1B 6D

MAC = 10 E1 F0 F1

– Chuỗi dữ liệu 1 với Phương pháp Đệm 3

khóa (K)	01 23 45 67 89 AB CD EF
H_1	4B B5 82 65 DD 87 B3 05
$D_2 \oplus H_1$	05 DA F5 45 B4 F4 93 71
H_2	40 C4 00 AD 74 2E 4F D6
$D_3 \oplus H_2$	28 A1 20 D9 1D 43 2A F6
H_3	23 7D 5F 95 0B F7 1F 57
$D_4 \oplus H_3$	45 12 2D B5 6A 9B 73 77
$G = H_4$	2C 58 FB 8F F1 2A AE AC

MAC = 2C 58 FB 8F

– Chuỗi dữ liệu 2 với Phương pháp Đệm 1

khóa (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 E4 B1 85 6A
$G = H_3$	E4 5B 3A D2 B7 CC 08 56

MAC = E4 5B 3A D2

– Chuỗi dữ liệu 2 với Phương pháp Đệm 2

khóa (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 E4 B1 85 6A
$G = H_3$	A9 24 C7 21 36 14 92 11

MAC = A9 24 C7 21

– Chuỗi dữ liệu 2 với Phương pháp Đệm 3

khóa (K)	01 23 45 67 89 AB CD EF
H_1	DF 9C D6 EA 7E 5A E1 62
$D_2 \oplus H_1$	91 F3 A1 CA 17 29 C1 16
H_2	C7 6F B0 02 94 A4 19 BE
$D_3 \oplus H_2$	AF 0A 90 76 FD C9 7C 9E
H_3	83 02 28 FD 78 D7 BE 71
$D_4 \oplus H_3$	E5 6D 5A DD 11 A3 BE 71
$G = H_4$	B1 EC D6 FC 8B 37 C3 92

MAC = B1 EC D6 FC

B.3 Thuật toán MAC 2

Các ví dụ được đưa ra sử dụng DEA như mã khối (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]). Hai giá trị khóa được sử dụng là K = 0123456789ABCDEF (hexa), và K' được tính bằng cách lấy bù luân phiên các chuỗi con 4 bit bắt đầu từ 4 bit đầu tiên. Độ dài m theo bit của MAC là bằng 32.

Các q bước đầu tiên là giống với các bước của Thuật toán MAC 1. Điểm khác duy nhất là ở chỗ Biến đổi Đầu ra 2 được áp dụng thay cho Biến đổi Đầu ra 1.

– Chuỗi dữ liệu 1 với Phương pháp Đệm 1

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	10 F9 BC 67 A0 3C D5 D8

MAC = 10 F9 BC 67

– Chuỗi dữ liệu 1 với Phương pháp Đệm 2

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	BE 7C 2A B7 D3 6B F5 B7

MAC = BE 7C 2A B7

– Chuỗi dữ liệu 1 với Phương pháp Đệm 3

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	8E FC 8B C7 C2 72 6E 5C

MAC = 8E FC 8B C7

– Chuỗi dữ liệu 2 với Phương pháp Đệm 1

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	21 5E 9C E6 D9 1B C7 FB

MAC = 21 5E 9C E6

– Chuỗi dữ liệu 2 với Phương pháp Đệm 2

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	17 36 AC 1A 63 63 0E FB

MAC = 17 36 AC 1A

– Chuỗi dữ liệu 2 với Phương pháp Đệm 3

khóa (*K'*)	F1 D3 B5 97 79 5B 3D 1F
G	05 38 26 96 27 4F B4 F0

MAC = 05 38 26 96

B.4 Thuật toán MAC 3

Các ví dụ được đưa ra sử dụng DEA như mã khối (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]). Hai giá trị khóa được sử dụng là *K* = 0123456789ABCDEF (hexa), và *K'* = FEDCBA9876543210 (hexa). Độ dài *m* theo bit của MAC là bằng 32.

Các *q* bước đầu tiên là giống với các bước của Thuật toán MAC 1. Điểm khác duy nhất là ở chỗ Biến đổi Đầu ra 3 được áp dụng thay cho Biến đổi Đầu ra 1.

– Chuỗi dữ liệu 1 với Phương pháp Đệm 1

khóa (*K'*)	FE DC BA 98 76 54 32 10
đầu ra của *d*	B4 8D 36 EC 7A D5 69 4F
G	A1 C7 2E 74 EA 3F A9 B6

MAC = A1 C7 2E 7

– Chuỗi dữ liệu 1 với Phương pháp Đệm 2

khóa (K')	FE DC BA 98 76 54 32 10
đầu ra của d	79 53 7F EE 18 CF 18 93
G	E9 08 62 30 CA 3B E7 96

MAC = E9 08 62 30

– Chuỗi dữ liệu 1 với Phương pháp Đệm 3

khóa (K')	FE DC BA 98 76 54 32 10
đầu ra của d	FE B3 B9 66 1D BE DE CD
G	AB 05 94 63 D7 A7 D1 70

MAC = AB 05 94 63

– Chuỗi dữ liệu 2 với Phương pháp Đệm 1

khóa (K')	FE DC BA 98 76 54 32 10
đầu ra của d	32 8A C7 8B A1 CA 0B 3F
G	2E 2B 14 28 CC 78 25 4F

MAC = 2E 2B 14 28

– Chuỗi dữ liệu 2 với Phương pháp Đệm 2

khóa (K')	FE DC BA 98 76 54 32 10
đầu ra của d	7A 71 AF 2F 5D 15 40 A7
G	5A 69 2C E6 4F 40 41 45

MAC = 5A 69 2C E6

– Chuỗi dữ liệu 2 với Phương pháp Đệm 3

khóa (K')	FE DC BA 98 76 54 32 10
đầu ra của d	20 97 B4 05 F1 9E 2D D8
G	C5 9F 7E ED 32 8D DD 69

MAC = C5 9F 7E ED

B.5 Thuật toán MAC 4

Các ví dụ được đưa ra sử dụng DEA như mã khối (được quy định trong Phụ lục A của TCVN 11367-3 (ISO/IEC 18033-3) và ANSI X3.92 [10]). Hai giá trị khóa được sử dụng là K = 0123456789ABCDEF (hexa), và K' = FEDCBA9876543210 (hexa). Các khóa dẫn xuất được tính bằng cách lấy bù luân phiên các chuỗi con 4 bit bắt đầu từ 4 bit đầu tiên. Độ dài m theo bit của MAC là bằng 32.

– Chuỗi dữ liệu 1 với Phương pháp Đệm 1

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	3F A4 0E 8A 98 4D 48 15
H_1	EA F0 4B F5 31 ED 33 5E
$D_2 \oplus H_1$	82 95 6B 81 58 80 56 7E
H_2	7E 7F 98 A0 C8 B1 65 6C
$D_3 \oplus H_2$	18 10 EA 80 A9 DD 09 4C
H_3	7B 93 0A AE 67 4A C9 24
G	AD 35 02 B7 AC 4A 48 A0

MAC = AD 35 02 B7

– Chuỗi dữ liệu 1 với Phương pháp Đệm 2

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	3F A4 0E 8A 98 4D 48 15
H_1	EA F0 4B F5 31 ED 33 5E
$D_2 \oplus H_1$	82 95 6B 81 58 80 56 7E
H_2	7E 7F 98 A0 C8 B1 65 6C
$D_3 \oplus H_2$	18 10 EA 80 A9 DD 09 4C
H_3	7B 93 0A AE 67 4A C9 24
$D_3 \oplus H_3$	FB 93 0A AE 67 4A C9 24
H_4	26 C4 FA D7 2E 6D D3 A2
G	61 C3 33 E3 42 C5 53 7C

MAC = 61 C3 33 E3

– Chuỗi dữ liệu 1 với Phương pháp Đệm 3

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	4B B5 82 65 DD 87 B3 05
H_1	71 5A F8 BE DA BE 90 44
$D_2 \oplus H_1$	3F 35 8F 9E B3 CD B0 30
H_2	50 2A 04 42 6A 80 B6 0B
$D_3 \oplus H_2$	38 4F 24 36 03 ED D3 2B
H_3	AF 13 8C 54 99 9B 84 30
$D_3 \oplus H_3$	C9 7C FE 74 F8 F7 E8 10
H_4	7F 90 05 61 B4 2C CE D2
G	95 2A F8 38 98 9B 5C 00

MAC = 95 2A F8 38

– Chuỗi dữ liệu 2 với Phương pháp Đệm 1

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	3F A4 0E 8A 98 4D 48 15
H_1	EA F0 4B F5 31 ED 33 5E
$D_2 \oplus H_1$	82 95 6B 81 58 80 56 7E
H_2	7E 7F 98 A0 C8 B1 65 6C
$D_3 \oplus H_2$	18 10 EA 80 A1 C5 65 6C
H_3	21 FC 35 F2 B2 26 6C 9A
G	05 F1 08 4C 1D E3 A3 3D

MAC = 05 F1 08 4C

– Chuỗi dữ liệu 2 với Phương pháp Đệm 2

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	3F A4 0E 8A 98 4D 48 15
H_1	EA F0 4B F5 31 ED 33 5E
$D_2 \oplus H_1$	82 95 6B 81 58 80 56 7E
H_2	7E 7F 98 A0 C8 B1 65 6C
$D_3 \oplus H_2$	18 10 EA 80 A1 C5 65 6C
H_3	8F 76 9B 55 48 42 23 FD
G	A1 BC 09 31 52 BB 3E 0F

MAC = A1 BC 09 31

– Chuỗi dữ liệu 2 với Phương pháp Đệm 3

khóa (K)	01 23 45 67 89 AB CD EF
khóa (K')	FE DC BA 98 76 54 32 10
khóa (K'')	0E 2C 4A 68 86 A4 C2 E0
đầu ra của e	DF 9C D6 EA 7E 5A E1 62
H_1	82 61 94 52 C7 6D 04 F1
$D_2 \oplus H_1$	CC 0E E3 72 AE 1E 24 85
H_2	ED 33 1C 07 37 D6 B8 26
$D_3 \oplus H_2$	85 56 3C 73 5E BB DD 06
H_3	7C A1 DE 70 BB 1F 7F 07
$D_3 \oplus H_3$	1A CE AC 50 D2 6B 7F 07
H_4	40 B7 45 2E F3 CF 71 49
G	AF DE E0 F9 50 39 66 3D

MAC = AF DE E0 F9

B.6 Thuật toán MAC 5

B.6.1 Các ví dụ của quá trình sinh MAC

Tại đây cung cấp 10 ví dụ của quá trình sinh MAC cho thuật toán này. Mã khối cơ sở trong các ví dụ này hoặc là AES hoặc bội ba DEA (TDEA) (cả hai được quy định trong TCVN 11367-3 (ISO/IEC 18033-3)). Hai ví dụ được cung cấp về từng độ dài khóa có thể có đối với hai phép mã hóa này, tức là 128, 196 và 256 bit cho AES và mã hóa TDEA dùng 2 khóa và 3 khóa. Trong mỗi cặp ví dụ, việc tính MAC cho hai thông điệp khác nhau được đề cập, cả hai sử dụng cùng một khóa. Việc sinh các khóa che giấu K_1 và K_2 từ khóa K được chỉ ra trong mỗi trường hợp, theo sau là 2 ví dụ sinh MAC.

Tất cả các chuỗi được biểu diễn ở dạng ký hiệu hexa.

B.6.2 AES dùng khóa 128-bit

Khóa 128-bit sau và các khóa che giấu dẫn xuất tương ứng được sử dụng trong cả hai ví dụ:

khóa (K)	2B 7E 15 16 28 AE D2 A6	AB F7 15 88 09 CF 4F 3C
$S = e_K(0^{128})$	7D F7 6B 0C 1A B8 99 B3	3E 42 F0 47 B9 1B 54 6F
K_1	FB EE D6 18 35 71 33 66	7C 85 E0 8F 72 36 A8 DE
K_2	F7 DD AC 30 6A E2 66 CC	F9 0B C1 1E E4 6D 51 3B

Các tính toán MAC là như sau

chuỗi dữ liệu (D)	chuỗi rỗng	
G	BB 1D 69 29 E9 59 37 28	7F A3 7D 12 9B 75 67 46

chuỗi dữ liệu (D)	6B C1 BE E2 2E 40 9F 96	E9 3D 7E 11 73 93 17 2A
G	07 0A 16 B4 6B 4D 41 44	F7 9B DD 9D D0 4A 28 7C

B.6.3 AES dùng khóa 192-bit

Khóa 192-bit sau và các khóa che giấu dẫn xuất tương ứng được sử dụng trong cả hai ví dụ:

khóa (K)	8E 73 B0 F7 DA 0E 64 52	C8 10 F3 2B 80 90 79 E5
	62 F8 EA D2 52 2C 6B 7B	
$S = e_K(0^{128})$	22 45 2D 8E 49 A8 A5 93	9F 73 21 CE EA 6D 51 4B
K_1	44 8A 5B 1C 93 51 4B 27	3E E6 43 9D D4 DA A2 96
K_2	89 14 B6 39 26 A2 96 4E	7D CC 87 3B A9 B5 45 2C

Các tính toán MAC là như sau:

Chuỗi dữ liệu (D)	Chuỗi rỗng	
G	D1 7D DF 46 AD AA CD E5	31 CA C4 83 DE 7A 93 67

Chuỗi dữ liệu (D)	6B C1 BE E2 2E 40 9F 96	E9 3D 7E 11 73 93 17 2A
G	9E 99 A7 BF 31 E7 10 90	06 62 F6 5E 61 7C 51 84

B.6.4 AES dùng khóa 256-bit

Khóa 256-bit sau và các khóa che giấu dẫn xuất tương ứng được sử dụng trong cả hai ví dụ:

khóa (*K*)	60 3D EB 10 15 CA 71 BE	2B 73 AE F0 85 7D 77 81
	1F 35 2C 07 3B 61 08 D7	2D 98 10 A3 09 14 DF F4
$S = e_K(0^{128})$	E5 68 F6 81 94 CF 76 D6	17 4D 4C C0 43 10 A8 54
K_1	CA D1 ED 03 29 9E ED AC	2E 9A 99 80 86 21 50 2F
K_2	95 A3 DA 06 53 3D DB 58	5D 35 33 01 0C 42 A0 D9

Các tính toán MAC là như sau:

Chuỗi dữ liệu (*D*)	chuỗi rỗng	
G	02 89 62 F6 1B 7B F8 9E	FC 6B 55 1F 46 67 D9 83

Chuỗi dữ liệu (*D*)	6B C1 BE E2 2E 40 9F 96	E9 3D 7E 11 73 93 17 2A
G	28 A7 02 3F 45 2E 8F 82	BD 4B F2 8D 8C 37 C3 5C

B.6.5 Bội ba DEA dùng 3 khóa

Khóa sau và các khóa che giấu dẫn xuất tương ứng được sử dụng trong cả hai ví dụ (trong đó *K* là các khóa của bội ba DEA):

khóa (*K*)	8A A8 3B F8 CB DA 10 62
	0B C1 BF 19 FB B6 CD 58
	BC 31 3D 4A 37 1C A8 B5
$S = e_K(0^{64})$	C8 CC 74 E9 8A 73 29 A2
K_1	91 98 E9 D3 14 E6 53 5F
K_2	23 31 D3 A6 29 CC A6 A5

Các tính toán MAC là như sau:

Chuỗi dữ liệu (*D*)	chuỗi rỗng
G	B7 A6 88 E1 22 FF AF 95

chuỗi dữ liệu (*D*)	6B C1 BE E2 2E 40 9F 96
G	8E 8F 29 31 36 28 37 97

B.6.6 Bội ba DEA dùng 2 khóa

Khóa sau và các khóa che giấu dẫn xuất tương ứng được sử dụng trong cả hai ví dụ (trong đó *K* là các khóa của bội ba DEA, với khóa thứ nhất và thứ ba là như nhau):

khóa (*K*)	4C F1 51 34 A2 85 0D D5
	8A 3DS 10 BA 80 57 0D 38
	4C F1 51 34 A2 85 0D D5
$S = e_K(0^{64})$	C7 67 9B 9F 6B 8D 7D 7A

K_1	8E CF 37 3E D7 1A FA EF
K_2	1D 9E 6E 7D AE 35 F5 C5

Các tính toán MAC là như sau:

Chuỗi dữ liệu (*D*)	chuỗi rỗng
G	BD 2E BF 9A 3B A0 03 61

Chuỗi dữ liệu (*D*)	6B C1 BE E2 2E 40 9F 96
G	4F F2 AB 81 3C 53 CE 83

B.7 Thuật toán MAC 6

B.7.1 Các ví dụ của quá trình sinh MAC

Tại đây cung cấp ba ví dụ của quá trình sinh ra MAC cho thuật toán này. Mã khối cơ sở trong các ví dụ này là AES (được quy định trong TCVN 11367-3 (ISO/IEC 18033-3)). Phương pháp Đệm 2 được sử dụng.

Một ví dụ được cung cấp cho từng độ dài khóa có thể có đối với mã pháp này, tức là 128, 192 và 256 bit. Việc sinh ra *K* và *K'* từ một khóa duy nhất *K** được chỉ ra trong mỗi trường hợp, nó được theo sau bởi ví dụ của việc sinh MAC.

Tất cả các chuỗi đều được biểu diễn ở dạng hexa.

B.7.2 AES dùng khóa 128-bit

Các khóa *K* và *K'* sau được sử dụng trong ví dụ (trong đó *K* và *K'* được rút ra từ *K** dùng phương pháp dẫn xuất khóa 1):

*K**	91 18 69 5B E6 B7 86 F2	81 7A BE FB 54 E2 58 29
K	0D D9 B7 C6 0C 9F 1E E0	63 D6 BB 3E 4F E5 6B D9
K'	B7 9F 0C 87 04 1F 68 18	B6 CE 3F 3B 77 EE BE 08

Tính toán MAC là như sau:

D_1	61 62 63 80 00 00 00 00	00 00 00 00 00 00 00 00
G	E7 A8 FD 3F 6A 4F DB 80	33 1E E2 6E 94 09 CB 22

B.7.3 AES dùng khóa 192-bit

Các khóa *K* và *K'* sau được sử dụng trong ví dụ (trong đó *K* và *K'* được rút ra từ *K** dùng phương pháp dẫn xuất khóa 1):

*K**	C6 D0 9C CE 02 F8 34 70	E0 CF AE 90 17 90 A0 92
	41 8A AC B1 28 72 FE 9D	
K	1A D9 8F 06 2C 00 46 81	01 97 1B C0 19 8C E5 F0
	58 42 E3 73 D4 D4 82 A5	
K'	95 31 D7 D1 2B 8F 3E 8C	F8 B6 A9 CE E9 97 6B 11
	37 83 9F 7C 5D C6 6A A3	

Tính toán MAC là như sau:

D_1	48 65 6C 6C 6F 20 57 6F	72 6C 64 80 00 00 00 00
G	A5 C5 AD EC D5 4B DA 85	4E A8 DD FF FD A5 05 1F

B.7.4 AES dùng khóa 256-bit

Các khóa K và K' sau được sử dụng trong ví dụ (trong đó K và K' được rút ra từ K^* dùng phương pháp dẫn xuất khóa 1) :

K^*	78 3D 99 0F 8A DA 0F E2	E2 EC 43 19 B4 90 F8 9D
	B2 9A D0 7A 41 ED 6D 75	E3 50 76 F2 C6 85 2E E1
K	64 76 71 37 61 40 3E FC	10 EC 83 5B EC 67 C3 EB
	FF 10 F3 82 BC 19 9A EB	8E E4 B6 66 71 6C C4 DC
K'	5B 59 59 9E D8 27 F9 2F	AC 0C F3 D4 69 AE 64 5B
	C6 40 1D 3C 32 0C 1D E9	2C 4C E2 F9 02 D3 E6 36

Tính toán MAC là như sau:

D_1	53 69 78 74 65 65 6E 20	4C 65 74 74 65 72 73 2E
D_2	80 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00
G	A8 3E 5B 7E D6 C8 FD 25	62 F2 7C C1 FA 3F 55 A2

Phụ lục C
(tham khảo):
Phân tích độ an toàn của các thuật toán MAC

C.1 Tổng quan

Phụ lục này đề cập về mức an toàn của các thuật toán MAC trong tiêu chuẩn này. Mục tiêu là để hỗ trợ người sử dụng tiêu chuẩn này trong việc lựa chọn một trong những cơ chế.

Giả thiết rằng độ dài khóa của các mã khối là k bit, trong khi độ dài khóa của thuật toán MAC là bằng k^* bit. Giá trị của k^* là bằng k hoặc bằng $2k$.

Trong Phụ lục này, $MAC_K(D)$ ký hiệu MAC cho chuỗi D được tính khi dùng khóa của thuật toán MAC là K.

Để xác định mức an toàn của một thuật toán MAC, hai chiến lược tấn công đã được xem xét:

- **Tấn công giả mạo**: Tấn công này bao gồm việc dự đoán giá trị của $MAC_K(D)$ cho chuỗi dữ liệu D mà không có hiểu biết lúc đầu về K. Nếu đối phương có thể làm điều này cho một chuỗi dữ liệu duy nhất, anh ta được gọi là có *khả năng giả mạo*. Các tấn công thực hành được thường yêu cầu rằng một giả mạo là xác minh được, tức là MAC bị giả mạo là tính được chuẩn xác từ trước với xác suất gần bằng 1. Hơn nữa, trong nhiều ứng dụng chuỗi dữ liệu có một khuôn dạng đặc biệt, tại đó áp đặt các ràng buộc thêm lên chuỗi dữ liệu D;
- **Tấn công khôi phục khóa**: Tấn công này bao gồm việc tìm ra bản thân khóa K của thuật toán MAC từ một số các cặp chuỗi dữ liệu/MAC. Một tấn công như vậy là mạnh hơn so với giả mạo, vì nó cho phép các giả mạo tùy ý. Chú ý rằng độ dài khóa bằng 56 bit (chẳng hạn thuật toán DEA) không còn đủ để đề xuất bảo vệ cho phần lớn các ứng dụng.

Tính khả thi của một tấn công phụ thuộc vào số lượng các cặp chuỗi dữ liệu/MAC đã biết và đã lựa chọn theo yêu cầu, và phụ thuộc vào số lượng các mã hóa ngoại tuyến.

Các tấn công được mô tả dưới đây có thể chống lại các thuật toán MAC; không có đảm bảo rằng danh sách này là đầy đủ toàn bộ. Hai tấn công đầu là tổng quát, tức là chúng áp dụng được vào bất kỳ thuật toán MAC nào. Tấn công tiếp theo áp dụng được vào bất kỳ thuật toán MAC lặp nào. Ba tấn công tiếp theo là đặc thù cho một hoặc nhiều hơn các thuật toán MAC được mô tả trong tiêu chuẩn này (thông tin chi tiết xem [15], [20], [21], [25], [26], [27]).

- **Đoán MAC**: Đây là một giả mạo không thể xác minh được, và có xác suất thành công bằng $\max(1/2^m, 1/2^{k^*})$. Tấn công này áp dụng được vào tất cả các thuật toán MAC, và chỉ có thể bị ngăn ngừa bằng cách lựa chọn thận trọng giá trị của m và k^*.
- **Khôi phục khóa vét cạn**: Tấn công này yêu cầu trung bình 2^{k^*-1} phép toán; việc xác minh một tấn công như vậy đòi hỏi theo thứ tự k^*/m cặp chuỗi dữ liệu/MAC. Một lần nữa kiểu tấn công này áp dụng được vào tất cả các thuật toán MAC. Kiểu tấn công này có thể bị ngăn chặn bằng cách lựa chọn cẩn thận giá trị k^*. Ngoài ra, người ta có thể ngăn cản ai đó nhận được k^*/m cặp chuỗi dữ liệu/MAC, đó là số cần thiết để nhận diện khóa một cách duy nhất. Ví dụ, nếu $k^* = 64$ và $m = 32$, xấp xỉ 2^{32} khóa tương

ứng với một cặp chuỗi dữ liệu/MAC đã cho; nếu khóa được thay đổi sau mỗi chuỗi dữ liệu, thì việc khôi phục khóa vét cạn sẽ không hiệu quả hơn so với việc đoán giá trị MAC.

- **Giả mạo ngày sinh** [25], [27]: nếu người ta thu thập xấp xỉ $2^{n/2}$ cặp chuỗi dữ liệu/MAC, thì tập này với xác suất cao sẽ chứa hai chuỗi dữ liệu D và D' sao cho $MAC_K(D) = MAC_K(D')$ và các giá trị của H_q trong cả hai tính toán là bằng nhau; việc này được gọi là va chạm trong. Nếu D và D' tạo nên một va chạm trong, thì $MAC_K(D \| Y) = MAC_K(D' \| Y)$ đối với chuỗi Y bất kỳ. Việc này cho phép giả mạo sau khi một chuỗi dữ liệu đã được chọn, vì đối phương có thể dự đoán MAC cho $D' \| Y$ sau khi đã quan sát MAC tương ứng cho $D \| Y$. Việc giả mạo này một lần nữa trên các chuỗi dữ liệu có dạng cụ thể, mà có thể không được quan tâm trong tất cả các ứng dụng, nhưng cần chú ý rằng các mở rộng của kiểu tấn công này tồn tại mà cho phép có tính mềm dẻo lớn hơn trong các chuỗi dữ liệu. Tấn công yêu cầu một chuỗi dữ liệu đã chọn và xấp xỉ $2^{n/2}$ chuỗi dữ liệu đã biết và $\min\{2^{n-m}, 2^{n/2}\}$ chuỗi dữ liệu đã lựa chọn.

  Chú ý rằng tấn công giả mạo kiểu ngày sinh không thể bị ngăn ngừa bởi tổ hợp của Phương pháp Đệm 3 và thêm trước một khối vào chuỗi dữ liệu có chứa một số seri (xem [16] để biết thêm chi tiết).
- **Giả mạo tầm thường:** Nếu Phương pháp Đệm 1 được sử dụng, thì đối phương thường có thể thêm vào hoặc xóa đi một số các bit '0' ở cuối của chuỗi dữ liệu mà không làm thay đổi MAC. Việc này kéo theo rằng Phương pháp Đệm 1 chỉ nên được sử dụng trong các môi trường mà tại đó độ dài của chuỗi dữ liệu D được biết bởi các bên từ trước, hoặc tại đó các chuỗi dữ liệu cùng với một số các bit '0' ở cuối khác có cùng ngữ nghĩa.
- **Giả mạo kiểu XOR:** Nếu Thuật toán MAC 1 được sử dụng cùng với Phương pháp Đệm 1 hoặc 2 và $m = n$, thì có thể sảy ra một giả mạo kiểu XOR đơn giản. Giả sử, để đơn giản, cho D có tính chất là phiên bản đã đệm của nó là $\overline{D}$ bao gồm một khối duy nhất (vì thế, nếu Phương pháp Đệm 2 được sử dụng, chúng ta giả thiết rằng D có độ dài nhỏ hơn n bit). Ngoài ra, đặt v ký hiệu ánh xạ mà loại bỏ bit 1 bên phải nhất khỏi một chuỗi bit, và tất cả các bit 0 mà theo sau bit này (và vì thế nếu $\overline{v(X)}$ ký hiệu phiên bản đã được đệm của $v(X)$ khi dùng Phương pháp Đệm 2, thì $\overline{v(X)} = X$).

  Giả sử rằng người ta biết $MAC_K(D)$. Nếu Phương pháp Đệm 1 được sử dụng thì suy ra ngay lập tức rằng $MAC_K(\overline{D} \| (\overline{D} \oplus MAC_K(D))) = MAC_K(D)$. Tương tự, nếu Phương pháp Đệm 2 được sử dụng, thì suy ra rằng $MAC_K(\overline{D} \| v(\overline{D} \oplus MAC_K(D))) = MAC_K(D)$. Việc này kéo theo rằng người ta có thể kiến thiết một thông điệp mới với cùng giá trị MAC, đó là một giả mạo.

  Chú ý rằng tấn công này áp dụng được thậm chí nếu khóa của thuật toán MAC chỉ được sử dụng 1 lần. Giả sử rằng người ta biết $MAC_K(D)$ và $MAC_K(D')$. Nếu Phương pháp Đệm 1 đã được sử dụng, thì một tính toán tương tự chỉ ra rằng $MAC(\overline{D} \| (\overline{D'} \oplus MAC_K(D))) = MAC_K(D')$ (ở đây D có thể có độ dài tùy ý nhưng D' cần phải dài 1 khối). Tương tự, nếu Phương pháp Đệm 2 được sử dụng, thì suy ra rằng $MAC(\overline{D} \| v(\overline{D'} \oplus MAC_K(D))) = MAC_K(D')$ (ở đây D có thể có độ dài tùy ý nhưng $\overline{D'}$ cần phải có độ dài 1 khối).

  Ngoài ra, đối với Phương pháp Đệm 1, nếu ta biết $MAC_K(D)$, $MAC_K(D \| Y)$, và $MAC_K(D')$, thì ta biết rằng $MAC_K(D' \| Y') = MAC_K(D \| Y)$ nếu $Y' = Y \oplus MAC_K(D) \oplus MAC_K(D')$ (nếu D và Y bắt đầu và kết

thúc tại các ranh giới khối). Việc này cũng cho phép một giả mạo, vì đối phương có thể giả mạo MAC trên $D' \| Y'$ khi đã biết các MAC đối với 2 chuỗi dữ liệu đã biết và 1 chuỗi dữ liệu đã chọn. Một tấn công giả mạo tương tự (nhưng phức tạp hơn một chút) cũng hoạt động đối với Phương pháp Đệm 2. Chú ý rằng tất cả các giả mạo ở trên là trên các chuỗi dữ liệu có dạng đặc biệt, trong đó có thể không liên quan đến tất cả các ứng dụng.

Tấn công này có thể được ngăn ngừa bằng cách dùng Phương pháp Đệm 3.

Tấn công này có thể được mở rộng ra cho trường hợp $m < n$, nhưng nó trở nên khó hơn: trong trường hợp này nó yêu cầu biết các MAC cho $2^{(n-m)/2}$ chuỗi dữ liệu đã chọn bổ sung [20].

Áp dụng cùng tấn công này khi Thuật toán MAC 2 được sử dụng với 2 khóa bằng nhau, tức là $K' = K$. Trong trường hợp này tấn công hoạt động khi Y chứa ít nhất 2 khối, và n bit đầu tiên của Y là các bit '0'.

– **Khôi phục khóa đường tắt:** Một số thuật toán MAC tiềm tàng bị tổn thương đối với các tấn công khôi phục khóa dựa trên va chạm trong. Các ví dụ về Thuật toán MAC 3 (xem [21], [22], [26]) và Thuật toán MAC 4 trong tổ hợp với Phương pháp Đệm 1 hoặc 2 [18] hoặc Phương pháp Đệm 3 [17]. Thuật toán MAC 5 cho phép một tấn công khôi phục khóa từng phần [23]; một khi một phần của khóa đã nhận được, việc tìm các giả mạo trở nên dễ dàng.

Các bảng sau trình bày một so sánh về mức an toàn của các thuật toán MAC đã mô tả trong tiêu chuẩn này. Giả thiết rằng mã khối không có điểm yếu. Bảng C.1 chỉ ra các tính chất chính của các thuật toán MAC. Do Phương pháp Đệm 1 cho phép một giả mạo tầm thường, việc so sánh bao gồm các Thuật toán MAC 1, 2, 3, 4 và 6 chỉ xem xét các Phương pháp Đệm 2 và 3. Các Bảng C.2 và C.3 trình bày các tấn công đã biết tốt nhất đối với các mã khối có $n = 64$ và $k = 56$ (ví dụ Thuật toán Mã hóa Dữ liệu [10]). Các Bảng C.4 và C.5 là cho các mã khối có $n = 64$ và $k = 128$; các Bảng C.6 và C.7 là cho các mã khối có $n = 128$ và $k = 128$. Trong các trường hợp này chỉ các Thuật toán MAC 1 và 2 được xem xét, vì không có nhu cầu gấp đôi độ dài khóa của thuật toán MAC. Mô tả của các tấn công này có thể được tìm thấy trong [15], [16], [17], [18], [19], [20], [21], [25], [26], [27]. Tấn công được xem xét như một bộ tứ $[\alpha, \beta, \gamma, \delta]$, trong đó α ký hiệu số các phép mã hóa bằng mã khối ngoại tuyến, β ký hiệu số các cặp chuỗi dữ liệu/MAC đã biết, γ ký hiệu số các cặp chuỗi dữ liệu/MAC đã chọn và δ ký hiệu số các lần xác minh MAC trực tuyến.

Trong [15], [24] một cận dưới được chứng minh trên mức an toàn của kiến thiết CBC-MAC cơ bản (Thuật toán MAC 1) dựa trên các thuộc tính có liên quan về mã khối. Đề nghị rằng cần thảo luận nhiều kiểu tấn công ngày sinh gần với các tấn công tốt nhất có thể có với giả thiết rằng mã khối là mạnh. Tài liệu tham khảo [24] cung cấp một lập luật an toàn cho Thuật toán MAC 2.

C.2 Cơ sở hợp lý

Điều này giải thích việc lựa chọn các thuật toán MAC trong tiêu chuẩn này. Một nhân tố quan trọng trong việc lựa chọn cơ chế đã đề xuất là để duy trì tính tương thích ngược với các chuẩn ANSI, ISO, và ISO/IEC trước đó, ngoại trừ các Thuật toán MAC 5 và 6 trong ISO/IEC 9797-1:1999 đã bị loại bỏ. Động cơ thúc đẩy cho việc loại bỏ này là do độ an toàn bổ sung được đề xuất là thấp hơn so với dự tính; nếu

một thuật toán MAC cùng với mức an toàn cao hơn được cần đến, thì khuyến nghị thực hiện 2 lần tính MAC cùng với các khóa độc lập và nối các kết quả lại (thay cho việc XOR với nhau).

Thuật toán MAC 4 cung cấp một cách cải tiến để tăng độ dài khóa so với Thuật toán MAC 3. *Khuyến nghị mạnh* rằng sử dụng thuật toán này kết hợp với Phương pháp Đệm 3; trong trường hợp này nó cung cấp mức an toàn tốt hơn so với Thuật toán MAC 3 với cùng chi phí.

Bảng C.1 – Các tính chất của các thuật toán MAC

Nhóm	Thuật toán MAC	Lặp Lần cuối	Biến đổi Đầu ra	Phương pháp Đệm	#Keys	Hiệu suất
1.1	1	1	1	2	1	*t*+1
1.2	1	1	1	3	1	*t*+2
2.1	2	1	2	2	1	*t*+2
2.2	2	1	2	2	2	*t*+2
3	3	1	2	2	2	*t*+3
4.1	4	1	2	2	2	*t*+3
4.2	4	1	2	3	2	*t*+4
5	5	3	1	4	1†	*t*†
6.1	6	2	1	2	1	*t*+1
6.2	6	2	1	2	2	*t*+1

#Keys ký hiệu số lượng các khóa độc lập của mã khối.

Độ hiệu quả ký hiệu số lần mã hóa để xử lý một chuỗi dữ liệu có *tn* bit.

† Thuật toán MAC 5 cần một tính toán trước là một phép mã và cần lưu trữ hai khóa cộng thêm *n*-bit.

Bảng C.2 – Các mức an toàn ước lượng cho n = 64, k = 56 và m = 64

Mức an toàn được xác định bởi 4 con số: số lần mã hóa mã khối ngoại tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC đã chọn và số lần xác minh MAC trực tuyến.

Nhóm	Khôi phục khóa		Giả mạo		
	Vét cạn	Đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{56}, 1, 0, 0]$	–	$[0, 0, 0, 2^{56}]$	$[0, 1, 0, 0]$	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
1.2	$[2^{56}, 1, 0, 0]$	–	$[0, 0, 0, 2^{56}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
2.1	$[2^{58}, 1, 0, 0]$	–	$[0, 0, 0, 2^{56}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
2.2	$[2^{112}, 2, 0, 0]$	$[2^{57}, 2, 0, 0]$	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
3	$[2^{112}, 2, 0, 0]$	$[2^{57}, 2^{32}, 0, 0]$ $[2^{56}, 1, 0, 2^{56}]$ $[2^{57}, 2, 0, 2^{63}]$	$[0, 0, 0, 2^{64}]$ $[0, 1, 0, 2^{56}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
4.1	$[2^{112}, 2, 0, 0]$	$[2^{58}, 2^{32}, 2, 0]$† $[2^{58}, 1, 1, 2^{56}]$†	$[0, 0, 0, 2^{64}]$ $[2^{56}, 1, 0, 2^{56}]$†	–	$[0, 2^{32}, 1, 0]$†
4.2	$[2^{112}, 2, 0, 0]$	$[2^{58}, 2^{33}, 2^{50}, 0]$†	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 0, 1, 2^{64}]$†
5	$[2^{56}, 1, 0, 0]$	$[0, 2^{32}, 0, 0]$‡	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{33}, 0, 0]$
6.1	$[2^{56}, 1, 0, 0]$	–	$[0, 0, 0, 2^{56}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$
6.2	$[2^{112}, 2, 0, 0]$	$[2^{57}, 2, 0, 0]$	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$† $[0, 1, 2^{32}, 0]$

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo tầm thường

Bảng C.3 – Các mức an toàn ước lượng cho n = 64, k = 56 và m = 32

Mức an toàn được xác định bởi 4 con số: số lần mã hóa mã khối ngoại tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC đã chọn và số lần xác minh MAC trực tuyến.

Nhóm	Khôi phục khóa		Giả mạo		
	vét cạn	đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{56}, 2, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	$[0, 2, 2^{16}, 0]$	$[0, 2^{32}, 2^{32}, 0]$†
1.2	$[2^{56}, 2, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
2.1	$[2^{56}, 2, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
2.2	$[2^{112}, 4, 0, 0]$	$[2^{57}, 2^{32}, 2^{32}, 0]$ $[2^{88}, 4, 0, 0]$	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
3	$[2^{112}, 4, 0, 0]$	$[2^{57}, 2^{32}, 2^{32}, 0]$† $[2^{89}, 2^{32}, 0, 0]$ $[2^{57}, 0, 0, 2^{48}]$	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
4.1	$[2^{112}, 4, 0, 0]$	$[2^{78}, 2^{32}, 2^{50}, 0]$†	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
4.2	$[2^{112}, 4, 0, 0]$	$[2^{78}, 0, 2^{58}, 2^{57}]$† $[2^{64}, 0, 2^{63}, 2^{57}]$†	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
5	$[2^{56}, 2, 0, 0]$	$[0, 2^{33}, 2^{33}, 0]$‡	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{33}, 2^{33}, 0]$
6.1	$[2^{56}, 2, 0, 0]$	-	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†
6.2	$[2^{112}, 4, 0, 0]$	$[2^{57}, 2^{32}, 2^{32}, 0]$ $[2^{88}, 4, 0, 0]$	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$†

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo tầm thường

Bảng C.4 – Các mức an toàn ước lượng cho n = 64, k = 128 và m = 64

Mức an toàn được xác định bởi 4 con số: số lần mã hóa mã khối ngoại tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC đã chọn và số lần xác minh MAC trực tuyến.

Nhóm	Khôi phục khóa		Giả mạo		
	vét cạn	đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	$[0, 1, 0, 0]$	$[0, 2^{32}, 1, 0]$ † $[0, 1, 2^{32}, 0]$
1.2	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$ † $[0, 1, 2^{32}, 0]$
2.1	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$ † $[0, 1, 2^{32}, 0]$
5	$[2^{128}, 2, 0, 0]$	$[0, 2^{33}, 0, 0]$ ‡	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{33}, 0, 0]$
6.1	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{32}, 1, 0]$ † $[0, 1, 2^{32}, 0]$

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo tầm thường

Bảng C.5 – Các mức an toàn ước lượng cho n = 64, k = 128 và m = 32

Mức an toàn được xác định bởi 4 con số: số lần mã hóa mã khối ngoại tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC đã chọn và số lần xác minh MAC trực tuyến.

Nhóm	Khôi phục khóa		Giả mạo		
	vét cạn	đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	$[0, 2, 2^{16}, 0]$	$[0, 2^{32}, 2^{32}, 0]$ †
1.2	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$ †
2.1	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$ †
5	$[2^{128}, 4, 0, 0]$	$[0, 2^{33}, 2^{33}, 0]$ ‡	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{33}, 2^{33}, 0]$
6.1	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{32}, 2^{32}, 0]$ †

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo thông thường

Bảng C.6 – Các mức an toàn ước lượng cho n = 128, k = 128 và m = 64

Mức an toàn được xác định bởi 4 con số: số lần mã hóa mã khối ngoại tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC đã chọn và số lần xác minh MAC trực tuyến.

	Khôi phục khóa		Giả mạo		
Nhóm	vét cạn	đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	$[0, 2, 2^{32}, 0]$	$[0, 2^{64}, 2^{64}, 0]$ †
1.2	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †
2.1	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †
5	$[2^{128}, 2, 0, 0]$	$[0, 2^{65}, 2^{65}, 0]$ ‡	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{65}, 2^{65}, 0]$
6	$[2^{128}, 2, 0, 0]$	–	$[0, 0, 0, 2^{64}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo tầm thường

Bảng C.7 – Các mức an toàn ước lượng cho n = 128, k = 128 và m = 32

Mức an toàn được xác định bởi 4 con số: số phép mã của mã khối không trực tuyến, số cặp chuỗi dữ liệu/MAC đã biết, số cặp chuỗi dữ liệu/MAC được lựa chọn và số lần xác minh MAC trực tuyến.

	Khôi phục khóa		Giả mạo		
Nhóm	vét cạn	đường tắt	đoán	kiểu XOR	kiểu ngày sinh
1.1	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	$[0, 2, 2^{48}, 0]$	$[0, 2^{64}, 2^{64}, 0]$ †
1.2	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †
2.1	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †
5	$[2^{128}, 4, 0, 0]$	$[0, 2^{65}, 2^{97}, 0]$ ‡	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{65}, 2^{97}, 0]$
6	$[2^{128}, 4, 0, 0]$	–	$[0, 0, 0, 2^{32}]$	–	$[0, 2^{64}, 2^{64}, 0]$ †

† Có thể bị ngăn chặn bằng cách thêm một số seri vào phía trước chuỗi dữ liệu trong tổ hợp cùng với Phương pháp Đệm 3

‡ Chỉ khôi phục các khóa che giấu mà cho phép các giả mạo tầm thường

Phụ lục D
(tham khảo)
So sánh với các chuẩn thuật toán MAC trước đó

Phụ lục này so sánh các thuật toán MAC trong tiêu chuẩn này với các chuẩn thuật toán MAC sớm hơn. Tính toán MAC như mô tả trong ISO 8731-1 và ANSI X9.9 là một trường hợp đặc biệt của tiêu chuẩn này khi $n = 64$, $m = 32$, Thuật toán MAC 1 và Phương pháp Đệm 1 được sử dụng, mã khối là DEA (ANSI X3.92:1981). Tính toán MAC như đã được mô tả trong ANSI X9.19 và ISO 9807 là một trường hợp đặc biệt của tiêu chuẩn này khi $n = 64$, $m = 32$, hoặc Thuật toán MAC 1 hoặc Thuật toán MAC 3 được sử dụng (cả hai cùng với Phương pháp Đệm 1), mã khối là DEA (ANSI X3.92:1981).

Thư mục tài liệu tham khảo

[1] ISO 7498-2:1989, Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture

[2] ISO 8731-1:1987[1], Banking – Approved algorithms for message authentication – Part 1: DEA

[3] ISO 8732:1988, Banking – Key management (wholesale)

[4] ISO/IEC 8825-1:2002[2], Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)

[5] ISO/IEC 9798-1:2010, Information technology – Security techniques – Entity authentication –Part 1: General

[6] ISO 9807:1991[3], Banking and related financial services – Requirements for message authentication (retail)

[7] ISO/IEC 10116:2006, Information technology – Security techniques – Modes of operation for an n-bit block cipher;

[8] TCVN 7817 (ISO/IEC 11770) (tất cả các phần), Công nghệ thông tin – Kỹ thuật an ninh – Quản lý khóa;

[9] ISO 11568 (all parts), Banking – Key management (retail)

[10] ANSI X3.92:1981, Data Encryption Algorithm

[11] ANSI X9.9:1986[2], Financial Institution Message Authentication (Wholesale)

[12] ANSI X9.19:1986, Financial Institution Retail Message Authentication

[13] ANSI X9.24-1:2004[4], Retail Financial Services Symmetric Key Management – Part 1: Using Symmetric Techniques

[14] NIST Special Publication 800-38B: 2005, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005

[15] M. Bellare, J. Kilian, and P. Rogaway, 'The security of cipher block chaining', Advances in Cryptology, Proceedings Crypto'94, LNCS 839, Y. Desmedt, Ed., Springer-Verlag, 1994, pp. 341-358

[16] K. Brincat and C.J. Mitchell, 'New CBC-MAC forgery attacks', Information Security and Privacy, ACISP 2001, LNCS 2119, V. Varadharajan and Y. Mu, Eds., Springer-Verlag, 2001, pp. 3-14

[17] D. Coppersmith, L.R. Knudsen, and C.J. Mitchell, 'Key recovery and forgery attacks on the MacDES MAC algorithm', Advances in Cryptology, Proceedings Crypto 2000, LNCS 1880, M. Bellare, Ed., Springer-Verlag, 2000, pp. 184-196

---

[1] Đã bị hủy bỏ và thay thế bằng ISO 16609:2004

[2] Đã bị hủy bỏ và thay thế bằng ISO/IEC 8825-1:2008

[3] Đã bị hủy bỏ và thay thế bằng ISO 16609:2004

[4] Đã bị hủy bỏ và thay thế bằng ANSI X9.24-1:2009

[18] D. Coppersmith and C.J. Mitchell, 'Attacks on MacDES MAC algorithm', Electronics Letters, Vol. 35, No. 19, 1999, pp. 1626-1627

[19] T. Iwata and K. Kurosawa, 'OMAC: One-key CBC MAC', Proceedings Fast Software Encryption 2003, LNCS 2887, T. Johansson, Ed., Springer-Verlag, 2003, pp. 129-153

[20] L. Knudsen, 'Chosen-text attack on CBC-MAC', Electronics Letters, Vol. 33, No. 1, 1997, pp. 48-49

[21] L. Knudsen and B. Preneel, 'MacDES: MAC algorithm based on DES', Electronics Letters, Vol. 34, No. 9, 1998, pp. 871-873

[22] C.J. Mitchell, 'Key recovery attack on ANSI retail MAC', Electronics Letters, Vol. 39, 2003, pp. 361-362

[23] C.J. Mitchell, 'Partial key recovery attack on XCBC, TMAC and OMAC', Cryptography and Coding: Proceedings 10th IMA International Conference, LNCS 3796, N. Smart, Ed., Springer-Verlag, 2005, pp. 155-167 (See also: Royal Holloway, University of London, Mathematics Department Technical Report RHUL-MA-2003-4, August 2003, 15 pages)

[24] E. Petrank and C. Rackoff, 'CBC MAC for real-time data sources', Journal of Cryptology, Vol. 13, No. 3, 2000, pp. 315-338

[25] B. Preneel and P.C. van Oorschot, 'MDx-MAC and building fast MACs from hash functions', Advances in Cryptology, Proceedings Crypto'95, LNCS 963, D. Coppersmith, Ed., Springer-Verlag, 1995, pp. 1-14

[26] B. Preneel and P.C. van Oorschot, 'A key recovery attack on the ANSI X9.19 retail MAC', Electronics Letters, Vol. 32, No. 17, 1996, pp. 1568-1569

[27] B. Preneel and P.C. van Oorschot, 'On the security of iterated Message Authentication Codes', IEEE Transactions on Information Theory, Vol. 45, No. 1, January 1999, pp. 188-199;