BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
-------o0o-------

TÌM HIỂU LƯỢC ĐỒ CHỮ KÝ SỐ CHỐNG CHỐI BỎ

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ Thông tin

Sinh viên thực hiện: Hoàng Văn Hiệp
Giáo viên hướng dẫn: TS. Hồ Văn Canh
Mã số sinh viên: 1351010042

HẢI PHÒNG - 2013

MỤC LỤC

MỞ ĐẦU

Cùng với sự phát triển mạnh mẽ của công nghệ thông tin và sự giao lưu thông tin ngày càng trở nên phổ biến trên các mạng truyền thông, thì vấn đề đảm bảo an toàn thông tin đã trở thành một yêu cầu chung của mọi hoạt động kinh tế, xã hội và giao tiếp của con người.

Để thực hiện yêu cầu về bảo mật thông tin thì cách hay dùng nhất là mã hoá thông tin trước khi gửi đi. Vì vậy mật mã đã được nghiên cứu và sử dụng từ rất lâu trong lịch sử loài người. Tuy nhiên chỉ vài ba chục năm gần đây, nó mới được nghiên cứu công khai và tìm được các lĩnh vực ứng dụng trong đời sống công cộng cũng với sự phát triển của kỹ thuật tính toán và viễn thông hiện đại. Và từ đó, ngành khoa học này đã phát triển rất mạnh mẽ, đạt được nhiều kết quả lý thuyết sâu sắc và tạo cơ sở cho việc phát triển các giải pháp bảo mật và an toàn thông tin trong mọi lĩnh vực hoạt động của con người trong thời đại mà công nghệ thông tin được ứng dụng rộng rãi.

Các hệ thống mật mã được chia làm hai loại: mật mã bí mật và mật mã khoá công khai.

Trong các hệ thống mật bí mật, hai người muốn truyền tin bí mật cho nhau phải thoả thuận một khóa mật mã chung K, K vừa là khóa để lập mã vừa là khóa để giải mã. Và khóa K phải giữ kín chỉ có hai người biết.

Đề tài dựa trên cơ sở là các hệ thống mật mã khóa công khai. Ở đây, quan niệm về bí mật được gắn với độ phức tạp tính toán: ta xem một giải pháp là bí mật, nếu để biết được bí mật thì cần phải thực hiện một quá trình tính toán cực kỳ phức tạp, phức tạp đến mức mà ta coi là "không thể được" trên thực tế. Với quan niệm đó, người ta đã cải tiến và tạo mới nhiều giải pháp mật mã chỉ có thể thực hiện được bằng các công cụ tính toán hiện đại. Mật mã khóa công khai là cống hiến mới của lý thuyết mật mã hiện đại và có nhiều ứng dụng mà các hệ thống mật mã cổ điển không thể có được. Mật mã khóa công khai dựa trên ý tưởng: có thể tách riêng khóa làm hai phần tương ứng với hai quá trình lập mã và giải mã. Bí mật là dành cho người nhận tin, nên phần khóa giải mã phải được giữ bí mật cho người nhận tin, còn phần khóa dành cho việc lập mã để gửi đến một người A có thể công khai để mọi người có thể dùng để gửi thông tin mật cho A. Ý tưởng đó được thực hiện nhờ vào các hàm cửa sập một phía. Tính ưu việt của các hệ thống mật mà này thể hiện ở chỗ: trong một hệ truyền tin bảo mật không ai phải trao đổi khóa bí mật trước với ai

cả, mỗi người chỉ giữ cái bí mật riêng của mình mà vẫn truyền tin bảo mật với mọi người khác. Điều này rất quan trọng khi việc truyền tin được phát triển trên các mạng rộng với số người sử dụng gần như không hạn chế.

Mật mã khóa công khai không chỉ có tác dụng bảo mật, mà còn có nhiều ứng dụng khác, một trong các ứng dụng đó là xác thực, chữ ký số. Trong cách giao thiệp truyền thống, một chữ ký viết tay của người gửi dưới một văn bản không có tẩy, xoá là đủ xác nhận người gửi là ai, người gửi có trách nhiệm về văn bản và sự toàn vẹn của văn bản và cũng không thể chối bỏ trách nhiệm về chữ ký của mình. Nhưng trong truyền tin điện từ, văn bản chỉ là một dãy bít, nên để đảm bảo được hiệu lực như truyền thống thì người ta phải dùng chữ ký số. Chữ ký số cũng có nhiệm vụ giống chữ ký tay nghĩa là nó dùng để thực hiện các chức năng xác nhận của một người gửi trên một văn bản. Nó phải làm sao vừa mang dấu vết không chối cãi được của người gửi, vừa phải gắn bó với từng bit của văn bản mà nếu thay đổi dù chỉ một bit của văn bản thì chữ ký cũng không còn được chấp nhận. May thay, những yêu cầu này có thể thực hiện được bằng phương pháp mật mã khoá công khai. Nói chung các sơ đồ chữ ký số thì không cần đối thoại. Tuy nhiên, trong một số trường hợp để tăng thêm trách nhiệm trong việc xác nhận, người ta dùng các giao thức có tính chất đối thoại (hay chất vấn)qua một vài lần hỏi đáp để chính thức xác nhận tính đúng đắn (hoặc không đúng đắn) của chữ ký, tính toàn vẹn của văn bản, hay để buộc chấp nhận (không thể thoái thác, chối bỏ) chữ ký của mình. Trên cơ sở đó, trong đề tài tốt nghiệp tôi tìm hiểu về lược đồ chữ ký số chống chối bỏ và việc áp dụng nó trong quản lý hành chính trên mạng của trường Đại học Dân Lập Hải Phòng.

CHƯƠNG 1: MẬT MÃ KHÓA CÔNG KHAI

1.1. Lịch sử phát triển

1.1.1. Giới thiệu.

Theo các nhà nghiên cứu lịch sử mật mã thì Hoàng đế Caesar là người đầu tiên sử dụng mật mã trong quân sự.

Trong năm 1949, bài báo của Claude Shannon lần đầu tiên đã được công bố với tiêu đề "lý thuyết thông tin của các hệ thống mật" (Communication Theory of Secret Systems) trong The Bell Systems Technical Joumal. Bài báo này đã đặt nền móng khoa học cho mật mã, nó có ảnh hưởng lớn đến việc nghiên cứu khoa học của mật mã.

Ý tưởng về một hệ mật khoá công khai đã được Diffie và Hellman đưa ra vào 1976, còn việc hiện thực hoá đầu tiên hệ mật khoá công khai thì do Rivest, Shamir và Adleman đưa ra vào năm 1977. Họ đã tạo nên hệ mật RSA nổi tiếng. Kể từ đó đã có nhiều hệ mật được công bố và được phân tích, tấn công.

Mục tiêu cơ bản của mật mã là giúp hai người (Bob và Alice) thường xuyên liên lạc với nhau qua một kênh không an toàn mà sao cho đối phương (Oscar) không thể hiểu họ đang nói gì. Kênh này có thể là một đường dây điện thoại hoặc mạng máy tính. Thông tin Alice muốn gửi cho Bob, mà chúng ta gọi là "thông báo rõ", có thể là văn bản tiếng Anh, các dữ liệu bằng số, hoặc bất kỳ tài liệu nào có cấu trúc tuỳ ý.Alice mã thông báo bằng cách sử dụng một khoá đã được xác định trước và gửi kết quả trên kênh. Oscar thu trộm mã trên kênh song không thể hiểu được thông báo rõ là gì, nhưng với Bob người biết khoá mã của Alice có thể giải mã và thu được thông báo rõ.

1.1.2. Định nghĩa hệ mật

Một hệ mật là một bộ 5 thành phần (P, **C,** K, E, D) thoả mãn các điều kiện sau:

1. P: là 1 tập hữu hạn các bản rõ có thể.
2. C: là 1 tập hữu hạn các bản mã có thể.
3. K: (không gian khoá): tập hữu hạn các khóa có thể
4. Đối với mỗi $k \in K$ có 1 quy tắc mã $e_k \in E$ và một quy tắc giải $d_k \in D$ tương ứng, trong đó:

Mỗi e_k: P → C và d_k: $C \rightarrow$P là các hàm thoả mãn:

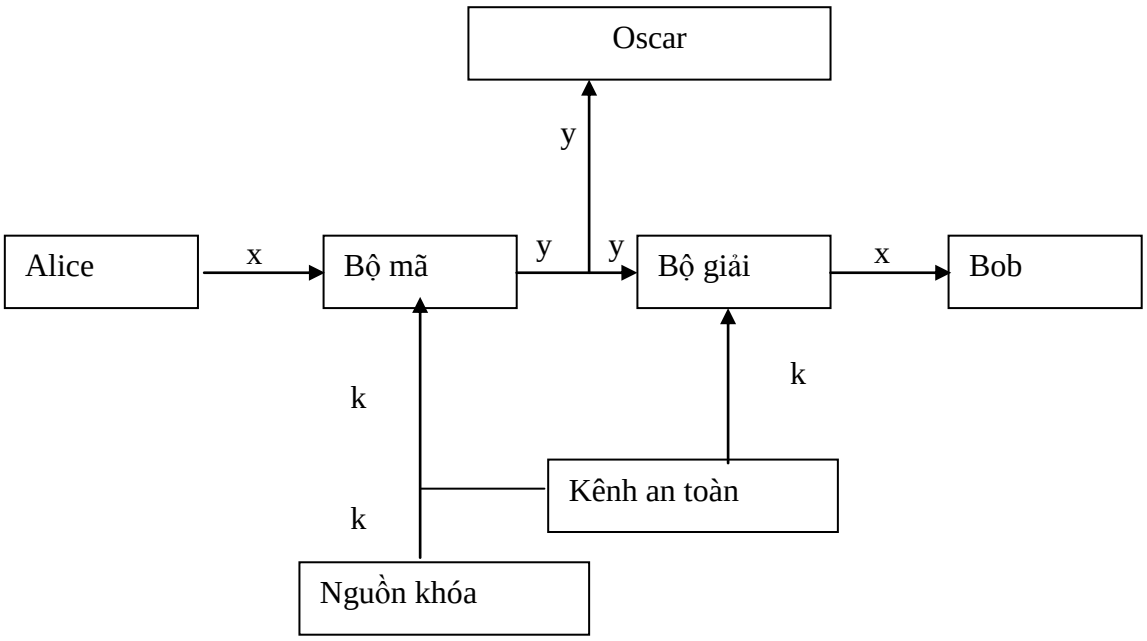
dk (e_k(x)) = x với mọi x∈ P

Tính chất quan trọng nhất là tính chất 4. Nội dung của nó là nếu một bản rõ x được mã hoá bằng e_k và bản mã được giải mã bằng d_k thì ta phải thu được bản rõ ban đầu x.

Alice và Bob sẽ áp dụng thủ tục sau để dùng hệ mật khoá riêng. Đầu tiên họ chọn một khoá ngẫu nhiên k ∈ K. Điều này được thực hiện khi họ ở cũng một chỗ và không bị theo dõi bởi Oscar, hoặc khi họ có một kênh an toàn trong trường hợp không cũng một chỗ. Sau đó Alice muốn gửi cho Bob một thông báo trên một kênh không an toàn, giả sử thông báo ấy là một chuỗi : x= x_1 x_2 ...x_n (với số nguyên n ≥1, ở đây mỗi bản rõ được ký hiệu là x_i∈ P, 1 ≤ i ≤ n). Alice mã mỗi x_i bằng quy tắc e_k với khoá xác định trước là k. Nghĩa là, Alice tính:

$y_i = e_k(x_i)$, 1 ≤ i ≤n, và kết quả là một chuỗi: y = y_1 y_2... y_n sẽ được gửi trên kênh. Khi Bob nhận được y_1 y_2 ... y_n , anh ta sẽ giải nó bằng hàm d_k và thu được thông bảo gốc x_1 x_2 ...x_n.

Ta có thể hình dung hệ thống liên lạc như sau:



Rõ ràng trong trường hợp này, hàm e_k phải là hàm đơn ánh (ánh xạ 1-1) nếu không việc giải mã sẽ không thực hiện được một cách tường minh. Ví dụ nếu: y = $e_K(x_1) = e_2(x_2)$, trong đó $x_1 \neq x_2$, thì Bob sẽ không biết giải mã thành x_1 hay x_2. Chú ý ràng nếu P=C thì mỗi hàm mã hoá sẽ là một phép hoán vị, tức là nếu tập các bản mã và tập các bản rõ đồng nhất thì mỗi một hàm mà sẽ là một hoán vị của các phần tử của tập này.

1.2. Một vài hệ mật đơn giản

1.2.1. Mã dịch chuyển

– Giả sử P = C = K = Z_{26} , với 0≤k ≤ 25, định nghĩa**:**

$e_K(x) = x + k \bmod 26$

và $d_K(x) = y - k \bmod 26$; (x, y € Z_{26})

– Ví dụ:

Cho k = 11 và bản rõ là: wewillmeetatmidnight

∗ Biến đổi bản rõ thành dày các số nguyên tương ứng và công với k theo modulo 26:

22 4 22 8 11 11 12 4 4 19 0 19 12 8 3 13 8 6 7 19

+

K =11

---

7 15 7 19 22 22 23 15 15 4 11 4 23 19 14 24 19 17 18 4

– Biến đổi dãy số nguyên này các ký tự ta được bản mã sau:

hphtwwxppelextoytrse

– Để giải mã thì ta chuyển bản mã thành dãy số nguyên ,sau đó trừ đi K theo modulo 26:

7 15 7 19 22 22 23 15 15 4 11 4 23 19 14 24 19 17 18 4

-

K =11

---

22 4 22 8 11 11 12 4 4 19 0 19 12 8 3 13 8 6 7 19

ta sẽ được bản rõ ban đầu là:

wewillmeetatmidnight

+ Nhận xét:

– Mã dịch chuyển là không an toàn vì nó có thể bị thám khóa bằng phương pháp vét cạn(vì chỉ có 26 khóa). Về trung bình, sẽ tính được thông báo sau 26/2 = 13 lần thử.

– Một hệ mật muốn sử dụng được trong thực tế thì nó phải thỏa mãn một số tính chất nhất định. Sau đây sẽ nêu ra hai trong số đó:

1. Mỗi hàm mã hóa e_k và mỗi hàm giải mã d_k phải có khả năng tính toán được một cách hiệu quả.
2. Đối phương dựa trên xâu bản mã phải không có khả năng xác định khóa k đã dùng hoặc không có khả năng xác định được xâu bản rõ x.

1.2.2. Mã thay thế.

+ Định nghĩa hệ mật:

Cho $P = C = Z_{26}$. K chứa mọi hoán vị có thể của 26 ký hiệu 0, 1, 2, ..., 25. Với mỗi hoán vị K, ta xác định phép thế Π và với mỗi phép thế Π đó ta định nghĩa:

$$e_K(x) = \Pi(x)$$

và $$d_K(y) = \Pi^{-1}(y)$$

Trong đó Π^{-1} là phép thế ngược của Π,

+ Ví dụ: K= defghijkolnmpqrswtuvyxbaze

$$\Pi = \begin{pmatrix} a\ b\ c\ d\ e\ i\ g\ h\ i\ j\ k\ 1\ m\ n\ o\ p\ q\ r\ s\ t\ u\ v\ w\ x\ y\ z \\ d\ e\ f\ g\ h\ i\ j\ k\ o\ l\ n\ m\ p\ q\ r\ s\ w\ t\ u\ v\ y\ x\ b\ a\ z\ c \end{pmatrix}$$

Hãy giải bản mã sau đây:

qdbqj vpybd mdifk yzhhq lfydt utsbo iacza

$$\Pi^{-1} = \begin{pmatrix} a\ b\ c\ d\ e\ f\ g\ h\ i\ j\ k\ l\ m\ n\ o\ p\ q\ r\ s\ t\ u\ v\ w\ x\ y\ z \\ x\ w\ z\ a\ b\ c\ d\ e\ f\ g\ h\ j\ l\ k\ i\ m\ n\ p\ o\ r\ t\ s\ q\ v\ u\ y \end{pmatrix}$$

Để giải bản mã trên, áp dụng Π^{-1} vào từng ký tự của bản mã, sau đó ghép lại ta sẽ được bản rõ tương ứng. Cụ thể như sau:

qdbqj => navvng => nắng

vpybd=> smuvva => mưa

mdifk => lafch => là

yzhhq => uyeen => chuyện

lfydt => jcuar => của

utsbo => trovvi => trời

– Nhận xét:

Mỗi khoá của mật mã thay thế là một phép hoán vị của 26 ký tự. Số hoán vị là 26!, lớn hơn 4 x 10^{26} . Đó là một số rất lớn. Bởi vậy, phép tìm khóa vét cạn không thể thực hiện được, thậm chí bằng máy tính.

1.2.3. Mã Affine

+ Trong mã Affine, ta xét các hàm mã có dạng:

e (x) = ax + b mod 26, (a, b $\in$ Z_{26}**).**

Các hàm này được gọi là hàm Affine (khi a = 1 => ta có mã dịch chuyển).

+ Để việc giải mã có thể thực hiện được, yêu cầu cần thiết là hàm Affine phải đơn ánh. Với bất kỳ y $\in$ Z_{26}, ta cần phương trình:

ax + b ≡ y (mod 26), phải có nghiệm duy nhất.

Đồng dư thức này tương đương với : ax = y - b (mod 26).

Vì y biến đổi trên Z_{26} nên (y - b) mod 26 cũng biến đối trên Z_{26} .Vì vậy, ta chỉ cần xét đồng dư thức: ax ≡ y (mod 26), y $\in$ Z_{26}

Ta biết rằng, phương trình này có 1 nghiệm duy nhất đối với mỗi y khi và chỉ khi (a, 26) = 1. Trước tiên, giả sử rằng (a, 26) = d > 1. Thế thì, ax≡0(mod 26) sẽ có ít nhất 2 nghiệm phân biệt trong Z_{26} là x = 0 và x =26/d. Khi đó, e(x) = ax + b mod 26 không phải hàm đơn ánh vì vậy nó không phải là một hàm mã hợp lệ.

+ Ta giả thiết (a, 26) = 1. Khi đó phương trình ax = y (mod 26) có đúng một nghiệm với mọi y $\in$ Z_{26} là x = a^{-1} y (mod 26).

+ Hệ mật:

Cho p = c = Z_{26} và giả sử:

K = { (a, b) $\in$ Z_{26} x Z_{26} : (a, 26) = 1}

với k = (a, b) ∈ K, ta định nghĩa**:**

e_k (x) = ax + b mod 26

và d_k (x) = a^{-1} (y - b) mod 26. x, y ∈ Z_{26}

+Ví dụ: k = (7, 3). Do (7, 26) = 1 nên có hàm mã là:

e_k (x) = 7x + 3

và hàm giải mã: d_k = 7^{-1} (y - 3) mod 26

= 15 (y - 3) = 15y - 19

Bản rõ: hot chuyền thành các số nguyên tương ứng là 7, 14, 19.

e_k(h) = 7x7 + 3 mod 26 = 0

e_k (o) = 7x14 + 3 mod 26 = 23

e_k (t) =7x19+3 mod *26=6*

Chuyền ba sổ 0, 23, 6 thành ký tự tương ứng ta được bản mã là AXG.

Giải ngược lại:

AXG => 0 23 6

d_k (A) = 15x0-19 mod 26 = 7 => H

d_k (X) = 15x23- 19 mod 26 = 14 => O

d_k (G) =15x6-19 mod 26 = 19 => T

1.2.4. Mã Vigenere.

+ Cho m là một số nguyên dương cố định nào đó. Định nghĩa P = C = K = $(Z_{26})^m$. Với khoá k = (k_1, k_2 ,..., k_m) ta xác định:

e_k (x_1, x_2,..., x_m) = (x_1 + k_1 , x_2 + k_2,

..., x_m + k_m);

và d_k (y_1, y_2,...,y_m) = (y_1 – k_1, y_2 - k_2, ,..., y_m - k_m);

trong đó tất cả các phép toán được thực hiện trong $Z_{26.}$

+ Ví dụ: m = 6. từ khoá k = CIPHER = (2, 8, 15, 7, 4, 17)

Thông báo x: thiscryprosystemisnotsecure Chuyển

từng khối 6 ký tự sang số:

thiscr	=> 19	7	8	18	2	17
	k 2	8	15	7	4	17
	21	15	23	25	6	8=>VPXZGI
yptosy	=> 24	15	19	14	18	24
	k 2	8	15	7	4	17
	0	23	8	21	22	15=>AXIVWP
stemis	=> 18	19	4	12	8	18
	k 2	8	15	7	4	17
	20	1	19	19	12	9=> UBTTMJ
notsec	=> 13	14	19	18	4	2
	k 2	8	15	7	4	17
	15	22	8	25	8	19 =>PWIZIT
ure	=> 20	17	4			
	k 2	8	15			
	22	25	19	=>WZT		

1.2.5. Mã hoán vị.

+ Ý tưởng của mật mã hoán vị là giữ các ký tự trên bản rõ không đổi nhưng thay đổi vị trí của chúng bằng cách sắp xếp lại các ký tự này.

+ Cho m là một số nguyên dương xác định nào đó. Cho P = **C** = $(Z_{26})^m$ và K gồm tất cả các hoán vị của {1, …, m} .Đối với một khoá K (tức là một hoán vị), ta xác định phép thế n tương ứng và xác định:

$e_K (x_1,..,x_m)=(x_{n(1)},x_{n(m)})$

và $d_K (y_1,\ldots,y_m) = (y_n{}^{-1}{}_{(1)},....,yn^{-1}{}_{(m)})$

Trong đó n^{-1} là hoán vị ngược của n.

+ Ví dụ: Cho m=6, K= 351642, khi đó ta có phép thế:

$$\Pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 5 & 1 & 6 & 4 & 2 \end{pmatrix}$$

Hãy giải bản mã:

wsstpa isorsw onugww difjad cdhajo laapan arjpih nfhsgo lmefax eklyxe iermen uojwwm suisaf wtnhma hlaafn jrautp wgwfno.

+ Tìm:

$$\Pi^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 6 & 1 & 5 & 2 & 4 \end{pmatrix}$$

+ Áp dụng Π^{-1} vào bản mã ta được bản rõ như sau:

wsstpa → sawpst	eklyxe → leexky
isorsw → owistr	iermen → rnieem
onugww → uwowng	uojwwm → jmuwow
difjad → fddaij	suisaf → ifsaus
cdhajo → hocjda	wtnhma → nawmth
laapan → anlaap	hlaafn → anhfla
arjpih → jhairp	jrautp → apjtru
nfhsgo → hongfs	wgwfno → wowngf
lmefax → exlamf	

Bản rõ thu lại được là: Sắp tới trường Đại học dân lập Hải Phòng sẽ làm lễ kỷ niệm mười sáu năm thành lập trường.

1.3. Mật mã khoá công khai.

Trong mô hình mật mã cổ điển mà cho tới nay vẫn còn đang được nghiên cứu, Alice (người gửi) và Bob (người nhận) chọn k một cách bí mật .Sau đó dùng k để mã hóa và giải mã. Các hệ mật thuộc loại này còn được gọi là các hệ mật khóa bí mật vì việc để lộ k sẽ làm cho hệ thống mất an toàn.

Nhược điểm của hệ mật này là nó yêu cầu phải có thông tin về khoá k giữa Alice và Bob qua một kênh an toàn trước khi gửi một bản mã bất kỳ. Trên thực tế, điều

này rất khó đảm bảo, chẳng hạn khi Alice và Bob ở rất xa nhau và liên lạc với nhau bằng thư tín điện tử (Email), thì việc xây dựng một kênh an toàn là rất khó khăn.

Ý tưởng xây dựng một hệ mật khóa công khai là tìm một hệ mật không có khả năng tính toán để xác định d_k nếu đã biết e_k. Nếu thực hiện được như vậy thì quy tắc e_k có thể được công khai bằng cách công bố nó. Ưu điểm của hệ mật khóa công khai là ở chỗ Alice (hoặc bất kỳ người nào đó) có thể gửi một thông báo đã được mã hóa(mà không cần phải thông tin trước về khóa bí mật) bằng quy tắc mã công khai e_k. Bob sẽ là người duy nhất có thể giải được bản mã này bằng quy tắc giải mã bí mật d_k của mình.

Ta cũng có thể hình dung hệ mật như sau: Alice đặt một vật vào một hộp kim loại và rồi khóa nó bằng một khóa bấm do Bob để lại. Chỉ có Bob là người duy nhất có thể mở được hộp vì chỉ có anh ta mới có chìa mở được khóa của mình.

1.3.1. Cơ sở của mật mã khóa công khai.

Hệ mật khóa công khai không bao giờ đảm bảo được độ mật tuyệt đối(an toàn vô điều kiện). Khi đối phương nghiên cứu bản mã y, thì anh ta có thể mã lần lượt các bản rõ có thể bằng quy tắc mã công khai e_k cho tới khi anh ta tìm được một bản rõ duy nhất x thỏa mãn $y = e_k(x)$. Bản rõ này chính là kết quả giải mã của y.

Các hàm một chiều đóng vai trò rất quan trọng trong mật mã. Trong mật mã khóa công khai người ta muốn rằng thuật toán mã hóa nhờ khóa công khai e_k của Bob là dễ tính toán song việc tính hàm ngược(tức là giải mã) phải rất khó đối với bất kỳ ai không phải là Bob.

- Hàm f(x) được gọi là hàm 1 chiều, nếu tính $y = f(x)$ là dễ nhưng việc tính ngược $x = f^{-1}(y)$ là rất khó. Có thể hiểu "dễ" là tính được trong thời gian đa thức(ví dụ đa thức bậc thấp) và "khó" theo nghĩa là không tính được trong thời gian đa thức. Cho đến nay, chưa có hàm nào được chứng minh là một chiều nhưng có một số hàm được tin là hàm một chiều.

Thí dụ: Hàm $f(x) = g^x \bmod p$ (p: số nguyên tố; g: phần tử nguyên thủy mod p) được tin là hàm một chiều. Hàm $f(x) = x^2 \bmod n$(n: là tích của 2 số nguyên tố lớn khác nhau n=p.q) cũng được người ta tin là hàm một chiều.

Tuy nhiên, để xây dựng một hệ mật khoá công khai thì việc tìm được hàm một chiều vẫn chưa đủ. Ta không muốn e_k là hàm một chiều đối với Bob vì anh ta phải có khả năng giải mã các bản mã nhận dược một cách có hiệu quả. Điều cần thiết là Bob cần phải có một cửa sập chứa thông tin bí mật cho phép dễ dàng tìm ngược e_k.

Như vậy, Bob có thể giải mã 1 cách hữu hiệu vì anh ta biết cái cửa sập nằm trong bí mật của K. Bởi vậy, hàm f(x) được gọi là hàm cửa sập một chiều, nếu f là hàm một chiều nhưng nếu biết cửa sập của nó thì việc tính $f^{-1}(y)$ là dễ.

Thí dụ: cho n = p.q là tích của hai số nguyên tố lớn, a là số nguyên, hàm $f(x) = x^a \pmod n$ là hàm cửa sập một chiều, nếu chi biết n và a thì tính $x=f^{-1}(y)$ là rất khó**,** nhưng nếu biết cửa sập, chẳng hạn hai thừa số của n thì sẽ tính được $f^{-1}(y)$ khá dễ.

1.3.2. Một số hệ mật điển hình

a. *Hệ mật RSA.*

+ Hệ mật RSA do Rives, Shamir và Adleman đề xuất năm 1977. Giá sử n là số nguyên, tích của hai số nguyên tố lớn khác nhau p và q, n = p.q. Ta chọn số a nguyên tố với $\phi(n) = (p - 1) (q -1)$ và tính $b \equiv a^{-1} \bmod \phi$; tức $ab \equiv \bmod \phi(n)$.

+ Hệ RSA được mô tả như sau:

Lấy n = p.q; p và q là hai số nguyên tố khác nhau: $P = C = Z_n$

$K = \{(n, p, q, b, a): ab \equiv 1 \bmod \phi(n)\}$

trong đó n, b: công khai; a, p, q: bí mật.

Với k = (n, p, q, a, b) ta định nghĩa:

$e_k(x) = x^b \bmod n, V\ x \in P$

$e_k(y) = y^a \bmod n,\ Vy \in C$

+ Ví dụ: Giá sử Bob chọn p = 101 và q = 113. Khi đó n = 11413 và $\phi(n)=100 \times 112 =11200$. Vì $11200 = 2^6.5^2.7$, nên có thế dùng một số nguyên b như một số mũ mã hoá khi và chỉ khi b không chia hết cho 2, 5 hoặc 7. Giá sử Bob chọn b = 3533 (vì$(\phi(n), b)=l$).

Khi đó: $a = b^{-1} \bmod <\phi(n)$

$= 3533^{-1} \bmod \phi(n) = 6597$

Bob sẽ công bố n = 11413 và b = 3533 trong thư mục công cộng. Bây giờ giá sử Alice muốn gửi bản rõ x = 9726 tới cho Bob. Cô ta sẽ tính:

$y = x^b \bmod n = 9\ 7\ 2\ 6^{\ 3533} \bmod 11413$

$= 5761$

Sau đó cô ta sẽ gửi bản mã 5761 trên kênh liên lạc. khi Bob nhận dược bản mã

5761. anh ta sử dụng khoá bí mật a để tính: $5761^{6597} \bmod 11413 = 9726$.

Độ mật của RSA được dựa trên giả thiết là hàm mã $e_k(x) = x^b \bmod n$ là hàm một chiều. Bởi vậy, thám mã sẽ không có khả năng về mặt tính toán để giải một bản mã. Cửa sập cho phép Bob giải mã được chính là thông tin về phép phân tích thừa số n = p.q. Các thuật toán phân tích hiện thời có khả năng phân tích các số khoảng 130 chữ số thập phân, vì vậy để đảm bảo an toàn nên chọn số p và số q lớn, chẳng hạn có chừng 100 chữ số, khi đó n sẽ có 200 chữ số. Trong khoảng 20 năm, người ta đã đưa ra nhiều sơ hở đề tấn công hệ mật RSA nhưng không có cách nào hiệu quả tuyệt đối mà chỉ đưa ra các sơ hở để người dùng hệ mật RSA tránh không mắc phải, do đó RSA vẫn là hệ mật an toàn.

b. *Hệ mật Elgamal*

+ Hệ mật Elgamal được xây dựng dựa trên bài toán logarithm rời rạc. Bài toán logarithm rời rạc trong Zp được xem là bài toán khó nếu số p được chọn cẩn thận .Cụ thể là không có thuật toán nào giải bài toán logarithm rời rạc trong thời gian đa thức. Số p được lựa chọn phải có ít nhất 150 chữ số thập phân và (p - 1) phải có ít nhất 1 thừa số nguyên tố lớn. Lợi thế của bài toán logarithm rời rạc là khó tìm được các logarithm rời rạc, song bài toán ngược lấy luỹ thừa lại có thể tính dễ dàng. Hay luỹ thừa theo modulo p là hàm 1 chiều với các số nguyên tố p thích hợp.

+ Mô tả bài toán logarithm rời rạc trong Zp.

Cho I = (p, α, β) trong đó p là số nguyên tố, $\alpha \in Zp$ là phần từ nguyên thuỷ, $\beta \in Z^*p$.

Bài toán đặt ra là: Hãy tìm 1 số nguyên duy nhất a, $0 \le a \le p - 2$ sao cho: $\alpha^a \equiv \beta \pmod p$; ta sẽ kí hiệu $a = \log_\alpha \beta$.

+ Định nghĩa hệ mật:

Cho p là số nguyên tố sao cho bài toán logarithm rời rạc trong Z_p là khó giải. Và $\alpha \in Z^*p$ là phần tử nguyên thuỷ. Giá sử $P = Z^*p$, $C = Z^*_p x Z^*_p$ .Ta định nghĩa:

$K = \{(p, \alpha, a, \beta): \beta \equiv \alpha^a \pmod p\}$

Các giá trị p, α,β được công khai, còn a là bí mật.

Với k = (p, α, a, β) và một số ngẫu nhiên bí mật $k' \in Z_{p-1}$, ta xác định:

$e_k(x, k) = (y_1, y_2)$; trong đó:

$y_1 = \alpha^{k'} \bmod p$

$y_2 = x\beta^{k'} \bmod p$.

và với $y_1 , y_2 \in Z^*p$, ta xác định: $d_k (y_1, y_2) = y_2 (y^a_1)^{-1} \bmod p$

+ Ví dụ:

Cho p = 2579, α = 2, a =765. Khi đó:

$\beta = 2^{765} \bmod 2579 = 949$

Bây giờ Alice muốn gửi thông báo x = 1299 tới Bob. Giả sử Alice chọn số ngẫu nhiên bí mật k' = 853. Sau đó cô tính:

$y_1 = 2^{853} \bmod 2579 = 435$

$y_2 = 1299.949^{853} \bmod 2579 = 2396$

Và cô gửi (435, 2396) trên kênh cho Bob. Khi Bob nhận được bản mã

y = (435, 2396), anh ta tính:

$x = 2396 \text{ x } (435^{765})^{-1} \bmod 2579 = 1299$

Và đây là bản rõ mà Alice đã mã hoá trước khi gửi cho Bob.

Như vậy, đối với hệ mật này thì độ dài của bản mã gấp đôi độ dài của bản rõ, thành phần bản mã phụ thuộc vào việc chọn ngẫu nhiên số k, việc chọn này làm tăng độ bí mật, nhưng lại không ảnh hưởng gì đến quá trình giải mã, do vậy ứng với một bản rõ có thể có nhiều bản mã khác nhau, phụ thuộc vào k khác nhau. Ta cũng thấy ràng y_1 không liên quan đến bản rõ vì toàn bộ thông tin liên quan đến bản rõ nằm trong y_2 .Nhưng y_1 lại cho biết thông tin cần thiết về k và việc giữ bí mật số k là rất quan trọng vì biết k' thì có thể tính được khoá bí mật a.

CHƯƠNG 2: CHỮ KÝ SỐ

2.1. Giới thiệu

Khái niệm về chữ ký đã khá quen thuộc trong đời sống hàng ngày .Chữ ký được sử dụng hàng ngày để viết thư, rút tiền ở nhà băng, ký hợp đồng, ... Chữ ký viết tay thông thường trên tài liệu dùng để xác nhận một người ký nó.

Lược đồ chữ ký số là một phương pháp ký một thông điệp lưu dưới dạng điện tử.

Ví dụ như thông điệp được ký có thể truyền trên mạng máy tính.

Giữa chữ ký tay và chữ ký số có một vài điều khác nhau cơ bản. Cụ thể như sau:

+ Với chữ ký thông thường, nó là một phần vật lý của tài liệu. Đối với chữ ký số thì không gắn theo kiểu vật lý vào tài liệu mà gắn theo kiểu logic với tài liệu.

+ Về việc kiểm tra chữ ký: Với chữ ký thông thường thì kiểm tra bằng cách so sánh nó với những chữ ký xác thực khác. Ví dụ, một người ký trên một tấm séc mua hàng, người bán phải so sánh chữ ký trên mảnh giấy với chữ ký nằm ở sau thẻ tín dụng để kiểm tra. Và ta có thể thấy đây không phải là phương pháp an toàn. Mặt khác, lược đồ chữ ký số có thể được kiểm tra bằng cách sử dụng thuật toán kiểm thử công khai. Vì vậy bất kỳ ai cũng có thể kiểm thử chữ ký số. Việc dùng một sơ đồ chữ ký số an toàn có thể ngăn chặn được khả năng giả mạo.

+ Còn một sự khác nhau cơ bản giữa chữ ký số và chữ ký thông thường là bản sao chép của chữ ký số đồng nhất với bản gốc. Còn của chữ ký thông thường có thể khác xa so với bản gốc .Điều này có nghĩa là phải cẩn thận ngăn chặn một thông điệp chữ ký số khỏi bị dùng lại. Ví dụ, nếu Bob ký bức điện số xác nhận Alice rút 100$ từ nhà băng, anh ta chỉ muốn Alice có thể làm điều đó một lần. Vì vậy, cần nghiên cứu những phương pháp để ngăn chặn việc chữ ký số bị dùng lại.

Một lược đồ chữ ký số bao gồm 2 phần: 1 thuật toán ký và 1 thuật toán kiểm thử. Bob có thể ký trên thông điệp x bằng một thuật toán ký an toàn. Kết quả của việc ký sig(x) có thể được kiểm thử bằng thuật toán công khai. Khi đưa 1 cặp (x,y), thuật toán kiểm thử trả lại câu trả lời là "True" hoặc "False" phụ thuộc vào việc chữ ký số là xác thực hay không xác thực.

2.2. Định nghĩa lược đồ chữ ký số:

Một lược đồ chữ ký số là một bộ năm phần tử (P, A, K, S, V) thoả mãn các điều kiện dưới đây:

+ K: không gian khoá, là tập hữu hạn các khoá có thể.
+ Với mỗi k thuộc K, có một thuật toán ký $sig_k \in S$ và thuật toán kiểm thử tương ứng $ver_k \in V$

Mỗi $sig_k : P \rightarrow A$ và $ver_k : P \times A \rightarrow \{True, False\}$ là những hàm sao cho mỗi bức điện $x \in P$ và mỗi chữ ký $y \in A$ thoả mãn

$ver(x,y) =$ True , nếu $y = sig(x)$

False, nếu $y \neq sig(x)$

Với mỗi khoá $k \in K$. hàm sig_k và ver_k là hàm có thời gian tính toán đa thức, ver_k sẽ là hàm công khai và sig_k là hàm bí mật. Điều đó có nghĩa là với x cho trước, chỉ có Bob mới tính được chữ ký y để ver(x,y) = True. Lược đồ chữ ký số không thể an toàn mà không có điều kiện vì Oscar có thể kiểm tra tất cả chữ ký số y có thể trên thông điệp x bằng thuật toán công khai ver cho đến khi tìm được chữ ký đúng. Vì thế nếu có đủ thời gian, Oscar sẽ luôn luôn có thể giả mạo được chữ ký của Bob. Cũng giống như hệ thống mật mã công khai, mục đích của chúng ta là tìm các lược đồ chữ ký an toàn về mặt tính toán.

Chú ý rằng, ai đó có thể giả mạo chữ ký của Bob trên 1 bức điện ngẫu nhiên x bằng cách tính $x = e_k(y)$ với y nào đó; khi đó $y = sig_k(x)$. Một biện pháp xung quanh vấn đề khó khăn này là yêu cầu các thông điệp chứa đủ phần dư thừa để chữ ký giả mạo kiểu này không tương ứng với bức điện đầy đủ x trừ một xác suất rất nhỏ.

2.3. Một số lược đồ chữ ký số

Trong phần này ta sẽ nghiên cứu một số lược đồ chữ ký số tốt, đứng vững trước các kiểu tấn công trong thời gian qua.

2.3.1. Lược đồ chữ ký RSA

Lược đồ chữ ký số RSA được định nghĩa như sau:

Cho n = p.q, p và q là các số nguyên tố lớn khác nhau. Cho $P = A = Z_n$ và định nghĩa:

$K = \{ (n, p, q, a, b): ab \equiv l(\mod \phi(n)) \}$

Các giá trị n, b là công khai; a, p, q là bí mật.

Với k = (n, p, q, a, b), ta định nghĩa:

$$sig_k(x) = x^a \bmod n$$

và $ver_k(x,y) = True <=> x \equiv y^b \pmod n ; x,y \in Z_n$

Kết hợp chữ ký với mã hoá sẽ làm cho độ an toàn của chữ ký tăng thêm.

Giả sử rằng, Alice sẽ tính chữ ký của cô ta là y = sigAlice(x), và sau đó mã hóa cả x và y bằng cách sử dụng mật mã công khai eBob của Bob, khi đó cô ta nhận được z = eBob(x,y). Bản mã z sẽ được truyền tới Bob. Khi Bob nhận được z, việc trước tiên là anh ta giải mã bằng hàm dBob để nhận được (x,y). Sau đó anh ta sử dụng hàm kiểm thử công khai của Alice để kiểm tra xem liệu verAlice(x,y) = True hay không?

Nếu Alice mã hoá x trước rồi sau đó mới ký lên bản mã đã được mã hóa thì sao? Khi đó cô ta tính:

$$y = sig_{Alice}(e_{Bob}(x))$$

Alice sẽ truyền cặp (z,y) cho Bob. Bob sẽ giải mã z, nhận được x và kiểm tra chữ ký y trên z bằng cách sử dụng ver_{Alice}. Một vấn đề tiềm ẩn trong biện pháp này là nếu Oscar có được cặp (z,y) kiểu này, anh ta có thể thay thế chữ ký y của Alice bằng chữ ký của anh ta:

$$y' = sig_{Oscar}(e_{Bob}(x))$$

Chú ý rằng Oscar có thể ký bản mã $e_{Bob}(x)$ ngay cả khi anh ta không biết bản rõ x.

Khi đó, nếu Oscar truyền (z,y) tới Bob, chữ ký của Oscar sẽ được kiểm thử vì Bob sử dụng ver_{Oscar} và Bob có thể suy ra rằng bản rõ x xuất phát từ Oscar. Điều này cũng làm cho người sử dụng hiểu rằng nên ký trước rồi sau đó mới tiến hành mã hoá.

Ví dụ:

Giá sử Bob dùng lược đồ chữ ký số RSA với n = 143 (p = 1 1, q = 1 3);

$\phi(n)$ =120. Khoá công khai của Bob là b=7 => $a = 7^{-1} \bmod 120 = 103$.

+ Bob có thông báo là x = 110 khi đó Bob sẽ ký trên thông báo x:

$$y = x^a \bmod n = 110^{103} \bmod 143 = 33$$

+ Bod gửi y=33 và x=l10 cho Alice. Alice sẽ kiểm thử bằng cách sử dụng khoá công khai của Bob như sau:

$y^b \bmod n = 33^7 \bmod 143 = 110 = X$

Và Alice chấp nhận y=33 là chữ ký hợp lệ.

2.3.2. Lược đồ chữ ký Elgamal

Lược đồ Elgamal đã được Viện tiêu chuẩn và Công nghệ quốc gia Mỹ sửa đổi thành chuẩn chữ ký số .Lược đồ Elgamal không tất định cũng giống như hệ thống mã khoá công khai Elgamal. Điều này có nghĩa là, có nhiều chữ ký hợp lệ cho một thông điệp bất kỳ. Thuật toán kiểm thử phải có khả năng chấp nhận bất kỳ chữ ký hợp lệ nào khi xác minh.

Lược đồ Elgamal được định nghĩa như sau:

+ Cho p là số nguyên tố sao cho bài toán log rời rạc trên Z_p là khó và giả sử $a \in Z_p$ là phân tử nguyên thuỷ. Cho $P = Z_p, A = Z^*_p x Z_{p-1}$

và định nghĩa: $K = \{ (p, \alpha, a, \beta): \beta \equiv \alpha^a \pmod p \}$

giá trị p, α, β là công khai; a là bí mật.

+ Với k= (p, α, a, β) và một số ngẫu nhiên bí mật $k' \in Z^*p_{-1}$,định nghĩa:

$sig_k(x,k') = (\gamma, \delta)$, trong đó: $\gamma = \alpha^k \bmod p$

$\delta = (x - a\gamma)k'^{-1} \bmod (p - 1)$

+ Với $x, y \in Z^*p$ và $\delta \in Z^*p_{-1}$, là định nghĩa:

$ver(x,\gamma, \delta)$ = True khi và chỉ khi $\beta^y \gamma^\delta \equiv \alpha^x \pmod p$

Chứng minh:

+ Nếu chữ ký được thiết lập đúng thi kiểm tra sẽ thành công vì:

$\beta^\gamma \gamma^\delta = a^{a\gamma} a^{k\delta} \pmod p$

$\equiv \alpha^x \pmod p$ (vì $a\gamma + k\delta = x \pmod{p - 1}$)

+ Bob tính chữ ký bằng cách dùng cả giá trị bí mật a (là một phần của khoá) lẫn số ngẫu nhiên bí mật k' (dùng để ký trên x). Việc kiểm thử có thể thực hiện duy nhất bằng thông tin công khai.

Vídụ: Giá sử p = 467; α = 2; a = 127

→$\beta = \alpha^a \bmod p = 2^{127} \bmod 467 = 132$

→Giá sử Bob có thông điệp x = 100, Bob chọn ngẫu nhiên k'=213 vì (213, 466) =1 và $213^{-1} \bmod 466 = 431$

-» Bob ký trên x như sau:

$$y = \alpha^k \bmod p = 2^{213} \bmod 467 = 29$$

và $\delta = (x - a\gamma)k'^{-1} \bmod (p - 1) = (100 - 127).\ 431 \bmod 4 = 51$

Bất kỳ người nào đó cũng có thể kiểm tra chữ ký này bằng cách:

$$132^{29}\ 29^{51} \equiv 189 \pmod{467}$$

$$2^{100} \equiv 189 \pmod{467}$$

Do đó chữ ký là hợp lệ

Xét độ an toàn của lược đồ chữ ký Elgamal. Giá sử Oscar thử giá mạo chữ ký trên bức diện x cho trước mà không biết a. Nếu Oscar chọn giá trị γ và thử tìm δ tương ứng, anh ta phải tính log rời rạc của $\log_\gamma \alpha^x \beta^{-\gamma}$. Mặt khác, nếu anh ta chọn δ trước và sau đó thử tìm γ thì anh ta phải giải phương trình $\beta^\gamma\ \gamma^\delta \equiv \alpha^x \pmod p$, trong đó γ là ẩn số. Bài toán này chưa có lời giải, tuy nhiên dường như nó liên quan đến bài toán đã nghiên cứu. Vẫn còn có khả năng là tìm δ và γ đồng thời để (δ, γ) là chữ ký. Hiện thời không ai tìm được cách giải song cũng không ai khẳng định được là nó không có lời giải.

Nếu Oscar chọn γ và δ và sau đó thử giải để tìm x**,** anh ta sẽ phải tính bài toán logarit rời rạc, tức phải tính $\log_\alpha \beta^\gamma \gamma^\delta$. Vì thế Oscar không thể ký một thông điệp ngẫu nhiên bằng cách này .Tuy nhiên có một cách để Oscar ký lên thông điệp ngẫu nhiên bằng việc chọn γ, δ và x đồng thời:

Giả thiết i và j là các số nguvên $0 \le i \le p - 2$; $0 \le j \le p - 2$ và $(j, p - 1) = 1$

Khi đó thực hiện các phép tính:

$$\gamma = \alpha^i \beta^j \bmod p$$

$$\delta = - yj^{-1} \bmod (p - 1)$$

$$x = \gamma\ i\ j^{-1} \bmod (p - 1) = i\delta \bmod (p-1)$$

trong đó j^{-1} được tính theo module (p - 1). Ta thấy rằng (γ, δ) là chữ ký hợp lệ của x. Điều này được chứng minh qua việc kiểm tra:

$$\beta^y y^{£} \equiv \beta^{-\gamma}(\alpha^i\beta^j)^{-\gamma 1 j-1} \bmod p$$

$$= \beta^y \alpha^{-ij-1y}\ \beta^{-y} \pmod p$$

$$= \alpha^{-\gamma ij-1} \pmod p$$

$=\alpha^x \pmod p$

Ví dụ: p = 467; $\alpha = 2$; $\beta = 123$. Giả sử Oscar chọn i = 99; j = 179, khi đó $j^{-1} \bmod (p - 1) = 151$. Anh ta tính:

$\gamma = 2^{99}\ 132^{179} \bmod 467 = 177$

$\delta = -177 \times 151 \bmod 466 = 41$

$x = 99 \times 41 \bmod 466 = 331$

Và (117, 41) là chữ ký trên x= 331

Kiểm thử:

$132^{117}\ 117^{41} \equiv 303 \pmod{467}$

và $2^{331} \equiv 303 \pmod{467}$

Do đó chữ ký là hợp lệ

Oscar có thể giả mạo chữ ký theo kiểu khác là bắt đầu từ thông điệp x đã được Bob ký. Giả sử (γ, δ) là chữ ký hợp lệ trên x. Khi đó Oscar có khả năng ký lên nhiều thông điệp khác nhau. Giả sử i, j, h là các số nguyên; $0 \le h; i; j \le p - 2$ và $(h\gamma - j\delta, p-1) = 1$. Thực hiện các phép tính:

$\lambda = \gamma^h \alpha^i \beta^j \bmod p$

$\mu = \delta\lambda(h\gamma - j\beta) \bmod (p - 1)$

$x = \lambda(hx + i\delta)(h\gamma - j\delta)^{-1} \bmod (p - 1)$

trong đó $(h\gamma - j\delta)^{-1}$ được tính theo module (p - 1)

Kiểm thử:

$\beta^{\lambda}\lambda^{\mu} \equiv \alpha^x \bmod (p) \Rightarrow (\lambda, \mu)$ là chữ ký hợp lệ của x.

Cả hai phương pháp trên đều tạo các chữ ký giả mạo hợp lệ song không xuất hiện khả năng đối phương giả mạo chữ ký trên thông điệp có lựa chọn của chính họ mà không phải giải bài toán logarit rời rạc. Vì thế không có gì nguy hiểm về độ an toàn của lược đồ Elgamal.

+ Một vài sơ xuất để phá lược đồ Elgamal:

1) Lộ k' $\Rightarrow$ giải phương trình đồng dư tuyến tính: $\delta a = (x - k'\gamma) \bmod (p-1)$ để tìm a. Nghiệm nào thoả mãn $\beta = \alpha^a \bmod p$ thì đó là giá trị đúng. Khi biết a thì Oscar dễ dàng giả mạo chữ ký.

2) Dùng một k' để ký hai thông điệp khác nhau. Giá sử (γ, δ1) là chữ ký trên xl; (γ, δ2) là chữ ký trên x2

Khi đó:

$\delta_1=(x_1-a\gamma)k'^{-1} \bmod (p-1)$

$\delta 2=(x_2-a\gamma)k'^{-1} \bmod (p-1)$

Từ đây nhận được phương trình tìm k chưa biết:

$k'(\delta_1-\delta_2)\equiv(x_1-x_2) \bmod (p-l)$

Phương trình này có nghiệm vì thực sự đã có k ,δ_1, δ_2, x_1, x_2 thoả mãn. Giải phương trình đồng dư tuyến tính với k’ là ẩn số ta có thể tìm được d=(δ_1 - δ_2, p-1) nghiệm. Kiểm tra điều kiện $\gamma\equiv\alpha^{k'} \pmod p$ để tìm 1 giá trị đúng duy nhất của k’. Sau khi có k’ ta trở về trường hợp trước để tìm a.

CHƯƠNG 3: HÀM HASH

3.1. Chữ ký và hàm Hash

3.1.1. Đặt vấn đề

Ta thấy rằng các cơ sở đồ chữ ký chỉ cho phép ký các thông báo nhỏ, thương có đọ dài xấp xỉ bằng bản thân chữ ký. Trên thực tế ta cần ký các thông báo có đọ dài lớn hơn nhiều chẳng hạn như có thể là một vài Megabyte. Vậy làm thế nào có thể có được chữ ký ngắn trên một thông báo có độ dài tùy ý. Vì chữ ký điện tử phải "ký " trên từng bit của thông báo, nên muốn có chữ ký độ dài cố định trên thông báo có độ dài tùy ý thì phải rút ngắn thông báo. Trên thực tế, việc rút ngắn văn bản là không thể được.

Vấn đề đặt ra là: chúng ta có thể có thể cắt thông báo x ra thành từng đoạn ký được có độ dài bằng nhau và cố định sau đó thực hiện ký trên từng đoạn và gửi từng đoạn đó đi. Nhưng giải pháp này gặp nhiều khó khăn vì:

- Vì phải sử lý quá nhiều đoạn.
- Có thể người nhận sẽ không sắp xếp lại được thông báo theo đúng trật tự ban đầu.
- Có thể mất từng đoạn khi truyền.

Giải pháp thứ 2 là dùng hàm HASH:

Cho thông báo x có độ dài tùy ý và hàm HASH biến đổi thành thông báo thu gọn z=h(x) có độ dài cố định 128 bit hoặc 160 bit. Sau đó ta thực hiện ký trên trên thông báo thu gọn: Chữ ký y=sigk(z).

Ta sẽ gửi cặp (x, y) nếu không cần bí mật. Còn nếu cần giữ bí mật thì ta sẽ mã hóa thông báo x thành x' và gửi (x', y).

Như vậy hàm HASH đã làm nhiệm vụ biến một thông báo x có độ dài bất kỳ thành một thông báo thu gọn có độ dài cố định và từ đó thực hiện ký trên thông báo thu gọn một cách dễ dàng và vẫn đảm bảo tính xác thực thông tin.

3.1.2. Định nghĩa hàm HASH

Hàm HASH là một hàm tính toán có hiệu quả khi ánh xạ các dòng nhị phân có độ dài tùy ý thành các dòng nhị phân có độ dài cố định nào đó.

+ Hàm HASH yếu:

Một hàm HASH được gọi là yếu nếu cho một thông báo x thì về mặt tính toán

không tìm ra được thông báo $x' \neq x$ và $h(x') = h(x)$.

+ Hàm HASH mạnh:

Một hàm HASH được gọi là mạnh nếu về mặt tính toán không tìm ra được 2 thông báo x và x' sao cho $x' \neq x$ và $h(x') = h(x)$.

+ Hàm HASH có tính chất một chiều:

Hàm HASH có tính chất một chiều nếu cho trước một thông báo rút gọn z thì về mặt tính toán không tìm ra được thông báo x sao cho h(x) = z.

Như vậy hàm HASH có tính chất như sau:

- Có thể tác động lên một khối dữ liệu có kích thước tùy ý và sản sinh một đầu ra có kích thước cố định.
- Có thể dễ dàng sản sinh ra bản tóm tắt đối với một thông báo bất kỳ.
- Tương đối dễ tính toán đối với một x bất kỳ khi thực hiện trên cả phần cứng cũng như phần mềm.
- Từ mã cho trước không thể sản sinh ra thông báo tương ứng với nó qua hàm HASH.
- Với một x đã cho không thể tính toán y khác x sao cho h(x)= h(y).
- Không tìm ra một cặp (x, y) sao cho h(y)= h(x).

Giải thích các tính chất trên:

Ba tính chất đầu là yêu cầu cần thiết cho một ứng dụng thực tế của hàm HASH trong việc xác thực thông báo.

Tính chất thứ tư là tính chất của hàm một chiều: Nó rất dễ dàng sản sinh ra một mã đối với một thông báo cho trước nhưng không thể sản sinh một thông báo từ mã cho trước. Tính chất này rất quan trọng.

Tính chất thứ năm bảo đảm rằng khi đã cho một thông báo thì không thể tính toán để tìm ra thông báo khác có cùng bản tóm tắt và do đó làm cho chữ ký số trở nên tin cậy giống như toàn bộ chữ ký lên toàn bộ thông báo.

Tính chất thứ sáu chống lại kẻ giả mạo tạo ra hai bản thông báo có nội dung khác nhau, sau đó thu nhận chữ ký hợp pháp cho một thông báo dễ được chấp nhận , rồi lấy nó giả mạo lấy nó làm chữ ký cho thông báo thứ hai.

3.2. Một số hàm HASH sử dụng trong chữ ký số

3.2.1. Các hàm HASH đơn giản

Tất cả các hàm HASH thực hiện sử dụng nguyên tắc chung dưới đây. Đầu vào (thông điệp, file...) được biểu diễn như m khối, mỗi khối n bit. Đầu vào sử lý mỗi khối tại một thời điểm trong một kiểu lặp đi lặp lại để xây dựng một hàm HASH n bit.

Một hàm HASH đơn giản nhất trong các hàm là thực hiện phép toán XOR từng bit một của mỗi khối. Nó được biểu diễn như sau:

$Ci= bi_1 \oplus bi_2 \oplus \ldots \oplus bi_m$

Trong đó:

Ci là bit thứ i của mã HASH, $1 \leq i \leq n$

m= số các khối đầu vào

bij = bit thứ i trong khối thứ j

$\oplus$ = phép cộng modulo

Hàm HASH đơn giản sử dụng phép XOR các bit

	bit1	bit2	.....	bit n
Khối 1	b_{11}	b_{21}	.....	bn_1
	b_{12}	b_{21}	.....	bn_2
.....				
.....				
.....				
Khối m	b_{1m}	b_{2m}	.....	bn_m
Mã HASH	C_1	C_2	.....	C_n

Khi thực hiện phép cộng modulo, nó sản sinh ra một bit parity đơn giản cho mỗi vị trí của từng bit và nó được biết như một sự kiểm tra ngẫu nhiên dọc theo chiều dài. Nó có tác động một cách ngẫu nhiên đến dữ liệu như một sự kiểm tra tổng thể tính toàn vẹn của dữ liệu.

3.2.2. Hàm HASH MD5:

MD5 là một phiên bản mạnh hơn MD4, có một sự khác biệt quan trọng đó là MD5 sử dụng 4 vòng với 4 hàm cơ bản chứ không phải là 3 vòng với 3 hàm cơ bản như MD4.

a. Giới thiệu thuật toán:

Thuật toán thực hiện đầu vào là một thông điệp có độ dài bất kỳ và xây dựng một đầu ra là một thông điệp 128 bit rút gọn. Đầu vào sử lý các khối bit có độ dài 512 bit.

b. Quá trình xây dựng thông điệp rút gọn thuật toán thực hiện một số bước sau:

Bước 1: Mở rộng và gắn thêm các bit

Thông diệp được chèn thêm sao cho độ dài là một chuỗi các bit đồng dư với 448 modulo 512 (độ dài ≡448 mod 512). Các bit được thêm vào không làm thay đổi nội dung của thông điệp đó là các số 0.

Bước 2 : Mở rộng độ dài

Một thông điệp nguyên thủy là một chuỗi các bit có đại diện 64 bit(trước khi gắn vào) thì sẽ được mở rộng sao cho nó phải phù hợp với kết quả của bước 1. Nếu độ dài nguyên thủy lớn hơn 2^{64} bit thấp sẽ được sử dụng.

Kết quả đầu tiên của hai bước đầu có hiệu quả cao ta được một thông điệp là một số nguyên nhân với 512 bit độ dài. Thông điệp mở rộng sẽ đưa ra một chuỗi các khối 512 bit Y_0, Y_1,..., YL_{-1} , vì vậy độ dài của thông điệp sẽ là L x 512 bit. Tương tự, kết quả là tích của 16 từ có độ dài 32 bit. Trong đó M[0...N-1] biểu diễn các từ của kết quả thông điệp, với N là một số nguyên và N= L*16.

Bước 3: Giá trị ban đầu(khởi nguồn) của bộ đệm MD

Bộ đệm 128 bit được sử dụng để lưu trữ trực tiếp và kết quả cuối cùng của hàm HASH. Bộ đệm được thể hiện là 4 thanh ghi 32 bit(A, B, C, D). Các thanh ghi này được nhận giá trị ban đầu là các giá trị thập lục phân dưới đây:

A=0x01234567

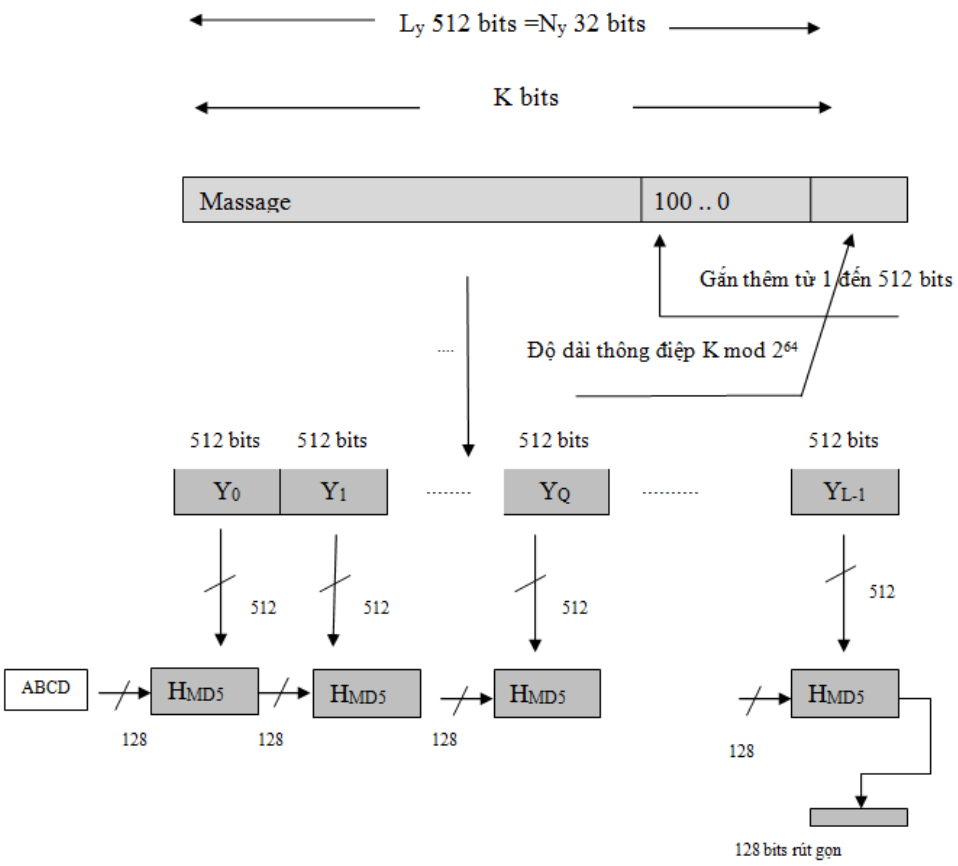
B=0x89abcdef

C=0xfedcba98

D=0x76543210

Bước 4: Xử lý thông điệp trong các khối 512 bit(16 từ).

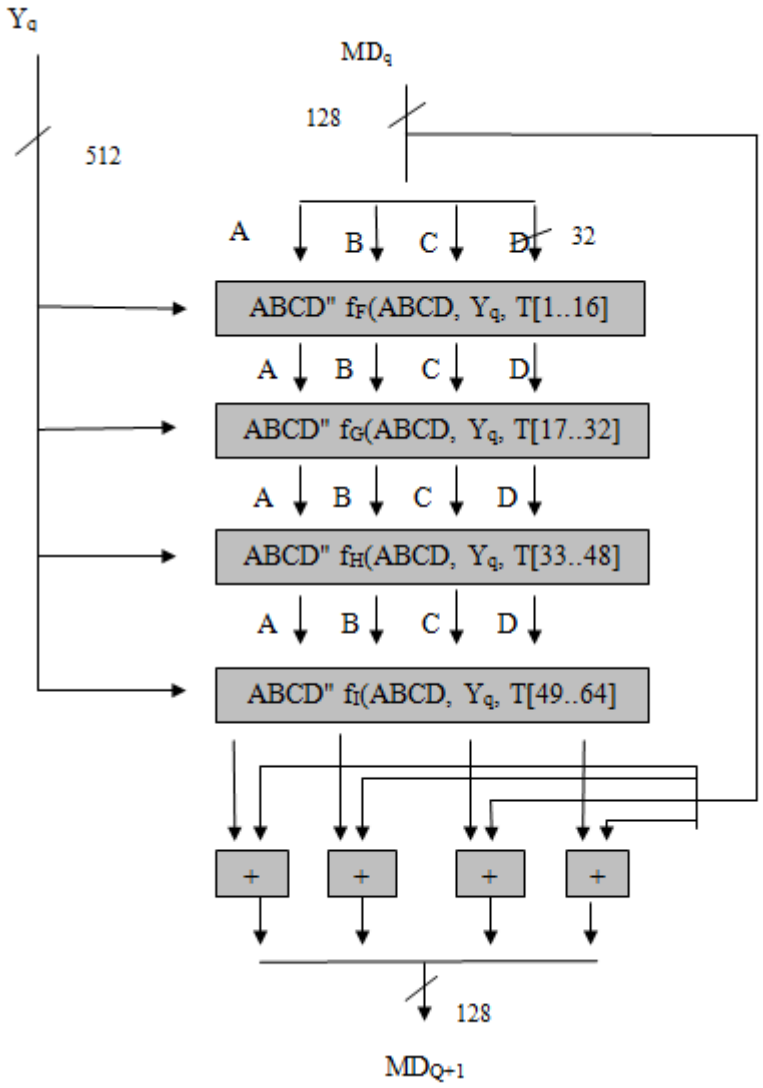
Trái tim của thuật toán là module nó bao gồm 4 vòng" 4 rounds" của quá trình xử lý. Mỗi module được đặt một nhãn H_{MD5}, chúng được thể hiện theo mô hình sau:



Mô hình tổng quát thông điệp rút gọn sử dụng MD5.

Bốn vòng có cấu trúc như nhau, nhưng mỗi vòng sử dụng hàm logic nguyên thủy khác nhau, như các hàm F, G, H và I được mô tả chi tiết dưới đây. Trong mô hình dưới đây 4 vòng là các nhãn f_F, f_G, f_H, f_I, các nhãn này chỉ ra rằng mỗi vòng có chức năng cấu trúc tổng quát là như nhau, nhưng f phụ thuộc vào các hàm nguyên thủy(F, G, H, I).

Mô hình biểu diễn công việc sử lý các khối đơn 512 bit(H_{MD5}):



Chú ý mỗi vòng thực hiện như đầu vào một khối hiện tại 512 bit được sử lý (Yq) và bộ đệm 128 bit co giá trị ABCD và cập nhật nội dung của bộ đệm. Mỗi vòng tạo ra sử dụng 1/4 của bộ table 64 phần tử T[1...64] được xây dựng từ hàm sine. Phần tử thứ i của T là T[i] có giá trị bằng một phần của số nguyên 2^{32}x abs(sin(i)), trong đó i tính bằng radians. Khi abs(sin(i)) là một số nằm giữa 0 và 1, thì mỗi phần tử của T là một số nguyên có thể biểu diễn trong 32 bit. Trong bảng xây dựng một cách ngẫu nhiên một cặp các mẫu 32 bit, cái loại ra một số tính hợp thức trong dữ liệu vào.

Bảng dưới đây là giá trị của T.

T[1]=D76AA478	T[17]=F61E5265	T[33]=FFFA3942	T[49]=F4292244
T[2]=E8C7B756	T[18]=C040B340	T[34]=8771F681	T[50]=432AFF97
T[3]=242070DB	T[19]=265E5A51	T[35]=69D96122	T[51]=AB9423A7
T[4]=C1BDCEEE	T[20]=E9B6C7AA	T[36]=FDE5380C	T[52]=FC93A039
T[5]=F57COFAF	T[21]=D62F105D	T[37]=A4BEEA44	T[53]=655B59C3
T[6]=4787C62A	T[22]=02441453	T[38]=4ADFCFA9	T[54]=8F0CCC92
T[7]=A8304613	T[23]=D8A1E681	T[39]=F6BB4B60	T[55]=FFEFF47D
T[8]=FD469501	T[24]=E7D3FBC8	T[40]=BEBFBC70	T[56]=85845DD1
T[9]=698098D8	T[25]=21E1CDE6	T[41]=28987EC6	T[57]=6FA87E4F
T[10]=8B44F7AF	T[26]=C33707D6	T[42]=EAA127FA	T[58]=FE2CE6E0
T[11]=FFFF5BB1	T[27]=F4D50D87	T[43]=DAEF3085	T[59]=A3014314
T[12]=895CD7EB	T[28]=455A14ED	T[44]=04881D05	T[60]=AE0811A1
T[13]=6B901122	T[29]=A9E3E905	T[45]=D4D9D039	T[61]=F7537E82
T[14]=FD987193	T[30]=FCEFA3F8	T[46]=6EDB99E5	T[62]=BD3AF235
T[15]=A679438E	T[31]=676F02D9	T[47]=1FA27CF8	T[63]=2AD7D2BB
T[16]=49B40821	T[32]=8D2A4C8A	T[48]=C4AC5665	T[64]=EB86D391
.....			

Bước 5: Đầu ra

Sau khi tất cả L khối 512 bit đã được sử lý, thì đầu ra từ tầng thứ L là thông điệp 128 bit rút gọn.

Chúng ta nghiên cứu chi tiết hơn việc sử lý mức logic trong 4 vòng của mỗi khối 512 bit. Mỗi vòng bao gồm một chuỗi 16 bước xử lý trên bộ đệm ABCD. Mỗi bước là một nguyên mẫu thể hiện dưới đây:

$$a \leftarrow b + CLS_s (a + g(b,c,d) + X[k] + T[i])$$

Trong đó:

a,b,c,d = 4 từ của bộ đệm, được chỉ rõ trật tự qua mỗi bước nhẩy.

g = là một trong 4 hàm nguyên thủy F, G, H, I

CLS_s =quay vòng dịch trái của 32 bit argument bởi s bit.

Round	Primitive functions g	G(b,c,d)
f_F	F(b,c,d)	(b.c) v ((~b).d)
f_G	G(b,c,d)	(b.d) v (c.(~d))
f_H	H(b,c,d)	b ⊕ c ⊕ d
f_I	I(b,c,d)	c ⊕ (b.(~d))

Các phép toán logic (AND, OR, NOT, XOR) được biểu diễn bằng các biểu tượng (., v, ~, ⊕). Hàm F là hàm điều kiện: if b then c else d. Tương tự, G có thể được xác định if d then b else c. Hàm H xây dựng bit kiểm tra chẵn lẻ.

Bảng IIIa là bảng chân lý của 4 hàm.

b	c	d	F	G	H	I
0	0	0	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	1	1	0
0	1	1	1	0	0	1
1	0	0	0	0	1	1
1	0	1	0	1	0	1
1	1	0	1	1	0	0
1	1	1	1	1	1	0

c. Thuật toán MD5

```
/* Process each 16 word( 512-bit) block*/
For q = 0 to (N/16) - 1 do
/* Copy block i into X*/
For  j=0 to 15 do
Set X[j] to M[q*16+j]
end/*of loop on*/
/* Save A as AA, B as BB, C as CC and D as DD*/
AA =A
BB = B
CC = C
DD= D
/* Round 1*/
/* Let [abcd k s i] denote the operation
a=b+((a + F(b,c,d) + X[k] + T[i] <<<s)
Do the following 16 operations.*/
[ABCD 0    7    1]
[DABC 1   12   2]
[CDAB 2   17   3]
[BCDA 3   22   4]
[ABCD 4    7    5]
[DABC 5   12   6]
[CDAB 6   17   7]
[BCDA 7   22   8]
[ABCD 8    7    9]
[DABC 9   12 10]
[CDAB 10 17 11]
```

[BCDA 11 22 12]

[ABCD 12 7 13]

[DABC 13 12 14]

[CDAB 14 17 15]

[BCDA 15 22 16]

/* Round 2*/

/* Let [abcd k s i] denote the operation

a=b+((a + G(b,c,d) + X[k] + T[i] <<<s)

Do the following 16 operations.*/

[ABCD 1 5 17]

[DABC 6 9 18]

[CDAB 11 14 19]

[BCDA 0 20 20]

[ABCD 5 5 21]

[DABC 10 9 22]

[CDAB 15 14 23]

[BCDA 4 20 24]

[ABCD 9 5 25]

[DABC 14 9 26]

[CDAB 3 14 27]

[BCDA 8 20 28]

[ABCD 13 5 29]

[DABC 2 9 30]

[CDAB 7 14 31]

[BCDA 12 20 32]

/* Round 3*/

/* Let [abcd k s i] denote the operation

a=b+((a + H(b,c,d) + X[k] + T[i] <<<s)

Do the following 16 operations.*/

[ABCD 5 4 33]

[DABC 8 11 34]

[CDAB 11 16 35]

[BCDA 14 23 36]

[ABCD 1 4 37]

[DABC 4 11 38]

[CDAB 7 16 39]

[BCDA 10 23 40]

[ABCD 13 4 41]

[DABC 0 11 42]

[CDAB 3 16 43]

[BCDA 6 23 44]

[ABCD 9 4 45]

[DABC 12 11 46]

[CDAB 15 16 47]

[BCDA 2 23 48]

/*Round 4*/

/* Let [abcd k s i] denote the operation

a=b+((a + I(b,c,d) + X[k] + T[i] <<<s)

Do the following 16 operations.*/

[ABCD 0 6 49]

[DABC 7 10 50]

[CDAB 14 15 51]

[BCDA 5 21 52]

[ABCD 12 6 53]

[DABC 3 10 54]

[CDAB 10 15 55]

[BCDA 1 21 56]

[ABCD 8 6 57]

[DABC 15 10 58]

[CDAB 6 15 59]

[BCDA 13 21 60]

[ABCD 4 6 61]

[DABC 11 10 62]

[CDAB 14 15 63]

[BCDA 5 21 64]

/* then increment it of four registers by the value it had before this block was started.*/

A = A + AA

B= B + BB

C = C + CC

D = D + DD

end./* of loop on q*/

d. Sức mạnh của MD5

Thuật toán MD5 có tính chất mỗi bit của mã hàm băm là một hàm của các bit đầu vào. Sự lặp lại phức tạp của 4 hàm cơ bản (F, G, H, I) tạo ra kết quả có sự hòa lẫn rất mạnh, do vậy mà nó không dễ dàng có hai thông điệp lựa chọn ngẫu nhiên thậm chí chúng có cùng nội dung tổng quát mà có cùng một mã Hash như nhau. Theo Rivest phỏng đoán trong RFC thì MD5 rất mạnh trong 128 bit mã băm, rất khó khăn trong việc xây dựng hai thông điệp có cùng một thông điệp rút gọn trong trật tự của 2^{64} toán hạng, bên cạnh đó cũng rất khó tìm ra một thông điệp với một điểm rút gọn trong trật tự 2^{128} toán hạng.

Cho đến nay chưa có một phân tích nào bác bỏ các nhận định trên. Tuy nhiên khi sử dụng các phân tích bí mật(cryptalalysis) khác nhau có thể trong thời điểm

phù hợp sẽ tìm ra hai thông điệp mà cùng đưa ra một rút gọn giống nhau trong một vòng đơn MD5.

Hàm Hash MD5 là một trong các hàm Hash mạnh và chưa thể tìm ra được trong thời điểm hiện nay. Hàm MD5 đã sử dụng một số hàm rất phức tạp trong 4 vòng và chưa tìm ra đụng độ trên các hàm này. Tuy nhiên, hàm MD5 chạy chậm hơn MD4 khoảng 30%(0.9Mbytes / sec) nhưng nó được coi là mạnh nhất hiện nay.

MD5 được phát triển và kế thừa từ MD4 do vậy về cơ bản thì MD5 có nhiều nét đặc trưng tương tự như MD4.

CHƯƠNG 4: CHỮ KÝ CHỐNG CHỐI BỎ

4.1. Giới thiệu

Các chữ ký chống chối bỏ do Chaum và Antwerpen đưa ra từ năm 1989. So với những chữ ký số khác chúng có một vài đặc điểm mới. Đặc điểm khác biệt nhất trong các chữ ký này là chữ ký không thể xác minh được nếu không có sự hợp tác của người ký là Bob. Như vậy, điều này sẽ bảo vệ được Bob trước khả năng các tài liệu được anh ta ký bị nhân bản và được phân phối bằng phương pháp điện tử mà không có sự đồng ý của anh ta.

Tuy nhiên, liệu có cần sự hợp tác của Bob để xác minh chữ ký không? (điều này nhằm ngăn chặn việc Bob từ chối nhận đã ký thông báo trước đó). Bob có thể tuyên bố một chữ ký hợp lệ là giả mạo và từ chối xác minh nó, hoặc thực hiện giao thức theo cách để chữ ký không thể được xác minh. Để ngăn chặn tình huống này xảy ra, sơ đồ chữ ký chống chối bỏ đã kết hợp giao thức từ chối(theo giao thức này Bob có thể chứng minh chữ ký là giả mạo. Như vậy, Bob có khả năng chứng minh trước tòa rằng chữ ký bị lừa dối trên thực tế là giả mạo.(Nếu anh ta không chấp nhận tham gia vào giao thức từ chối thì điều này được xem như là bằng chứng chứng tỏ chữ ký trên thực tế là thật và anh ta đang cố gắng từ chối chữ ký của mình).

Như vậy sơ đồ chữ ký chống chối bỏ gồm 3 thành phần: thuật toán ký, giao thức xác minh và giao thức từ chối.

Ta có thể xét hai ví dụ sau:

+ Ví dụ 1: Một khách hàng A muốn truy nhập đến một miền bí mật được giám sát bởi khách hàng B, B yêu cầu A ký một lần vào văn bản trước khi truy nhập. Nếu A sử dụng chữ ký không chối bỏ được thì B không thể chứng minh(sau đó) với mọi người là A đã sử dụng tài liệu này mà không có sự tham gia trực tiếp của B trong quá trình xác minh chữ ký.

+ Ví dụ 2: Giả sử rằng một vài công ty lớn A nào đó tạo một sản phẩm phần mềm. A ký vào sản phẩm và bán cho B, người có ý định tạo ra một bản sao của sản phẩm này và bán lại cho C. C không thể xác minh tác giả của sản phẩm phần mềm đó mà không có sự hợp tác của A. Tất nhiên, điều này không ngăn chặn được B ký lại lên sản phẩm với chữ ký riêng nhưng với sự hợp tác của A thì dễ dàng lần ra dấu vết hành động của B.

4.2. Sơ đồ chữ ký chống chối bỏ.

4.2.1. Thuật toán ký

Cho p = 2q+1 là một số nguyên tố sao cho q là số nguyên tố và bài toán logarithm rời rạc trong Z_p là không giải được.

Giả sử $\alpha \in Z^*_p$ là phần tử bậc q. Cho $1 \le a \le q-1$ và định nghĩa $\beta = \alpha^a \bmod p$.

Giả sử G biểu thị nhóm con nhân bậc q của Z^*_p (G gồm các thặng dư bậc hai modulo p). Cho P=A=G và định nghĩa

$K=\{(p,\alpha,a,\beta):\beta = \alpha^a \bmod p\}$

Trong đó: các giá trị p,α,β là công khai; a là bí mật.

Với $k=(p,\alpha,a,\beta)$ và $x \in G$, định nghĩa:

$y=sig_k(x)= x^a \bmod p$

và y là chữ ký trên văn bản x;

4.2.2. Thuật toán xác minh:

Với $x,y \in G$, việc xác minh được thực hiện như sau:

1. Alice chọn ngẫu nhiên $e_1,e_2 \in Z^*_p$

2. Alice tính $c = y^{e1}\beta^{e2} \bmod p$ và gửi nó cho Bob

3. Bob tính $d = c^{a-1 \bmod q} \bmod p$ và gửi nó cho Alice.

4. Alice chấp nhận y là chữ ký hợp lệ khi và chỉ khi:

$d \equiv x^{e1}\alpha^{e2} \bmod p$

4.2.3. Giao thức từ chối:

1. Alice chọn ngẫu nhiên $e_1, e_2 \in Z^*_p$

2. Alice tính $c= y^{e1}\beta^{e2} \bmod p$ và gửi nó cho Bob

3. Bob tính $d = c^{a-1 \bmod q} \bmod p$ và gửi nó cho Alice.

4. Alice xác minh xem có $d \equiv x^{e1}\alpha^{e2} \bmod p$ không

5. Alice chọn ngẫu nhiên $f_1, f_2 \in Z^*_q$

6. Alice tính $C= y^{f1}\beta^{f2} \bmod p$ và gửi nó cho Bob

7. Bob tính $D = C^{a-1 \bmod q} \bmod p$ và gửi nó cho Alice.

8. Alice xác minh xem có $D \cong x^{f1}\alpha^{f2} \bmod p$ không

9. Alice kết luận rằng y là giả mạo khi và chỉ khi $(d\alpha^{-e2})^{f1} \equiv (D\,\alpha^{-f2})^{e1} \pmod p$

Ta xét vai trò của p và q trong sơ đồ này. Sơ đồ này tồn tại trong Z^*_p; tuy vậy cần có khả năng tính toán theo nhóm nhân con G của Z^*_p có bậc nguyên tố. Đặc biệt ta cần có khả năng tính được các phần tử nghịch đảo modulo |G| - đây là lý do giải thích tại sao |G| phải là số nguyên tố. Để thuận lợi lấy p = 2q + 1, trong đó q là số nguyên tố.

Trước hết ta chứng minh rằng, Alice sẽ chấp nhận một chữ ký hợp lệ. Trong các tính toán sau đây, tất cả các số mũ được rút gọn theo modulo q. Đầu tiên nhận xét rằng:

$d \equiv c^{\alpha-1} \bmod p$

$\equiv y^{e1a-1}\beta^{e2a-1} \bmod p$

vì $\beta \equiv \alpha^a \bmod p$, ta có:

$\beta^{a-1} \equiv \alpha \pmod p$

Tương tự:

$y \equiv x^a \bmod p$, ta có:

$y^{a-1} \equiv x \pmod p$

Vì thế $d \equiv x^{e1}\alpha^{e2} \bmod p$.

∗ Tiếp theo ta chứng minh rằng Bob không thể lừa Alice chấp nhận chữ ký giả mạo như một chữ ký hợp lệ ngoại trừ một xác suất rất bé. Kết quả này không phụ thuộc vào bất kỳ giả thiết tính toán nào, điều đó có nghĩa là độ an toàn là vô điều kiện.

∗ Định lý 1: nếu $y \cong x^a \bmod p$ thì Alice sẽ nhận y là một chữ ký hợp lệ trên x với xác suất 1/q.

Chứng minh:

Trước hết, nhận xét rằng mỗi yêu cầu c có thể tương ứng chính xác với q cặp được sắp(e_1,e_2) (đó là tại vì cả y và β đều là các phần tử của nhóm nhân G có bậc nguyên tố q). Vì $y = \alpha^{-1} \bmod p$, $\beta = \alpha^a \bmod p$ nên $c = y^{e1}\beta^{e2} = \alpha^{-1e1}\alpha^{ae2} \bmod p$

$= \alpha^{-1e1+ae2} \bmod p$,

c cố định nên $(-1e_1+ae_2) \bmod q = u$ cố định.

$\Rightarrow e_2 = a^{-1}(u - 1e_1) \bmod q$ vi $e_1 \in Z^*_q$

$\Rightarrow$ có q giá trị $e_1 \Rightarrow$ có q cặp (e_1, e_2) cho cùng cùng 1 giá trị c.

Bây giờ, khi Bob nhận được yêu cầu c, anh ta không có cách biết cặp được sắp (e1,e2) nào trong q cặp co thể mà Alice đã dùng để xây dựng c. Ta khẳng định, nếu $y \cong x^a \bmod p$ thì đáp ứng $d \in G$ mà Bob có thể tạo ra là tương ứng với chính xác một trong q cặp được sắp(e1,e2).

Vì α sinh ra G nên ta có thể viết một phần tử bất kỳ thuộc G như một số mũ của α , trong đó số mũ được xác định duy nhất theo modulo q. Vì thế có thể viết $c=\alpha^i$, $d=\alpha^j$, $x=\alpha^k$ và $y=\alpha^{-1}$; với $i,j,k,l \in Z_q$ và mọi phép tính số học theo modulo q. Xét hai đồng dư thức sau:

$c \equiv y^{e1}\beta^{e2} \bmod p)$

$d \equiv x^{e1}\alpha^{e2} (\bmod p)$

Hệ này tương đương với hệ đồng dư thức sau:

$i \equiv -le_1 + ae_2 (\bmod)q$

$j \equiv ke_1 + e_2 (\bmod) q$

Bây giờ giả thiết rằng: $y \cong x^a \pmod p$ thì ta rút ra : $1 \cong a\ k \pmod q$.

Vì thế, ma trận hệ số của hệ các đồng dư thức theo modulo q này có định thức khác 0 và như vậy tồn tại nghiệm duy nhất cho hệ thống. Nghĩa là, mỗi $d \in G$ là một đáp ứng đúng với một trong q cặp (e1,e2) được sắp có thể. Hệ quả là xác suất để Bob đưa cho Alice một đáp ứng d sẽ được xác minh đúng là 1/q.

Bây giờ ta chứng minh hai vấn đề:

1. Bob có thể thuyết phục Alice rằng chữ ký không hợp lệ là giả mạo

2. Bob không thể làm Alice tin rằng một chữ ký tin cậy là giả mạo trừ một xác suất rất bé.

∗ Định lý 2: Nếu $y \cong x^a \bmod p$ và cả Alice và Bob thực hiện theo giao thức từ chối thì $(d\alpha^{-e2})^{f1} \equiv (D\alpha^{-f2})^{e1} \bmod p$

Chứng minh:

Do cả Alice và Bob đều thực hiện giao thức nên ta có thể sử dụng các yếu tố:

$d \equiv c^{a-1} \bmod p$

$c \equiv y^{e1}\beta^{e2} \bmod p$

và $\beta \equiv \alpha^{a} \bmod p$

Ta có: $(d\alpha^{-e2})^{f1} \equiv ((y^{e1}\beta^{e2})^{a-1}\alpha^{e-2})^{f1} (\bmod p)$

$\equiv (y^{e1f1a-1}\beta^{e2a-1f1}\alpha^{-e2f1} (\bmod p)) \equiv y^{e1f1a-1}\alpha^{e2f1}\alpha^{-e2f1} \bmod p$

$\equiv y^{e1f1a-1} \bmod p$

Tương tự, dùng các yếu tố $D \equiv C^{a-1} \bmod p$, $C \equiv y^{f1}\beta^{f2} \bmod p$ và $\beta \equiv \alpha^{a} \bmod p$, ta có:

$(D\alpha^{-f2})^{e1} \equiv y^{e1f1a-1} \bmod p$

vì thế phép kiểm tra tính phù hợp trong bước 9 thành công.

Bây giờ xét xác suất để Bob có thể từ chối một chữ ký hợp lệ. Trường hợp này không giả thiết Bob thực hiện theo thủ tục. Nghĩa là Bob có thể không xây dựng D và d như trong giao thức. Vì thế trong định lý tiếp theo chi giả thiết rằng, Bob có thể tạo ra các D và d thỏa mãn điều kiện trong các bước 4,8,9 của giao thức chối bỏ.

∗ Định lý 3: Giả sử $y \equiv x^{a} \bmod p$ và Alice thực hiện theo giao thức từ chối. Nếu $d \cong x^{e1}\alpha^{e2} \bmod p$ và $D \cong x^{f1}\alpha^{f2} (\bmod p)$ thì xác suất để $(d\alpha^{-e2})^{f1} \cong (D\alpha^{-f2})^{e1} (\bmod p)$ là 1-1/q.

Chứng minh:

Giả sử rằng các đồng dư thức sau được thỏa mãn

$y \equiv x^{a} (\bmod p)$

$d \cong x^{e1}\alpha^{e2} (\bmod p)$

$D \cong x^{f1}\alpha^{f2} (\bmod p)$

$(d\alpha^{-e2})^{f1} \equiv (D\alpha^{-f2})^{e1} (\bmod p)$

ta sẽ đưa được ra mâu thuẫn như sau:

Trong giao thức chối bỏ, bước 9 có thể viết lại như sau:

$D \equiv d_0^{f1}\alpha^{f2} (\bmod p)$

trong đó, $d_0 = d^{1/e1}\alpha^{-e2/e1} \bmod p$ là giá trị chỉ phụ thuộc vào các bước từ 1-4 trong giao thức.

Áp định lý 1, ta kết luận y là chữ ký hợp lệ đối với d_0 với xác suất 1-1/q. Song ta đang giả thiết y là chữ ký hợp lệ đối với x, nghĩa là ta có $x^{a} \equiv d_0^{a} (\bmod p)$ với

xác suất cao, điều này kéo theo là $x = d_0$.

Tuy nhiên do $d \equiv x^{e1}\alpha^{e2} \pmod p$ có nghĩa là $x \equiv d^{1/e1}\alpha^{-e2/e1} \pmod p$

Và từ biểu thức $d_0 \equiv d^{1/e1}\alpha^{-e2/e1} \pmod p$, suy ra $x \neq d_0 \Rightarrow$ ta nhận được mâu thuẫn.

Như vậy Bob co thể lừa dối Alice theo cách này với xác suất là 1/q.

+ Ví dụ: Giả sử lấy p=467, vi 2 là phần tử nguyên thủy nên $2^2 = 4$ là phần tử sinh của G, gồm các thặng dư bậc hai modulo 467. Vì thế ta có thể lấy α=4. Giả thiết a = 101, Khi đó:

$\beta = \alpha^a \bmod 467 = 449$

Bob sẽ ký thông báo x=119 với chữ ký:

$y = 119^{101} \bmod 467 = 129$

Bây giờ giả sử Alice muốn xác minh chữ ký y, cô ta chọn các số ngẫu nhiên chẳng hạn e_1=38, e_2=397. Cô tính c= 13 và ngay lúc đó Bob sẽ trả lời với d= 9. Alice kiểm tra câu trả lời bằng việc xác minh rằng: $119^{38}4^{397} = 9 \bmod 467$. Vì thế mà Alice chấp nhận y là chữ ký hợp lệ.

+ Ví dụ: Giả sử p= 467, α=4, a=101 và β=449. Giả sử thông báo x=286 được ký với chữ ký y = 83 và Bob muốn thuyết phục Alice rằng chữ ký này không hợp lệ.

Giả sử Alice bắt đầu bằng việc chọn các giá trị ngẫu nhiên e_1=45, e_2=237. Alice tính c=305 và Bob trả lời với d=109. Sau đó Alice tính:

$246^{45}4^{237} \bmod 467 = 149$

vì $149 \neq 109$ nên Alice thực hiện bước tiếp theo là chọn các giá trị ngẫu nhiên f_1=125, f_2=9. Alice tính C=270 và Bob trả lời với D=68. Alice tính:

$286^{125}4^{9} \bmod 467 = 25$

Vì $25 \neq 68$ nên Alice thực hiện bước cuối cùng là kiểm tra tính phù hợp.

Bước kiểm tra này thành công vì:

$(109 \times 4^{-237})^{125} \equiv 188 \pmod{467}$

và: $(68 \times 4^{-9})^{45} \equiv 188 \pmod{467}$

Vì thế Alice tin rằng chữ ký y trên x là không hợp lệ.

CHƯƠNG 5 : ÁP DỤNG CHỮ KÝ CHỐNG CHỐI BỎ VÀO QUẢN LÝ HÀNH CHÍNH CỦA TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

5.1. Đặt vấn đề.

Với bất kỳ một cơ quan tổ chức nào thì hệ thống quản lý hành chính đóng vai trò rất quan trọng. Trường Đại Học Dân Lập Hải Phòng có cơ sở 1 tại số 36-Đường dân lập-Dư Hàng Kênh- Lê Chân- Hải Phòng, cơ sở 2(đang xây dựng) Tại Xã Minh Tân- Kiến Thụy- Hải Phòng và khu Khách Sạn Sinh Viên. Do khoảng cách đia lý nên việc quản lý hành chính khá phức tạp và khó khăn.

Trường Đại Học Dân Lập Hải Phòng gồm các phòng ban, các khoa, các trung tâm nên việc ban hành các quyết định, văn bản từ Nhà trường tới các phòng ban đều do bộ phận văn thư đảm nhiệm nên rất khó khăn và mất thời gian. Mặt khác, những yêu cầu, kiến nghị của các khoa, trung tâm muốn lấy ý kiến, chỉ thị của lãnh đạo nhà trường cũng rất phức tạp.

5.2. Giải quyết vấn đề.

Ngày nay với sự phát triển mạnh mẽ của công nghệ cao, công nghệ internet không còn mới mẻ đối với mọi người, việc trao đổi thông tin trên mạng trở nên phổ biến hơn. Vì vậy, giải pháp thuận lợi nhất là chuyển các quyết định, văn bản qua mạng.

Tuy nhiên việc trao đổi tài liệu qua mạng không phải không có khó khăn: ví dụ trên đường truyền văn bản, nghị quyết bị sửa đổi, điều này rất nguy hiểm vì đây không phải là những tài liệu thông thường mà là các nghị quyết, văn bản quy chế của Nhà trường. Cũng có những vấn đề khác cần giải đáp, ví dụ như văn bản đó có đáng tin không? văn bản đó có chính xác không?

Chúng ta biết rằng, trong các nghị quyết, văn bản,... thông thường trên giấy thì bao giờ cũng kết thúc bằng chữ ký người quyết định. Đó là bằng chứng để chứng minh hiệu lực của văn bản. Nếu chữ ký bị tẩy xóa thì ta co thể nghi ngờ tài liệu đã bị sửa đổi, hoặc chữ ký là giả mạo. Nhưng khi đã trao đổi trên mạng thì chữ ký viết tay không thể thực hiện được như trên văn bản thông thường.

Hiện nay có rất nhiều chữ ký điện tử đã được đem vào sử dụng trong thực tiễn. Do tính chất và đặc điểm của việc quản lý hành chính, tôi chọn chữ ký chống chối bỏ để áp dụng. Chữ ký chống chối bỏ là một chữ ký với ba giao thức: giao thức ký, giao thức kiểm thử và giao thức chối bỏ, do đó phù hợp với việc trao đổi trên mạng, hơn nữa nếu tất cả cùng áp dụng các giao thức chữ ký chống chối bỏ thì việc

ban hành các quyết định có chữ ký của Nhà trường hoàn toàn đáng tin cậy. Người ký muốn phủ nhận chữ ký của mình cũng không được. Hơn nữa, một người có thể chối bỏ chữ ký nếu nó không phải chữ ký của mình. Đặc biệt chữ ký chống chối bỏ cũng không thể nhân bản được.

Khi sử dụng chữ ký chống chối bỏ, tất cả những người tham gia trong mạng sẽ có một số nguyên tố p, phần tử nguyên thủy α và khóa β trên thư mục công khai chung của Trường và một khóa bí mật a cho riêng mình, trong đó α là phần tử có bậc q trong Z^*_p, p=2q + 1 (q là số nguyên tố) và $\beta = \alpha^a \bmod p$.

+ Ví dụ: Khoa Công Nghệ Thông Tin muốn gửi danh sách những sinh viên đủ điều kiện bảo vệ đồ án tốt nghiệp cho Hiệu trưởng duyệt. Trước khi gửi danh sách cho Hiệu trưởng thì Trưởng khoa Công Nghệ Thông Tin sẽ tiến hành việc ký như sau:
  1. Ông sẽ lấy số nguyên tố p trên thư mục công khai và một khóa bí mật a chỉ dùng cho riêng mình.
  2. Tiến hành ký trên văn bản: $y=x^a \bmod p$(Ở đây x là văn bản thu được sau hai phép biến đổi thông báo cần ký là: Hash và đưa về nhóm con G).
  3. Gửi văn bản kèm theo chữ ký trên mạng.

Về phía mình, Hiệu trưởng sẽ kiểm tra lại chữ ký có đúng là của Trưởng khoa Công Nghệ Thông Tin hay không?. Ông ta sẽ tiến hành giao thức kiểm thử cùng với sự hợp tác của Trưởng khoa Công Nghệ Thông Tin như sau:

1. Hiệu trưởng lấy ngẫu nhiên hai số e_1, e_2 và tính $c=y^{e1}\beta^{e2} \bmod p$ rồi gửi lại cho Trưởng khoa Công Nghệ Thông Tin.
2. Trưởng khoa Công Nghệ Thông Tin sẽ tính $d = a^{-1 \bmod q} \bmod p$ rồi gửi lại cho Hiệu trưởng.
3. Hiệu trưởng kiểm tra nếu $d \equiv x^{e1}\alpha^{e2} \bmod p$ thì chấp nhận là chữ ký hợp lệ.

Sau khi kiểm tra chữ ký, Hiệu trưởng sẽ ra quyết định cho các sinh viên mà khoa Công Nghệ Thông Tin đã gửi được bảo vệ đồ án tốt nghiệp và ký vào quyết định trước khi gửi cho khoa Công Nghệ Thông Tin. Khoa Công Nghệ Thông Tin cũng sẽ kiểm tra chữ ký của Hiệu trưởng như cách thức trên. Nếu cả hai bên không có vấn đề gì thì thời gian tiến hành bảo vệ do khoa Công Nghệ Thông Tin ấn định. Nếu trong bản danh sách có sai sót gì thì người chịu trách nhiệm là Trưởng khoa Công Nghệ Thông Tin vì danh sách đã có chữ ký của Trưởng khoa. Tuy nhiên, Trưởng khoa có thể tuyên bố chữ ký đó là giả mạo và từ chối xác minh nó, hoặc thực hiện giao thức theo cách để chữ ký không thể xác minh. Để ngăn chặn tình

huống này, Hiệu trưởng sẽ thực hiên giao thức từ chối. Nếu Trưởng khoa Công Nghệ Thông Tin không chấp nhận tham gia vào giao thức từ chối thì điều này xem như chữ ký trên thực tế là thật và ông ta đang cố gắng từ chối chữ ký cua mình. Nếu Trưởng khoa tuân thủ các bước của giao thức và kết quả là việc Hiệu trưởng kết luận chữ ký giả mạo thi ông sẽ có quyền chối bỏ chữ ký kia một cách thuyết phục. Giao thức từ chối được tiến hành như sau:

1. Hiệu trưởng chọn ngẫu nhiên e_1, $e_2 \in Z^*_q$.
2. Hiệu trưởng tính $c = y^{e1}\beta^{e2} \bmod p$ và gửi nó cho Trưởng khoa CNTT.
3. Trưởng khoa CNTT tính $d= c^{a-1 \bmod q} \bmod p$ và gửi nó cho Hiệu trưởng.
4. Hiệu trưởng xác minh xem có $d \cong x^{e1}\alpha^{e2} \bmod p$ không?
5. Hiệu trưởng chọn ngẫu nhiên f_1, $f_2 \in Z^*_q$
6. Hiệu trưởng tính $C= y^{f1}\beta^{f2} \bmod p$ và gửi nó cho Trưởng khoa CNTT.
7. Trưởng khoa CNTT tính $D = C^{a-1 \bmod q} \bmod p$ và gửi nó cho Hiệu trưởng.
8. Hiệu trưởng xác minh xem có $D \cong x^{f1}\alpha^{f2} \bmod p$ không?
9. Hiệu trưởng kết luận rằng y là giả mạo khi và chỉ khi $(d\alpha^{-e2})^{f1} \equiv (D\alpha^{-f2})^{e1} \pmod p$.

Như vậy bằng cách sử dụng chữ ký số chống chối bỏ thì việc trao đổi các quyết định, giấy tờ rất nhanh và chính xác. Hơn nữa, tất cả các chữ ký và văn bản đều được lưu giữ lại do đó có thể kiểm thử lại bất cứ lúc nào thấy cần thiết.

Tuy nhiên, khi cài đặt lược đồ chữ ký vào mạng thông tin máy tính của trường tôi thấy cần nghiên cứu thêm ba vấn đề nữa:

1. Có thể Hiệu trưởng hoặc Trưởng khoa Công Nghệ Thông Tin bận, nếu cần ủy quyền cho người khác tham gia vào giao thức hỏi đáp, thì phải thực hiện như thế nào?
2. Cũng có trường hợp, bên nhận thấy văn bản gửi cho mình có nội dung mà mình chưa muốn thực hiện nên không tiến hành kiểm tra chữ ký và lờ đi. Vậy làm thế nào để ngăn chặn sự cố này?
3. Nghiên cứu việc lưu trữ trên máy tính các văn bản và các chứng cứ tương ứng như thế nào cho an toàn thuận tiện.

CHƯƠNG 6: CHƯƠNG TRÌNH

6.1. Giải thích chương trình

+ Để cài đặt được sơ đồ chữ ký chông chối bỏ, tôi sử dụng ngôn ngữ C++ và để an toàn tôi thực trên các số cỡ lớn hệ cơ số 256. Mỗi số cỡ lớn được cài đặt như sau: độ dài các số cỡ lớn được lưu vào một biến unsigned int và giá trị của nó lưu trong một mảng unsigned char.
+ Để mô phỏng chữ ký chống chối bỏ, tôi chọn số p có độ dài 19 byte và các giá trị như sau: 56 29 43 85 112 151 142 150 211 7 90 78 106 77 226 71 208 21 191.
+ Khóa α được tạo theo thủ tục sau:
  1. Chọn ngẫu nhiên một phần tử $e \in Z^*_p$ và tính $\alpha = e^2 \bmod p$
  2. Nếu $\alpha = 1$, quay lại bước 1.
+ Điều kiện để thực hiện chữ ký chống chối bỏ là x, $y \in G$.Để ký được trên các văn bản bất kỳ, tôi thực hiện như sau:
  1. Tiến hành Hash văn bản gốc(giả sử là x1) theo thuật toán MD5 thành một văn bản gồm 16 byte (giả sử là m)
  2. Tính $x = \alpha^m \bmod p$. Và giao thức ký được tiến hành trên x để thu được chữ ký là y và y được xem như là chữ ký trên văn bản gốc. Điều này đảm bảo x, $y \in G$.
+ Chương trình được chia làm hai modun sau:
  1. Tạo danh sách những người sử dụng trên mạng trong một cơ sở dữ liệu: Modun này cho phép đăng ký tham ra bằng cách nhập user name và password.
  2. Tạo khóa bí mật: Trước tiên, người sử dụng phải đăng nhập mạng bằng cách nhập user name và password(dạng mã hóa) của mình. Sau khi đã đăng nhập mạng thành công thì có thể tiến hành thủ tục tạo khóa với số p cho trước. Khóa α, β sẽ được đưa vào thư mục công khai, còn khóa bí mật a thì người sử dụng giữ cho riêng mình.
  3. Cuối cùng là modun chứa các giao thức ký, kiểm thử và từ chối.

6.2. Các phép toán hỗ trợ

a. Thuật toán cộng

Input: x,y là số nguyên, mỗi số gồm n+1 chữ số hệ 256

Output: $x+y = (w_{n+1}w_n...w_1w_0)$ hệ 256.

1. c←0(c là số nhớ)

2. For i=0 to n do

   2.1 $w_i \leftarrow (x_i + y_i + c) \bmod 256$

   2.2 if($x_i + y_i + c < 256$) then $c \leftarrow 0$

3. else $c \leftarrow 1$.
4. $w_{n+1} \leftarrow c$
5. Return $(w_{n+1} w_n ... w_1 w_0)$

b. Thuật toán trừ

Input: x, y là số nguyên, mỗi số gồm n+1 chữ số hệ 256; và $x \geq y$

Output: $x-y = (w_{n+1} w_n ... w_1 w_0)$ hệ 256.

1. $c \leftarrow 0$ (c là số nhớ)
2. For i=0 to n do
   a. $w_i \leftarrow (x_i - y_i + c) \bmod 256$
   b. if($x_i - y_i + c \geq 0$) then $c \leftarrow 0$
   else $c \leftarrow -1$
3. Return $(w_{n+1} w_n ... w_1 w_0)$

c. Thuật toán nhân

Input: x, y là số nguyên gồm n+1 và t+1 chữ số hệ 256;

Output: $x*y = (w_{n+1} w_n ... w_1 w_0)$

1. For i=0 to (n+t+1) do $w_i \leftarrow 0$
2. For i =0 to t do

   2.1. $c \leftarrow 0$

   2.2. For j =0 to n do

Tính $(uv)_{256} = w_{i+j} + x_j * y_i + c$;

Đặt $w_{i+j} \leftarrow v$; $c \leftarrow u$;

   2.3. $w_{i+n+1} \leftarrow u$.

3. Return $(w_{n+t+1} ... w_1 w_0)$

d. Thuật toán chia

Input: $x = (x_n ... x_1 x_0)_{256}$, $y = (y_t ... y_1 y_0)$

với $n \geq t \geq 1$; $y_t \neq 0$

Output: thương $q = (q_{n-t} ... q_1 q_0)_{256}$; số dư $r = (r_t ... r_1 r_0)_{256}$;

sao cho $x = q.y + r$; $0 \leq r \leq y$.

1. For j=0 to (n-1) do $q_j \leftarrow 0$
2. While($x \geq y.256^{n-t}$) do q_{n-t} +1; $x \leftarrow x-y.256^{n-t}$
3. For i=n downto (t+1) do
   3.1. if $x_i = y_t$ then $q_{i-t-1} \leftarrow 256-1$
   else $q_{i-t-1} \leftarrow \lfloor (x_i.256 + x_{i-1})/y_t \rfloor$
   3.2. While (q_{i-t-1} ($y_t 256+y_{t-1}$)>$x_i 256^2+x_{i-1}256+x_{n-2}$) do $q_{i-t-1} \leftarrow q_{i-t-1}-1$
   $x \leftarrow x-q_{i-t-1} y 256^{i-t-1}$.
   3.3. While x<0 do $x \leftarrow x+y256^{i-t-1}$; $q_{i-t-1} \leftarrow q_{i-t-1}-1$.
4. $r \leftarrow x$
5. Return (q, r)

e. Thuật toán tính AB mod M

Input: A,B,M là ba số nguyên hệ 256; A, B <M;

Output: P= AB mod M

1. Đổi $B \leftarrow (B_n B_{n-1} ... B_1 B_0)_2$; H là số chứ số của B ở hệ nhị phân
2. $T \leftarrow H-1$; $D \leftarrow M-A$
3. While (B(T) =0) do $T \leftarrow T-1$
4. $P \leftarrow A$
5. For i=T-1 downto 0 do
   5.1 if(P<M) then $P \leftarrow 2*P$
   else $P \leftarrow 2(P-M)$
   5.2 if (B(i)=1) then
   if (P<M) then $P \leftarrow P+A$
   else $P \leftarrow P-D$
6. if ($P \geq M$) $P \leftarrow P-M$
7. Return P.

f. Thuật toán nhị phân y^n mod N

Input: y,n,N là ba số nguyên hệ 256;

Output: $b=y^n$ mod N

1. $a \leftarrow n$; $b \leftarrow 1$; $c \leftarrow y$;
2. if (a chẵn) then $d \leftarrow 1$; else $d \leftarrow 0$; $a \leftarrow [a/2]$;
3. if d=1 then goto 6
4. $b \leftarrow b*c$ mod N
5. if (a=0) then return 0
6. $c \leftarrow c*c$ mod N; goto 2;

7. Return (b);

g. Thuật toán tính phần tử nghịch đảo: n^{-1} mod m

Input: m, n là hai số nguyên hệ 256;

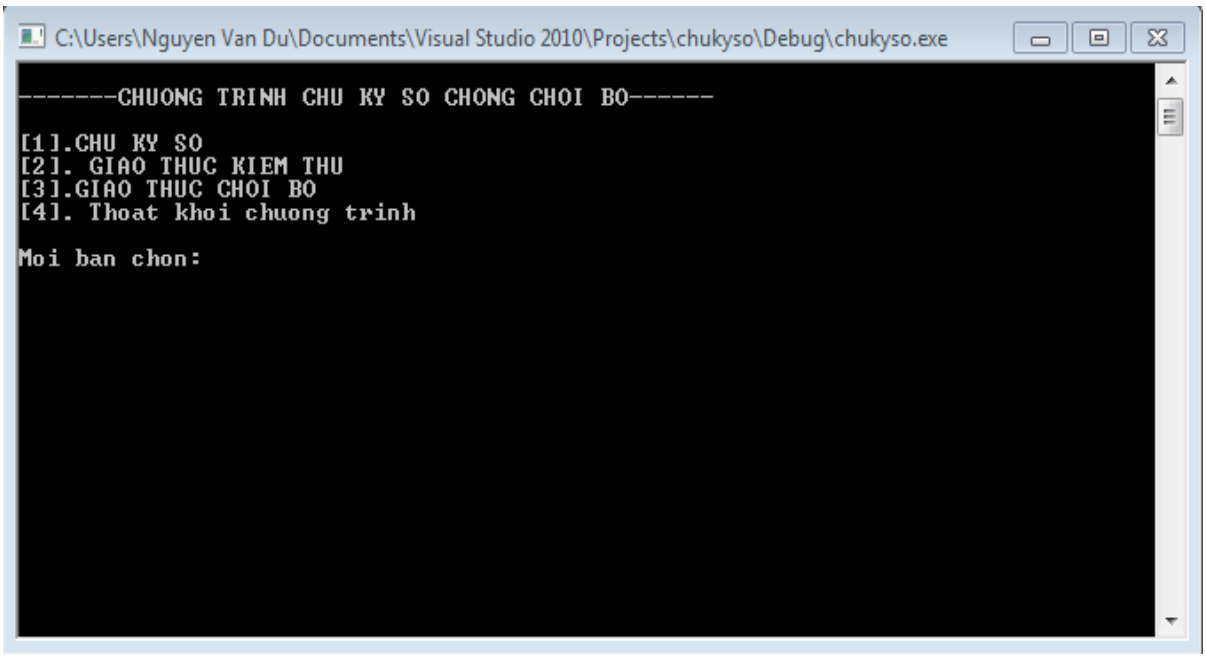
(a_1, a_2, a_3); (b_1, b_2, b_3); (c_1, c_2, c_3) là ba tập vector hệ 256.

Output: y= n^{-1} mod m

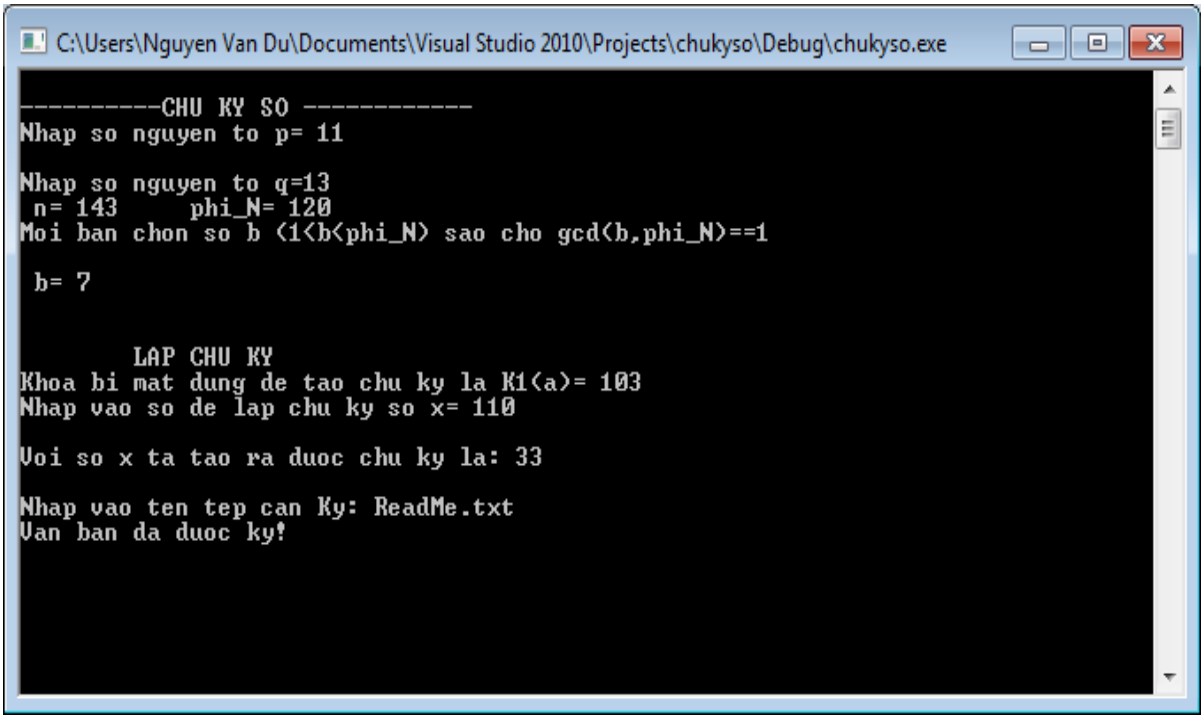
1. (a_1, a_2, a_3)←(1, 0, m); (b_1, b_2, b_3)←(0, 1, n);
2. if(b_3=0) then y=a_2 ; return 0;
3. q←[a_3/b_3] và
   (c_1,c_2, c_3)←(a_1, a_2, a_3) -q(b_1, b_2, b_3)
   (a_1, a_2, a_3)← (b_1,b_2, b_3)
   (b_1, b_2, b_3)←(c_1, c_2, c_3)
   goto 2;
4. Return (y);

6.3. Demo chương trình.

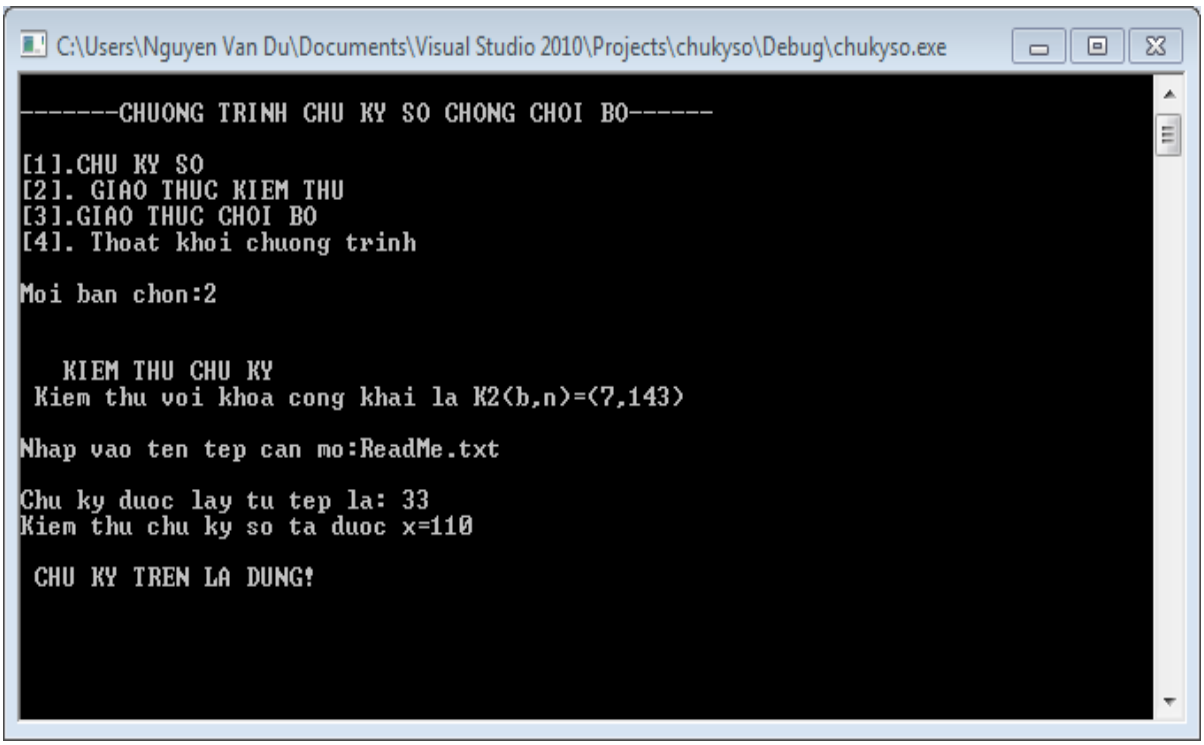
a. Khởi động chương trình:



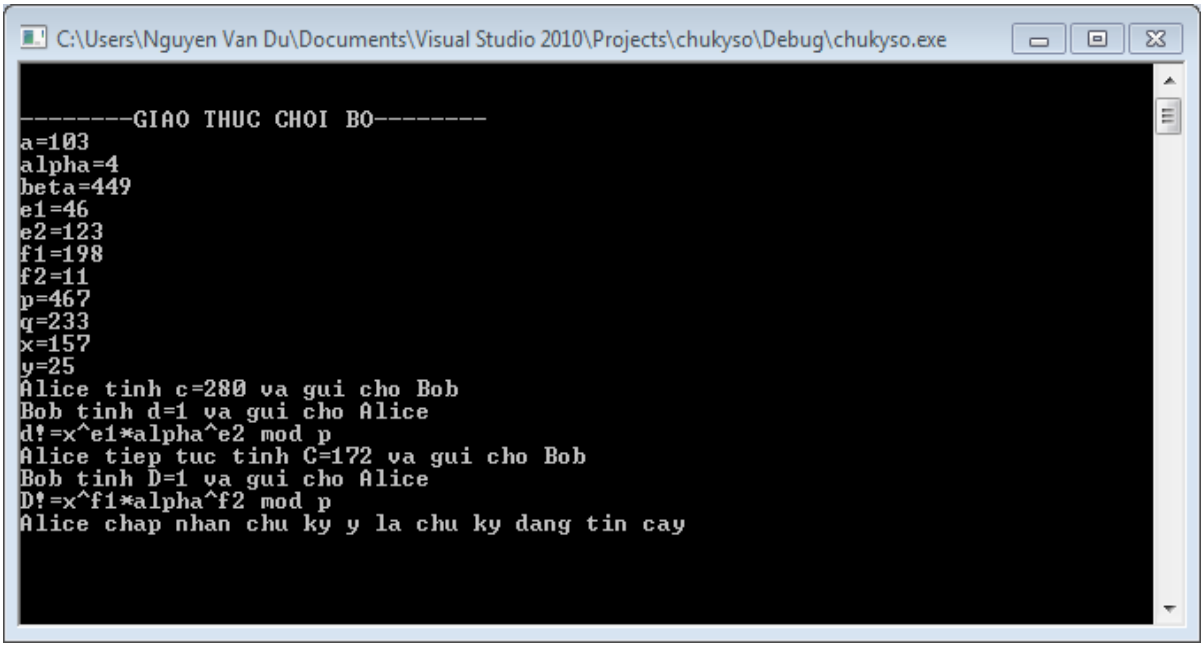
b. Giao thức ký:



c. Giao thức kiểm thử:



d. Giao thức chối bỏ:



KẾT LUẬN

Ngày nay cùng với sự phát triển của khoa học kỹ thuật hiện đại, công nghệ thông tin đã giúp nhiều trong các lĩnh vực đời sống của con người. Mạng Internet với tốc độ nhanh, lượng thông tin trao đổi có thể rất lớn và đặc biệt không hạn chế người sử dụng, giúp cho con người có thể trao đổi với nhau nhanh hơn, chính xác hơn và hiệu quả hơn.

Trong đồ án này em đã đi sâu tìm hiểu về lược đồ chữ ký số chống chối bỏ, lược đồ chữ ký người xác nhận được chỉ định và lược đồ chữ ký người xác nhận không thể chối bỏ.

Với lược đồ chữ ký số chống chối bỏ đã giải quyết được yêu cầu của chữ ký số đó là chống sao chép không hợp pháp. Vì chữ ký số chống chối bỏ chỉ có thể kiểm tra khi có sự hợp tác của người ký thông qua giao thức hỏi- đáp. Tuy nhiên đó cũng là nhược điểm của chữ ký số chống chối bỏ vì nếu trường hợp người ký không hợp tác hoặc người ký cố tình không thực hiện đúng giao thức thì khả năng kiểm tra là không thể và có thể chối bỏ chữ ký đó.

Tuy vậy, chữ ký số chống chối bỏ cũng là một sự lựa chọn rất hợp lý trong truyền tải dữ liệu số yêu cầu bảo mật và tính xác thực và một số ứng dụng khác.

Để hoàn thành được đồ án này, tôi đã nhận được sự chỉ bảo, hướng dẫn tận tình của thầy giáo TS. Hồ Văn Canh. Tuy nhiên, đồ án không tránh khỏi thiếu xót, rất mong sự góp ý của các Thầy, Cô giáo.

TÀI LIỆU THAM KHẢO

[1] Phan Đình Diệu(2002): Lý thuyết mật mã và an toàn thông tin. NXB Đại học Quốc gia Hà Nội, 2002.

[2]Nguyễn Đ. Cương(2007) : Mật mã và an toàn thông tin(sách điện tử).

[3]Hồ Văn Canh : Vai trò của chữ ký số chống chối bỏ trong thương mại điện tử(2011).

[4]Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone : Handboof of Applied Cryptography CRC press : Boca Raton, New York, London, Tokyo(1997).