VỀ MỘT BACKDOOR ĐỐI XỨNG TRONG SINH KHÓA RSA

Bạch Nhật Hồng*, Lê Quang Huy+
** Trường Đại học Sư phạm kỹ thuật Hưng Yên*
+Ban Cơ yếu Chính phủ

Abstract: Bài báo trình bày một đề xuất về thuật toán sinh khóa RSA chứa backdoor dựa trên ý tưởng của thuật toán Hidden Prime Factor trong [1]. Thuật toán đề xuất sử dụng kết quả của Coppersmith trong [7] để giảm lượng thông tin backdoor cần nhúng.

Keywwords: Mật mã, sinh khóa, RSA, backdoor**.**

I. MỞ ĐẦU

Với hạ tầng mật mã khóa công khai (PKI), một vấn đề được quan tâm là việc khôi phục cặp khóa (dùng để mã) của người dùng khi có yêu cầu từ người dùng (mất khóa) hoặc từ cơ quan nhà nước (liên quan đến an ninh). Để giải quyết vấn đề trên người ta sử dụng hai cách. Cách thứ nhất là sử dụng giải pháp cơ sở dữ liệu (CSDL) khóa và các biện pháp đảm bảo an toàn cho CSDL khóa như giải pháp mềm (mã mật CSDL khóa) và giải pháp vật lý (bảo vệ máy tính…). Giải pháp sử dụng CSDL khóa có ưu điểm là các tính chất của khóa không bị giới hạn (so với thuật toán chuẩn) nhưng có nhược điểm là chi phí lớn và vận hành phức tạp. Cách thứ hai là sử dụng thuật toán sinh khóa chứa backdoor (viết gọn là backdoor). Ưu điểm của giải pháp sử dụng backdoor là chi phí thấp, vì khi sử dụng, hệ thống không cần lưu giữ khóa riêng của người dùng (không duy trì CSDL khóa) mà chỉ cần lưu giữ và bảo vệ khóa của người thiết kế. Nhược điểm của giải pháp sử dụng backdoor là các tính chất khóa bị thu hẹp so với thuật toán sinh khóa chuẩn.

Với giải pháp thứ hai, nhiều thuật toán sinh khóa chứa backdoor trên một số hệ mật đã được công bố như: RSA, Elgamal, Eliptic Curve. Các thuật toán sinh khóa RSA chứa backdoor trong [1], [2] được đánh giá là có lực lượng thấp. Các thuật toán sinh khóa RSA chứa backdoor trong [3], [4] có độ phức tạp gần với bậc hai và cần sử dụng bộ nhớ không mất dữ liệu để lưu thông tin.

Do vậy mục tiêu nghiên cứu ngoài việc tăng thêm hiểu biết để có giải pháp phòng vệ tốt hơn trong việc sử dụng các sản phẩm mật mã khi không làm chủ được mà còn có khả năng ứng dụng, bài báo tập trung nghiên cứu các RSA backdoor và đề xuất một RSA backdoor mới dựa trên ý tưởng của thuật toán Hidden Prime Factor trong [1]. Thực hiện mục tiêu trên, bài báo được tổ chức thành 4 phần: mục 1 Đặt vấn đề, nêu lên sự cần thiết nghiên cứu và một số kết quả nghiên cứu của các tác giả đi trước; mục 2, Các định nghĩa và cơ sở phục vụ cho việc phân tích backdoor; mục 3 Đề xuất backdoor mới; mục 4 Kết luận tóm tắt các kết quả nghiên cứu và hướng phát triển.

II. CÁC ĐỊNH NGHĨA VÀ CƠ SỞ

A. Thuật toán sinh khóa chứa backdoor

Định nghĩa về thuật toán sinh khóa chứa backdoor và các thuộc tính có liên quan (tiêu chuẩn đánh giá) trình bày trong phần này sử dụng các định nghĩa trong [5].

1) Một số ký hiệu

Ký hiệu G_0, G_1 lần lượt là thuật toán sinh khóa trung thực (chuẩn) và thuật toán sinh khóa chứa backdoor. Ký hiệu (k_{priv} , k_{pub}) lần lượt là khóa riêng và khóa công khai được tạo bởi G_0 hoặc G_1. Ký hiệu k_{pub*} là khóa công khai hoặc một phần của khóa công khai. Ký hiệu ℓ là tham số an toàn của hệ mật. Thuật toán sinh khóa chứa backdoor được biểu diễn thông qua 3 hàm sau và các hàm nghịch đảo tương ứng:

Hàm thứ nhất: gọi là hàm trích thông tin, ký hiệu $\mathcal{I}$. Hàm $\mathcal{I}$ trích từ khóa riêng k_{priv} các thông tin sẽ được giấu trong khóa công khai k_{pub} sao cho thông tin được trích và nhúng vào khóa công khai đủ để khôi phục một cách hiệu quả k_{priv}.

Hàm thứ hai: gọi là hàm che giấu thông tin, ký hiệu $\mathcal{E}$. Hàm $\mathcal{E}$ thực hiện mã mật hóa thông được trích, $\mathcal{I}(k_{priv})$, tạo kết quả là thông tin che giấu $\mathcal{E}(\mathcal{I}(k_{priv}))$. Hàm $\mathcal{E}$ có thể dựa trên hệ mật đối xứng hoặc hệ mật bất đối xứng sao cho phân phối đầu ra của hàm $\mathcal{E}$ không thể phân biệt được với phân phối đều.

Hàm thứ ba: gọi là hàm nhúng, ký hiệu $\mathcal{M}$. Hàm $\mathcal{M}$ nhúng các thông tin backdoor đã mã mật hóa, $\mathcal{E}(\mathcal{I}(k_{priv}))$, vào trong khóa công khai. Khóa công khai có dạng cuối là $k_{pub*} = \mathcal{M} \circ \mathcal{E} \circ \mathcal{I}(k_{priv})$, (1).

2) Định nghĩa thuật toán sinh khóa chứa backdoor

Ký hiệu B_0 , B_1 lần lượt là thiết bị hộp đen được cài đặt thuật toán sinh khóa G_0, G_1. Ký hiệu R_1 là thuật toán khôi phục cặp khóa được tạo bởi G_1. Các cặp khóa được tạo ra bởi

Tác giả liên hệ: Lê Quang Huy
Email: lequanghuyabc@gmail.com
Đến tòa soạn: 5/2017, chỉnh sửa: 8/2017, chấp nhận đăng: 9/2017

G_1 là các cặp khóa chứa backdoor an toàn nếu $G_1(\ell)$ tạo ra cặp khóa (k_{pub}, k_{priv}) với các thuộc tính sau được thỏa mãn:

1. Tính bảo mật:

a) người thiết kế nhúng một phần khóa riêng vào trong khóa công khai tương ứng, $k_{pub*} = \mathcal{M} \circ \mathcal{E} \circ \mathcal{I}(k_{priv})$ (2).

b) người dùng, kẻ tấn công không thể tính toán được khóa riêng từ khóa công khai tương ứng, $k_{priv} \neq \mathcal{I}^{-1} \circ \mathcal{E}^{-1} \circ \mathcal{M}^{1}$ (k_{pub}) (3).

2. Tính hoàn chỉnh: Tồn tại thuật toán R_1 , để người thiết kế có thể khôi phục được khóa riêng từ khóa công khai tương ứng, $k_{priv} = R_1(k_{pub})$. Hay các hàm $\mathcal{M}$, $\mathcal{E}$, $\mathcal{I}$ khả nghịch để người thiết kế tính được $k_{priv} = \mathcal{I}^{1} \circ \mathcal{E}^{-1} \circ \mathcal{M}^{1}$ (k_{pub}), (4).

3. Khả năng ẩn giấu (khả năng không thể phân biệt được):

a) Kết quả đầu ra của B_0 và B_1 là không thể phân biệt được về mặt thống kê, hoặc về mặt tính toán.

b) Các đo đạc bên ngoài B_0 và B_1 không thể phân biệt được một cách rõ ràng cái này với cái kia.

B. Một số tiêu chuẩn đánh giá thuật toán sinh khóa chứa backdoor

Các tiêu chuẩn đánh giá thuật toán sinh khóa chứa backdoor (mục 5 trong [5]) bao gồm các tính chất liên quan đến bản thân backdoor như: tính bảo mật và tính hoàn chỉnh; các thuộc tính liên quan đến khóa gồm lực lượng khóa, các thuộc tính phân phối và tương quan giữa các thành phần khóa; các thuộc tính liên quan đến thuật toán gồm độ phức tạp tính toán, và bộ nhớ sử dụng. Các tiêu chuẩn đánh giá thuật toán sinh khóa chứa backdoor được cho trong bảng 1.

Bảng I. Các tiêu chuẩn đánh giá thuật toán sinh khóa chứa backdoor

Đánh giá / Tiêu chuẩn	Tốt	Trung bình	Kém (thất bại)
Bảo mật	$l_{\mathcal{E}} >= l_{G1}$	$l_{G1} >= l_{\mathcal{E}} >= l_{G1}/2$	$l_{\mathcal{E}} < l_{G1}/2$
Hoàn chỉnh	$\forall k_{pub}$ kẻ tấn công $k_{priv} \neq \mathcal{F}^{-1}(k_{pub})$	-	$\exists k_{pub}$ kẻ tấn công $k_{priv} = \mathcal{F}^{-1}(k_{pub})$
Lực lượng khóa	c >= - ½	-1/2 > c >= -3/2	c < -3/2
Tính phân phối	$\mathcal{D}_{G1} \approx 0$	$\mathcal{D}_{G1} \approx 0$	$\mathcal{D}_{G1} > 0$
Tính tương quan	Không tương quan	-	$\exists$ thành phần khóa tương quan
Độ phức tạp	Tuyến tính (a =< 1 và c =< 1)	< bậc 2 (2 > a > 1 hoặc 2 > c > 1)	>= bậc 2 (a >= 2 hoặc c >= 2)
Bộ nhớ	Không dùng VM, NM	Chỉ dùng VM	Dùng NM

C. Một số kết quả về hệ mật RSA

1) Định lý về số các số nguyên tố

Ký hiệu $\pi(n)$ là số lượng các số nguyên tố nhỏ hơn hoặc bằng n.

Thì khi n lớn, ta có: $\pi(n) \sim \frac{n}{\ln n}$ (5) (mục 4.1.2 trong [6])

Giả sử p là số nguyên tố k bít, số lượng các số nguyên tố k-bit

$$\#\{p\} = \pi(2^k) - \pi(2^{k-1}) = \frac{(k-2)2^{k-1}}{k(k-1)\ln 2} \approx \frac{2^{k-1}}{k \ln 2}$$

$$= \frac{2^k}{k\, 2\ln 2} \approx 2^{k-log_2 k} \text{ (6)}$$

Xác xuất một số nguyên k bít là số nguyên tố:

$$\Pr[\text{một số } k \text{ bit là số nguyên tố}] = \frac{\pi(2^k) - \pi(2^{k-1})}{2^k - 2^{k-1}}$$

$$\approx \frac{2^{k-log_2 k}1}{2^{k-1}} = 2^{1-log_2 k} = \frac{2}{k} \text{ (7)}$$

2) Số lượng khóa có thể sinh được

Ta có $\#\{p\} = \#\{q\} \approx 2^{k-log_2 k}$

Với mỗi tham số e xác định một tham số d duy nhất, tham số e có thể được chọn ngẫu nhiên trong $Z^*_{\varphi(n)}$. Số các số nguyên như vậy là số các số nguyên nguyên tố cùng nhau với $\varphi(n)$, và bằng $\varphi(\varphi(n))$. Từ [6, Fact 2, p102], ta có:

$$\frac{n}{6 \ln\ln n} < \varphi(n) < n, \text{ (8)}$$

$$\frac{n}{36\,(\ln\ln n)\ln\ln(n/(6\ln\ln n))} < \varphi(\varphi(n)) < n$$

Trong đó ln ln n là viết tắt của ln(ln(n)).

Chia n cho ln n không ảnh hưởng nhiều đến kết quả nên ta xấp xỉ $\frac{n}{\ln n} \approx n$. $\varphi(n)$ hoặc $\varphi(\varphi(n))$ có thể được xấp xỉ bằng $n = 2^{2k}$. Với mỗi giá trị e xác định duy nhất một nghịch đảo d, nên ta có lực lượng khóa của thuật toán sinh khóa RSA là:

$\#\{(p,q,d,e)\} = \#\{(p, q)\} \bullet \#\{e\} \approx (2^{k - \log_2 k})^2 \bullet \varphi(\varphi(n))$

$\approx 2^{2k - 2\log_2 k} \cdot 2^{2k} = 2^{4k - 2\log_2 k}$.

Đối với thuật toán sinh khóa có yêu cầu về tiêu chuẩn đối với các tham số (p,q,d,e) lực lượng khóa của thuật toán sẽ nhỏ hơn.

3) Định lý Coppersmith (Theorem 4, 5 trong [7]*)*

Trong thời gian đa thức, có thể tìm được phân tích nhân tử của $n = p.q$ nếu biết ¼ $\log_2 n$ (khoảng ½ độ dài bit của p) các bít thấp (cao) của p.

III. ĐỀ XUẤT THUẬT TOÁN SINH KHÓA RSA CHỨA BACKDOOR MỚI

A. Bổ đề 1

Cho p, q là hai số nguyên tố, có cùng chiều dài k bit và giả sử $p < q$. Ký hiệu $n = p.q$, n có độ dài $2k$ bit. Ký hiệu $\lfloor\sqrt{n}\rfloor$ là phần nguyên của căn bậc 2 của n. Ký hiệu s là số nguyên nhỏ nhất thỏa mãn $(q - p) \leq 2^s$, $s < k$. Ký hiệu *div* là phép chia lấy thương nguyên. Ta luôn có : $p < \lfloor\sqrt{n}\rfloor < q$ và $p\ div\ 2^s = \lfloor\sqrt{n}\rfloor\ div\ 2^s$ (các bit cao từ s tới k trong p và $\lfloor\sqrt{n}\rfloor$ là giống nhau).

Chứng minh:

Ta có: $(q - p) \leq 2^s \Leftrightarrow q \leq p + 2^s$

$\Rightarrow q\ div\ 2^s \leq (p + 2^s)\ div\ 2^s$

$\Leftrightarrow q\ div\ 2^s \leq (p\ div\ 2^s) + 1$

Vì s là số nguyên nhỏ nhất thỏa mãn $(q - p) < 2^s$ nên $q\ div\ 2^s = (p\ div\ 2^s) + 1$

Ta có: $p < \lceil\sqrt{n}\rceil < q \Rightarrow p \; div \; 2^s \leq \lceil\sqrt{n}\rceil \; div \; 2^s \leq q \; div \; 2^s = (p \; div \; 2^s) + 1$

$\Rightarrow p \; div \; 2^s = \lceil\sqrt{n}\rceil \; div \; 2^s$. Điều này có nghĩa là các bit cao từ s tới k trong p và $\lceil\sqrt{n}\rceil$ là giống nhau hoặc (k - s) bít cao trong p và $\lceil\sqrt{n}\rceil$ giống nhau.

B. Bổ đề 2

Cho p, q là các số nguyên tố, có cùng chiều dài bit là k bit và giả sử $p < q$. Ký hiệu $n = p.q$, ký hiệu *nlen* là độ dài theo bit của n, $nlen = 2k$. Ký hiệu $\lceil\sqrt{n}\rceil$ là phần nguyên của căn bậc 2 của n. Ký hiệu s là số nguyên nhỏ nhất thỏa mãn $(q - p) \leq 2^s$. Ký hiệu v là một số nguyên. Ký hiệu $p\rfloor^v$ là v bit cao của p và ký hiệu $p\rfloor^v$ là v bit thấp của p. Nếu biết được $(p\rfloor^{k/2})\rfloor^{s - k/2}$ ta luôn tìm được p.

Chứng minh:

Từ kết quả của bổ đề 1 ta luôn có $p\rfloor^{k-s} = (\lceil\sqrt{n}\rceil)\rfloor^{k-s}$

Từ giả thiết của bổ đề 2 ta biết được $(p\rfloor^{k/2})\rfloor^{s - k/2}$ và kết hợp với kết quả của bổ đề 1 ta biết được $p\rfloor^{k/2}$.

Sử dụng định lý của Coppersmith (2.3.3) ta tính được p.

C. Bổ đề 3

Cho n' là một số nguyên có độ dài $2k$ bit. Cho p là một số nguyên tố có độ dài k bit, đặt $q = [n' / p]$, ký hiệu $n = p.q$. Thì ta luôn có n và n' giống nhau ở k bit cao hay n div $2^k = n'$ div 2^k hay $n\rfloor^k = n'\rfloor^k$

Chứng minh

Ta có: $q = [n' / p]$ và $n = p.q$, nên $n' > n$, và $n' - n < p \Rightarrow (n' - n) \; div \; 2^k \leq p \; div \; 2^k \Leftrightarrow (n' - n) \; div \; 2^k \leq 0$.

Vì $p < 2^k$ nên $(n' - n) \; div \; 2^k = 0$ hay $n' \; div \; 2^k = n \; div \; 2^k$.

D. Giới thiệu thuật toán đề xuất

Thuật toán đề xuất sử dụng ý tưởng của thuật toán Hidden Prime Factor: thông tin backdoor là một phần của p được mã mật hóa bởi hệ mật đối xứng F. Điểm khác biệt là phần thông tin backdoor có thể được rút gọn với chiều dài s - k/2 bit, nên có lực lượng lớn hơn. Với thông tin backdoor rút gọn, thuật toán có thể thỏa mãn một phần ràng buộc tham số của p, q; ($|p - q| > 2^{nlen/2 - 100}$) trong [8], mục B.3.1 (chọn $k > s \geq k - 100$).

Các tham số thuật toán:

+ G_1 = Thuật toán đề xuất; $\mathcal{Z}(k_{priv}) = ((p\rfloor^{k/2})\rfloor^{s - k/2})$; $\mathcal{E}$ = AES, $\ell_{\mathcal{E}} = \log_2 n = 2k$, $k(\mathcal{E}) = (K)$; $\mathcal{M} = n$, với $\log_2 n = 2k$; $\log_2 p = \log_2 q = k$.

+ Hàm F: là hàm mã mật hóa đối xứng ví dụ AES

+ Hàm G: $\{0, 1\}^{2k} \times \{0, 1\}^{k/2} \times \{0, 1\}^{k/2} \rightarrow \{0, 1\}^k$, hàm này thực hiện kết quả của định lý Coppersmith (mục 2.3.3), ví dụ: $p = G(n, 2^{k/2}, p \; div \; 2^{k/2})$

+ Hàm C: là hàm kiểm tra điều kiện ($p < q$) và $(q - p) \leq 2^s$ với s cho trước (có thể chọn s thỏa mãn $k > s \geq k - 100$).

Thuật toán đề xuất [sinh khóa RSA]

Input (k, K)
Output (p, q, d, e)
1: Generate a random prime p
2: Generate a random q' //$\log_2 q' = k$
3: $n_1 = p.q'$
4: $\tau = n_1\rfloor^{k/2}$; $\mu = F_K((p\rfloor^{k/2})\rfloor^{s - k/2})$;
5: **repeat**
6: Generate a random r //$\log_2 r = 3k/2 - s$
7: $n_2 = \tau : \mu : r$
8: $q = [n_2 / p]$
9: **until** (q is prime) and (C)
10: $n = p.q$ //Bổ đề 3
11: Generate a random odd e;
//$\log_2 e \leq 2k$ and $\gcd(e, \varphi(n)) = 1$
12: $d \equiv e^{-1} \bmod \varphi(n)$
13: Output (p, q, d, e)

Thuật toán đề xuất [khôi phục khóa RSA]

Input (n, e, K)
Output (d)
1: $\mu = (n\rfloor^{3k/2})\rfloor^{s - k/2}$ // Bổ đề 3
2: $p_1 = (\lceil\sqrt{n}\rceil)\rfloor^{k-s}$ // Bổ đề 1
3: $p_2 = F^{-1}_K(\mu)$
4: $p\rfloor^{k/2} = p_1 : p_2$
5: $p = G(n, 2^{k/2}, p\rfloor^{k/2})$ //Bổ đề 2
6: $d \equiv e^{-1} \bmod \varphi(n)$.
7: Output (d).

E. Đánh giá thuật toán đề xuất

Tính bảo mật: Tính bảo mật được đánh giá ở mức tốt vì người thiết kế sử dụng hệ mật đối xứng có độ dài tham số an toàn tương đương với độ dài tham số an toàn của người dùng.

Tính hoàn chỉnh: Tính hoàn chỉnh được đánh giá ở mức tốt. Với thuật toán đề xuất người thiết kế luôn tính được khóa riêng từ khóa công khai tương ứng.

Lực lượng khóa:

Xét việc tạo p (bước 1) Vì p được tạo ngẫu nhiên nên theo định lý về số lượng các số nguyên tố (mục 2.3.), ta có $\#\{p\} = 2^{k - \log_2 k}$

Xét việc tạo q (từ bước 5 đến bước 9). Với mỗi p có thể chọn ngẫu nhiên τ và r, do đó tạo được $2^{2k - s}$ giá trị n_2. Vì q được tính theo n_2 và p (bước 5), nên $\#\{ q \} = \#\{n_2 / p\}$. $\Pr[q \; là \; số \; nguyên \; tố]$

Theo định lý về số các số nguyên tố (mục 2.3), một số nguyên k-bit là số nguyên tố với xác suất khoảng $2^{1 - \log_2 k}$. Vì vậy:

$\#\{ q \} = \#\{ n_2 / p\}$. $\Pr[q \; là \; số \; nguyên \; tố] = 2^{2k - s - k + \log_2 k} . 2^{1 - \log_2 k} = 2^{k - s + 1}$

Vậy số lượng n là: $\#\{n\} = \#\{p\} . \#\{q\} = 2^{k - \log_2 k} . 2^{k - s + 1} = 2^{2k - s + 1 - \log_2 k}$

Khi áp dụng một ràng buộc (tiêu chuẩn) đối với các tham số, lực lượng của thuật toán G_0 và G_1 đều giảm một lượng xác định, tuy nhiên tỷ lệ lực lượng giữa chúng không thay đổi.

Xét tỷ lệ giữa lực lượng của G_1 và G_0, vì e được sinh tự do trong G_1 giống trong G_0 nên hạng tử $\#\{e\}$ có thể bỏ qua, ta có:

$$R_{G_1} = \frac{N_{G_1,n}}{N_{G_0,n}} \approx \frac{2^{2k-s+1-\log_2 k}}{2^{2k-2\log_2 k}} = 2^{1-s+\log_2 k}$$

Vì $s > k/2 \Rightarrow -s + 1 + \log_2 k < -k/2 + 1 + \log_2 k$. Vậy $R_{G_1} < 2^{-k/2}.\frac{2}{k} < -1/2$. Vì hằng số $c < -1/2$ trong R_{G_1} nên lực lượng của G_1 được đánh giá là tốt.

Tính chất phân phối: Thông tin backdoor được mã mật hóa và nhúng vào vị trí cố định trong *n*. Tuy nhiên vì thông tin backdoor được mã mật hóa bằng mã khối đối xứng (gần với phân bố đều) nên phần nhúng thông tin backdoor cũng có phân phối gần với phân phối đều. Việc sinh *p* ngẫu nhiên và *q* được tạo thông qua việc chia n_2 cho *p* với giá trị n_2 phụ thuộc một phần vào *p*, tham số ngẫu nhiên *r* và tham số ngẫu nhiên *q'*. Do vậy phân phối của *n* có thể gần tới phân bố đều và do vậy khoảng cách thống kê giữa thành phần *n* của G_1 và G_0 là xấp xỉ bằng 0, $\mathcal{D}_{G1} \approx 0$. Vậy tính chất phân phối của G_1 được đánh giá là tốt.

Tương quan giữa các thành phần khóa: Vì thông tin backdoor được nhúng vào tham số *n* nên tham số *e* có thể được tạo độc lập. Theo cách tạo *q* (từ bước 2 đến bước 6), *q* phụ thuộc vào *p*, tham số ngẫu nhiên *r* và tham số ngẫu nhiên *q'*. Nếu người dùng cố định *p* và yêu cầu sinh lại *q* thì việc sinh lại khóa tổng quát thực hiện được. Tuy nhiên, nếu người dùng yêu cầu ngược lại, tức là cố định *q* và sinh lại *p* là không khả thi. (Việc sinh lại khóa trong trường hợp này chỉ khả thi nếu thuật toán cho phép hoán đổi được vai trò của *p* và *q*). Do vậy tính tương quan giữa các thành phần khóa của G_1 được đánh giá đạt mức trung bình.

Độ phức tạp tính toán:

Vì *p* được tạo giống như trong G_0 nên ta có $t_p(G_0) = t_p(G_1)$.

Trong vòng lặp tạo *q* (từ bước 2 đến bước 6) có thêm việc mã mật hóa thông tin backdoor bằng hệ mật đối xứng (AES). Tuy nhiên độ phức tạp của hệ mật đối xứng AES không đáng kể so độ phức tạp của việc tạo *q* nên độ phức tạp của việc tạo *q* trong G_1 cũng gần giống như trong G_0 nên ta có: $t_q(G_1) = t_q(G_0)$.

Và độ phức tạp tạo *n* là: $t_n(G_1) = t_p + t_q = t_n$.

Cách tạo *e* của G_1 cũng giống với G_0, nên độ phức tạp tạo *e*: $t_e(G_1) = t_e(G_0)$.

Vậy độ phức tạp của thuật toán là: $T(G_1) = t_n + t_e$. Do đó độ phức tạp của G_1 được đánh giá là "tốt" vì nó tuyến tính đối với độ phức tạp của G_0.

Bộ nhớ sử dụng: Thuật toán không sử dụng bộ nhớ NM và VM nên nó có thuộc tính bộ nhớ sử dụng được đánh giá là tốt.

F. Tóm tắt các điểm cải tiến của thuật toán đề xuất

Bảng II Tóm tắt các so sánh đánh giá giữa thuật toán đề xuất với thuật toán Hidden Prime Factor:

Thuộc tính \ Thuật toán	Thuật toán đề xuất	Hidden Prime Factor
Bảo mật	Tốt	Tốt
Hoàn chỉnh	Tốt	Tốt
Lực lượng khóa	Tốt	Tốt
Thuộc tính phân phối	Tốt	Tốt
Tính tương quan	Trung bình	Trung bình
Độ phức tạp	Tốt	Tốt
Bộ nhớ	Tốt	Tốt

IV. KẾT LUẬN

Trên cơ sở thuật toán *Hidden Prime Factor*, một thuật toán sinh khóa chứa backdoor mới được đề xuất với các ưu điểm về lực lượng khóa tốt hơn so với thuật toán *Hidden Prime Factor* và thuật toán này có thể sinh các cặp khóa thỏa mãn một phần ràng buộc tham số *p, q* ($|p - q| > 2^{nlen/2 - 100}$) trong [8], mục B.3.1. Thuật toán đề xuất có thể ứng dụng tốt trong phần sinh khóa của thiết bị PKI Token hoặc HSM (Hardware Security Module). Ngoài ra thuật toán có thể được xem xét thêm theo hướng rút bớt thông tin backdoor hoặc xem xét sự phù hợp của các tham số do thuật toán sinh ra với tiêu chuẩn tham số ứng dụng cho một hạ tầng PKI cụ thể.

TÀI LIỆU THAM KHẢO

[1]. Claude Crepeau and Alain Slakmon, Simple Backdoors for RSA Key Generation, Available online at https://pdfs.semanticscholar.org/0428/db02e8c46cd4b1b8a43359503bd9509034f6.pdf (2003)

[2]. Young A and Yung M, Kleptography: Using Cryptography Against Cryptography, Available online at https://cryptome.org/2013/09/klepto-crypto.pdf (1997)

[3]. Young A and Yung M, Malicious Cryptography: Kleptographic Aspects, Available online at https://pdfs.semanticscholar.org/6c9c/9bb21f1b52480df05ce7a9266436ff594535.pdf (2005)

[4]. Young A and Yung M, A Space Efficient Backdoor in RSA and Its Applications, Available online at http://link.springer.com/chapter/10.1007%2F11693383_9#page-1 (2005)

[5]. G.Arboit, Two mathematical security aspects of the rsa cryptosystem, Available online at http://crypto.cs.mcgill.ca/~crepeau/PDF/these-Genevieve.pdf (2008)

[6]. A. Menezes, P. van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, CRC Press (2008)

[7]. D Coppersmith, Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities, Available online at https://www.di.ens.fr/~fouque/ens-rennes/coppersmith.pdf (1995)

[8]. FIPS, *FIPS PUB 186-4;* Digital Signature Standard, Available online at http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf (2013)

A SYMETRIC BACKDOOR IN RSA KEY GENERATION

This paper presents a propose of a symmetric backdoored RSA key generation algorithm based on improvement of Hidden Prime Factor algorithm in [1]. *The proposed algorithm use use the Coppersmith*'s theorem *in* [7] *to reduce backdoor information for embedding.*

Keywords: Cryptography, key generation, RSA, backdoor

Bạch Nhật Hồng, Nhận học vị Tiến sĩ năm 1989, học hàm PGS năm 2001. Hiện công tác tại Trường Đại học Sự phạm kỹ thuật Hưng yên. Lĩnh vực nghiên cứu: Mật mã, An toàn thông tin, Điện tử, Tự động.

Lê Quang Huy, Nhận học vị Thạc sỹ năm 2008. Hiện công tác tại Ban Cơ yếu Chính phủ. Lĩnh vực nghiên cứu: Mật mã, PKI, RSA, ECC.