# MỘT SỐ KHUYẾN NGHỊ VỀ ĐỘ AN TOÀN CỦA HỆ MẬT RSA (PHẦN 1)

**ThS. Phạm Quốc Hoàng, Đặng Tuấn Anh**
*Học viện Kỹ thuật mật mã*

*Trong hệ mật RSA, mô hình hệ mật, cấu trúc thuật toán của các nguyên thủy mật mã là công khai. Tuy nhiên, việc lựa chọn và sử dụng các tham số cho hệ mật này sao cho an toàn và hiệu quả là một vấn đề đã và đang được nhiều tổ chức quan tâm nghiên cứu. Trong bài viết này, nhóm tác giả đã tổng hợp một số khuyến nghị cho mức an toàn đối với độ dài khóa RSA được Lenstra, Verheul và ECRYPT đề xuất.*

## GIỚI THIỆU

Hệ mật RSA được đặt tên theo các chữ cái viết tắt tên ba tác giả phát minh ra hệ mật này là Ron Rivest, Adi Shamir và Leonard Adleman [1], bao gồm hai nguyên thủy là hệ mật khóa công khai RSA và lược đồ chữ ký số RSA đã và đang được tích hợp trong hầu hết các sản phẩm bảo mật và an toàn thông tin hiện nay. Để triển khai sử dụng các nguyên thủy mật mã RSA, các bộ tham số RSA được sinh, phân phối dưới dạng các chứng thư số (khóa công khai) và khóa riêng thông qua các hệ thống cơ sở hạ tầng khóa công khai (PKI).

Các nguyên thủy mật mã RSA cùng bộ tham số đi kèm có thể được sử dụng cho nhiều mục đích trong các hệ thống thông tin dùng trong cả lĩnh vực kinh tế xã hội cũng như an ninh quốc phòng, như: bảo mật, xác thực thư tín điện tử, bảo mật và xác thực web, bảo mật và xác thực giao thức truyền tệp (FTP). Dưới đây là thuật toán sinh bộ tham số khóa cho các nguyên thủy mật mã RSA. Mỗi thực thể trong hệ thống trước khi muốn sử dụng các nguyên thủy mật mã RSA sẽ tạo hoặc được tạo một bộ tham số RSA theo các bước sau [1]:

1. Sinh ngẫu nhiên hai số nguyên tố lớn khác nhau $p$ và $q$.

2. Tính $N = pq$, $\varphi(N) = (p-1)(q-1)$.

3. Chọn $e$ thỏa mãn $1<e<\varphi(N)$ sao cho $(e,\varphi(N)) = 1$.

4. Tính số nguyên dương $d$ thỏa mãn: $1< e < \varphi(N)$ và $ed = 1 \bmod \varphi(N)$.

5. Đầu ra là bộ tham số: $N, e, d$.

Trong đó: $N$ là RSA modulo, $e$ là khóa công khai, $d$ là khóa bí mật (thông thường khi nói bộ

tham số RSA 2048 bit có nghĩa là bộ tham số RSA với modulo *N* sẽ có độ dài là 2048 bit). Độ dài modulo trong thuật toán RSA rất quan trọng, vì độ an toàn của hệ mật RSA dựa trên bài toán phân tích số nguyên.

## MỘT SỐ KHUYẾN NGHỊ

### Khuyến nghị của Lenstra và Verheul

Giả sử rằng kích thước khóa phải được xác định hiện tại ít nhất tương đương với kích thước khóa đối xứng *d*.

Lưu ý rằng các công thức kết quả phải độc lập với các giả định của các tác giả về biên độ an toàn, tiến bộ phần cứng hoặc tiến bộ trong thám mã. Các điều chỉnh duy nhất được sử dụng ở đây là *P* và *v*, vì chúng là các điều chỉnh duy nhất phù hợp với hoàn cảnh hiện tại.

So với việc thám DES 56 bit với chi phí dự kiến là $5x10^5$ Mips-Years, việc phá khóa kích thước *d* được sử dụng cùng với hệ mật khóa đối xứng chậm hơn v lần so với DES có thể mất:

$$EMY(d) = 2^{(d-56)} x 5x10^5 x v \text{ Mips-Years}$$

Trong đó, EMY là viết tắt của "Số lượng Mips-Years tương đương" (Equivalent number of Mips-Years) [2].

Nếu kích thước khóa bất đối xứng cổ điển *k* được chọn sao cho:

$$\frac{L[2^k]}{EMY(d)} \geq \frac{L[2^{512}]}{10^4}$$

thì độ an toàn được đưa ra bởi các hệ mật bất đối xứng cổ điển hiện tại ít nhất tương đương về mặt tính toán với độ an toàn được đưa ra bởi khóa đối xứng có kích thước *d*.

Tuy nhiên, nếu kích thước khóa bất đối xứng cổ điển *k'* được chọn sao cho:

$$\frac{L[2^{k'}]}{EMY(d)} \geq \frac{L[2^{512}]}{10^4 \times 26 \times P}$$

thì độ an toàn được đưa ra bởi các hệ mật bất đối xứng cổ điển hiện có chi phí ít nhất tương đương với độ an toàn được đưa ra bởi khóa đối xứng có kích thước *d*.

Từ các công thức được đưa ra tại đây, rõ ràng các công thức thu được cho các kích thước khóa tương đương với kích thước khóa đối xứng đã cho trong một năm xác định: sử dụng các công thức bằng IMY(y) (Infeasible number of Mips-Years) được thay thế bằng EMY(d) [2].

*Tra cứu các kích thước khóa tương đương về mặt tính toán hiện tại*

Với kích thước khóa đối xứng d, các kích thước khóa bất đối xứng hiện tương đương về mặt tính toán với nó có thể được tra cứu như sau. Đối với các hệ thống bất đối xứng cổ điển, hãy tra cứu kích thước khóa bất đối xứng cổ điển cho năm $y' = 30 \times d/43 + 1950,8$. Công thức này theo sau bằng cách giải phương trình:

$$EMY(d) = IMY(y') \times 2^{(12(y'-1999))/y}$$

Đối với các hệ mật khác, hãy để *y* là năm trong [2], trong đó *d* xuất hiện trong cột kích thước khóa đối xứng.

Với kích thước khóa cổ điển bất đối xứng $k$, kích thước khóa đối xứng tương đương về mặt tính toán hiện tại có thể được tìm thấy bằng cách tra cứu năm $y$ xảy ra $k$ và bằng cách sử dụng kích thước khóa đối xứng $43 \times y/30 - 2796,2$; nó theo ngay sau $y' = 30 \times d/43 + 1950,8$.

Ví dụ, đối với khóa đối xứng có kích thước $d=85$, chúng ta thấy rằng $y=2019$ và $y' = 30 \times 85/43 + 1950,8 = 2010,1$. Kích thước khóa tương đương về mặt tính toán hiện tại là: khoảng 1375 bit đối với khóa bất đối xứng cổ điển, nhóm con có kích thước 150+2=152 trên 1375 trường bit và hệ thống EC 160 bit. Tương tự, đối với khóa bất đối xứng cổ điển có kích thước $k'=1024$, chúng ta thấy rằng $y=2002$ và kích thước khóa đối xứng tương đương về mặt tính toán hiện tại được cho bởi $43 \times 2002/30 - 2796,2 \approx 74$.


*Tra cứu các kích thước khóa tương đương với chi phí hiện tại*

Với kích thước khóa đối xứng *d*, kích thước khóa bất đối xứng hiện có giá trị tương đương với nó có thể được tra cứu theo cách rất giống nhau: chỉ cần thay thế 1950,8 và 2796,2 bằng 1942,9 và 2784,9 tương ứng. Công thức này theo sau bằng cách giải phương trình:

$$\frac{EMY(d)}{26 \times P} = IMY(y') \times 2^{12(y'-1999)/r}$$

Ví dụ, đối với khóa đối xứng có kích thước *d=85*, chúng ta thấy rằng *y=2019* và $y'=30\times85/43+1942,9=2002,2$. Kích thước khóa tương đương với chi phí hiện tại là: khoảng 1036 bit cho khóa bất đối xứng cổ điển, nhóm con có kích thước 150+2=152 trên các trường 1036 bit và hệ mật EC 160 bit.

Tương tự, đối với khóa bất đối xứng cổ điển có kích thước *k=1024*, chúng ta thấy rằng *y=2002* và kích thước khóa đối xứng tương đương chi phí hiện tại được cho là 43×2002/30-2784,9≈85.

**Khuyến nghị của ECRYPT**

Việc cung cấp các kích thước khóa để sử dụng lâu dài có phần hơi khó khăn, vì ngay từ đầu, nó giả định rằng thuật toán mà ta đang chọn kích thước khóa không bị phá vỡ trong thời gian đó. Vì vậy, khi cung cấp các kích thước khóa cho các miền ứng dụng cụ thể, ta đưa ra một giả định ngầm định rằng nguyên mẫu, lược đồ hoặc giao thức sử dụng kích thước khóa này sẽ không bị phá vỡ trong tương lai gần. Tất cả các nguyên mẫu và lược đồ được đánh dấu là phù hợp để sử dụng trong tương lai được tin tưởng sẽ bảo mật trong một khoảng thời gian đáng kể.

Tuy nhiên, việc đưa ra giả định này vẫn bao hàm một mức độ lựa chọn đối với kích thước khóa. Mã khối AES có thể vẫn an toàn trong 50 năm tới, nhưng người ta có thể muốn sử dụng kích thước khóa lớn hơn cho dữ liệu mà họ muốn bảo mật trong 50 năm thay vì với số năm ít hơn, chẳng hạn như 5 năm. Do đó, khi cung cấp các hướng dẫn về kích thước khóa, ta đưa ra hai trường hợp riêng biệt cho các lược đồ phù hợp để sử dụng trong tương lai. Trường hợp đầu tiên là bảo mật đảm bảo trong ít nhất mười năm (gọi là ngắn hạn) và thứ hai là bảo mật từ ba mươi đến năm mươi năm (gọi là dài hạn). Các nhà khoa học lưu ý rằng đây hoàn toàn là các nguyên tắc về kích thước khóa và chúng không đảm bảo tính bảo mật cũng như không đảm bảo chống lại các cuộc tấn công vào các nguyên tắc toán học cơ bản.

Trong Bảng 1 sẽ trình bày các hướng dẫn về kích thước khóa một cách chi tiết. Về cơ bản chúng đã tuân theo sự tương đương của NIST giữa các

*Bảng 1. Phân tích kích thước khóa*

| | Tham số | Kế thừa | Hệ sử dụng trong tương lai | |
|---|---|---|---|---|
| | | | Ngắn hạn | Dài hạn |
| Kích thước khóa đối xứng | $k$ | 80 | 128 | 256 |
| Đầu ra hàm băm | $m$ | 160 | 256 | 512 |
| Đầu ra MAC | $m$ | 80 | 128 | 256 |
| Bài toán RSA | $l(n) \geq$ | 1024 | 3072 | 15360 |
| Bài toán logarit rời rạc trên trường hữu hạn | $l(p^n) \geq$ | 1024 | 3072 | 15360 |
| | $l(p), l(q) \geq$ | 160 | 256 | 512 |
| ECDLP | $l(q) \geq$ | 160 | 256 | 512 |
| Ghép | $l(p^{k.n}) \geq$ | 1024 | 6144 | 15360 |
| | $l(p), l(q) \geq$ | 160 | 256 | 512 |

kích thước khóa khác nhau. Tuy nhiên, các kích thước khóa tương đương này cần được hiểu là chỉ áp dụng cho thuật toán tốt nhất cho mật mã khối, hàm băm, tham số RSA,... Rõ ràng là mật mã khối có độ an toàn 128 bit không cung cấp độ an toàn 128 bit do các cuộc tấn công thám mã.

Trong tài liệu [3] đã tập trung vào độ an toàn 128-bit để hướng dẫn sử dụng trong tương lai, nó có thể cung cấp bảo mật tốt về lâu dài. Khuyến cáo tương tự (giả sử) ở độ an toàn 112 bit (tương ứng với các khóa RSA khoảng 2048 bit) có thể được đưa ra. Giới hạn phải được vạch rõ ở đâu đó và có sự đồng thuận chung rằng nó phải ở trên mức 100 bit, cho dù chọn 112 bit hay 128 bit làm mức tiêu chuẩn là do sở thích. Do nhu cầu bảo vệ dữ liệu dài hạn, các nhà khoa học đã đưa ra lựa chọn thận trọng và giải quyết bài toán ở 128 bit, với mức độ cao hơn để sử dụng lâu dài.

Lưu ý, trong trường hợp kích thước đầu ra MAC, giá trị được cung cấp là giá trị cần thiết để bảo vệ chống lại việc tấn công vét cạn tiền ảnh. Tuy nhiên, trong nhiều ứng dụng, MAC là mã thông báo xác thực tồn tại trong thời gian ngắn và do đó, người ta chỉ cần bảo vệ chống lại kẻ tấn công yếu hơn nhiều hoặc thậm chí chỉ ở mức bảo mật thống kê. Do đó trong nhiều ứng dụng, kích thước đầu ra MAC là 48 bit là đủ.

## KẾT LUẬN

Bài báo đã tổng hợp một số khuyến nghị về độ dài modulo trong thuật toán RSA, trong đó có trình bày khuyến nghị của Lenstra, Verheul và ECRYPT. Các nội dung được trình bày trong bài báo là cơ sở để tính toán độ dài modulo RSA trong hiện tại và tương lai gần. Tuy nhiên, các tính toán trong bài báo này là dựa trên năng lực tính toán của máy tính thông thường, nếu máy tính lượng tử ra đời và đủ mạnh sẽ đòi hỏi các lập luận mới. Trong phần II, nhóm tác giả sẽ tiếp tục trình bày những khuyến nghị của BSI (Cơ quan An ninh thông tin của Cộng hòa Liên bang Đức) và của RFC 3766.❖


**TÀI LIỆU THAM KHẢO**

[1]. *Hoàng Văn Thức, Đinh Quốc Tiến, Xem xét mức an toàn với độ dài khóa RSA, Tạp chí An toàn thông tin, 2023.*

[2]. *A. Lenstra, E. Verheul, Selecting Cryptographic Key Sizes, 2001.*

[3]. *Ecrypt-CSA, Algorithm, Key size and Protocols Report, 2018.*

[4]. *BSI-Technical Guideline, Cryptographic Mechanisms: Recommendation and Key Lengths.*

[5]. *RFC 3766, H. Orman and P. Hoffman Determining strengths for public keys used for exchanging symmetric keys, 2004.*