Một số phân tích về độ an toàn của cấu trúc xác thực thông điệp dựa trên hàm băm theo mô hình hàm giả ngẫu nhiên

Nguyễn Bùi Cương, Triệu Quang Phong, Nguyễn Tuấn Anh

***Tóm tắt*— Cấu trúc NMAC (Nest Message Authentication Code) và biến thể HMAC (Hash MAC) được đưa ra bởi Mihir Bellare, Ran Canetti và HugoKrawczyk vào năm 1996 ([1]). Tuy nhiên cho đến nay cấu trúc HMAC chỉ được phát biểu với một số nhận xét liên quan mà chưa có chứng minh về tính an toàn cụ thể nào cho mô hình này. Trong bài báo này, chúng tôi đánh giá mô hình an toàn NMAC theo cách tiếp cận của cấu trúc Băm-rồi-MAC (Hash then MAC) và đưa ra một giả thiết khác đối với hàm nén để chứng minh chi tiết độ an toàn của HMAC.**

***Abstract*— The NMAC structure and the variant HMAC was proposed by Mihir Bellare, Ran Canetti and HugoKrawczy in 1996 ([1]). However, the HMAC structure was only expressed with some related comments and it was not proved until now. In this paper, we assess the safety model NMAC follow Hash-then-MAC constructions's approach and provide a different hypothesis for the compression function to give detailed proof for the security of HMAC.**

***Từ khóa*— Hàm giả ngẫu nhiên; Mã xác thực thông điệp; Cấu trúc băm-rồi-MAC; Cấu trúc NMAC; Cấu trúc HMAC.**

***Keywords*— Pseudorandom functions; Message authentication; Hash-then-MAC construction; NMAC construction; HMAC construction.**

I. GIỚI THIỆU

Xác thực tính toàn vẹn và độ chính xác của thông tin là vấn đề thiết yếu trong trao đổi thông tin trên môi trường mạng máy tính. Đặc biệt, khi giao tiếp trên một kênh không an toàn, thì hai bên tham gia cần một phương pháp giúp xác thực tính toàn vẹn và chính xác của thông tin được gửi đi. Để giải quyết vấn đề này, người ta thường sử dụng các mã xác thực như một phương pháp để xác thực thông điệp. Điều này có nghĩa là khi người dùng X gửi một thông điệp cho người dùng Y thì X sẽ gửi kèm một giá trị gọi là mã xác thực thông điệp, được tính bởi thuật toán MAC dưới một khóa bí mật chung. Thuật toán này là một hàm của thông điệp và khóa bí mật. Sau khi nhận được thông điệp kèm theo mã xác thực từ X, Y sẽ tính lại mã xác thực cho thông điệp thông qua hàm MAC (theo giá trị bí mật chung), sau đó kiểm tra xem giá trị đó có trùng với mã xác thực mà anh ta nhận được hay không.

Ban đầu, các lược đồ mã xác thực thông điệp được đề xuất dành cho các thông điệp có độ dài cố định và thường được xây dựng dựa trên các hàm giả ngẫu nhiên. Sau đó, chúng được cải tiến sử dụng cho các lược đồ MAC cho phép xác thực các thông điệp có độ dài bất kỳ. Một trong những ý tưởng để thực hiện việc này là sử dụng hàm băm để băm thông điệp có độ dài bất kỳ xuống độ dài cố định, sau đó dùng các lược đồ xác thực cho thông điệp có độ dài cố định đã biết. Để thực hiện điều này, trước hết ta phải xây dựng được một hàm băm với đầu vào có độ dài tùy ý, bằng cách sử dụng phép biến đổi Merkle-Damgard. Ngoài ra, phép biến đổi này còn bảo toàn được tính kháng va chạm của hàm băm ban đầu, hay còn gọi là hàm nén.

Các công trình nghiên cứu cấu trúc NMAC và biến thể HMAC là những nghiên cứu tiêu biểu cho ý tưởng xây dựng các lược đồ MAC như trên. Hai cấu trúc này được đưa ra lần đầu tiên trong [1] vào năm 1996 bởi ba tác giả Mihir Bellare, Ran Canetti và HugoKrawczyk. Các tác giả đã chứng minh cấu trúc NMAC an toàn dựa theo tính kháng va chạm yếu của hàm băm và tính an toàn của lược đồ MAC, với đầu vào có độ dài cố định. Trong khi đó, tính an toàn của HMAC có thể được suy ra (chưa được chứng minh chi tiết) từ tính an toàn của NMAC bằng cách thêm giả thiết "hàm nén là giả ngẫu nhiên" ([1]). Ngoài ra, trong [2] các tác giả đã đưa ra một ý tưởng chứng minh cho tính an toàn của HMAC với giả thiết yếu hơn cho hàm nén là bộ sinh giả ngẫu nhiên.

Trong [4], tác giả Mihir Bellare đã chứng minh, nếu như hàm nén là giả ngẫu nhiên thì HMAC cũng có tính chất như vậy. Điều này

Bài báo được nhận ngày 23/11/2016. Bài báo được nhận xét bởi phản biện thứ nhất vào ngày 07/12/2016 và được chấp nhận đăng vào ngày 29/12/2016. Bài báo được nhận xét bởi phản biện thứ hai vào ngày 07/12/2016 và được chấp nhận đăng vào ngày 29/03/2017.

khẳng định rằng cấu trúc HMAC là an toàn chỉ cần dựa theo giả thiết giả ngẫu nhiên của hàm nén.

Trong bài báo này, dựa theo các kết quả của hàm giả ngẫu nhiên và lược đồ Merkle-Damgard, chúng tôi trình bày một số phân tích về giả thiết đảm bảo tính an toàn của cấu trúc NMAC, cũng như cấu trúc HMAC. Thay vì giả thiết về tính an toàn của hàm MAC với đầu vào có độ dài cố định trong [1], chúng tôi đưa ra giả thiết "giả ngẫu nhiên đều" của hàm nén để chứng minh độ an toàn của HMAC.

Bố cục của bài báo gồm 4 mục. Sau Mục Giới thiệu, Mục II trình bày một số định nghĩa cơ bản được sử dụng trong bài báo. Mục III đưa ra đánh giá độ an toàn của cấu trúc NMAC và HMAC dựa theo mô hình hàm giả ngẫu nhiên của chúng tôi. Và cuối cùng là Mục Kết luận.

II. MỘT SỐ ĐỊNH NGHĨA CƠ BẢN

Trong phần này, chúng tôi nhắc lại một số định nghĩa cơ bản, trong đó có sử dụng một số khái niệm cụ thể như sau:

Tham số an toàn dùng để chỉ độ dài của khóa.

Hàm không đáng kể f được gọi là một hàm không đáng kể nếu tồn tại số nguyên dương N sao cho với mọi $n \geq N$ thì $f(n) < n^{-c}$.

Bộ tiên tri được coi như một hộp đen mà khi kẻ tấn công đưa vào một giá trị đầu vào nó sẽ đưa ra một giá trị đầu ra. Việc tính toán trong bộ tiên tri phụ thuộc vào cơ chế của nó, có thể sử dụng một hàm, một thuật toán.…

A. Hàm giả ngẫu nhiên

Một hàm phụ thuộc khóa là một ánh xạ $F: \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$, trong đó $\mathcal{K}$ là tập khóa của F còn $\mathcal{X}$ và $\mathcal{Y}$ tương ứng là miền xác định và ảnh của F. Tập khóa và ảnh là hữu hạn, các tập trên là khác rỗng. Hàm hai đầu vào F lấy một khóa k và một đầu vào x sau đó trả lại y, ta ký hiệu là $F(k, x)$. Với mọi khóa $k \in \mathcal{K}$ ta định nghĩa ánh xạ $F_k: \mathcal{X} \to \mathcal{Y}$ như sau $F_k(x) = F(k, x)$.

Định nghĩa 1 ([3], tr65). *Cho $F: \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ là một hàm phụ thuộc khóa và hiệu quả, gọi A là một thuật toán tấn công thời gian đa thức xác suất được quyền truy cập lên bộ tiên tri là hàm $g: \mathcal{X} \to \mathcal{Y}$ và trả lại một bit. Ta xét hai thí nghiệm dưới đây:*

$\mathbf{Exp}_F^{\mathrm{prf}-1}(A)$	$\mathbf{Exp}_F^{\mathrm{prf}-0}(A)$
$k \xleftarrow{\$} \mathcal{K}$	$g \xleftarrow{\$} Func(\mathcal{X}, \mathcal{Y})$
$b \leftarrow A^{F_k}$	$b \leftarrow A^{g}$
Trả về b	*Trả về b*

Lợi thế giả ngẫu nhiên của A được định nghĩa là

$$\mathrm{Adv}_F^{\mathrm{prf}}(A) = \Pr\left[\mathrm{Exp}_F^{\mathrm{prf}-1}(A) = 1\right] - \Pr\left[\mathrm{Exp}_F^{\mathrm{prf}-0}(A) = 1\right]. \quad (1)$$

F được gọi là giả ngẫu nhiên nếu như với mọi A tồn tại hàm không đáng kể negl *sao cho*

$$\mathrm{Adv}_F^{\mathrm{prf}}(A) \leq \mathrm{negl(n)},$$

với n là tham số an toàn.

Trong một số trường hợp ta xét hàm phụ thuộc khóa với đầu vào thứ hai là tập khóa, khi đó ta cũng có định nghĩa tương tự hàm giả ngẫu nhiên như ở trên. Để tránh nhầm lẫn trong trường hợp này, ta ký hiệu là $F^k(x) \coloneqq F(x, k)$. Đặc biệt, chúng tôi đưa thêm khái niệm *hàm giả ngẫu nhiên đều* nếu như nó là một hàm giả ngẫu nhiên khi xét tập khóa là đầu vào thứ nhất đồng thời nó cũng là một hàm giả ngẫu nhiên khi xét tập khóa là đầu vào thứ hai.

B. Hàm băm

Hàm băm thông thường được hiểu đơn giản là hàm đưa ra một chuỗi có độ dài ngắn hơn chuỗi đầu vào. Một trong những yêu cầu quan trọng của hàm băm là tính kháng va chạm, nghĩa là rất khó để tìm được hai đầu vào khác nhau sao cho chúng có cùng giá trị băm. Trong phần này, chúng tôi sẽ nhắc lại định nghĩa tổng quát về hàm băm. Lưu ý rằng, ở đây chúng tôi xem xét đến các hàm băm phụ thuộc khóa. Điều này có nghĩa hàm băm H với hai đầu vào là khóa s và chuỗi x, đưa ra chuỗi $H^s(x) \coloneqq H(s, x)$. Với khóa s được sinh ngẫu nhiên (thường được công khai) thì rất khó để tìm được một va chạm của hàm băm H^s.

Định nghĩa 2 ([2] tr154). Một hàm băm (với đầu ra độ dài l) là một cặp thuật toán thời gian đa thức xác suất (Gen, H) thỏa mãn:

- Gen là thuật toán xác suất với đầu vào là tham số an toàn 1^n và đưa ra khóa s. Ta giả sử rằng 1^n được ẩn trong s.
- H với đầu vào là khóa s và chuỗi $x \in \{0,1\}^*$ đưa ra chuỗi $H^s(x) \in \{0,1\}^{l(n)}$ (với n là giá trị của tham số an toàn ẩn trong s).

Nếu H^s được xác định chỉ với đầu vào là $x \in \{0,1\}^{l'(n)}$ và $l'(n) > l(n)$ thì ta nói rằng (Gen, H) là hàm băm cố định độ dài với đầu vào độ dài $l'(n)$. Trong trường hợp này ta cũng gọi H là hàm nén.

Tiếp theo, ta định nghĩa rõ hơn về hàm băm kháng va chạm, xét thí nghiệm cho hàm băm $\Pi = (\mathrm{Gen}, H)$, thuật toán tấn công A và tham số an toàn n.

Thí nghiệm tìm va chạm $\mathbf{HashColl}_{A,\Pi}(n)$ ([2]):

1. Chạy thuật toán Gen(1^n) đưa ra khóa s.
2. Kẻ tấn công A với s cho trước đưa ra x, x'. (Nếu Π là một hàm băm cố định độ dài với đầu vào là $l'(n)$ thì ta yêu cầu $x, x' \in \{0,1\}^{l'(n)}$.)
3. Thí nghiệm đưa ra 1 khi và chỉ khi $x \neq x'$ và $H^s(x) = H^s(x')$. Trong trường hợp này ta nói rằng A tìm được một va chạm.

Định nghĩa 3 ([2], tr155). Một hàm băm Π =(Gen, H) là kháng va chạm nếu như với mọi thuật toán tấn công thời gian đa thức xác suất A luôn tồn tại một hàm negl thỏa mãn:

$$Pr\left[HashColl_{A,\Pi}(n) = 1\right] \leq negl(n). \quad (2)$$

Chú ý: Trong một vài trường hợp ta chỉ cần dựa vào yêu cầu an toàn yếu hơn tính kháng va chạm. Tức là:

- *Kháng tiền ảnh thứ 2*: Một hàm băm H là kháng tiền ảnh thứ 2 nếu như với s và một chuỗi ngẫu nhiên đều (được chọn ngẫu nhiên theo phân bố đều) x thì rất khó để một thuật toán tấn công thời gian đa thức xác suất tìm được $x' \neq x$ sao cho $H^s(x') = H^s(x)$.
- *Kháng tiền ảnh*: Một hàm băm là kháng tiền ảnh nếu như với s và y ngẫu nhiên đều thì rất khó để một thuật toán tấn công thời gian đa thức xác suất tìm được x sao cho $H^s(x) = y$.

Một hàm băm kháng va chạm thì cũng kháng tiền ảnh thứ 2 ([2]). Thật vậy, nếu với x ngẫu nhiên đều cho trước, một thuật toán tấn công có thể tìm được $x' \neq x$ sao cho $H^s(x') = H^s(x)$, thì rõ ràng rằng có thể tìm được một va chạm x và x'. Tương tự, một hàm kháng tiền ảnh thứ 2 thì cũng kháng tiền ảnh ([2]). Thật vậy, nếu với y ta có thể tìm được x sao cho $H^s(x) = y$, thì với x ngẫu nhiên đều ta tính $y := H^s(x)$ sau đó đưa ra x' với $H^s(x') = y$. Với xác suất cao $x \neq x'$ (dựa vào hàm H là một hàm nén, nên có nhiều giá trị đầu vào nhưng cùng đưa ra một kết quả), điều này có nghĩa là một tiền ảnh thứ 2 được tìm thấy.

Trong một số trường hợp, hàm băm thường đi kèm với một khóa bí mật k mà kẻ tấn công không được biết, ta ký hiệu là H_k^s. Khi đó kẻ tấn công chỉ có quyền truy vấn lên bộ tiên tri băm để đưa ra một va chạm. Điều này được thể hiện qua thí nghiệm sau:

Thí nghiệm tìm va chạm yếu $\mathbf{WHashColl}_{A,\Pi}(n)$ ([4]):

1. Chạy thuật toán Gen(1^n) đưa ra khóa s, và khóa bí mật k.
2. Kẻ tấn công A truy vấn lên bộ tiên tri băm cho giá trị x và nhận lại $H_k^s(x)$. A đưa ra x và x'.
3. Thí nghiệm này đưa ra 1 khi và chỉ khi $x \neq x'$ và $H_k^s(x) = H_k^s(x')$.

Định nghĩa 4 ([4], tr6). Một hàm băm $\Pi = (Gen, H)$ sử dụng khóa bí mật k được gọi là kháng va chạm yếu nếu như với mọi thuật toán tấn công thời gian đa thức xác suất A luôn tồn tại một hàm không đáng kể negl sao cho:

$$Pr\left[WHashColl_{A,\Pi}(n) = 1\right] \leq negl(n). \quad (3)$$

Từ định nghĩa trên ta dễ dàng thấy rằng một hàm băm là kháng va chạm thì kháng va chạm yếu.

C. Mã xác thực thông điệp

Mã xác thực thông điệp là một công cụ giúp các bên giao tiếp có thể nhận biết được rằng thông điệp có bị giả mạo hay không. Mục đích của mã xác thực thông điệp là ngăn chặn một kẻ tấn công có thể giả mạo một thông điệp mới bằng cách sửa đổi những thông điệp hợp lệ. Cũng giống như trong các trường hợp mã hóa thông thường, điều này chỉ có thể thực hiện được nếu các bên giao tiếp chia sẻ khóa bí mật chung mà kẻ tấn công không thể biết được. Khi một bên muốn gửi một thông điệp tới bên còn lại một thông điệp m thì anh ta gắn nhãn MAC t (hay đơn giản là một nhãn t) được tính theo thông điệp và khóa bí mật được chia sẻ trước đó. Nhãn t được tính bằng thuật toán sinh nhãn MAC, được ký hiệu là $t \leftarrow \mathrm{Mac}_k(m)$, sau đó gửi (m, t). Sau khi nhận được (m, t) thì bên còn lại xác thực xem có phải t là giá trị nhãn tương ứng với m hay không. Việc xác thực này được chạy với một thuật toán thời gian đa thức tất định.

Định nghĩa 5 ([2], tr111). Một mã xác thực thông điệp (MAC) gồm có 3 thuật toán thời gian đa thức (Gen, Mac, Vrfy) thỏa mãn:

1. Thuật toán thời gian đa thức xác suất sinh khóa Gen với đầu vào là tham số an toàn 1^n và đưa ra khóa k với $|k| \geq n$.

2. Thuật toán thời gian đa thức xác suất sinh nhãn Mac lấy đầu vào là k và thông điệp $m \in \{0,1\}^*$ và đưa ra nhãn t. Ký hiệu $t \leftarrow Mac_k(m)$.

3. Thuật toán thời gian đa thức tất định xác thực Vrfy lấy đầu vào là khóa k, thông điệp m và nhãn t. Thuật toán đưa ra một bit b, với $b = 1$ là

hợp lệ còn $b = 0$ *thì ngược lại. Ta viết lại* $b \coloneqq \text{Vrfy}_k(m, t)$.

Với mọi n, mọi khóa k sinh bởi Gen(1^n) và mọi $m \in \{0,1\}^*$ thì ta luôn có $\text{Vrfy}_k(m, \text{Mac}_k(m)) = 1$.

Nếu có một hàm l sao cho mọi khóa k sinh bởi Gen(1^n) thuật toán Mac_k chỉ xác định cho thông điệp $m \in \{0,1\}^{l(n)}$, thì ta gọi lược đồ MAC cố định độ dài cho thông điệp có độ dài $l(n)$.

Trong mật mã khóa bí mật, Gen(1^n) thông thường chọn ngẫu nhiên đều một khóa $k \in \{0,1\}^n$.

Phần tiếp theo sẽ nhắc lại định nghĩa an toàn cho mã xác thực thông điệp. Ý tưởng của định nghĩa này là không có một thuật toán tấn công hiệu quả nào có thể đưa ra một giá trị nhãn cho bất kỳ thông điệp mới m bất kỳ, mà chưa từng được sử dụng để trao đổi trước đây.

Thí nghiệm xác thực thông điệp $\text{MacForge}_{A,\Pi}(n)$ ([2]):

1. Chạy thuật toán Gen(1^n) sinh ra khóa k.
2. Thuật toán tấn công A với đầu vào là tham số an toàn 1^n và truy cập vào bộ tiên tri $\text{Mac}_k(\cdot)$. Gọi $Q = \{m_i\}$ là tập tất cả các truy vấn mà A yêu cầu lên bộ tiên tri.
3. Thuật toán tấn công đưa ra (m, t). A thành công khi và chỉ khi (1) $\text{Vrfy}_k(m, t) = 1$ và (2) $m \notin Q$. Trong trường hợp này thí nghiệm đưa ra 1.

Định nghĩa 6 (xem [2], tr113). Một mã xác thực thông điệp Π =(Gen, Mac, Vrfy) được gọi là MAC an toàn nếu như với mọi thuật toán tấn công thời gian đa thức xác suất A thì luôn tồn tại một hàm không đáng kể negl thỏa mãn:

$$Pr[MacForge_{A,\Pi}(n) = 1] \leq negl(n).$$

D. Phép biến đổi Merkle-Damgard

Ta giả sử hàm nén (Gen, h) nén độ dài đầu vào xuống còn một nửa, nghĩa là đầu vào độ dài $2n$ thì nó đưa ra thông điệp độ dài n. Ta xây dựng một hàm băm kháng va chạm (Gen, H) có thể đưa một thông điệp độ dài bất kỳ về độ dài n. Phép biến đổi Merkle-Damgard được định nghĩa trong Cấu trúc 1 dưới đây.

Cấu trúc 1 ([2]):

Gọi (Gen, h) là hàm băm độ dài cố định với đầu vào có độ dài $2n$ và đưa ra độ dài n. Ta xây dựng hàm băm (Gen, H) như sau:

- Gen: thuật toán xác suất có đầu vào là tham số an toàn 1^n và đưa ra khóa s.
- H: với đầu vào là khóa s và chuỗi $x \in \{0,1\}^*$ có độ dài $L < 2^n$, ta thực hiện như sau:
  1. Đặt $B \coloneqq \left\lceil \frac{L}{n} \right\rceil$ (số khối của x). Thêm các phần tử 0 vào x sao cho độ dài của nó là bội của n. Phân tích chuỗi vừa thu được thành các khối n-bit $x_1, \ldots, x_B$. Đặt $x_{B+1} \coloneqq L$, với L được biểu diễn như một chuỗi n-bit.
  2. Đặt $z_0 \coloneqq 0^n$ (cũng được gọi là *IV*).
  3. Với $i = 1, \ldots, B+1$ tính $z_i \coloneqq h^s(z_{i-1} \| x_i)$.
  4. Đưa ra z_{B+1}.

Giá trị z_0 sử dụng trong Bước 2 của cấu trúc được gọi là vectơ khởi tạo hay IV, là bất kỳ và có thể được thay thế bằng hằng số bất kỳ.

Định lý 1 ([2]). Nếu (Gen, h) là kháng va chạm thì (Gen, H) cũng vậy.

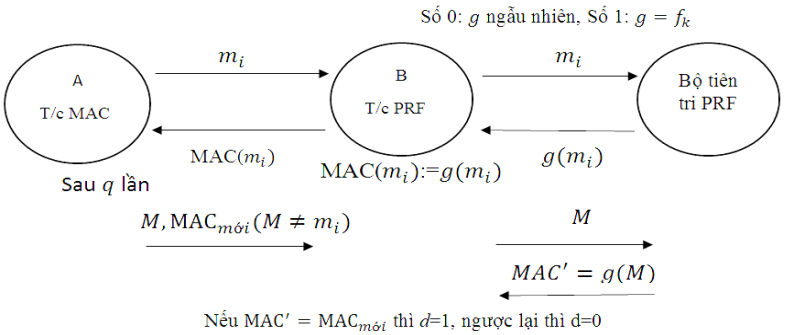
Từ chứng minh của Định lý 1 ta nhận thấy rằng nếu (Gen, h) là kháng va chạm thì (Gen, H) cũng kháng va chạm yếu.

III. ĐỘ AN TOÀN CỦA CẤU TRÚC XÁC THỰC THÔNG ĐIỆP DỰA TRÊN HÀM BĂM

A. Hàm giả ngẫu nhiên và mã xác thực an toàn

Hàm giả ngẫu nhiên là một trong những công cụ hữu ích để xây dựng mã xác thực thông điệp. Một cách trực quan, nếu nhãn t được sinh bằng cách sử dụng hàm giả ngẫu nhiên cho thông điệp m, thì việc giả mạo đòi hỏi kẻ tấn công phải đoán giá trị của hàm giả ngẫu nhiên với một thông điệp đầu vào mới. Xác suất để hai giá trị ngẫu nhiên bằng nhau là 2^{-n} (với tham số an toàn là n).

Cấu trúc 2 sau đây sử dụng ý tưởng trên để xây dựng một mã xác thực thông điệp với độ dài cố định. (Để cho đơn giản trong việc trình bày cũng như chứng minh, ta giả sử hàm giả ngẫu nhiên được sử dụng ở đây có khóa, đầu vào và đầu ra có cùng một độ dài là n).



Hình 1. Mô tả thuật toán tấn công B

Cấu trúc 2 ([2]).

Cho f là một hàm giả ngẫu nhiên. Định nghĩa MAC độ dài cố định cho thông điệp n-bit như sau:

- Mac: Với đầu vào là khóa $k \in \{0,1\}^n$ và thông điệp $m \in \{0,1\}^n$, đưa ra nhãn $t := f_k(m)$.
- Vrfy: Với đầu vào là khóa $k \in \{0,1\}^n$, thông điệp $m \in \{0,1\}^n$ và nhãn $t \in \{0,1\}^n$ đưa ra 1 khi và chỉ khi $t = f_k(m)$.

Định lý sau đây đã được chứng minh trong [2] nhưng chúng tôi đưa ra một chứng minh khác ngắn gọn hơn.

Định lý 2 ([2]). Giả sử f là một hàm giả ngẫu nhiên, khi đó nếu lược đồ MAC sử dụng f là hàm sinh nhãn như trong Cấu trúc 2 thì lược đồ đã cho là MAC an toàn và có độ dài cố định.

Chứng minh. Gọi Π là lược đồ MAC sử dùng hàm f giả ngẫu nhiên làm thuật toán sinh nhãn, A là một thuật toán tấn công thời gian đa thức xác suất lên Π. Ta sẽ xây dựng B là thuật toán tấn công sử dụng A như là một chương trình con.

Thuật toán tấn công *B*

1. Người thách thức chọn một số 0 hoặc 1, và giữ nguyên trong quá trình. Chạy $A(1^n)$, khi A yêu cầu nhãn cho thông điệp m_i, B gửi m_i lên người thách thức. Người này sẽ trả lại giá trị $g(m_i)$ và B trả lại nhãn $t_i = \text{MAC}(m_i) = g(m_i)$ cho A, trong đó $g(\cdot)$ phụ thuộc vào số mà thách thức chọn, : số 0 thì $g \in \text{Func}_n$, số 1 thì $g = f_k$.
2. A đưa ra $(M, \text{MAC}_{mới})$ (giả sử $M \neq m_i$). B gửi giá trị M cho người thách thức và nhận lại giá trị $\text{MAC}' = g(M)$. B so sánh $\text{MAC}' \stackrel{?}{=} \text{MAC}_{mới}$, nếu bằng thì đưa ra $d = 1$ còn ngược lại đưa ra $d = 0$.

Nếu số 1 được chọn thì cách nhìn của A khi chạy trong thuật toán trên giống như cách nhìn của A chạy độc lập trong thí nghiệm $\text{MacForge}_{A,\Pi}$. Khi A đưa ra một giả mạo hợp lệ $(\text{M}, \text{MAC}_{mới})$ và B gửi M cho thách thức sau đó nhận lại giá trị $g(M)$. Do $(\text{M}, \text{MAC}_{mới})$ là giả mạo hợp lệ nên $g(M) = \text{MAC}_{mới}$, mặt khác nếu như $(\text{M}, \text{MAC}_{mới})$ không hợp lệ thì $g(M) \neq \text{MAC}_{mới}$, điều này có nghĩa là xác suất để B đưa ra số 1 trong trường hợp số 1 được chọn chính là bằng xác suất để giả mạo thành công.

Nếu số 0 được chọn thì khi đó A được trả lại giá trị MAC không chính xác cho nên khi A đưa ra giả mạo cũng chỉ giống như đưa ra một giá trị ngẫu nhiên. Vì vậy để B đưa ra 1 trong trường hợp này có nghĩa là hai giá trị ngẫu nhiên trùng nhau. Xác suất này xảy ra đúng bằng $\frac{1}{2^n}$.

Ta có:

$$\begin{aligned}\text{Adv}_f^{\text{prf}}(B) &= \Pr\left[B^{f_k(\cdot)}(1^n) = 1\right] \\ &\quad - \Pr\left[B^{g(\cdot)}(1^n) = 1\right] \\ &= \Pr\left[\text{MacForge}_{A,\Pi}(n) = 1\right] - \frac{1}{2^n}\end{aligned}$$

Suy ra:

$$\Pr\left[\text{MacForge}_{A,\Pi}(n) = 1\right] = \text{Adv}_f^{\text{prf}}(B) + \frac{1}{2^n}.$$

Do f là hàm giả ngẫu nhiên nên ta có điều phải chứng minh □

B. Cấu trúc băm- rồi-MAC

Cấu trúc băm-rồi-MAC là một ý tưởng dùng để xây dựng cho một mã xác thực thông điệp với độ dài thông điệp bất kỳ, ý tưởng này dựa trên tính kháng va chạm yếu của hàm băm.

Đầu tiên, với thông điệp có độ dài bất kỳ m được băm xuống độ dài cố định $H^s(m)$ bằng cách sử dụng một hàm băm kháng va chạm yếu. Sau đó sử dụng MAC cho thông điệp đã được băm.

Cấu trúc 3 ([2]):

Với Π=(Mac, Vrfy) là một MAC cho thông điệp có độ dài $l(n)$, và Π_H=(Gen_H, H) là một hàm băm với đầu ra có độ dài $l(n)$. Ta xây dựng một MAC Π'=($\text{Gen}', \text{Mac}', \text{Vrfy}'$) với thông điệp có độ dài bất kỳ như sau:

- Gen': Với đầu vào là tham số an toàn 1^n, chọn ngẫu nhiên đều $k \in \{0,1\}^n$ và chạy thuật toán $\text{Gen}_H(1^n)$ thu được s, khóa là $k' := \langle k, s\rangle$.
- Mac': Với đầu vào là khóa $\langle k, s\rangle$ và một thông điệp $m \in \{0,1\}^*$, sau đó đưa ra $t \leftarrow \text{Mac}_k\big(H^s(m)\big)$.
- Vrfy': Với đầu vào là khóa $\langle k, s\rangle$, thông điệp $m \in \{0,1\}^*$ và nhãn t, đưa ra 1 khi và chỉ khi $\text{Vrfy}_k(H^s(m), t) = 1$.

Cấu trúc 3 là an toàn nếu Π là MAC an toàn cho thông điệp có độ dài cố định và (Gen, H) là kháng va chạm yếu. Vì hàm băm trên là kháng va chạm yếu nên việc xác thực $H^s(m)$ cũng như xác thực m: nếu người gửi có thể đảm bảo rằng người nhận thu được giá trị đúng $H^s(m)$, tính kháng va chạm yếu đảm bảo rằng kẻ tấn công không thể tìm được thông điệp m' khác mà khi băm có cùng giá trị. Nếu một người sử dụng Cấu trúc 3 để xác thực một tập thông điệp Q thì thuật toán tấn công A có thể giả mạo nhãn cho một thông điệp $m^* \notin Q$ khi các trường hợp sau xảy ra:

- Tồn tại thông điệp $m \in Q$ sao cho $H^s(m) = H^s(m^*)$. Khi đó A tìm được một va chạm của H^s, mâu thuẫn với tính kháng va chạm yếu của (Gen_H, H).
- Với mọi thông điệp $m \in Q$ ta đều có $H^s(m^*) \neq H^s(m)$, đặt $H^s(Q) = \{H^s(m) | m \in Q\}$ khi đó $H^s(m^*) \notin H^s(Q)$. Trong trường hợp này A đưa ra nhãn giả cho thông điệp mới $H^s(m^*)$- có độ dài là $l(n)$. Điều này mâu thuẫn với giả thiết Π là MAC an toàn.

Định lý 3. ([1]) Nếu Π là MAC an toàn với thông điệp độ dài l và Π_H là kháng va chạm yếu thì cấu trúc 3 là MAC an toàn (cho thông điệp có độ dài bất kỳ).

Chứng minh. (sử dụng ý tưởng trong chứng minh Định lý 5.6 trong [2])

Gọi Π' là cấu trúc 3, A' là một thuật toán tấn công thời gian đa thức xác suất lên Π'.

Trong thí nghiệm $\text{MacForge}_{A',\Pi'}(n)$, gọi $k' = \langle k, s\rangle$ là khóa MAC, Q là tập các thông điệp mà A' truy vấn, gọi (m^*, t) là kết quả đưa ra cuối cùng của A'. Không mất tính tổng quát, ta giả sử $m^* \notin Q$. Định nghĩa coll là sự kiện mà trong thí nghiệm $\text{MacForge}_{A',\Pi'}(n)$, tồn tại $m \in Q$ sao cho $H^s(m^*) = H^s(m)$. Ta có:

$$\begin{aligned}&\Pr\big[\text{MacForge}_{A',\Pi'}(n) = 1\big]\\ &= \Pr\big[\text{MacForge}_{A',\Pi'}(n) = 1 \wedge \text{coll}\big] + \\ &\Pr[\text{MacForge}_{A',\Pi'}(n) = 1 \wedge \overline{\text{coll}}]\\ &= \Pr[\text{coll}] + \Pr\big[\text{MacForge}_{A',\Pi'}(n) = 1 \wedge \overline{\text{coll}}\big].\end{aligned} \quad (4)$$

Thuật toán tấn công tìm va chạm C (coll):

Thuật toán với đầu vào là s (n ẩn)

- Chọn ngẫu nhiên đều $k \in \{0,1\}^n$.
- Chạy $A'(1^n)$. Khi A' yêu cầu nhãn cho thông điệp thứ i là $m_i \in \{0,1\}^*$, đầu tiên yêu cầu lên bộ tiên tri băm giá trị của $H^s(m_i)$ sau đó tính $t_i \leftarrow \text{Mac}_k(H^s(m_i))$ và trả lại t_i cho A'.
- Khi A' đưa ra (m^*, t), nếu tồn tại i sao cho $H^s(m^*) = H^s(m_i)$ thì đưa ra (m^*, m_i).

Chúng ta sẽ chỉ ra rằng cả hai xác suất của (4) đều là không đáng kể. Xác suất đầu tiên không đáng kể là do tính kháng va chạm yếu của hàm Π_H còn xác suất thứ hai là do sự an toàn của Π.

Xác suất thứ nhất tương ứng với điều kiện là coll xảy ra, khi đó ta xét thuật toán tấn công C tìm một va chạm yếu của hàm Π_H mà sử dụng A' như là một chương trình con.

Dễ thấy rằng, thuật toán C chạy với thời gian đa thức. Khi đầu vào của C được sinh bởi thuật toán $\text{Gen}_H(1^n)$ là s, cách nhìn A' như là một chương trình con của C cũng giống như cách nhìn A' trong thí nghiệm $\text{MacForge}_{A',\Pi'}(n)$. Đặc biệt, nhãn đưa cho A' bởi C có cùng phân phối như nhãn mà A' nhận từ $\text{MacForge}_{A',\Pi'}(n)$. Bởi vì C đưa ra một va chạm chính khác khi coll xảy ra nên ta có:

$$\Pr\big[\text{WHashColl}_{C,\Pi_H}(n) = 1\big] = \Pr[\text{coll}].$$

Vì Π_H là kháng va chạm yếu nên ta có $\Pr[\text{coll}]$ là không đáng kể.

Tiếp theo ta chứng minh xác suất thứ hai của (4) là không đáng kể tương ứng với sự kiện coll không xảy ra. Thật vậy, nếu thuật toán tấn công A lên Π trong thí nghiệm $\text{MacForge}_{A,\Pi}(n)$ và sử dụng A' như là một chương trình con.

Thuật toán tấn công A:

Kẻ tấn công với quyền truy cập vào bộ tiên tri MAC là $\text{Mac}_k(\cdot)$:

- Tính $\text{Gen}(1^n)$ để thu s.
- Chạy $A'(1^n)$. Khi A' yêu cầu nhãn cho thông điệp thứ i $m_i \in \{0,1\}^*$, thì (1) tính $\widehat{m}_i \coloneqq H^s(m_i)$; (2) lấy nhãn t_i của $\widehat{m}_i$ từ bộ tiên tri MAC, và (3) trả lại t_i cho A'.
- Khi A' đưa ra (m^*, t) thì đưa ra $(H^s(m^*), t)$.

Dễ thấy rằng A chạy với thời gian đa thức. Trong thí nghiệm $\text{MacForge}_{A,\Pi}(n)$, khi A chạy $\text{Gen}(1^n)$ để thu được s, cách nhìn A' như là chương trình con của A giống như cách nhìn A' chạy độc lập trong thí nghiệm $\text{MacForge}_{A',\Pi'}(n)$. Hơn nữa, khi $\text{MacForge}_{A',\Pi'}(n) = 1$ và sự kiện coll không xảy ra, cho nên A giả mạo được trong thí nghiệm $\text{MacForge}_{A,\Pi}(n)$. (Trong trường hợp này t là giá trị nhãn của $H^s(m^*)$ trong lược đồ Π tương ứng với k. Sự kiện va chạm không xảy ra có nghĩa rằng $H^s(m^*)$ chưa từng được A truy vấn lên bộ tiên tri MAC). Vì vậy ta có:

$$\Pr\left[\text{MacForge}_{A,\Pi}(n) = 1\right] = \Pr\left[\text{MacForge}_{A',\Pi'}(n) = 1 \wedge \overline{\text{coll}}\right], \quad (5)$$

và do Π là an toàn nên xác suất $\Pr\left[\text{MacForge}_{A',\Pi'}(n) = 1 \wedge \overline{\text{coll}}\right]$ là không đáng kể. Từ đây ta suy ra điều phải chứng minh □

Hệ quả 1 (Định lý 5.6 tr159 [2]). Nếu Π là một MAC an toàn với thông điệp độ dài l và Π_H là kháng va chạm thì khi đó cấu trúc băm rồi MAC là MAC an toàn với độ dài thông điệp bất kỳ.

Chứng minh. Hệ quả này được suy ra trực tiếp từ việc Π_H là kháng va chạm suy ra kháng va chạm yếu và Định lý 3 □

C. Cấu trúc NMAC

Tất cả các mã xác thực thông điệp được công bố trước đây đều dựa trên một số mã khối. Một câu hỏi được đặt ra là liệu có thể xây dựng một MAC an toàn dựa trên hàm băm. Suy nghĩ đầu tiên là nếu như ta thêm một khóa bí mật vào hàm băm liệu có làm cho $\text{Mac}_k(m) \coloneqq H(k\|m)$ là an toàn. Chúng ta mong muốn rằng nếu H là một hàm băm đủ tốt thì sẽ khó khăn cho một kẻ tấn công có thể đoán được giá trị của $H(k\|m')$ nếu biết trước giá trị của $H(k\|m)$ với mọi $m' \neq m$, giả sử rằng k được chọn ngẫu nhiên đều và kẻ tấn công không biết được. Nhưng nếu như hàm H được xây dựng dựa trên phép biến đổi Merkle-Damgard thì MAC được thiết kế như vậy là không an toàn. Thật vậy, với $m' = m\|\|m|$, với $|m|$ là độ dài của m thì kẻ tấn công có thể đưa ra một giả mạo hợp lệ nếu như truy vấn vào bộ tiên tri băm $\text{Mac}_k(m)$. Thay vào đó chúng ta sử dụng hai lớp hàm băm. Cấu trúc 4 dưới đây gọi là NMAC thể hiện cho ý tưởng trên.

Cấu trúc 4 (NMAC) ([1]):

Gọi (Gen_H, H) là hàm băm được xây dựng bằng cách sử dụng phép biến đổi Merkle-Damgard từ hàm nén (Gen_H, h) với đầu vào có độ dài là $n + n'$. Định nghĩa MAC như sau:

- Gen: chạy thuật toán $\text{Gen}_H(1^n)$ thu được s. Chọn ngẫu nhiên đều $k_1, k_2 \in \{0,1\}^n$. Đưa ra khóa $\langle s, k_1, k_2 \rangle$.
- Mac: với đầu vào là khóa $\langle s, k_1, k_2 \rangle$ và thông điệp $m \in \{0,1\}^*$ đưa ra
$$t \coloneqq H^s_{k_1}\left(H^s_{k_2}(m)\right).$$
- Vrfy: với đầu vào là khóa $\langle s, k_1, k_2 \rangle$, thông điệp $m \in \{0,1\}^*$ và nhãn t đưa ra 1 khi và chỉ khi $t = H^s_{k_1}\left(H^s_{k_2}(m)\right)$.

Trong đó $H^s_{k_1}$ và $H^s_{k_2}$ hiểu là được xây dựng từ phép biến đổi Merkle-Damgard nhưng thay vì dùng vectơ khởi tạo là 0^n ta dùng là k_1, k_2 tương ứng.

Chú ý rằng do đầu ra của $H^s_{k_2}(m)$ chỉ có độ dài là n nên ta có thể viết lại $H^s_{k_1}\left(H^s_{k_2}(m)\right) = h^s\left(k_1\|H^s_{k_2}(m)\right) = h^s_{k_1}\left(H^s_{k_2}(m)\right)$.

Định lý 4 ([1]). Nếu $h^s_{k_1}$ là một MAC an toàn và $H^s_{k_2}$ là một hàm kháng va chạm yếu thì khi đó NMAC là một MAC an toàn.

Chứng minh. Xem Định lý 3 □

Chúng tôi thấy rằng việc chọn hai khóa như trên có thể được thay thế bằng cách chọn một khóa ngẫu nhiêu đều duy nhất và sử dụng hàm giả ngẫu nhiên. Chúng tôi đưa ra mệnh đề sau:

Mệnh đề 1. Giả sử tồn tại hàm f giả ngẫu nhiên sao cho $f^k(\cdot)$ sinh ra hai khóa k_1 và k_2. Khi đó NMAC vẫn là một MAC an toàn.

Chứng minh. Ta sẽ chứng minh bằng phương pháp phản chứng, giả sử với k_1, k_2 được sinh bởi hàm giả ngẫu nhiên f với tập khóa là đầu vào thứ hai và khóa k được chọn ngẫu nhiên đều mà cấu trúc NMAC không phải là một MAC an toàn, hay có nghĩa là có một thuật toán tấn công B nào đấy đưa ra được giả mạo với một xác suất đáng kể.

Ta định nghĩa thuật toán tấn công A lên hàm giả ngẫu nhiên f như sau:

- Yêu cầu lên bộ tiên tri O thu về hai giá trị k_1 và k_2.
- Dùng k_1, k_2 làm khóa cho cấu trúc NMAC hay là $H^s_{k_1}\left(H^s_{k_2}(\cdot)\right)$.
- Nếu cấu trúc NMAC trên là an toàn thì đưa ra 0, còn ngược lại thì đưa ra 1.

Khi đó trong trường hợp với bộ tiên tri O là hàm $f^k(\cdot)$ ta có:

$$\Pr\left[\mathrm{Exp}_f^{\mathrm{prf}-1}(A) = 1\right] \geq \epsilon(n),$$

trong đó, $\epsilon(n)$ là một hàm đáng kể.

Mặt khác với k_1, k_2 được chọn ngẫu nhiên đều ở Cấu trúc 4 thì NMAC là một MAC an toàn trừ một xác suất không đáng kể hay là

$$\Pr\left[\mathrm{Exp}_f^{\mathrm{prf}-0}(A) = 1\right] \leq \mathrm{negl}(n).$$

Từ đây suy ra:

$$\Pr\left[\mathrm{Exp}_f^{\mathrm{prf}-1}(A) = 1\right] - \Pr\left[\mathrm{Exp}_f^{\mathrm{prf}-0}(A) = 1\right] \geq \epsilon(n) - \mathrm{negl}(n).$$

Do $\epsilon(n) - \mathrm{negl}(n)$ là một hàm đáng kể nên mâu thuẫn với giả thiết f là một hàm giả ngẫu nhiên □

Chú ý: Bằng những lập luận tương tự, chúng tôi chứng minh được một lược đồ thỏa mãn tính chất mật mã a nào đó với khóa được lấy ngẫu nhiên thì tính a của lược đồ vẫn được đảm bảo khi khóa được sinh bởi một hàm giả ngẫu nhiên.

D. Cấu trúc HMAC

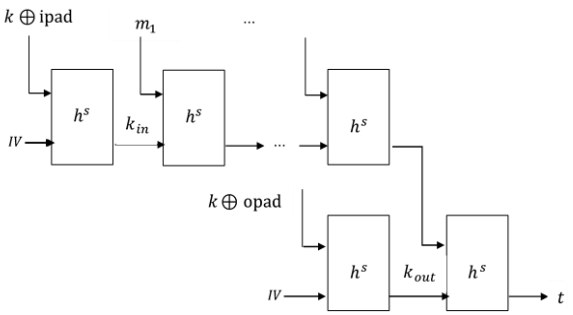
Sau đây ta sẽ xét một biến thể của NMAC là HMAC. Điểm khác biệt của cấu trúc này so với NMAC là nó chỉ sử dụng một khóa duy nhất k được mô tả như sau:

Cấu trúc 5 (HMAC) ([1,2]):

Gọi (Gen_H, H) là một hàm băm xây dựng bằng cách sử dụng phép biến đổi Merkle-Damgard từ hàm nén (Gen_H, h) với đầu vào có độ dài là $n + n'$. Gọi opad và ipad là các hằng số có độ dài n'. Định nghĩa MAC như sau:

- Gen: chạy thuật toán $\mathrm{Gen}_H(1^n)$ thu được s. Chọn ngẫu nhiên đều $k \in \{0,1\}^{n'}$. Đưa ra khóa $\langle s, k\rangle$.
- Mac: với đầu vào là khóa $\langle s, k\rangle$ và thông điệp $m \in \{0,1\}^*$ đưa ra

  $t := H^s\left((k \oplus \mathrm{opad}) \| H^s\left((k \oplus \mathrm{ipad}) \| m\right)\right)$.
- Vrfy: với đầu vào là khóa $\langle s, k\rangle$, thông điệp $m \in \{0,1\}^*$ và nhãn t đưa ra 1 khi và chỉ khi $t = H^s\left((k \oplus \mathrm{opad}) \| H^s\left((k \oplus \mathrm{ipad}) \| m\right)\right)$.



Hình 2. Mô hình HMAC

Về đặc điểm, nhìn vào Hình 2 ta thấy rằng cấu trúc HMAC bao gồm băm một thông điệp độ dài bất kỳ thành một chuỗi ngắn $y \overset{def}{=} H^s((k \oplus \mathrm{ipad}) \| m)$ sau đó tính $H^s((k \oplus \mathrm{opad}) \| y)$.

Chú ý rằng bên trong tính toán $\widetilde{H}^s(m) \overset{def}{=} H^s((k \oplus \mathrm{ipad}) \| m)$ là kháng va chạm (từ giả thiết hàm h) với mọi giá trị của $k \oplus$ ipad. Hơn nữa, bước đầu tiên trong tính toán $H^s((k \oplus \mathrm{opad}) \| y))$ là tính giá trị $k_{out} \overset{def}{=} h^s(IV \| (k \oplus \mathrm{opad}))$. Sau đó, ta tính $h^s(k_{out} \| \hat{y})$ với $\hat{y}$ là giá trị sau khi đệm thêm vào y.

Vì vậy, nếu ta xem k_{out} như là ngẫu nhiên đều và giả sử rằng $\widetilde{\mathrm{Mac}}_k(y) \overset{def}{=} h^s(k \| \hat{y}) = h^s_k(\hat{y})$ là một MAC với độ dài cố định, thì HMAC có thể xem như cấu trúc Băm-rồi-MAC: $\mathrm{HMAC}_{s,k}(m) = \widetilde{\mathrm{Mac}}_{k_{out}}(\widetilde{H}^s(m))$ với $k_{out} = h^s(IV \| (k \oplus \mathrm{opad}))$.

Trong HMAC, khóa đơn k được sử dụng trong việc kết hợp với ipad và opad để tạo ra hai khóa k_{in}, k_{out} như sau $G^s(k) = h^s(IV \| (k \oplus \mathrm{opad})) \| h^s(IV \| (k \oplus \mathrm{ipad})) = k_{out} \| k_{in}$.

Độ an toàn của HMAC được quy về độ an toàn của cấu trúc sau đây:

$$\mathrm{Mac}_{s,k_{in},k_{out}}(m) = h^s\left(k_{out} \| H^s_{k_{in}}(m)\right) = h^s_{k_{out}}\left(H^s_{k_{in}}(M)\right).$$

Cấu trúc này có thể được chứng minh an toàn nếu như $H^s_{k_{in}}$ là kháng va chạm yếu và thuật toán $\widetilde{\mathrm{Mac}}_k$ là MAC an toàn độ dài cố định.

Chúng tôi nhận thấy rằng trong cấu trúc HMAC ở trên ta cần phải chọn ipad ≠ opad để sinh ra được hai khóa k_{in} và k_{out} gần như độc lập với nhau. Trong [1], hai giá trị này được chọn cố định lần lượt là 0x36…36, và 0x5c…5c. Nếu như ipad = opad thì $k' := k_{in} = k_{out}$, thì lược đồ HMAC trở thành $h^s_{k'}\left(H^s_{k'}(M)\right)$. Khi đó chúng tôi đưa ra một điểm yếu của lược đồ như sau: nếu kẻ

tấn công được quyền truy vấn bộ tiên tri $h^s_{k'}(\cdot)$ hai lần thì anh ta có thể đưa ra được giá trị xác thực cho một thông điệp $m = m_1\|m_2\| \dots \|m_B$ bất kỳ theo cách hợp lệ.

Thật vậy, đầu tiên kẻ tấn công truy vấn lên bộ tiên tri cho thông điệp m_1, nhận lại được $z_1 := h^s_{k'}(m_1)$ thì kẻ tấn công sẽ tính $z_2 := h^s(z_1\|m_2), \dots, z_{B+1} := h^s(z_B\||m|)$ và thu được $m^* = H^s_{k'}(m) = z_{B+1}$. Sau đó kẻ tấn công này lại truy vấn lên bộ tiên tri cho thông điệp m^* và được trả về giá trị m_{MAC}. Khi đó, m_{MAC} chính là nhãn hợp lệ cho thông điệp m. (Vì kẻ tấn công không truy vấn đến hàm băm $H^s_{k'}(\cdot)$ nên cách giả mạo này được coi là hợp lệ). Còn trong trường hợp ipad ≠ opad (khác nhau như đã nói ở trên) thì k_{in} và k_{out} gần như độc lập, vì vậy cho dù kẻ tấn công có một năng lực mạnh hơn là được quyền truy cập vào bộ tiên tri $h^s_{k_{out}}$ thì việc phá vỡ được lược đồ HMAC vẫn không phải là dễ, nếu anh ta không được truy vấn tới hàm băm. Từ đây ta cũng thấy rằng trong lược đồ NMAC thì điều kiện hai khóa k_1 và k_2 ngẫu nhiên và độc lập là cần thiết.

Định lý 5 (Định lý 5.8 [2]). Giả sử G^s được định nghĩa trong phương trình ở trên là bộ sinh giả ngẫu nhiên với mọi s, mã xác thực thông điệp $\widetilde{Mac}_k$ là MAC an toàn và (Gen_H, H) là kháng va chạm yếu. Khi đó HMAC là MAC an toàn (với độ dài bất kỳ).

Trong [1] cũng đưa ra giả thiết an toàn gần tương tự như Định lý 5, nhưng trong cả hai đều chỉ nhận xét rằng cấu trúc HMAC là một trường hợp đặc biệt của NMAC nếu như khóa k_{out}, k_{in} được coi như là ngẫu nhiên và độc lập với nhau dưới điều kiện thích hợp của G^s mà không đưa ra bất kỳ chứng minh an toàn nào. Nhận thấy rằng các điều kiện để đảm bảo tính an toàn cho cấu trúc trên chỉ xuất phát từ hàm nén ban đầu, chúng tôi đưa ra giả thiết mới cho cấu trúc HMAC như sau:

Hệ quả 2. Giả sử hàm nén h^s là giả ngẫu nhiên đều $\{0,1\}^n \times \{0,1\}^{n'} \to \{0,1\}^n$ và h^s kháng va chạm. Khi đó HMAC được xây dựng như ở cấu trúc 5 là MAC an toàn (với mọi thông điệp có độ dài bất kỳ).

Chứng minh. Do giả thiết h^s là giả ngẫu nhiên với tập khóa là đầu vào thứ hai khi đó k_{in}, k_{out} được sinh từ một hàm giả ngẫu nhiên. Kết hợp với giả thiết h^s là ngẫu nhiên với tập khóa là đầu vào thứ nhất thì cấu trúc $\widetilde{\text{Mac}}_k$ là một MAC an toàn.

Hơn nữa do tính chất kháng va chạm của hàm h^s nên (Gen, H) là kháng va chạm yếu. Bài toán đến đây đưa về Định lý 4 và Mệnh đề 1 □

Chú ý: Trong hệ quả trên, vì chỉ muốn sử dụng các tính chất của hàm nén ta đã làm mạnh giả thiết bằng cách thêm tính kháng va chạm của hàm nén. Từ cách lập luận trên ta vẫn chứng minh được cấu trúc HMAC là an toàn nếu như tính ngẫu nhiên đều của hàm nén và tính kháng va chạm yếu của hàm băm lặp được đảm bảo.

IV. KẾT LUẬN

Trong bài báo này chúng tôi phân tích độ an toàn cho cấu trúc HMAC thông qua giả thiết giả ngẫu nhiên đều cho hàm nén. Mặc dù giả thiết này mạnh hơn so với các giả thiết trong [1, 2], song chúng tôi đã đưa ra được chứng minh chi tiết và tường minh cho độ an toàn của cấu trúc HMAC theo giả thiết này. Tuy vậy, các giả thiết trong bài báo của chúng tôi và [1, 2] vẫn bao gồm tính kháng va chạm yếu của hàm nén, trong khi [4] lại không nhắc tới giả thiết này. Nhưng điều này cho phép lược đồ HMAC (hay NMAC) vẫn an toàn trước những phương thức tấn công có năng lực mạnh hơn như truy vấn vào hàm băm hoặc có khả năng tính toán được việc băm thông điệp.

TÀI LIỆU THAM KHẢO

[1]. Bellare, M., R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication". in Annual International Cryptology Conference, Springer, 1996.

[2]. Katz, J. and Y. Lindell, "Introduction to modern cryptography" CRC press, 2014.

[3]. Bellare, M. and P. Rogaway, "Introduction to modern cryptography", Ucsd Cse, 2005.

[4]. Bellare, M, "New proofs for NMAC and HMAC: Security without collision-resistance". in Annual International Cryptology Conference. Springer, 2006.

SƠ LƯỢC VỀ TÁC GIẢ

ThS. Nguyễn Bùi Cương

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

Email: nguyenbuicuong@gmail.com

Quá trình đào tạo: Nhận bằng cử nhân chuyên ngành Toán học tại Đại học Sư phạm Hà Nội, Đại học Quốc gia Hà Nội năm 2004. Nhận bằng Thạc sĩ Toán học tại Đại học Khoa học tự nhiên, Đại Học Quốc gia Hà Nội năm 2008.

Hướng nghiên cứu hiện nay: Khoa học mật mã; Mã hóa đối xứng.

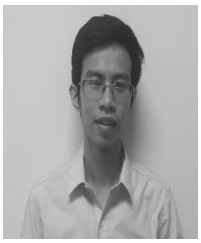
CN. Triệu Quang Phong

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

Email: phongtrieu53@gmail.com

Quá trình đào tạo: Nhận bằng cử nhân chuyên ngành Toán học, Đại học Khoa học tự nhiên, Đại học Quốc gia Hà Nội năm 2014.

Hướng nghiên cứu hiện nay: Khoa học mật mã.



CN. Nguyễn Tuấn Anh

Email: tuananhnghixuan@gmail.com

Quá trình đào tạo: Nhận bằng cử nhân chuyên ngành Toán tài năng tại Đại học Khoa học tự nhiên, Đại học Quốc gia Hà Nội năm 2016.

Hướng nghiên cứu hiện nay: Mã hóa đối xứng.