ADVANCED ELGAMAL THUẬT TOÁN MẬT MÃ KHOÁ BẤT ĐỐI XỨNG TƯƠNG LAI

PHẠM HỒNG LIÊN, NGUYỄN THÀNH SƠN

I. GIỚI THIỆU

Tác giả xây dựng thuật toán mới thuộc hệ mã bất đối xứng, dựa trên độ khó của bài toán logarit với các đối số cực lớn (ý tưởng của Elgamal), nên lấy tên là Advanced Elgamal. Mô hình mật mã này không những khắc phục những nhược điểm của Elgamal mà còn thêm nhiều ưu điểm vượt trội về độ bảo mật và tối ưu kích thước dữ liệu sau mã hóa.

II. THUẬT TOÁN

Thuật toán Elgamal còn nhược điểm khá lớn là tạo ra các văn bản mã giống nhau nếu cùng khối văn bản gốc. Điều này là một yếu điểm chung của phương pháp mật mã khoá bất đối xứng, làm giảm tính an toàn của thuật toán vì có thể sử dụng phương pháp thám mã theo xác suất [1, 2]. Mặt khác, các khối dữ liệu sau mã hoá đi trên mạng, do chủ quan hay khách quan, một vài khối có thể bị mất đi hoặc thêm vào hoặc bị thay đổi nội dung. Nơi nhận hoàn toàn không phát hiện được. Thuật toán sau giải quyết tốt vấn đề này.

Chọn p là số nguyên tố lớn có chiều dài n byte sao cho việc giải bài toán trong miền Z_p^* là đủ khó. Có thể chọn n bằng 8, 16, 32, 64 hoặc 128 byte.

- Khoá công khai $K_{pu} = (p,\alpha,\beta)$, trong đó: p: một số nguyên tố lớn bất kì; α: số nguyên bất kì là phần tử sinh; $\beta = \alpha^a \bmod p$, với a nguyên bất kì thỏa $1 \le a \le p-2$.

- Khoá bí mật $K_{pr} = a$

1. Quá trình mã hóa

Chia dữ liệu cần mã hóa thành các khối X[i] có kích thước n byte. Khối cuối cùng có kích thước nhỏ hơn n byte sẽ không được mã hóa.

Bước 1: Tính $A[i] = (\beta^k \bmod p)$ XOR X[i]

k là số nguyên bất kỳ thỏa $1 \le p \le p-2$

Bước 2: Thực hiện dịch vòng trái LCS (Left Circular Shift) từng byte của A[i] theo vectơ dịch SV (Shift Vector) thu được B[i].

$$B[i][j]=A[i][j]<<<SV$$

SV là ma trận hàng gồm n phần tử, mỗi phần tử thoả điều kiện:

$$0 \le SV[i] \le 7.$$

Bước 3: Thu được văn bản sau mã hoá bằng cách:

$$C[i]=B[i] \text{ XOR } C[i-1]$$

trong đó C[i-1] là văn bản mã liền trước. Sử dụng vectơ khởi tạo IV (Initial Vector) cho lần đầu tiên.

2. Quá trình giải mã

Nhận được C[i], C[i-1] và biết trước khoá bí mật $K_{pr} = a$.

Bước 1: Tìm B[i] = C[i] XOR C[i-1], sử dụng vectơ khởi tạo IV (Initial Vector) cho lần đầu tiên.

Bước 2: Sử dụng vectơ liên hiệp dịch $\overline{SV}$ dịch vòng trái LCS từng byte B[i] để thu A[i]. Vectơ liên hiệp dịch là ma trận hàng được suy ra từ vectơ SV, với $\overline{SV}$ [i] = (8-SV[i]) XOR 8.

$$A[i][j]=B[i][j]<<<\overline{SV}.$$

Bước 3: Ta thu được X[i]= A[i] XOR (α^{ak} mod p).

Chứng minh thuật toán

Trước tiên ta cần chứng minh:

Nếu a XOR b = c thì c XOR b = a; (1)

a XOR b XOR c = a XOR c XOR b; (2)

xy mod z = [x.(y mod z)] mod z. (3)

Chứng minh (1): Xét bảng chân trị sau

a	b	a XOR b = c	c XOR b
0	0	0	0
0	1	1	0
1	0	1	1
1	1	0	1

So sánh cột 1 và 4, ta thấy (1) đúng với số 1 bit. Vì phép XOR thực hiện trên từng bit, nên (1) cũng đúng trong trường hợp a và b là số nhiều bit. Vậy (1) đã được chứng minh.

Chứng minh (2): Tương tự, xét bảng chân trị sau

a	b	c	a XOR b XOR c	a XOR c XOR b
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

So sánh cột 4 và 5, ta thấy (2) đã được chứng minh.

Chứng minh (3): xy mod z = [x(y mod z)] mod z

Đặt: $xy \bmod z = r_1$ với $0 \le r_1 < z$,

$y \bmod z = r_2$ với $0 \le r_2 < z$,

$[x(y \bmod z)] \bmod z = xr_2 \bmod z = r_3$ với $0 \le r_3 < z$.

Dễ thấy rằng:

$$xy = nz + r_1 \quad (1')$$

$y = mz+r_2$ với m, n, k là các số nguyên không âm (2')

$$xr_2 = kz + r_3. \quad (3')$$

Rút r_2 ở biểu thức (2') thay vào biểu thức (3') ta được:

$$xy = (k + mx)z + r_3.$$

So sánh với biểu thức (1') ta được

$$(k + mx)z + r_3 = nz + r_1.$$

Dễ dàng nhận thấy $r_1 = r_3$. Hay xy mod z = [x(y mod z)] mod z (đpcm).

Chứng minh thuật toán giải mã

Theo bước 3 của quá trình mã hoá, ta có:

$$C[i] = B[i] \text{ XOR } C[i-1],$$

dựa vào (1) suy ra B[i] = C[i] XOR C[i-1].

Do B[i] = A[i]<<<SV (bước 2 quá trình mã hoá), mặt khác, $\overline{SV}$ là vector liên hiệp dịch của SV, việc dịch vòng trái từng byte của khối B[i] theo $\overline{SV}$ [i] bit chính là trả về trị ban đầu A[i]. Nghĩa là:

$$A[i]=B[i]<<<\overline{SV}.$$

Ta chỉ cần chứng minh X[i]= A[i] XOR (α^{ak} mod p). Thay A[i] ở bước 1 của quá trình mã hoá vào ta có:

$$\begin{aligned} VP &= (\beta^k \bmod p) \text{ XOR } X[i] \text{ XOR } (\alpha^{ak} \bmod p) \\ &= [(\alpha^a \bmod p)^k \bmod p] \text{ XOR } X[i] \text{ XOR } (\alpha^{ak} \bmod p). \quad (*) \end{aligned}$$

Mặt khác, ta lại có:

$$\begin{aligned} \alpha^{ak} \bmod p &= \alpha^{a(k-1)}.\alpha^a \bmod p \\ &= [\alpha^{a(k-1)}.(\alpha^a \bmod p)] \bmod p \text{ (theo (3))} \\ &= (\alpha^a \bmod p).\alpha^{a(k-1)} \bmod p \\ &= (\alpha^a \bmod p).\alpha^{a(k-2)}.\alpha^a \bmod p \\ &= [(\alpha^a \bmod p).\alpha^{a(k-2)}.(\alpha^a \bmod p)] \bmod p \text{ (theo (3))} \\ &= (\alpha^a \bmod p)^2.\alpha^{a(k-2)} \bmod p \\ &... \\ &= (\alpha^a \bmod p)^k \bmod p. \end{aligned}$$

Thay vào (*), ta thu được

$$\begin{aligned} VP &= [(\alpha^a \bmod p)^k \bmod p] \text{ XOR } X[i] \text{ XOR } [(\alpha^a \bmod p)^k \bmod p] \\ &= [(\alpha^a \bmod p)^k \bmod p] \text{ XOR } [(\alpha^a \bmod p)^k \bmod p] \text{ XOR } X[i] \\ &= X[i] \\ &= VP \text{ (đpcm).} \end{aligned}$$

III. ĐÁNH GIÁ THUẬT TOÁN

Thuật toán đã được thông qua bởi Hội đồng Khoa học tại Đại học Bách khoa Tp. Hồ Chí Minh.

Thuật toán phát triển dựa trên độ khó của bài toán logarit trong Elgamal nên vẫn giữ được ưu điểm khó thám mã tương đương với RSA và Elgamal.

Để thám mã thành công thuật toán Elgamal độ dài 64 byte, với máy tính đơn có bộ vi xử lý PIV 2.6 GHz, cần thời gian 300000 giờ (khoảng 34 năm). Thế nhưng nếu sử dụng mạng gồm 100000 máy thì thời gian thám mã chỉ còn hơn 3 giờ (theo tài liệu tính toán của RSA Inc)[5].

Thuật toán Advanced Elgamal giải quyết tốt vấn đề bảo mật, nhờ sử dụng vector dịch SV theo ma trận hàng. Một số tính năng ưu việt nổi bật của thuật toán này như sau:

Độ bảo mật được tăng cường rất lớn so với các thuật toán khoá mã công khai hiện tại. Với cùng kích thước bài toán 64 byte như trên, vector dịch SV là ma trận 1x 64, mỗi phần tử của SV có giá trị từ 0 đến 7. Để thám mã thành công thuật toán Advanced Elgamal, ngoài việc vượt qua độ khó của bài toán logarit như trên, cần phải tìm được chính xác SV. Tập không gian SV là $8^{64} = 2^{192} = 10^{192lg2} \approx 10^{58}$ vectơ. Theo trung tâm ứng dụng siêu quốc gia Mỹ, (12/2003) [4], một hệ thống siêu mạnh với 1500 máy chủ có thể thực hiện được 20 nghìn tỉ (2.10^{13}) phép tính trên giây. Với hệ thống siêu mạnh này, theo ước tính của tác giả, thời gian để tìm ra chính xác SV bằng phương pháp vét cạn để thám mã là $10^{58}/ (2.10^{13})=5.10^{44}$ (giây) $\approx 1,6.10^{37}$ (năm). Rõ ràng độ bảo mật tăng lên vô cùng lớn.

Kích thước dữ liệu sau mã hoá không thay đổi. So với thuật toán Elgamal, ứng với mỗi dữ liệu x sẽ cho ra văn bản mã c gồm y_1 và y_2 [1]. Riêng thuật toán Advaned Elgamal, chỉ sinh ra văn bản mã C[i] có kích thước bằng với kích thước văn bản gốc X[i].

Chống thám mã theo xác suất xuất hiện. Các phương pháp mã hoá theo mô hình khoá đối xứng đều có cùng nhược điểm là tạo ra các khối văn bản mã giống nhau với cùng văn bản gốc. Nhờ vào phép XOR với văn bản mã liền trước, Advanced Elgamal sẽ tạo ra các văn bản mã khác nhau cho dù văn bản gốc đầu vào giống nhau. Điều này loại bỏ hoàn toàn phép thám mã theo xác suất.

Nhận ra sự thay đổi dữ liệu trên đường truyền. Một ai đó cố tình phá hoại hệ thống bảo mật bằng cách tạo ra các khối giống với khối văn bản mã, hay cố tình sửa đổi nội dung văn bản mã trên đường truyền. Theo thuật toán Elgamal và RSA, nơi nhận không thể phát hiện điều này. Kĩ thuật XOR các văn bản mã với nhau trong thuật toán Advanced Elgamal giúp giải quyết triệt để vấn đề này.

Tốc độ thực thi cao nhờ sử dụng các phép gần với ngôn ngữ máy (phép dịch vòng, phép XOR.

Hiệu quả trong thiết kế phần cứng: sử dụng chung khoảng 2/3 kiến trúc phần cứng cho quá trình mã hoá và giải mã.

IV. KẾT LUẬN

Tuy tốc độ mã hoá và giải mã được cải thiện rõ rệt, nhưng không ngoại lệ, thuật toán Advanced Elgamal cũng giống như các thuật toán thuộc hệ mã công khai, vẫn còn cồng kềnh so với thuật toán thuộc hệ bí mật [1]. Vì vậy, thuật toán rất thích hợp cho các ứng dụng có kích thước nhỏ, đòi hỏi độ bảo mật cao, không cần định danh trước đối tác sử dụng khoá [3]. Do đó, khả năng ứng dụng trong thương mại, giao dịch điện tử, thư tín điện tử là rất lớn.

TÀI LIỆU THAM KHẢO

1. A. J. Menezes, P. C. Van Oorsschot, S. A. Vanston - Handbook of Applied Cryptography, CRC, 2001.
2. Carl H.Meyer Stephen M.Matyas - Crytography: a new dimension in computer data security, John Wiley & Son, Prentice-Hall, 1999.

3. Kamlesh Kbaijaj – Debjani Nag, Ecommerce the cutting edge of business, McGraw-Hill, 2002.
4. http://csrc.nst.gov
5. http://www.rsa.com

SUMMARY

ADVANCED ELGAMAL ASYMMETRIC KEY MODEL IN THE FUTURE

Living in the information period, protection of secret data is very important because data are always traffic on the network. Therefore, safety and security in information system is not only needed in the field of military but also all of fields such as politic, economy, electronic commerce...What do we do to prevent illegal attacks from these thieves? Nowadays, thank to the standard cryptography theories, the problem is solved but not perfect, these algorithms are still a lot of disadvantages. The paper presents a new encryption algorithm namely Advanced Elgamal, improved from Elgamal asymmetric cryptography model, with many advantages. With these advantages, we hope the algorithm will become an encryption standard in the future.

Địa chỉ:

Trường Đại học Bách khoa Tp. Hồ Chí Minh.

Nhận bài ngày 12 tháng 4 năm 2004