

TCVN

TIÊU CHUẨN QUỐC GIA

TCVN 11816-4:2017

ISO/IEC 10118-4:1998 WITH AMENDMENT 1:2014

Xuất bản lần 1

**CÔNG NGHỆ THÔNG TIN -
CÁC KỸ THUẬT AN TOÀN - HÀM BẮM -
PHẦN 4: HÀM BẮM SỬ DỤNG SỐ HỌC ĐỒNG DƯ**

*Information technology - Security techniques - Hash-functions -
Part 4: Hash-function using modular arithmetic*

HÀ NỘI - 2017

Mục Lục

Lời nói đầu.....	4
1 Phạm vi áp dụng.....	5
2 Tài liệu viện dẫn.....	5
3 Các thuật ngữ và định nghĩa.....	5
3.1 Thuật ngữ từ TCVN 11816-1.....	5
3.2 Duy nhất cho TCVN 11816-4.....	6
3.3 Các quy ước.....	7
3.4 Các định danh của hàm băm (hash-function Identifier).....	7
4 Các ký hiệu và các thuật ngữ viết tắt.....	7
4.1 Từ TCVN 11816-1:2017 (ISO/IEC 10118-1).....	8
4.2 Duy nhất cho TCVN 11816-4.....	8
5 Các yêu cầu.....	9
6 Các biến và giá trị cần thiết cho phép băm.....	9
6.1 Độ dài của mã băm và của N	9
6.2 Đồng dư của hàm vòng.....	10
6.3 Giá trị khởi tạo.....	10
6.4 Số mũ.....	10
6.5 Số nguyên tố của hàm rút gọn.....	10
7. Quy trình băm.....	10
7.1 Chuẩn bị xử lý dữ liệu.....	10
7.2 Ứng dụng của hàm vòng.....	11
7.3 Hàm rút gọn.....	11
8 Các hàm băm.....	12
8.1 MASH-1.....	12
8.2 MASH-2.....	12
Phụ lục A (Tham khảo) Các ví dụ.....	14
A.1 Quy trình băm.....	14
A.2 Các ví dụ tính toán băm.....	16
A.3 Thông điệp kiểm tra mẫu và giá trị băm của chúng.....	22
Phụ lục B (Tham khảo) Thông tin bổ sung.....	27
B.1 Xác định các giá trị của tham số.....	27
B.2 Sự khác nhau giữa MASH-1 và MASH-2.....	27
B.3 Hàm rút gọn.....	27
Phụ lục C (Quy định) Định danh đối tượng.....	28
Thư mục tài liệu tham khảo.....	29

Lời nói đầu

TCVN 11816-4:2017 hoàn toàn tương đương với ISO/IEC 10118-4:1998 và sửa đổi 1:2014.

TCVN 11816-4:2017 do Cục Quản lý mật mã dân sự và Kiểm định sản phẩm mật mã biên soạn, Ban Cơ yếu Chính phủ đề nghị, Tổng cục Tiêu chuẩn Đo lường Chất lượng thẩm định, Bộ Khoa học và Công nghệ công bố.

Bộ tiêu chuẩn TCVN 11816 (ISO/IEC 10118) *Công nghệ thông tin – Các kỹ thuật an toàn – Hàm băm* gồm các tiêu chuẩn sau:

- TCVN 11816-1 : 2017 (ISO/IEC 10118-1:2016), Phần 1: Tổng quan.
- TCVN 11816-2 : 2017 (ISO/IEC 10118-2:2010), Phần 2: Hàm băm sử dụng mã khối n-bit.
- TCVN 11816-3 : 2017 (ISO/IEC 10118-3:2004), Phần 3: Hàm băm chuyên dụng.
- TCVN 11816-4 : 2017 (ISO/IEC 10118-4:1998), Phần 4: Hàm băm sử dụng số học đồng dư.

Công nghệ thông tin - Các kỹ thuật an toàn - Hàm băm

Phần 4: Hàm băm sử dụng số học đồng dư

Information technology - Security techniques - Hash-functions - Part 4: Hash-functions using modular arithmetic

1 Phạm vi áp dụng

TCVN 11816-4 đặc tả 2 hàm băm sử dụng số học đồng dư. Những hàm này, được cho là kháng va chạm, nên thông điệp có độ dài bất kì, nhưng giới hạn độ dài mã băm, độ dài đó được xác định bởi độ dài của số nguyên tố được sử dụng trong hàm rút gọn được định nghĩa trong 7.3. Do đó, mã băm dễ dàng được mở rộng (scaled) với độ dài đầu vào của bất kỳ cơ chế nào (ví dụ thuật toán chữ ký, lược đồ định danh).

Các hàm băm được đặc tả trong tiêu chuẩn này là MASH-1 và MASH-2 (hàm băm an toàn số học đồng dư) phù hợp riêng cho các môi trường thực hiện số học đồng dư với đủ các chiều dài. Hai hàm băm chỉ khác nhau ở số mũ được sử dụng trong hàm vòng.

2 Tài liệu viện dẫn

Tiêu chuẩn sau bao gồm các điều khoản mà thông qua tài liệu viện dẫn này chứa đựng các điều khoản của TCVN 11816-4. Vào thời điểm xuất bản, ấn phẩm này là hợp lệ. Tất cả các tiêu chuẩn được hiệu chỉnh và thỏa thuận dựa trên TCVN 11816-4 đều được khuyến khích nghiên cứu khả năng áp dụng ấn phẩm mới nhất của tiêu chuẩn được chỉ rõ dưới đây. Các phần của IEC và ISO duy trì đăng ký của các tiêu chuẩn quốc tế hợp lệ hiện hành.

TCVN 11816-1:2016 (ISO/IEC 10118-1:2016), *Công nghệ thông tin – Các kỹ thuật an toàn – Hàm băm – Phần 1: Tổng quan*.

3 Các thuật ngữ và định nghĩa

Trong phần này các thuật ngữ và định nghĩa sau đây được áp dụng.

3.1 Thuật ngữ từ TCVN 11816-1

3.1.1

Hàm băm kháng va chạm (collision-resistant hash-function)

TCVN 11816-4 : 2017

Hàm băm thỏa mãn tính chất sau: không thể tìm được 2 đầu vào khác nhau với cùng một giá trị đầu ra.

3.1.2

Xâu dữ liệu (data string)

Xâu bit được dùng làm đầu vào cho hàm băm.

3.1.3

Mã băm (hash-code)

Xâu bit là đầu ra của hàm băm.

3.1.4

Hàm băm (hash-function)

Hàm mà ánh xạ một xâu bit tới một xâu bit có độ dài xác định thỏa mãn 2 tính chất sau:

- Không thể tìm được một giá trị đầu vào ứng với một giá trị đầu ra cho trước.
- Không thể tìm được một đầu vào thứ 2 khác với đầu vào cho trước mà có cùng đầu ra.

3.1.5

Giá trị khởi tạo (initializing value)

Một giá trị sử dụng để định nghĩa điểm bắt đầu của một hàm băm.

3.1.6

Đệm (padding)

Bit mở rộng đính kèm cho xâu dữ liệu.

3.2 Duy nhất cho TCVN 11816-4

3.2.1

Khối (block)

Một xâu bit có độ dài L_b là bội số nguyên của 16 (xem trong điều 6.1).

Ví dụ: Độ dài đầu ra H_i của hàm vòng.

3.2.2

Nửa khối (half-block)

Một xâu bit có độ dài $L_b/2$

Ví dụ: Nửa độ dài của khối H_i

3.2.3

Định danh của hàm băm (hash-function identifier)

Một byte xác định một hàm băm cụ thể

3.2.4

Số đồng dư (modulus)

Tham số là một số nguyên dương và là tích của 2 số nguyên tố riêng biệt.

3.2.5

Hàm rút gọn (reduction-function)

Một hàm RED được áp dụng cho khối H_q có độ dài L_q để tạo ra mã băm H với độ dài L_p .

3.2.6

Hàm vòng (round-function)

Hàm $\phi(\cdot)$ chuyển 2 xâu nhị phân độ dài L_q thành một xâu nhị phân độ dài L_p .

CHÚ THÍCH: Nó được dùng lặp đi lặp lại như một phần của hàm băm, trong đó, nó kết hợp lại một khối dữ liệu 'mở rộng' có độ dài L_q với đầu ra trước đó có độ dài L_p .

3.3 Các quy ước

3.3.1 Sự sắp thứ tự bit (bit ordering)

Trong TCVN 11816-4, thứ tự bit được mô tả trong điều 3 của TCVN 11816-1.

3.3.2 Biến đổi một số thành một xâu (converting a number to a string)

Trong quá trình tính hàm vòng, các số nguyên được biến đổi thành các xâu L bit. Ở nơi mà điều này được yêu cầu, xâu bit được tạo ra bằng sự biểu diễn nhị phân của số nguyên, với bit ở bên trái nhất của xâu tương ứng với bit quan trọng nhất của sự biểu diễn nhị phân. Nếu xâu bit cuối cùng có ít hơn L bit thì xâu sẽ được bổ sung vào bên trái số 0 phù hợp để tạo ra xâu có độ dài L .

3.3.3 Biến đổi một xâu thành một số nguyên (converting a string to a number)

Trong quá trình tính hàm vòng, các xâu bit cần được chuyển thành số nguyên. Ở đó, số nguyên sẽ được tạo ra bằng số mà có biểu diễn nhị phân bằng xâu nhị phân, ở đó bit trái nhất của xâu được coi như bit có nghĩa nhất của biểu diễn nhị phân.

3.4 Các định danh của hàm băm (hash-function identifier)

Các định danh được định nghĩa cho 2 hàm băm MASH được đặc tả trong TCVN 11816-4. Các định danh hàm băm cho các hàm băm quy định trong điều 8.1 và 8.2 là tương đương 41 và 42 (hệ thập lục phân). Dải các giá trị từ 43 tới 4f (hệ thập lục phân) được dành riêng để sử dụng sau này như định danh các hàm băm của TCVN 11816.

4 Các ký hiệu và các thuật ngữ viết tắt

Trong TCVN 11816-4, các ký hiệu sau đây được sử dụng.

4.1 Từ TCVN 11816-1:2017 (ISO/IEC 10118-1)

D	Data	Dữ liệu
H	Hash-code	Mã băm
IV	Initializing value	Giá trị khởi tạo
$X \oplus Y$	Exclusive-or of strings of bits X and Y	Phép toán XOR của chuỗi bit X và Y

4.2 Duy nhất cho TCVN 11816-4

B_j	Khối thứ j được chia từ chuỗi dữ liệu D sau quá trình đệm, tách và mở rộng.
D_j	Một nửa khối thứ j từ chuỗi dữ liệu D sau quá trình đệm và tách. D_{q+1} đến D_{q+8} là các khối dữ liệu bổ sung được tính toán trong hàm rút gọn.
e	Số mũ được dùng trong hàm vòng.
E	Một khối hằng số tương đương 4 khối (ở vị trí trái nhất) theo sau $L_\phi - 4$ số 0.
H_j	Đầu ra vòng thứ j của hàm vòng. H_j có độ dài L_ϕ .
L_D	Độ dài của chuỗi đầu vào D tính theo bit.
L_ϕ	Độ dài đầu ra H_j của hàm vòng ϕ . Nó là bội số nguyên của 16.
L_N	Độ dài của số đồng dư N sử dụng trong hàm vòng.
L_p	Độ dài của số nguyên tố p sử dụng trong hàm rút gọn.
mod	Nếu Z_1 là một số nguyên và Z_2 là một số nguyên dương thì $Z_1 \bmod Z_2$ thì số nguyên duy nhất Z_3 thỏa mãn
a)	a) $0 \leq Z_3 < Z_2$, và
b)	b) $Z_1 - Z_3$ là một số nguyên và là bội của Z_2 .
N	Một số nguyên hợp số được dùng như số đồng dư trong hàm vòng. CHÚ THÍCH: Để xác định giá trị N , xem điều 5.
p	Một số nguyên tố được sử dụng trong hàm rút gọn. CHÚ THÍCH: Để xác định giá trị p , xem điều 5.
q	Số lượng nửa khối trong chuỗi dữ liệu D sau các quá trình đệm và tách, và cũng là số các khối sau quá trình đệm, tách và mở rộng.
RED	Hàm rút gọn, mà được áp dụng như thao tác cuối cùng của quá trình băm để rút gọn khối H_q có độ dài L_ϕ để nhận được mã băm H có chiều dài L_p .
Y_j	Chuỗi con thứ j có độ dài $L_\phi/4$ bit được dùng trong hàm rút gọn.
ϕ	Một hàm vòng. Nếu X và Y biểu thị các chuỗi L_ϕ bit, thì $\phi(X, Y)$ biểu thị một chuỗi L_ϕ bit thu được bằng cách áp dụng ϕ tới X và Y .
$\vee$	Phép toán OR trên các chuỗi bit, tức là nếu X và Y là các chuỗi có cùng độ dài thì $X \vee Y$ biểu thị chuỗi có được là kết quả phép OR của X và Y .

Một ký hiệu biểu thị phép toán cắt. Nếu X là một xâu bit thì $X \sim j$ biểu thị xâu bit có được là j bit bên phải tận cùng của X .

$:=$ Một ký hiệu biểu thị phép toán "thiết lập bằng". Nó được sử dụng trong bản đặc tả kỹ thuật theo thủ tục của hàm vòng và của hàm rút gọn, trong đó nó chỉ ra rằng khối ở bên trái của ký hiệu sẽ được thay đổi để bằng với giá trị biểu thức ở bên phải của ký hiệu.

$X \parallel Y$ Phép nối các xâu bit X và Y theo thứ tự xác định.

5 Các yêu cầu

5.1. Để sử dụng một trong các hàm băm được đặc tả trong TCVN 11816-4, hai số nguyên sẽ được lựa chọn: số đồng dư N được sử dụng trong hàm vòng và số nguyên tố p được sử dụng trong hàm rút gọn.

Cả hai số nguyên, N và p được xác định bởi các yêu cầu an toàn của ứng dụng sử dụng những hàm băm này.

5.1.1 Việc phân tích thành thừa số của N được chọn là không khả thi về mặt tính toán.

5.1.2 N được tạo ra bằng cách duy trì các thừa số bí mật. Điều này có thể được thực hiện bởi một bên thứ 3 đáng tin cậy hoặc bởi sự tính toán nhiều bên an toàn.

CHÚ THÍCH 1: Việc tạo ra một số N với thuộc tính mà các thừa số của nó được giữ bí mật có thể được thực hiện bằng việc sử dụng một bên thứ ba đáng tin cậy, phần cứng đáng tin, và/hoặc sự tính toán nhiều bên an toàn. Các ví dụ có thể được tìm thấy trong Boneh [1], Cocks [2] và Frankel [3].

CHÚ THÍCH 2: Nếu các thừa số của số đồng dư được giữ bí mật, và nếu kích thước của số nguyên tố p là đủ lớn thì thuật toán tốt nhất để tìm sự va chạm xấp xỉ $2^{L_p/2}$ các ước lượng của hàm vòng và thuật toán tốt nhất để tìm một tiền ảnh thứ 2 cần xấp xỉ 2^{L_p} các ước lượng của hàm vòng. Do vậy, trong các tình huống đó, MASH-1 và MASH-2 này được tin tưởng là các hàm băm không va chạm.

5.1.3 Số nguyên tố p của hàm rút gọn sẽ không phải là một thừa số của N của hàm vòng.

5.1.4 Độ dài L_p của số nguyên tố p lớn nhất là bằng một nửa độ dài của N , $L_p \leq L_N/2$.

5.1.5 Ba bit bậc cao của số nguyên tố p phải bao gồm các số 1.

5.2 Để sử dụng một trong các hàm băm, MASH-1 và MASH-2, người dùng phải chọn một trong hai số mũ e được dùng trong hàm vòng ϕ .

5.3 MASH-1 và MASH-2 có thể được áp dụng cho tất cả các xâu dữ liệu D chứa nhiều nhất $2^{L_p/2} - 1$ bit.

6 Các biến và giá trị cần thiết cho phép băm

6.1 Độ dài của mã băm và của N

Độ dài của N và độ dài của các khối H_i có liên quan theo cách thức sau:

$$L_p + 1 \leq L_N \leq L_p + 16$$

Độ dài L_p của khối H_q là bội số nguyên của 16.

CHÚ THÍCH 1: Nếu độ dài L_p được chọn, thì độ dài L_N ràng buộc theo bất đẳng thức ở trên. Nếu độ dài L_N được chọn thì độ dài L_p sẽ là bội số lớn nhất của 16 mà nhỏ hơn L_N .

CHÚ THÍCH 2: Sự hiểu biết về N đủ để xác định L_N và hệ quả là L_p .

6.2 Đồng dư của hàm vòng

Số đồng dư N được dùng trong hàm vòng là một số nguyên hợp số được tạo ra bởi một tích của hai số nguyên tố có cùng độ dài sao cho việc tính toán để tìm thừa số của N là không khả thi.

CHÚ THÍCH 1: Ngoài tính không khả thi của việc tìm thừa số đồng dư, độ an toàn của hàm băm MASH được dựa trên một phần độ khó của việc khai căn các số đồng dư.

CHÚ THÍCH 2: Việc lựa chọn một số đồng dư riêng N với độ dài phù hợp nằm ngoài phạm vi của TCVN 11816-4.

6.3 Giá trị khởi tạo

Giá trị khởi tạo IV được định nghĩa là xâu của L_p các số nhị phân 0.

6.4 Số mũ

Đối với MASH-1, giá trị của số mũ e trong hàm vòng bằng 2. Đối với MASH-2, giá trị của số mũ e trong hàm vòng bằng 257.

6.5 Số nguyên tố của hàm rút gọn

Hàm rút gọn được đặc tả trong Điều 7.3 yêu cầu một số nguyên tố p . Độ dài L_p của số nguyên tố p được xác định bởi các yêu cầu an toàn, và bằng độ dài đầu vào của bất kỳ cơ chế nào sử dụng mã băm. Độ dài L_p lớn nhất là bằng một nửa độ dài của N , $L_p \leq L_N/2$.

CHÚ THÍCH 1: Việc lựa chọn một số nguyên tố cụ thể p với độ dài thích hợp nằm ngoài phạm vi của TCVN 11816-4.

CHÚ THÍCH 2: Để tránh các kết quả không cân bằng bởi rút gọn p , số nguyên tố p phải được chọn với ba bit cao là các số 1.

7. Quy trình băm

Mã băm H của xâu dữ liệu D được tính toán theo các bước sau (xem Hình 1):

7.1 Chuẩn bị xâu dữ liệu

Xâu dữ liệu D được chuyển thành một chuỗi các khối đầu vào cho hàm vòng ϕ . Việc chuẩn bị gồm việc đệm, tách và khai triển chi tiết trong các phần sau.

7.1.1 Đệm xâu dữ liệu

Nếu độ dài L_D của xâu dữ liệu D không phải là một bội số nguyên của $L_p/2$, thì D được đệm bên phải với các bit 0 nhị phân tuân theo phương pháp đệm 1 được mô tả trong Phụ lục B của TCVN 11816-1:2017 (ISO/IEC 10118-1).

7.1.2 Nối thêm độ dài

Một nửa khối bổ sung là được đệm thêm bên phải của xâu dữ liệu. Nó bao gồm biểu diễn nhị phân với độ dài L_D của xâu dữ liệu gốc D (chưa đệm thêm), được đệm bên trái với các bit nhị phân 0 (xem 3.3.2).

CHÚ THÍCH: Nếu khối dữ liệu D rỗng, thì chỉ có độ dài khối là đầu vào của quy trình băm.

7.1.3 Tách xâu dữ liệu

Xâu kết quả được chia thành một chuỗi các nửa khối q : $D_1, D_2, \dots, D_q$.

7.1.4 Khai triển

Mỗi nửa khối D_j , $j = 1, 2, \dots, q$ được nhân đôi độ dài từ $L_D/2$ bit thành L_ϕ bit. Điều này đạt được bằng cách chia D_j thành các nửa byte và đặt vào trước mỗi nửa byte của D_j với một nửa byte gồm 4 bit 1 (1111), cho $j = 1, 2, \dots, q$. Kết quả của quá trình này khi áp dụng với nửa khối D_j được ký hiệu là B_j , $j = 1, 2, \dots, q$.

7.2 Ứng dụng của hàm vòng

Hàm vòng ϕ , hàm mà các hàm băm MASH dựa vào để lấy hai khối đầu vào H_{j-1} và B_j , cả hai có độ dài L_ϕ . Nó trả về một khối H_j độ dài L_ϕ . Được định nghĩa như sau:

$$\Phi(B_j, H_{j-1}) = (((H_{j-1} \oplus B_j) \vee E)^s \bmod N) \sim L_\phi \oplus H_{j-1}$$

Hàm vòng được áp dụng lần lượt liên tục tới các khối dữ liệu B_j như sau:

$$H_0 := IV$$

$$H_j := \phi(B_j, H_{j-1}) \quad j = 1, 2, \dots, q$$

7.3 Hàm rút gọn

Hàm rút gọn RED gồm tám ứng dụng của hàm vòng với một đầu vào dữ liệu được lấy từ H_q . Mã băm H được tính toán bởi bốn bước sau: tách khối H_q , mở rộng xâu dữ liệu, xử lý các khối dữ liệu bổ sung và rút gọn khối H_{q+8} .

7.3.1 Việc tách khối H_q

Khối H_q được chia thành 4 xâu độ dài $L_\phi/4$ bit, kí hiệu $H_{q1}, H_{q2}, H_{q3}, H_{q4}$

$$H_q := H_{q1} \parallel H_{q2} \parallel H_{q3} \parallel H_{q4}$$

7.3.2 Mở rộng xâu dữ liệu

Định nghĩa $Y_0 := H_{q3}$, $Y_1 := H_{q1}$, $Y_2 := H_{q4}$, và $Y_3 := H_{q2}$. Với $i = 4$ tới 15 đặt:

$$Y_i := Y_{i-1} \oplus Y_{i-4}$$

Sau đó xác định 8 nửa khối dữ liệu bổ sung D_{q+1} tới D_{q+8} như sau:

Với $i = 1$ đến 8 đặt

$$D_{q+i} := Y_{2i-2} \parallel Y_{2i-1}$$

7.3.3 Xử lý các nửa khối

Tám nửa khối D_{q+1} tới D_{q+8} được xử lý như trong 7.1.4 và 7.2, nhưng với $IV = H_q$, nhận được kết quả là H_{q+8} .

7.3.4 Rút gọn

Mã băm H với độ dài L_p được tính toán như sau:

$$H := H_{q+s} \bmod p$$

8 Các hàm băm

Hai hàm băm được đặc tả trong TCVN 11816-4 khác nhau về giá trị của số mũ e được sử dụng trong hàm vòng ϕ .

8.1 MASH-1

Đối với MASH-1, hàm vòng ϕ đã quy định trong điều 7 trở thành:

$$\phi(B_i, H_{i-1}) = (((H_{i-1} \oplus B_i) \vee E)^2 \bmod N) \sim L_\phi \oplus H_{i-1}$$

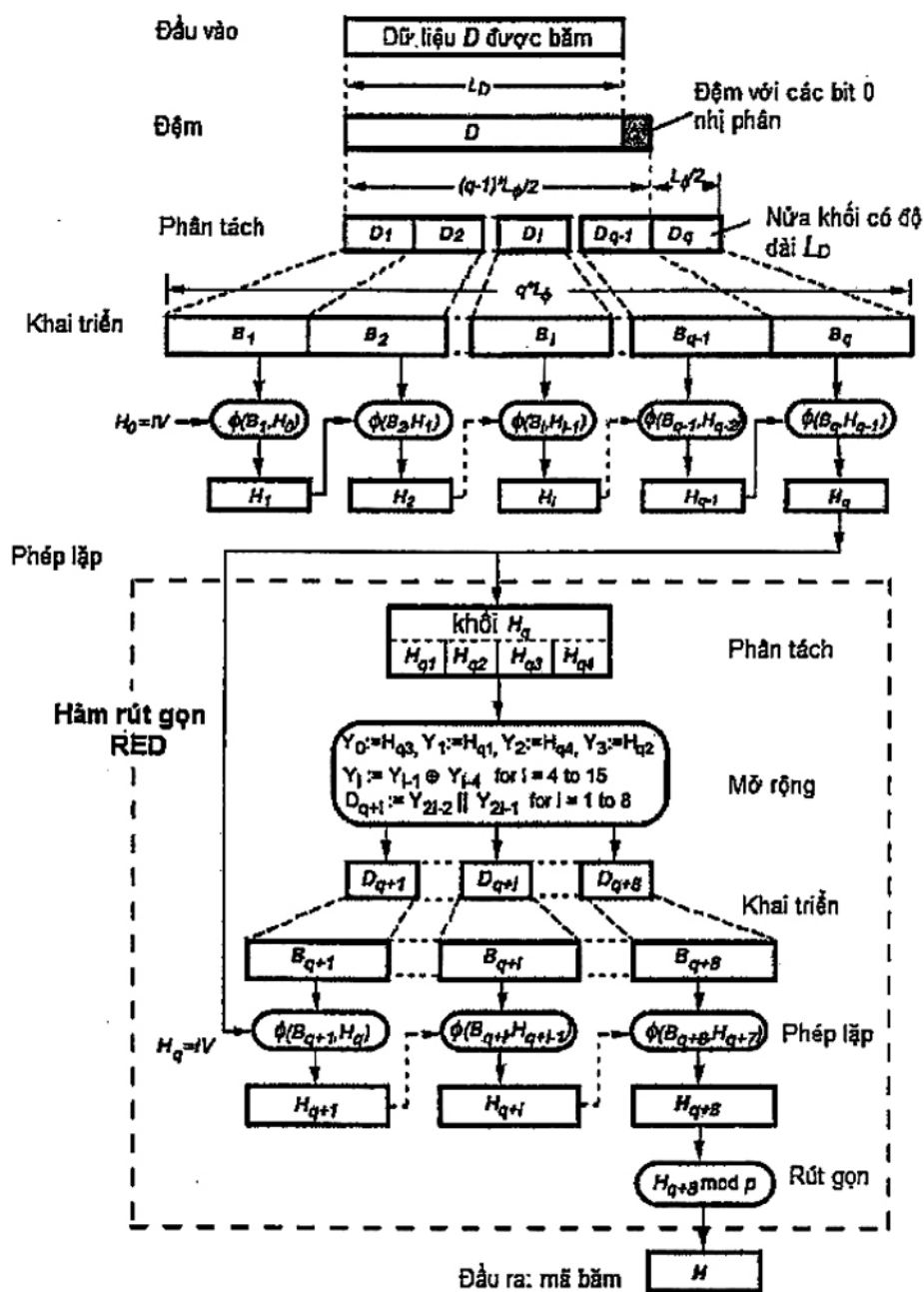
Định danh của hàm băm MASH-1 là 41 (hệ thập lục phân).

8.2 MASH-2

Đối với MASH-2, hàm vòng ϕ đã quy định trong điều 7 trở thành

$$\phi(B_i, H_{i-1}) = (((H_{i-1} \oplus B_i) \vee E)^{257} \bmod N) \sim L_\phi \oplus H_{i-1}$$

Định danh của hàm băm MASH-2 là 42 (hệ thập lục phân).



Hình 1: Hàm băm MASH

Phụ lục A (Tham khảo)

Các ví dụ

A.1 Quy trình băm

Thực tế, xâu dữ liệu có thể được cung cấp như một luồng. Độ dài tổng thể L_D có thể không được biết trước kết thúc luồng cuối cùng. Khi luồng đi qua, việc tính toán hàm băm được thực hiện. Để thuận lợi cho việc mô tả, một hằng số $k = L_D/2$ được đưa ra. Chỉ bốn thành ghi sau đây được yêu cầu cho các biến: i , A , B , C .

- i bộ đếm bit cho độ dài L_D .
- A bộ đệm để lưu giữ mã băm.
- B bộ cộng tích lũy để giữ các kết quả trung gian.
- C giữ nửa khối dữ liệu

Mã băm H của dữ liệu D được tính toán theo các bước sau:

A.1.1 Bước 1 (Khởi tạo)

- Bộ đệm A được thiết lập về 0: $A := 0$ (giá trị khởi tạo $IV = H_0 = 0$)
- Bộ cộng tích lũy B được đặt về 0: $B := 0$
- Bộ đếm bit được đặt về 0: $i := 0$ (đếm số các bit dữ liệu hiệu dụng đã được băm)

A.1.2 Bước 2a (Đọc một khối dữ liệu)

Nếu dữ liệu còn lại ít nhất k bit, đọc k bit của dữ liệu D vào C . Nếu dữ liệu còn lại là k' (với $0 < k' < k$) đọc k' bit của dữ liệu D vào C và lấp đầy C với $k-k'$ các bit 0 nhị phân vào bên phải. k (tương ứng với k') được cộng vào i . Nếu không có thêm dữ liệu nào ($k' = 0$) thì đi tới bước 3a.

Bước 2b (Khải triển)

Mỗi byte của C được tách thành các nửa và mỗi nửa được đứng trước với bốn bit 1 nhị phân. Kết quả được đặt vào trong bộ cộng tích lũy B .

Bước 2c (Kết hợp với giá trị băm trước đó)

Tính: $B := B \oplus A$ (kết hợp)

Bước 2d

$B := B \vee E$ (bốn bit giá trị cao nhất được đặt về 1)

Bước 2e (Nâng lũy thừa)

Tính: $B := B^e \bmod N$ trong đó $e = 2$ (phép bình phương) với MASH-1 và $e = 257$ với MASH-2

Bước 2f (Phép cắt)

(Các bit bên trái cùng của C vượt quá độ dài L_f bị xóa)

Bước 2g (Tiếp với giá trị băm trước đó)

Tính: $A := B \oplus A$

Bước 2h: Quay trở lại bước 2a.

A.1.3 Bước 3a (Đọc bộ đếm độ dài)

$C := i$ (nội dung của bộ đếm độ dài được đặt vào trong C).
Số nguyên được biến đổi thành một xâu như đã quy định ở 3.3.2.

Bước 3b (Băm với bộ đếm độ dài)

Thực hiện các bước 2b, 2c, 2d, 2e, 2f và 2g, sau đó chuyển sang bước 4.

A.1.4 Bước 4 (Đưa ra kết quả)

Khối H_q được chứa trong bộ đệm A là $L\phi$ các bit tận cùng bên phải.

A.1.5 Quá trình rút gọn

Các thanh ghi sau được yêu cầu cho các biến: $A, B, C, C_0, C_1, C_2, C_3, i$.

A	lưu giữ khối H_q để rút gọn.
B	bộ đệm được dùng trong hàm vòng.
C	bộ cộng tích lũy của độ dài $L\phi/2$ để lưu giữ nửa khối.
C_0, C_1, C_2, C_3	bốn bộ đếm độ dài $L\phi/4$.
i	bộ đếm.

Mã băm H được tính toán theo các bước sau.

A.1.5.1 Bước 4a (Khởi tạo)

Bộ đếm được đặt về 8: $i := 8$ (số nửa khối được xử lý)

A.1.5.2 Bước 4b (Tách khối H_q)

Khối H_q được chứa trong bộ đệm A được chia thành 4 phần $H_{q1}, H_{q2}, H_{q3}, H_{q4}$ mỗi phần độ dài $L\phi/4$ (xem Hình 1) và được lưu:

$C_0 := H_{q3}$

$C_1 := H_{q1}$

$C_2 := H_{q4}$

$C_3 := H_{q2}$

A.1.5.3 Bước 4c (Mở rộng và bước lặp)

Tính $C := C_0 \parallel C_1$ (phép nối)

Và thực hiện các bước 2b tới 2g (áp dụng hàm vòng ϕ) (số vòng $q+1$)

Tính $i := i - 1$. giảm bộ đếm

Bước 4d (Mở rộng và bước lặp)

Tính $C := C_2 \parallel C_3$ (phép nối)

Và thực hiện các bước 2b tới 2g (áp dụng hàm vòng ϕ) (số vòng $q+2$)

Tính $i := i - 1$. giảm bộ đếm

A.1.5.4 Bước 4e (Kết hợp, mở rộng và phép lặp)

Tính $C_0 := C_0 \oplus C_3$ (kết hợp)

$C_1 := C_1 \oplus C_0$

Tính $C := C_0 \parallel C_1$ (phép nối)

Và thực hiện các bước 2b tới 2g (áp dụng hàm vòng ϕ) (số vòng $q+3, q+5, q+7$)

Tính	$i := i - 1.$	giảm bộ đếm
Bước 4f	(Kết hợp các khối đã tách)	
Tính	$C_2 := C_2 \oplus C_1$	(kết hợp)
	$C_3 := C_3 \oplus C_2$	
Tính	$C := C_2 C_3$	(phép nối)
Và thực hiện các bước 2b tới 2g	(áp dụng hàm vòng ϕ) (số vòng $q+4, q+6, q+8$)	
Tính	$i := i - 1.$	Nếu i khác 0, quay lại bước 4e.

A.1.6 Bước 5 (Phép rút gọn cuối cùng)

Tính: $A := A \bmod p$

A.1.7 Bước 6 (Đưa ra kết quả)

Mã băm H được chứa trong bộ đệm A là L_p bit ở tận cùng bên phải.

A.2 Các ví dụ tính toán băm

Trong ví dụ sau, N là một hợp số, lớn hơn L_f 12 bit. L_f là một bội số của 16 ngay dưới L_N . Độ dài của số đồng dư N được chọn chỉ để minh họa quy trình băm từng bước một và không nhất thiết đáp ứng các yêu cầu an toàn. Để phân biệt giữa các số thập phân và các số hệ thập lục phân, ký hiệu d hay h được thêm vào tương ứng.

Dữ liệu sau được dùng:

Độ dài của số đồng dư N , $L_N = 10c_h = 268_d$

Độ dài của khối H_f , $L_f = 100_h = 256_d$

Độ dài của số nguyên tố p , $L_p = 80_h = 128_d$

Độ dài của mã băm H , $L_p = 80_h = 128_d$

số đồng dư $N = \text{eec19b75218d08ad5516deceec1cf4af3824a95691ab41806865e09ffd51fbfa54d}_h$

$E = \text{f000}_h$

số đồng dư $p = \text{ceced8f8b6b854189f8d6b39b75c1329}_h$

Xâu dữ liệu $D = \text{'Now is the time for all'}$

Xâu dữ liệu D được mã hóa theo ISO 646 trong định dạng cơ số 16:

$4e6f77206973207468652074696d6520666f7220616c6c20_h$

Độ dài của xâu dữ liệu D , $L_D = c0_h = 192_d$

CHÚ THÍCH: – Mã ISO 646 thường được gọi chung là mã ASCII vì nó tương đương với mã ASCII 7 bit. Mã ASCII 7 bit được mở rộng thành 8 bit bằng cách chèn một bit 0 ở vị trí đầu tiên.

Mỗi bước được ghi lại để chỉ ra các giá trị thay đổi của các biến i, A, B, C và C_0, C_1, C_2, C_3 trong suốt giai đoạn tính toán. Cột đầu tiên tham chiếu tới các bước được mô tả trong phụ lục A.1.

A.2.1 Ví dụ tính toán băm dùng MASH-1

Bước khởi tạo:

$A := 0$

$B := 0$

$i := 0$

Vòng đầu

(2a)	C	=	4e6f7206973207468652074696d6520 _h	16 _d byte đầu tiên của chuỗi dữ liệu D
	i	=	80 _h	
(2b)	B	=	f4feef6ff7f7f2f0e6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6f6f5f2f0 _h	
(2c)	B	=	f4feef6ff7f7f2f0e6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6f6f5f2f0 _h	
(2d)	B	=	f4feef6ff7f7f2f0e6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6f6f5f2f0 _h	
(2e)	B	=	00000d87836ef6fbfe7f050ffa9c69d6c5ebab11b7f42df62f56904a924f169bf41b39f1 _h	
(2f)	A	=	836ef6fbfe7f050ffa9c69d6c5ebab11b7f42df62f56904a924f169bf41b39f1 _h	
(2g)	A	=	836ef6fbfe7f050ffa9c69d6c5ebab11b7f42df62f56904a924f169bf41b39f1 _h	

Vòng thứ 2

(2a)	C	=	666f7220616c6c20 _h	8 _d byte cuối cùng của chuỗi dữ liệu D
(2a)	C	=	666f7220616c6c200000000000000000 _h	(...được đệm với các bit 0 nhị phân)
	i	=	c0 _h	
(2b)	B	=	f6f6f6ff7f7f2f0e6f1f6fcf6fcf2f0f0f0f0f0f0f0f0f0f0f0f0f0f0f0 _h	
(2c)	B	=	75980004098df7ff0c6d9f2a331759e14704dd06dfa660ba62bfe66b04ebc901 _h	
(2d)	B	=	f5980004098df7ff0c6d9f2a331759e14704dd06dfa660ba62bfe66b04ebc901 _h	
(2e)	B	=	000004ebdaecc33c684979d2695461839008637c449e153cd57ef76036361b335bc941f _h	
(2f)	A	=	daecc33c684979d2695461839008637c449e153cd57ef76036361b335bc941f _h	
(2g)	A	=	598235c796367cdd93c8085555e3c86df36a38ca62017f3c912c7728c1a7adee _h	

Vòng thứ q (Bộ đếm độ dài)

(3a)	C	=	00000000000000000000000000000000 _h	8 _d byte cuối cùng của chuỗi dữ liệu D
(2b)	B	=	f0 _h	
(2c)	B	=	a972c53766c68c2d6338f8a5a513389d039ac83a92f18fcc61dc87d83157511e _h	
(2d)	B	=	f972c53766c68c2d6338f8a5a513389d039ac83a92f18fcc61dc87d83157511e _h	
(2e)	B	=	0000025b1e0a335d8cd76a299d717e7a2bf111d5ea3b3ff6cb56afae0b84cf3ee98c818a _h	
(2f)	A	=	1e0a335d8cd76a299d717e7a2bf111d5ea3b3ff6cb56afae0b84cf3ee98c818a _h	
(2g)	A	=	4788069a1ae116f40eb9762f7e12d9b81951073ca957d0929aa8b816282b2c64 _h	
	H _q	=	4788069a1ae116f40eb9762f7e12d9b81951073ca957d0929aa8b816282b2c64 _h	

Hàm rút gọn RED

Phân tích

(4a)	H _{q1}	=	4788069a1ae116f4 _h	(4b)	C ₀	=	H _{q3}	=	1951073ca957d092 _h
	H _{q2}	=	0eb9762f7e12d9b8 _h		C ₁	=	H _{q1}	=	4788069a1ae116f4 _h
	H _{q3}	=	1951073ca957d092 _h		C ₂	=	H _{q4}	=	9aa8b816282b2c64 _h
	H _{q4}	=	9aa8b816282b2c64 _h		C ₃	=	H _{q2}	=	0eb9762f7e12d9b8 _h

Mở rộng

(4c)	C	=	C ₀ C ₁	=	1951073ca957d0924788069a1ae116f4 _h
------	---	---	----------------------------------	---	---

Vòng thứ q+1

(2b)	B	=	f1f9f5f1f0f7f3fcfa9f5f7fd0f9f2f4f7f8f8f0f6f9faf1fafef1f1f6ffef4 _h	
(2c)	B	=	b671f36bea16e508f44083d883e2204aeda6fffc459a129686b5246e7d9ddd390 _h	
(2d)	B	=	f671f36bea16e508f44083d883e2204aeda6fffc459a129686b5246e7d9ddd390 _h	
(2e)	B	=	000004beda0f78385d1a959a02687ab92ca6051b91e092f60b72105a2c3fa4059d239311 _h	
(2f)	A	=	da0f78385d1a959a02687ab92ca6051b91e092f60b72105a2c3fa4059d239311 _h	
(2g)	A	=	9d877ea247fb836e0cd10c9652b4dca388b195caa223c0c8b6971c13b508bf75 _h	

Mở rộng

(4d) C = C2 || C3 = 9aa8b816282b2c640eb9762f7a12d9b8b

Vòng thứ $q+2$

(2b) B = f9fafaf8fbf8f1f6f2f8f2fbf2fcf6f4f0f6fbf9f7f6f2fff7fe1f2fdcf9fbf8h
(2c) B = f47d4845abc037298f8e29f6e0a482a57784f6e3355d32374169ede14811448dh
(2d) B = 6474845abc037298f8e29f6e0a482a57784f6e3355d32374169ede14811448dh
(2e) B = 0000741c0abe2c27ac0bf2384d377ab7b07f3374c402f87ec26bnafabdf2e360f7000bh
(2f) A = c0abe2c27ac0bf2384d377ab7b07f3374c402f87ec26bnafabdf2e360f7000bh
(2g) A = 5d2c9c603d3b3c4d88027b3cd29b32f94c4f1ba4d4e037a671d41eeef0d5fbb7eh

Vòng thứ $q+3$

(4a) $C_0 = C_0 \oplus C_3 = 17e87113d745092e_h$
 (4a) $C_1 = C_1 \oplus C_0 = 50607789cda1fde_h$
 (4a) $G = C_0 \parallel C_1 = 17e87113d745092e50607789cda1fde_h$

```

(2b) B = f1f7fe5f8f7f1f1f3fd7f4f5f0f9f2fa5f0f6f0f7f7f8f9fcfdfaF4f1ffdfdeh
(2c) B = acdb6298cacacdbb75f58fc8d94add6e31014cbdb9f48299ealbc140424004280h
(2d) B = fcd6298cacacdbb75f58fc8d94add6e31014cbdb9f48299ealbc140424004280h
(2e) B = 0000085f0e394019c43439b100f3865b24c54196d0a6612bc8298ae325583501f5fe139bh
(2f) A = 0e394019c43439b100f3865b24c54196d0a6612bc8298ae325583501f5fe139bh
(2o) A = 3315dc79f90f03fc88f1fd66d0766e021457db66862af4843819dbf12D001ace5h

```

Vòng thứ $q+4$

(4f) C₂ = C₂ ⊕ C₁ = cac8cf9fe58f33ba_h
(4f) C₃ = C₃ ⊕ C₂ = c471b9b09b9dea02_h
(4f) C = C₂ || C₃ = cac8cf9fe58f33bac471b9b09b9dea02_h

(2b) B = fcfefcfc58fefeff9ffef5f8ffff3f3fbfafcf4f721fbfb9fbf0f9fbf9fdfeaf0f0f2_h
(2c) B = afef208105f0fc0376040599fef8595f8e8a32cp77d30f74cle2220cdcfb5c17_h
(2d) B = ffe208105f0fc0376040599fef8595f8e8a32cp77dd30f74cle2220cdcfb5c17_h
(2e) B = 000009ad067af1cb50e126496355f4ef8033d62330dd0c151703d398534f4ale89e28d4ah_h
(2f) A = 067af1cb50e126496355f4ef8033d62330dd0c151703d398534f4ale89e28d4ah_h
(2g) A = f37273654dc0117981ec4a2298d45b821248ad7739129271c6bc5691efa9e321af_h

Vòng thứ $q+5$

(4e) $C_0 = C_0 \oplus C_3 =$ d399c8a34cd8e328h
(4e) $C_1 = C_1 \oplus C_0 =$ 83f9bf2a817cfcf6h
(4e) $C = C_0 \parallel C_1 =$ d399c8a34cd8e32883f9bf2a817cfcf6h
(2b) $B =$ fdf3f9f9cf8faf3f4fcfd8fe3f32f8f8f3fff9fbfff2faf8f1f7fcffcfcff6h
(2c) $B =$ 0e818a9cb0f9ed6bea385fd173b64ad9dc79288a6ad6d5e693a76613561fde59h
(2d) $B =$ fe818a9cb0f9ed6bea385fd173b64ad9dc79288a6ad6d5e693a76613561fde59h
(2e) $B =$ 0000013eb89ff8f208f8dd6af54caf0c5783bcb901c9ed5dc70ef255eb64da4fbafe19a2h
(2f) $A =$ b89ff8f208f8dd6af54caf0c5783bcb901c9ed5dc70ef255eb64da4fbafe19a2h
(2g) $A =$ 4bedfcf744f9cacf25ab880d25dac6049825433a2e5627d54980324ba0131d380d

Vòng thứ $q+6$

(4f) $C_2 = C_2 \oplus C_1 = 493170b564f3cf4c_h$
 (4f) $C_3 = C_3 \oplus C_2 = 8d40c905ff6e254e_h$
 (4f) $C = C_2 \parallel C_3 = 493170b564f3cf4c8d40c905ff6e254e_h$

(2b) B = f4f9f3f1f7f0fbf5f6f4fff3fcff4f4cf8fdf4f0fcf9f0f5fffff6fef2f5f4feh
(2c) B = bf140f06b30931071d7cf2d62639f064ddbcbcedaead25bc7fcdbbd5ee1e8ccf3h
(2d) B = ff140f06b30931071d7cf2d62639f064ddbcbcedaead25bc7fcdbbd5ee1e8ccf3h
(2e) B = 000002cd631f7f2804f90933a27240f80ea77ae668d319c6157c23f7b11684f2d7080aceh
(2f) A = 631f7f2804f90933a27240f80ea77ae668d319c6157c23f7b11684f2d7080aceh
(2g) A = 28f283df4000c3c149fa4d4dd46177a0e4d9023e8435b76be3124cf52c41532c3h

Vòng thứ 6+7

```
(4a) C0 = C0 ⊕ C3 = 5ed901a6b3b6c666h
(4a) C1 = C1 ⊕ C0 = dd20be8c32ca3a90h
(4a) C = C0 || C1 = 5ed901a6b3b6c666dd20be8c32ca3a90h

(2b) B = f5fecdf2f0f1fa66fb3fbf6fcf6f6f6fdfd2f0fbf8f8fc3f2fcfaf3fa9f90h
(2c) B = dd0c7e26b0f13937b209b62b28978cf8b06dd118b8a50e42c2d633a837efcb33h
(2d) B = fd0c7e26b0f13937b209b62b28978cf8b06dd118b8a50e42c2d633a837efcb33h
(2e) B = 00000b5c780401af81fcafb61d0af544bb9533b074260c5b9536bc8cc8d89c021c8cc326h
(2f) A = 780401af81fcafb61d0af544bb9533b074260c5b9536bc8cc8d89c021c8cc326h
(2a) A = 50f68279c1cf6c67754f0b8996ff449ba39b62fb3d66d4a32f9fc33d0899f1e5h
```

Vòng thứ $q+8$

```
(4f) C2 = C2 ⊕ C1 = 9411ce395639f5dch  

(4f) C3 = C3 ⊕ C2 = 1951073ca957d092h  

(4f) C = C2 || C3 = 9411ce395639f5dc1951073ca957d092h  

(2b) B = f9f4f1f1fcfe3f9f5f6f3f9ff5fdcf1f9f5f1f0f7f3fcfa9z5f75df0f9f2h  

(2c) B = a90273813d029f8ea1064b609001b442c84fda42269ab9ce0305a6a725690817h  

(2d) B = f90273813d029f8ea1064b609001b442c84fda42269ab9ce0305a6a725690817h  

(2e) B = 00000d67969376fb27ef9563bd5bd2b08156dd01904bc7ac090552607e69252ba9265edh  

(2f) A = 7969376fb27ef9563bd5bd2b08156dd01904bc7ac090552607e69252ba9265edh  

(2g) A = 299fb51f738295216f2505b267e1246e20b293c916fd1f14fe1ac102620b9408h  

Hn+8 = 299fb51f738295216f2505b267e1246e20b293c916fd1f14fe1ac102620b9408h
```

Rút gọn mod p

(5) A = 497a26f552e76e98ea6eaae0e54b6d7d_h Mã băm được tính toán bởi MASH-1
(6) H = 497a26f552e76e98ea6eaae0e54b6d7d_h

A.2.2 Ví dụ tính giá trị băm sử dụng MASH-2

Bước khởi tạo:

$$A := 0$$
$$B := 0$$
$$f \equiv 0$$

Vòng đầu

(2a) C = 4a6f77206973207468652074696d6520h, 16_d byte đầu tiên của xâu dữ liệu D
I = 80_h

(2b) B = f4fe6ff7f7f72f0f6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6fdf6f5f2f0h

(2c) B = f4fe6ff7f7f72f0f6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6fdf6f5f2f0h

(2d) B = f4fe6ff7f7f72f0f6f9f7f3f2f0f7f4f6f8f6f5f2f0f7f4f6f9f6fdf6f5f2f0h

(2e) B = 0000083b71d3c375922382b7996401171c44350ffc63f85a1laebbf07e182108ba9d1f3fh

(2f) A = 71d3c375922382b7996401171c44350ffc63f85a1laebbf07e182108ba9d1f3fh

(2g) A = 71d3c375922382b7996401171c44350ffc63f85a1laebbf07e182108ba9d1f3fh

Vòng thứ 2

[illegible]

Vòng thứ q (Bỏ đếm đồ dài)

[illegible]

Hàm rút gọn RED

Phân tích

(4a)	$H_{q1} = 7c032998b0ba9a3f_h$ $H_{q2} = a27d0232928e1359_h$ $H_{q3} = adc25ceb217f1mc2_h$ $H_{q4} = 14f40fee6821cf2c_h$	(4b)	$G_0 = H_{q3} = adc25ceb217f1mc2_h$ $G_1 = H_{q1} = 7c032998b0ba9a3f_h$ $G_2 = H_{q4} = 14f40fee6821cf2c_h$ $G_3 = H_{q2} = a27d0232928e1359_h$
------	--	------	--

Mở rộng

(4c) $C = C_0 \parallel C_1 = \text{adc25ceb217f1ac27c032998b0ba9a3fh}$

Vòng thứ $q+1$

(2b) B = fdfcf2f5fcefbb2f1f7fff1fafcf2f7fc0f3f2f9f9f8fb0fbfaf9faf3ffh
(2c) B = 86fed56a454664c4508cf5cd6374efab5a3eac18d386e33aef04f41491db3cd3h
(2d) B = f6fed56a454664c4508cf5cd6374efab5a3eac18d386e33aef04f41491db3cd3h
(2e) B = 00000632c4c63d04f673f195bd0d7d79212c0b1a6d1037d07c20d31f946c07a954b480ecdh
(2f) A = 4c63d04f673f195bd0d7d79212c0b1a6d1037d07c20d31f946c07a954b480ecdh
(2g) A = 3060f9d7d785836472aad5a0804ea2ff7cc121eccc3722b355234757b236961e1h

Mở rộng

(4d) $C \equiv C_2 \parallel C_3 =$ 14f40fea6821cf2ca27d0232928e1359_h

Vòng thứ $\sigma+2$

(2b) B = f1f4ffff4f0ffefef6f8f2f1fcff2f2cfaf2f7fd0f0f2f3f2f9f2f8fef1f3f5f9h
(2c) B = c1940623277a7d9a945227517cbl50038633d6111380d8c9abc66d8d5429a3418h
(2d) B = f1940623277a7d9a945227517cbl50038633d6111380d8c9abc66d8d5429a3418h
(2e) B = 00000c2b4825db0a9091297c530abd55dffcf5b93a28f2c0c65cc50299a52bf7602d872beh
(2f) A = 4825db0a9091297c530abd55dffcf5b93e28f2c0c65cc50299a52bf7602d872beh
(2o) A = 784522d24714aa1821a068f55f1f96c94ade0d86ba7b12d866ca2cd21b1b35f5h

Vòng thứ $q+3$

```
(4e) C0 = C0 ⊕ C3 = 0fbf5ed9b3f1099bh  

(4e) C1 = C1 ⊕ C0 = 73bc7741034b93a4h  

(4e) C = C0 || C1 = 0fbf5ed9b3f1099b73bc7741034b93a4h  

(2b) B = f0fffbfxfxfxfdf9fbf3fff1f0f9f9fbf7f3fbfczf7f7f4f1f0f3f4fbf9f3fa4h  

(2c) B = 88bad922b2aa57e1da539704af48009769bdf61c71498fe338953ef6d84249abh  

(2d) B = fb9bd922b2aa57e1da539704af48009769bdf61c71498fe338953ef6d84249abh  

(2e) B = 000002595aaa22afa6266de434fa9395ae40ff1dc667735f1e6e5f603e5c1ea79c94d8e9h  

(2f) A = 5aa22afa6266de434fa9395aa40ff1dc667735f1e6e5f603e5c1ea79c94d8e9h  

(2g) A = 22ef0072e132c7fc755afb60f1f1067158297ebf98d02472f63ad4aabdb36bb6h
```

Vòng thứ $q+4$

(4f) $C_2 = C_2 \oplus C_1 = 674878af6b6a5c88h$
 (4g) $C_3 = C_3 \oplus C_2 = c5357a9df9e44fd1h$
 (4f) $C = C_2 \parallel C_3 = 674878af6b6a5c88c5357a9df9e44fd1h$

(2b) $B = f6f7f4f8f7f8fafef6fb6faf5fcf8f8fcf5f3f5f7faf9fdff9fef4f4ffdf1h$
 (2c) $B = d418f48a16ca3d0383a10d9a040dfe89a4dc8d4a6f2add8f09c32a5e49da9647h$
 (2d) $B = f418f48a16ca3d0383a10d9a040dfe89a4dc8d4a6f2add8f09c32a5e49da9647h$
 (2a) $B = 00000644434f5a752c3635d88ea144e3d28a4613eb15035e1b9c47c8ec14e50b95623ebh$
 (2f) $A = 4434f3a752c3635d88ea144e3d28a4613eb15035e1b9c47c8ec14e50b95623ebh$
 (2n) $A = 66dbf5d3b3f1a4a1fdb0ef2eccd9e21066982e8a7969e00e78fb9afad073485dh$

Vòng thứ $q+5$

```
(4a) C0 = C0 ⊕ C3 = ca8a2444a15464a9h
(4a) C1 = C1 ⊕ C0 = b9365305495ed5ee9h
(4a) C = C0 || C1 = ca8a2444a15464ab9365305495ed5ee9h

(2b) B = fcfaf8fa2f24f4f4f4f4f1f5f4f6f4fabf9f3f6f5f3f0f5f4f9f5fefdf5fefeh
(2c) B = 9a210d2f41055055094a1edb382f16ea9d61dd7c8c9a10fb8c026f04f986b6a3h
(2d) B = fa210d2f41055055094a1edb382f16ea9d61dd7c8c9a10fb8c026f04f986b6a3h
(2e) B = 00000b472e240ae982562be4723c0f7818292619525274b24a8b64f89ea41d1d89a764a0h
(2f) A = 2e240ae982562be4723c0f7818292619525274b24a8b64f89ea41d1d89a764a0h
(2g) A = 486ff13c31a78f485f8ce056d4f0c40934ca5a3833e284f6e65f47e78dd42cfdh
```

Vòng thứ $q+6$

```
(4f) C2 = C2 ⊕ C1 = de7e2baa22348966h
(4f) C3 = C3 ⊕ C2 = 1b4b5137dbd0c6b7h
(4f) C = C2 || C3 = de7e2baa223489661b4b5137dbd0c6b7h

(2b) B = fdfe7f7ef2fbfafef2f2f3f4f8f9f6f5f1fbf4fbf5f1f3f7fdfbfd0fc6f6fbf7h
(2c) B = b50106c2c35c75bf7d7e13a22c0932ffc531aec3c61377011ba47e177122d70ah
(2d) B = f50106c2c35c75bf7d7e13a22c0932ffc531aec3c61377011ba47e177122d70ah
(2e) B = 000002cac6f3b0a9d6b41c106b1ef6465385af56258886f78d80f016914d005f66f51db2h
(2f) A = c6f3b0a9d6b41c106b1ef6465385af56258886f78d80f016914d005f66f51db2h
(2g) A = 8e0c4195e7139355e492161087756b5f1142dccfbc6274e0771287b8eb21314fh
```

Vòng thứ $q+7$

```
(4a) C0 = C0 ⊕ C3 = d1c1757391c580fd6h
(4a) C1 = C1 ⊕ C3 = 68f72676d89b5513h
(4a) C = C0 || C1 = d1c1757391c580fd68f72676d89b5513h

(2b) B = fdf1fcf1f7f5f7f3f9f1fcf5f8f0ffdf6f8fff7f2f6f7f6fdf8f9fbf5f5f1f3h
(2c) B = 73fdbd6410e664a61d63ae57f8594a2e7ba23384c9483168aa7a431ed4c0bc h
(2d) B = f3fdbd6410e664a61d63ae57f8594a2e7ba23384c9483168aa7a431ed4c0bc h
(2e) B = 0000013ef76551c6e3b17dfb5ecb837566eac9c7fb39f16c39351b9fd4bdac352eaf01dh
(2f) A = f76551c6e3b17dfb5ecb837566eac9c7fb39f16c39351b9fd4bdac352eaf01dh
(2g) A = 7969105304a2eaaaba399565e19ba298aa7b2da387576f7fa3af2b0bb9bcb152h
```

Vòng thứ $q+8$

```
(4f) C2 = C2 ⊕ C1 = b6890ddcfafdc75h
(4f) C2 = C3 ⊕ C2 = adc25ceb217f1ac2h
(4f) C = C2 || C3 = b6890ddcfafdc75adc25ceb217f1ac2h

(2b) B = fbf6f8f9f0fdffdfcfefafafffdcfcf7f5fafdfcf2f5fcfefbf2f1f7fff1fafcf2h
(2c) B = 829fe8aaf45f135245a36f9a1c67556d1086d15172ab9184515edcf448313da0h
(2d) B = f29fe8aaf45f135245a36f9a1c67556d1086d15172ab9184515edcf448313da0h
(2e) B = 00000a8bf084975e7ac8e096b464172ed9a5cbffeb32c5512faec0b3702f427608497adbh
(2f) A = 834975e7ac8e096b464172ed9a5cbffeb32c5512faec0b3702f427608497adbh
(2g) A = 89ed870d7e6a0e380e3d824b383e69670149e8f2a8f9afcccd380697db182bbeafh
H... = 89ed870d7e6a0e380e3d824b383e69670149e8f2a8f9afcccd380697db182bbeafh
```

Rút gọn mod p

(5) A = 8ad87c2de674c2e82de5769806e1bb28h
(6) H = 8ad87c2de674c2e82de5769806e1bb28h

Mã băm được tính toán bởi MASH-2

A.3 Thông điệp kiểm tra mẫu và giá trị băm của chúng

Theo tính toán của mã băm cho tập hợp các thông điệp kiểm tra (A.3.1 tới A.3.9), các số nguyên sau đây được sử dụng:

1) Hợp số đồng dư

$N = 8b251fa16f8b7a3c8a1ac50da421de6bdfc4db6cf4452d0df98ad327b9f6feca66422e1434938ffe3576d7b7a76b8c94a90dc9d0cc576bd6f9a128f4af1d907e3b4dbb67f52683e7992a4cf031f885533d21d07c7e14811adb600fb78d62a789f_h$

2) Số nguyên tố

$$p = cde7e6f6e432331d896a7b02d031a09d7b2c77a1_h$$

Chiều dài của số đồng dư N sử dụng trong hàm vòng là $L_N = 780_d = 30C_h$

Chiều dài đầu ra của hàm vòng là $L_\phi = 768_d = 300_h$

Chiều dài của số nguyên tố p sử dụng trong hàm rút gọn là $L_p = 160_d = A0_h$

Chiều dài của mã băm là $L_\rho = 160_d = A0_h$

Những nội dung của các trường:

<i>length of text</i>	Số bit của thông điệp. Được đưa ra theo dạng thập phân đầu tiên (với kí hiệu là d) tiếp theo là biểu diễn dạng thập lục phân tương đương (với ký hiệu là h)
<i>text</i>	Chứa thông điệp sẽ được băm phân định bởi các dấu ngoặc đơn
<i>text in ASCII</i>	Chứa thông điệp theo mã ASCII. Những khoảng trống được chèn vào cho dễ đọc
H_q	Chứa đầu ra của hàm vòng theo định dạng thập lục phân sau q ứng dụng
H_{q+8}	Chứa đầu ra của hàm vòng theo định dạng thập lục phân sau $q+8$ ứng dụng
H	Chứa mã băm theo định dạng thập lục phân sau phép rút gọn modulo p

A.3.1 Ví dụ 1

<i>length of text</i>	$0_d = 0_h$
<i>text</i>	"
<i>text in ASCII</i>	(empty string)
MASH-1 H_q	a69e5191337162387c3a2701bc8a49ad385d648194fa1e188ca2d425f149fdf89cea0e9c1324fd96270a1fee9a77e00925a014432896dd6a54232947f222b28b786259c45de5496b166e4955cd0c84f2539c49fb61f6fc66763915dd006e8b8c
H_{q+8}	c42af288cf830875401c6e622a417058076b4869183cf5d79721a33c5d297937f90506891cbd0e0d307d8515ec3dd5d64bd6257e90ba06acf53c14c47cb007e54ae429dfa0329fff22cd1fdb2702598258d081d4f0d25a612084f906c2f5627
H	b7231fa49ce0249adef0ce0a1429796e8f19ad8b

MASH-2	H_q	499bcae76a9b15f8061fc0fc2da49d7d8f1ca7436a70c36d564ac35875cdcf310f3a8e8791e358774669341bd07f7917cd35928f9ee60a88f7c53cecb2d5e82f9778049b24243b896a3e66448801ab6f0b975018a7732a87baa3c1ac07e95eaf
	H_{q+8}	416000041591700f319a8e2bbc2bab56e88cd39dd24194da16e81b5b6db15d9ca22849f7ad7ae6d83987d92c1cdfdc65775c610740b935d1258f568af3038a79d8a7b97000aec763b960c27a63245d91d3c58f21bd386fb9b8600f4765d94efe
	H	bcce0b6f5646e7eb0cfb1ed6dbe787016ce4b18e

A.3.2 Ví dụ 2

	length of text	$8_d = 8_h$
	text	'a'
	text in ASCII	61
MASH-1	H_q	843d8fbd095cb55514b0017843541ea3003c03b6b9753501ad28e28f8e3d5a125461f8a6db9fc6b6db3475916915bb50dd71a833647710db0b71c4696570b37de678d185c99e285e496be7c5415130c2ee51108e4fe38f0e80ec4cfed7bb3c9f2
	H_{q+8}	b7cd378165eb4ed4ca34f1372643d57d395810e696959eb44f3514be6413c9131817ac827a8f501961ac3cb0a51e2ecb32c990443f2eac32890b81ecf5874eabccc97ef292870ab945c00eb0555a4ede7ab9e23cca450bfdafe46921e2a40d25
	H	c7c5e8262a0067a3f0479fdc5c814e90e636610b
MASH-2	H_q	4851adcd1809c0b66018a4f62a0008ea08e25fa5a6a8f970d0f09e3be81276301d68e5da78314ecae818a32589132a22caacdbe5f4e06afe7273632ef218255f4bd3c58d3fc3fd4357411c21047e77017c0358401a013f251638f3f4764ef5a4
	H_{q+8}	597292939e469da0fa542227c5e8c6a4eade4497433cc3ca8db32e473c23287af7dfcfdab7e1fe83723697772acc19ef5206d681bdab2b6fa3253fec58980284dda2fc077fbd0492e125c204af1f4625f439968115619e517ab05b74ee60eb0c
	H	c8ae72da006544d524a6ca09c9d7702628bfb683

A.3.3 Ví dụ 3

	length of text	$24_d = 18_h$
	text	'abc'
	text in ASCII	61 62 63
MASH-1	H_q	329f3abba8d9ffde27478023b0fba35bf1abaa31fdf7e32a9a314fb43769a3809bb654a94838382b544702263e6ff3bd8a21fc38f5ff5f06859d11aabd49af96912fed50ed7755ed3d1da0776ede4985ae1fd56dbe8cfe52e37eb7b724ec694
	H_{q+8}	5910c9e2d3fa19bda8c4143aaeeaac65f7acdf8d067b43891ee2d4626c0fc570188083e886d86d5fe89fded34e2b3ee75dfc5229bccc85b250b312339355b74c62e9429021b5a965e028ea3f5477687ae5d4858b46a480976b7e741d8ba0c8d0
	H	9e76dd088a4e88ea72bb9ce3727f34bd27c75912
MASH-2	H_q	3bbd7e23260830c085a6282c5ef2f2d35d7e78a74d0f3da4964bba9a74780b43662d9a666daba357abef2c3186129e3202f80a015de4ddc6811db472cf36420e0ccb17a9a937e1a89647eb7f84105c023617b914ba2faf8113070acb36edf6
	H_{q+8}	344d4e0f7627fbb5affdaa62610d096ce6951f9a17f191e56a25ba5d3dcc4b08a361ffe917b8ee53d0e0a39193fd43274b34f931b96a469518d5e7d60b1d794fe78c1740b0c42c7273e2b7b1fa7d8092d7a0c590957eb76caf7868339e6aa6b
	H	18086f7c5c96e1bdc659c1c7b5c957f3dfbae7c8

A.3.4 Ví dụ 4

length of text 112_d = 70_h
 text 'message digest'
 text in ASCII 6b 65 73 73 61 67 65 20 64 69 67 65 73 74

MASH-1 *H_q* 59ac98f8064c76e567e10fc19227fdb75e1711d60b1af2a51dd9788894daf4d543c541c
 30bd9e927e52196c0c3610330cd8004cdc06965b09d6c3ed5b5a454a2d05ade60620966
 5677d72b5a9a5e0aeebd565e1e99be2279a3208368cc02be99

H_{q+8} bc3f82db3e8b727264f2d6e1c5e078f6499309e7d15ccdf8ad78c356e5bc5835bcc934
 4c15efae7c08364c84f3f99d10df8e1a14bc74da4c52cab171ac01d0c69f751c825ce7e
 6eb2f5fcf878f2944d67e96a5cb14af0d83d85f720ad44689c

H 777c17eb3136b5db760261570210ae7f658cf6da

MASH-2 *H_q* 0c02d1efc59f9e78f68c35b76ff94247bbba0f9f92caff6f924d87e92239ad89078cde0
 9687a7573ff13bd13bfc4ad556514ea69f67aecf80cd1bf29347e0c5d0d54ed4ae876c
 26678454a0161a82f99ca6248a91d812ffe10ebc0750af3a99

H_{q+8} f64758641fe517e72d0ed551a2cc97c4bfacc36284895d5fe3621cf661bb2b972fcdde
 f06b35c5505aaeeab2ed41cc1cf7e9d7854b7441fd384b357982c098ed0b14c212ca3bf
 df7e5bfc7f671433503522fb88ded60af24d7b6b19c7c1215c

H 7bb3d180903bba2af00221ede0241343287320a8

A.3.5 Ví dụ 5

length of text 208_d = D0_h
 text 'abcdefghijklmnopqrstuvwxyz'
 text in ASCII 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 78
 79 7A

MASH-1 *H_q* 6441dc60fcc5a7790b7c587a9a40e631459b80e1acc252b7437178e0f8083af592f76d7
 f0a1bb0830ea9c85c2fb277b04cade9a2fbb97b6b7e69405ed8abda84d63cd563d834d
 95298ad99b7012266def8de2609c0590c7fbf6ef862ca3d7db

H_{q+8} 377dfad4021e44f5260f4122767909b7b4c18e742fd2a977f5978e19a997ae5bd3fb69c
 e1fb5d18b289dc040e8fd3aebeaa1e82e0117fec890db0a3511ee8b5cd0b5e51d2e9bbf
 12e43b795e50b8420d08d16b91c185b0f5ae80aeaf89fe2923

H 3a3f4bb5a818bff7cc2f0fdebceb379ba1251292

MASH-2 *H_q* 5d88e7fa3aa47dc288f8aea161bdeb66663d02eee5bf1c7669ce3d558b8983128bf37e0
 16f7bd8ad601c1b400c9af181c7da732aa6eece982440cbdd49f7813ba99cb6cb6e645e
 85c9246352c9d0680ca252aaf27021b3382b7ed396c49055d2

H_{q+8} 8c99eee3174263cd5c01d947ee41d6b2e28e8b8dae7e161bd4ab1fe5895e8171d780e
 a3691c123ffd39fd87315eac112a6195ecaec99a40256bd73eacc09ebfa92516f9ccced
 4a7e699bd91529c30585860f7b3dcd4d2dc68a002587383e56

H a1c36bfe55cc6fa3417ebd33d7161954282114f7

A.3.6 Ví dụ 6

length of text	496 _d = 1F0 _h
text	'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789'
text in ASCII	41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55 56 57 58 59 5A 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70 71 72 73 74 75 76 77 78 79 7A 80 81 82 83 84 85 86 87 88 89
MASH-1	<i>H_q</i> 3e456a50fb7c0b5b18a033bc92f2ef093bb6522cead219c5cce5a74956cd306b06ee2cc5 25f53739f934c95f0c1f8951e97724270d8c8754c2589f0c2bfff3f1f3358f7615bcde0c ce61c1703a3232d4403b4c7cbbfdbc676bffd91d777eabfc3f9
	<i>H_q+8</i> 1595f9efe2b51054969cd1f2071ea6bd1d31b3769d5ca908bb82d462dfaa0dcefdc80cc 8d29e33887006df0394ec8f6e6a2d96cfd9cae578362d0bb8ae0086907f4f6dfc96fabe 712ec826930a5d967b204cb27b246d80a1dcb0b564971521b0
	<i>H</i> a31a4e4699656d1abb4adfdc7a121ef29261315f
MASH-2	<i>H_q</i> edff2a68bba84db66268c92f5d65f24fb35aac2b63cfaac5a35b8de70f1a2542f0e91d0 89046cd071549611df7e71d735a29889c4a289929e5c2630da517874111e82777e11cf32 c1edf78a26638ca490cd8b754482e5f944c2bed7c094738b57
	<i>H_q+8</i> 07e467c92561ca22867eb55fc34a4d321b5ca2bc8ab327a6e3bc3d9bf2125e90da22ba7 a09a4b467618d151f0882c3fe29c06f0e1549ec070ac7bcbfac48a4cb95226dea668745 9749044c48670e9a65ec059f371f0f3c1f93211ca2677f0f67
	<i>H</i> 7cf0c17aa2f0f5fb29714855ad3046f075e1155f

A.3.7 Ví dụ 7

<i>length of text</i>		640 _d = 280 _h
<i>text</i>		'123456789012345678901234567890123456789012345678901234567890123 45678901234567890'
<i>text in ASCII</i>		31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30
MASH-1	H _q	95eb91b577348fe631954848ed02a368c880d09a890a676cc62250632b2fa70f8b13938 f994d0d9aca7e40303eab9e5863e28a47cb5880df7284b1d5e77443ff85bd85e1c50aa6 5a9276e84cd36755dc0404071cd78b2ab47dc5d53171e189984
	H _{q+8}	3bf6e2278383231af9f25f965a0a2e011c219861948c342b2245949e433579daac27407 3d51d3d9159738a5bc82741e3da635935bf1fff559da453378868dd18fd00536c32a146 1ed8b9f091b6ce88f6f021688602ae258b90bd64065cba02f3
	H	076e2907c13d0f4ead894d0f6ff3c90afc0ca668
MASH-2	H _q	bfe9db3bcc03590cf8495318ea5565c2368e6926795028b20b7ebbfb8a590831a9b296763 214426ad4501e1e56e9837ef0afe2bc0ad83abe0c21cc17c10933341a49c0195b5288ae 3197a21a145612dedb806ed28620ffe8162e857728f3dee328
	H _{q+8}	9691ac1b650337a854333015a957e0b3538633d9b4b1fe03fb809e032be712fee8ce618 efdb161bd7fe07715fcfed8019d1808a102f6e07d536e0501c9620a5b1a851ed91ddeff8 ac2991b46f0b0b279a70ddf08222b273e5271669b8db2397a4
	H	9f84f2ea61150871771fcbfd874a474aa73433a4

A.3.8 Ví dụ 8

```
length of text      448d = 100h
text                'abcdbcdecdefdefgfgfghfghighijhijkijklklmklmnlmnomnopnopq'
text in ASCII      61 62 63 64 62 63 64 65 63 64 65 66 64 65 66 67 65 66 67 68 66 67 68 69
                   67 68 69 6a 68 69 6a 6b 69 6a 6b 6c 6a 6b 6c 6d 6b 6c 6d 6e 6c 6d 6e 6f
                   6d 6e 6f 70 6e 6f 70 71

MASH-1  Hq      843f5939e80e0d705771a8c4acee8f9fa8b74fe416e230d7b5c43a016991421c258e08d
              c7d52116d06b81e6ebd4e7f7a737fbc583fb5f226ef07c8b3aacdb7265e145a9ac58b60
              874fdb0b9bfb75bba9893fefaab3a3bc138991c949cee290d

              Hq+s  aeabb3a0878961b9e7bd60e32686ac107e16b45265625b1962fca8e23fba867e0c4d910
                   2d77ecb81e4a5fcd4c2f143abde30a10bc741765f9d5c4c83a2d0f74c93df8466ddb62
                   6ff6da624c840bfcd9815e6a0f9c4ee57bc057af70e0309e09

              H      6bb2634d580719a727b1e4facef2c9facfbaec8a

MASH-2  Hq      299281d7e35cb7c90185ef5ca7844de96313f7d1a3add59ca7cd183cc9878fc46ec5d1
              9e50bfb10a9b9564949c75cbbfb32eecd3c5d332a34b23f1ee329d766841a5b5832fe4f
              a4bb0c84fc00918487132b9e0f12075590fd396ed7e15a8fc4

              Hq+s  3a94710994f16147aa1eb379abee459646708e524d3d0e7d31a540f8e83be310756468a
                   0cb41320354013d6792eb059db7c9e774b8b31ae973d0ca194d0ff2230ef425464744
                   657061107f6d040343b04883a5abbc4441e84901c023247097

              H      56f0b1aef4d275d46c0d50557af7b763b0ba180c2c
```

A.3.9 Ví dụ 9

		length of text text	8000000 _d = 7A1200 _h 1 Million 'a's
		text in ASCII	61 61 _ 61 61
MASH-1	H _q	8b5fe6058557f0122a98b60671f581e590b7a956a38aa3983964e3a46d47d9366cae18b f03d6a0c7beafb5ab1498582775b2391445fd4f6158997ac9b251be7cd005e8b29dda10 768589a4120be597228434dfe4c46d2995fab4a0b335b9b419	
	H _{q+8}	ee94b24125fbdccbb1f642e8038585f74847a57aa927b985bf923fff12637131ccf16c102 074f1bcdcf4346c164a61ff11a8788a13386f50e2b09a62027b9b25c4125bc5ab8ab36f4 c23f950450b776322a99158cafcbf11d2ac8c55eef2984272f	
	H	2b0b4c65e5aae33ac77687e81ce8fb3b29fd8069	
MASH-2	H _q	6f4c965b9a5c96fa1e9d808930b0a3c271e81ca0367a9f48dc1925a315ebfb8c3f1047e 0f44d83d90372f7fc313c307cdo98a38e0f364a0ab616d30a88148ebaa5fb9d720a58f8 c4fc0f0163981ce444722f984278e9f3674b4ee98bc89752929	
	H _{q+8}	8d73e319c037c294f5e4b2e2f94893cca708702ad52c9cad013ebcaf891f4b13998b01ec c907944a8ad49ffcf9f7c70b66778db2ec64314057afdf082a52068d18addca5637d68a31 4de1a29701da4c5b2e7b8e3339e147f6f3dc03719fc4860ac8	
	H	40262999c12dbe1cdc0913968266cbdfc6d508c	

Phụ lục B
(Tham khảo)

Thông tin bổ sung

B.1 Xác định các giá trị của tham số

TCVN 11816-4 không chỉ rõ các giá trị sẽ được sử dụng để tiếp cận đến một mức an toàn đã đề ra. Tuy nhiên, TCVN 11816-4 được đặc tả theo cách để tối thiểu hóa những sự thay đổi yêu cầu trong việc sử dụng nó nếu một trong các tham số này phải được sửa đổi.

Số nguyên tố p nên được lựa chọn cẩn thận, vì chiều dài bit của nó xác định kích cỡ của mã băm.

B.2 Sự khác nhau giữa MASH-1 và MASH-2

Lý do cho việc giới thiệu một phiên bản bổ sung của hàm băm với số mũ cao hơn (MASH-2) là để triệt tiêu bất kỳ tính chất thống kê nào có thể có với MASH-1. Đối với các yêu cầu an toàn rất cao, nên lựa chọn MASH-2 thay cho MASH-1

B.3 Hàm rút gọn

Hàm rút gọn RED được định nghĩa trong 7.3 để điều chỉnh chiều dài mã băm cho phù hợp với các yêu cầu về ứng dụng tiếp theo của mã băm H , và để tăng cường mức an toàn.

Phụ lục C
(Quy định)

Định danh đối tượng

Phụ lục này liệt kê các định danh đối tượng chỉ định cho hàm băm sử dụng số học đồng dư được đặc tả trong TCVN 11816-4.

```
--
-- ISO/IEC 10118-4 ASN.1 Module
--
HashFunctionsUsingModularArithmetic {
iso(1) standard(0) hash-functions(10118) part4(4)
asn1-module(1) hash-functions-using-modular-arithmetic(0))
DEFINITIONS EXPLICIT TAGS ::= BEGIN
-- Xuất ra toàn bộ; --
-- Nhập vào là None; --
OID ::= OBJECT IDENTIFIER -- Alias
-- Đồng nghĩa --
id-hfma OID ::= {
iso(1) standard(0) hash-functions(10118) part4(4) algorithms(0) }
-- Quy định --
id-hfma-mash1 OID ::= { id-hfma mash1(65) }
id-hfma-mash2 OID ::= { id-hfma mash2(66) }
-- Chú ý: Quy định cho OID mới có trên 66 thuật toán bổ sung --
}
END -- HashFunctionsUsingModularArithmetic --
```

Thư mục tài liệu tham khảo

- [1] BONEH D., & FRANKLIN M. *Efficient Generation of Shared RSA Keys*. Advances in Cryptology - CRYPTO '97 (BURTON S., & KALISKI Jr. ed.) Lecture Notes in Computer Science, Vol, 1294, Springer-Verlag, 1997, pp. 425-439.
 - [2] COCKS C. Split knowledge generation of RSA parameters. Cryptography and Coding. (DARELL M. ed.). Lecture Notes in Computer Science, Vol. 1355, Springer-Verlag, 1997, pp. 89-95.
 - [3] FRANKEL Y., MACKENZIE P.D., YUNG M. Robust efficient distributed RSA-key generation. Proceedings of the 30th Annual ACM Symposium on the Theory of Computing [STOC '98], ACM, 1998, pp. 663- 672
-