Mục Lục

LỜI CẢM ƠN

Trước hết, em xin gửi lời cảm ơn sâu sắc tới PGS. TS. Trịnh Nhật Tiến đã hướng dẫn em phát triển khóa luận đi từ lý thuyết đến ứng dụng. Sự hướng dẫn của thầy trong suốt thời gian qua đã giúp em tiếp cận tới một hướng nghiên cứu khoa học mới: đó là nghiên cứu trong lĩnh vực an toàn thông tin. Qua đó, những lý thuyết về an toàn thông tin đã lôi cuốn em và sẽ trở thành hướng nghiên cứu tiếp của em sau khi tốt nghiệp.

Em xin bày tỏ lòng biết ơn đến các thầy cô trong trường Đại Dân Lập Hải Phòng đã giảng dạy và cho em những kiến thức quý báu, làm nền tảng để em hoàn thành khóa luận cũng như thành công trong nghiên cứu, làm việc trong tương lai.

Cuối cùng, cho em gửi lời cảm ơn sâu sắc tới gia đình, bạn bè đã động viên kịp thời để em học tập tốt và hoàn thành được khóa luận.

Em xin chân thành cảm ơn!

Hải Phòng, tháng 12 năm 2012

Sinh viên

Trần thị chiên

GIỚI THIỆU

Những năm gần đây, nhu cầu trao đổi thông tin từ xa của con người ngày càng lớn, các ứng dụng trao đổi thông tin qua mạng diễn ra ngày càng nhiều.

Tuy nhiên, mỗi loại ứng dụng có những đòi hỏi riêng khác nhau, ví dụ như ứng dụng bầu cử từ xa cần phải che dấu được thông tin người bỏ phiếu, hoặc những văn bản đã được ký nhưng không muốn ai cũng có thể xác thực chữ ký khi chưa được sự đồng ý của người ký. Chữ ký mù đã ra đời để giải quyết vấn đề nêu trên. Ý tưởng chính của ký mù là người ký không biết mình đang ký trên nội dung gì.

Chương 1. MỘT SỐ KHÁI NIỆM CƠ BẢN

1.1. CÁC KHÁI NIỆM CƠ SỞ

1.1.1. Một số khái niệm trong toán học

1.1.1.1. Khái niệm về số nguyên tố

1./ Khái niệm

Số nguyên tố là số nguyên dương chỉ chia hết cho 1 và chính nó

Ví dụ

2, 3, 5… Các hệ mật mã thường sử dụng các số nguyên tố ít nhất là lớn hơn 10^{150}.

2./ Một số định lý về số nguyên tố

Định lý về số nguyên dương > 1

Mọi số nguyên dương n > 1 đều có thể biểu diễn được duy nhất dưới dạng: $n= p_1^{e}{}_1 \ p_2^{e}{}_{2\ldots} p_k^{e}{}_k$ . ... trong đó: k, ni (i =1,2,..,k) là các số tự nhiên, Pi là các số nguyên tố, từng đôi một khác nhau.

Định lý Mersenne

Cho $p = 2^k-1$, nếu p là số nguyên tố thì k phải là số nguyên tố.

Chứng minh

Giả sử k không là nguyên tố. Khi đó k = a.b với 1 < a, b < k.

Như vậy $p = 2^k-1 = 2^{ab}-1 = (2^a)^b - 1= (2^a-1).E$

(Trong đó E là một biểu thức nguyên - áp dụng công thức nhị thức Newton).

Điều này mâu thuẫn giả thiết p là nguyên tố. Vậy giả sử là sai, hay k là số nguyên tố.

1.1.1.2. Ước số chung lớn nhất

1./ Ước số

Khái niệm:

Cho hai số nguyên a, b $\in$ Z, b $\neq$ 0. Nếu có một số nguyên q sao cho a=b*q, thì ta nói rằng a chia hết cho b, ký hiệu b|a. Ta nói b la ước của a, và a là bội của b.

Ví dụ:

*Cho a=6, b=2, ta có 6=2*3, ký hiệu 2|6. Ở đấy 2 là ước của 6 và 6 là bội của 2.*

Tính chất: Cho a, b, c $\in$ Z

+a|a.

+ a|b, b|c $\Rightarrow$ a|c.

+a|b, b|a $\Rightarrow$ a= $\pm$b.

2./ Ước chung lớn nhất

Khái niệm:

Số nguyên d được gọi là ước chung của các số nguyên $a_1, a_2,\ldots,a_n$,nếu nó là ước của tất cả các số đó.

Một ước chung d > 0 của các số nguyên $a_1,a_2,\ldots,a_n$, trong đó mọi ước chung của $a_1, a_2,\ldots,a_n$, đều là ước của d, thì d được gọi là ước chung lớn nhất (ƯCLN) của $a_1, a_2,\ldots,a_n$. Ký hiệu d = gcd ($a_1, a_2,\ldots,a_n$) hay d = ƯCLN ($a_1, a_2,\ldots,a_n$).

Ví dụ:

Cho a = 12, b = 15, gcd (12,15) = 3.

Tính chất:

+d = gcd ($a_1, a_2,\ldots,a_n$) khi và chỉ khi tồn tại các số $x_1, x_2,\ldots,x_n$ sao cho:

$$d = a_1x_1 + a_2x_2 + \ldots + a_nx_n.$$

Đặc biệt: $a_1, a_2,\ldots,a_n$ nguyên số cùng nhau $\Leftrightarrow$ tồn tại các số $x_1, x_2,\ldots,x_n$ sao cho: $1 = a_1x_1 + a_2x_2 + \ldots + a_nx_n$.

+ d = gcd ($a_1, a_2, \ldots, a_n$) $\Leftrightarrow$ gcd ($a_1/d, a_2/d,\ldots,a_n/d$) = 1.

+ gcd ($m.a_1,m.a_2,\ldots,m.a_n$) = m* gcd ($a_1, a_2,\ldots,a_n$) (với m $\neq$ 0).

+ Nếu b > 0, a= b.q + r thì gcd (a,b) = gcd (b,r).

3./ Thuật toán Euclide tìm ước chung lớn nhất

Bài toán:

* Dữ liệu vào: Cho hai số nguyên không âm a,b, a $\geq$ b.

*Kết quả gcd(a,b).

Thuật toán: input(a,b);

While b>0

r = a mod b;

a = b;

b = r;

output(a);

Ví dụ: *a= 30, b= 18*

gcd(30,18) = gcd(18,12) = gcd(12,6) = gcd(6,0) = 6

Bảng 1: Mô tả các bước tính gcd(30,18)

a	b	r	a = b.q + r
30	18	12	30 = 18*1 + 12
18	12	6	18 = 12*1 + 6
12	6	0	12 = 6*2 + 0

1.1.1.3. Hàm Ø Euler

Định nghĩa:

Cho $n \geq 1$. $\emptyset(n)$ được định nghĩa là các số nguyên trong khoảng từ *[1.n]* nguyên tố cùng nhau với *n* . hàm Ø được gọi là hàm phi Euler.

Tính chất:

Nếu *p* là số nguyên tố thì $\emptyset(n) = p-1$.

Hàm phi Euler là hàm có tính nhân:

Nếu $n= p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$ là thừa số nguyến tố của n thì:

$$\emptyset(n)= n\left(1-\frac{1}{p1}\right)\left(1-\frac{1}{p2}\right)\dots.\left(1-\frac{1}{pk}\right)$$

1.1.1.4. Khái niệm số nguyên tố cùng nhau

Khái niệm:

Hai số m và n được gọi là số nguyên tố cùng nhau nếu ước số chung lớn nhất của chúng bằng 1 , ký hiệu gcd (m,n)=1

Ví dụ:

9 và 14 là 2 số nguyên tố cùng nhau.

1.1.1.5. Khái niệm số đồng dư

1./ Khái niệm

Cho a và b là các số nguyên , khi đó a được gọi là đồng dư với b theo modulo n, ký hiệu là a ≡ b mod n nếu a, b chia cho n có cùng số dư. Số nguyên n được gọi là modulo của đồng dư.

Kí hiệu: a ≡ b (mod n).

Ví dụ: 5 ≡ 7 mod 2 vì: 5 mod 2=1 và 7 mod 2 =1 .

2./ Tính chất của đồng dư

a ≡ b mod n nếu và chỉ nếu a và b có cùng số dư khi chia cho n.
Tính chất phản xạ: a ≡ a mod n.
Tính đối xứng: Nếu a ≡ b mod n thì b ≡ a mod n.
Tính bắc cầu: Nếu a ≡ b mod n và b ≡ c mod n thì a ≡ c mod n .
Nếu a ≡ a_1 mod n ,b ≡ b_1 mod n thì a + b ≡ (a_1+ b_1)(mod n) và ab ≡ (a_1 b_1) (mod n)

3./ Lớp tương đương

Lớp tương đương chủa một số nguyên a là tập hợp các số nguyên đồng dư với a theo modulo n.

Cho n cố định đồng dư với n trong không gian Z vào các lớp tương đương. Nếu a = qn + r, trong đó 0 ≤ r ≤ n thì a ≡ r mod n. vì vậy mỗi số nguyên a là đồng dư theo modulo n với duy nhất một số nguyên trong khoảng từ 0 đến *n-1* và được gọi là thặng dư nhỏ nhất của a theo modulo n. Cũng vì vậy, a và r cùng thuộc một lớp tương. Do đó r có thể đơn giản được dùng để thể hiện lớp tương đương.

1.1.2. Một số khái niệm trong đại số

1.1.2.1. Khái niệm nhóm, nhóm con, nhóm Cyclic

-Nhóm là bộ phận các phần tử (G, *) thõa mãn các tính chất sau:
+Tính chất kết hợp: (x * y) * z = x * (y * Z)
+Tính chất tồn tại phần tử trung gian e ∈ G: e * x=x *e =x, ∀ x ∈ G
+Tính chất tồn tại của phần tử nghịch đảo x’ ∈ G: x’ * x=x * x’ = e
-Nhóm con là bộ các phần tử (S, *) là nhóm thõa mãm các tính chất sau:
+ S ∈ G là phần tử trung gian e ∈ S
+x, y ∈ S => x * y ∈ S
-Nhóm Cyclic: là nhóm mà mọi phần tử của nó được sinh ra từ một phần tử đặc biệt g ∈ G . Phần tử này được gọi là phần tử nguyên thủy, tức:
Với ∀ x ∈ G: $\exists\, n \in \mathbb{N}$ mà $g^n = x$.
Ví Dụ: $(Z^+, *)$ là một nhóm cyclic có 1 phần tử sinh là 1.

1.1.2.2. Phần tử nghich đảo

1./ Định nghĩa

Cho a ∈ Z_n. Nghịch đảo nhân của a theo modulo n là một số nguyên x ∈ Z_n sao cho a*x $\equiv 1 \bmod n$. Nếu tồn tại, thì đó là giá trị duy nhất và a được gọi là khả nghịch. Nghịch đảo của a ký hiệu là a^{-1}.

2./ Tính chất

-Cho a, b ∈ Z_n. Phép chia của a cho b theo modulo *n* là tích của a và b^{-1} theo modulo *n* và chỉ được xác định khi *b* có nghịch đảo theo modulo *n*.
-Cho a ∈ Z_n, **a** nghịch đảo khi và chỉ khi (a,n) =1.
-Giả sử d= (a,n). Phương trình đồng dư $ax \equiv b \bmod n$ có nghiệm x nếu và chỉ nếu d chia hết cho b, trong trường hợp các nghiệm d nằm trong khoảng 0 đến n-1 thì các nghiệm đồng dư theo modulo n/d.

3./ Ví dụ:

Z_{25} = {0, 1, 2, 3,…, 24}. Trong Z_{25}: 13 + 16 = 4, vì 13 +16 = 29 $\equiv$ 4 (mod 25).

Tương tự: 13*16 = 8 trong Z_{25}.

4./ Các phép toán trong không gian modulo

Cho n là các số nguyên dương. Các phần tử trong Z_n được thể hiện bởi các số nguyên{0, 1, 2, 3,…,n-1}. Nhận xét rằng: nếu a, b $\in Z_n$ thì:

$$(a + b) \bmod n = \begin{cases} a+b & if \quad a+b < n \\ a+b-n & if \quad a+b \geq n \end{cases}$$

Vì vậy, phép cộng modulo (và phép trừ modulo) có thể được thực hiện mà không cần thực hiện các phép chia dài. Phép nhân modulo của a và b được thực hiện bằng phép nhân thông thường a với b như các số nguyên bình thường, sau đó lấy phần dư của kết quả sau khi chia cho n. Phép tính nghịch đảo trong Z_n có thể được thực hiện nhờ sử dụng thuật toán Euclid mở rộng.

1.1.2.3. Phần tử nghịch đảo trong Z_n

Định nghĩa:

Cho a $\in Z_n$, nghịch đảo của a là số nguyên b $\in Z_n$ sao cho a.b $\equiv$ 1 (mod n), ta nói b là phần tử nghịch đảo của a trong Z_n và ký hiệu là a^{-1} .

Một phần tử có phần tử nghịch đảo, gọi là khả nghịch.

Tính chất:

+ Cho a, b $\in Z_n$, a/b (mod n) = $a.b^{-1}$ (mod n) được xác định khi và chỉ khi b là khả nghịch theo modulo của n.

+ a $\in Z_n$, a là khả nghịch khi và chỉ khi gcd(a, n) = 1.

Chứng minh:

Nếu $a.a^{-1} \equiv 1 \pmod n$ thì $a.a^{-1} \equiv 1 + kn \leftrightarrow a.a^{-1} - kn = 1 \rightarrow (a,n) = 1$.

Nếu (a,n) = 1, ta có $a.a^{-1} = 1 + kn \rightarrow a.a^{-1} + kn$, do đó $a.a^{-1} \equiv 1 \pmod n$.

Ví dụ: Các phần tử khả nghịch trong Z_9 là 1, 2, 4, 5, 7 và 8.

$1^{-1} = 1$ vì $1*1 \equiv 1 \pmod 9$.

$2^{-1} = 5$ vì $2*5 \equiv 1 \pmod 9$.

$4^{-1} = 7$ vì $4*7 \equiv 1 \pmod 9$.

$8^{-1} = 8$ vì $8*8 \equiv 1 \pmod 9$.

Cho d = gcd(a,n).Khi đó phương trình đồng dư có dạng $a.x \equiv b \pmod n$ sẽ có nghiệm x khi và chỉ khi d chia hết cho b.

Tìm phần tử nghịch đảo bằng thuật toán Euclid mở rộng:

Bài toán:

+ Dữ liệu vào: $a \in Z_n$, n.

+ Kết quả: Phần tử nghịch đảo của a

Thuật toán:

Input(a,n);

Begin

$g_0 = n$; $g_1 = a$; $u_0 = 1$; $u_1 = 0$; $v_0 = 0$; $v_1 = 1$;

i = 0;

while $g_i \neq 0$

{

$y = g_{i-1} / g_i$; $g_{i+1} = g_{i+1} - y.g_i$;

$u_{i+1} = u_{i+1} - y.u_i$; $v_{i+1} = v_{i+1} - y.v_i$;

i = i +1;

}

$t = v_{i+1}$;

if t > 0 then $a^{-1} = t$ else $a^{-1} = t + n$;

End;

Ví dụ:
Tìm phần tử nghịch đảo của 3 trong Z_7

Tức là ta phải giải phương trình $3.x \equiv 1 \pmod 7$, x sẽ là phần tử nghịch đảo của 3.

i	**g_i**	**u_i**	**v_i**	**y**
0	7	1	0	
1	3	0	1	2
2	1	1	-2	3
3	0			

Vì $t = v_2 = -2 < 0$ do đó $x = x = a^{-1} = t + n = -2 + 7 = 5$.

Vậy 5 là phần tử nghịch đảo của 3 trong Z_7 .

Chú ý: Số mũ modulo có thể được tính một cách hiệu quả bằng thuật toán bình phương và nhân liên tiếp, nó được sử dụng chủ yếu trong nhiều giao thức mã hóa. Một phiên bản của thuật toán này như sau: Giả sử biểu diễn nhị phân của k là:

$$\sum_{i=0}^{1} k_i \, 2^i \quad \text{với } k_i \in \{0,1\}$$

Thuật toán bình phương liên tiếp để tính số mũ modulo trong Z_n:

Bài toán:

+ Dữ liệu vào: a $\in Z_n$ và số nguyên dương $0 \leq k \leq n$ trong đó k có biểu diễn nhị phân là:

$$k = \sum_{i=0}^{t} k_i 2^i$$

+ Kết quả: a^k mod n.

Thuật toán:

```
Readln(a,n);
Begin
    b:=1;
    if k = 0 then writeln(b);
    A:= a;
    if  k0 = 1 then b:= a;
    for i = 1 to n
        begin
            A:= A*A mod n;
            if  ki=0 then b:= A*b mod n;
        end;
    writeln(b);
End;
```

Ví dụ: Tính số mũ modulo

Bảng 2: Mô tả các bước tính 5^{596} mod (1234) = 1013

I	0	1	2	3	4	5	6	7	8	9
k_i	0	0	1	0	1	0	1	0	0	1
A	5	25	625	681	1011	369	421	799	947	925
B	1	1	625	625	67	67	1059	1059	1059	1013

1.1.2.4. Nhóm nhân Z_n^*

1/.Định nghĩa

Nhóm nhân (phép nhân) của tập Z_n ký hiệu là $Z_n^* = \{a \in Z_n \mid gcd(a,n) = 1\}$.

Đặc biệt, nếu n là một số nguyên tố thì $Z_n^* = \{a|1 \leq a \leq n-1\}$.

2/.Định nghĩa cấp của Z_n^*

Cấp của Z_n^* được định nghĩa là số phần tử trong Z_n^*, ($|Z_n^*|$). Theo định nghĩa hàm phi – Euler ta có $|Z_n^*| = \emptyset(n)$.

3/.Tính chất

Cho $n \geq 2$ là số nguyên:

+ Định lý Euler: Nếu $a \in Z_n^*$ thi $a^{\emptyset(n)} \equiv 1 \pmod n$.

+Nếu n là tích của các số nguyên tố phân biệt và nếu $r \equiv s \pmod{\emptyset(n)}$ thì $a^r \equiv a^s \pmod n$ với mọi số nguyên a. Nói cách khác, làm việc với các số theo modulo nguyên tố p thì số mũ có thể giảm theo modulo $\emptyset(n)$.

Cho p là số nguyên tố:

+ Định lý Fermat: Nếu gcd(a,p) = 1 thì $a^{p-1} \equiv 1 \pmod p$.

+ Nếu $r \equiv s \pmod{p-1}$ thì $a^r = a^s \pmod p$ với mọi số nguyên a.Nói cách khác, làm việc với các số theo modulo nguyên tố p thì số mũ có thể giảm theo modulo p-1.

+ $a^p \equiv a \pmod p$ với mọi số nguyên a.

1.1.2.5. Phần tử sinh

1/. Định nghĩa

Cho a ∈ Z_n^* , nếu cấp của α là Ø(n) ,khi đó α gọi là phần tử sinh hay phần tử nguyên thủy của Z_n^* ,và nếu Z_n^* có một phần tử sinh,thì Z_n^* được gọi là nhóm cyclic.(chú ý nếu là số nguyên tố thì Ø(n) = n-1).

2/. Tính chất:

+ Nếu α là phần tử sinh của Z_n^* thì Z_n^* = {α^i (mod n) | 0 ≤ i ≤ Ø(n)-1}.

+ Giả sử α là một phần tử sinh của Z_n^* .Khi đó , b = α^i mod n cũng là một phần tử sinh của Z_n^* khi và chỉ khi gcd(i, Ø(n)) = 1 .Và sau đó nếu Z_n^* là nhóm cyclic thì số phần tử sinh sẽ là Ø(Ø(n)).

+ α ∈ Z_n^* là phần tử sinh của Z_n^* khi và chỉ khi $\alpha^{\emptyset(n)/p}$! ≡ 1 (mod n) với mỗi số chia nguyên tố của Ø(n).

+ Z_n^* có phần tử sinh khi và chỉ khi n = 2, 4, p^k hay $2p^k$ khi p là số nguyên tố lẻ và k ≥ 1. Còn nếu p là số nguyên tố thì chắc chắn có phần tử sinh.

1.1.2.6. Thặng dư

1./ Định nghĩa

Cho a ∈ Z_n^* , a được gọi là thặng dư bậc hai theo modulo n hoặc bình phương theo modulo n, nếu tồn tại một x ∈ Z_n^* , sao cho x^2 ≡ a (mod n), và nếu không tồn tại x như vậy thì a được gọi là bất thặng dư bậc hai theo modulo n. Tập các thặng dư bậc hai ký hiệu là $\overline{Q_n}$.

Chú ý: Vì định nghĩa 0 ∉ Z_n^* nên 0 ∉ Q_n và 0 ∉ $\overline{Q_n}$.

2./ Tính chất

Cho n là tích của hai số nguyên tố p và q. Khi đó, a ∉ Z_n^* là một thặng dư bậc 2 theo modulo n khi và chỉ khi a ∉ Q_n và a ∉ $\overline{Q_n}$. Ta có:

$|Q_n| = |Q_p|.|Q_q| = (p-1)(q-1)/4$ và $|\overline{Q_n}| = 3(p-1)(q-1)/4$.

3./ Ví dụ

Cho n = 21. Khi Q_{21} = {1, 4, 16} và $\overline{Q_{21}}$ = {2, 5, 8, 10, 11, 13, 17, 19, 20

1.1.2.7. Hàm một phía và hàm một phía có cửa sập

Hàm một phía:

Một hàm một phía là hàm mà dễ dàng tính toán ra quan hệ một chiều nhưng rất khó để tính ngược lại.

Ví dụ: Biết giả thiết x thì có thể dễ dàng tính ra f(x), nhưng nếu biết f(x) thì khó tính ra được x. Trong trường hợp này "khó" có nghĩa là để tính ra được kết quả thì phải mất nhiều thời gian để tính toán.

Ví dụ: Tính $y = f(x)=a^x \bmod p$ là dễ tính nhưng tính ngược lại $x = \log_a y$ là bài toán "khó" (bài toán logarit rời rạc).

Hàm một phía có cửa sập:

F(x) được gọi là hàm một phía có cửa sập nếu tính xuôi y = f(x) thì dễ tính ngược $x= f^{-1} (y)$ thì khó tuy nhiên nếu có " cửa sập" thì vấn đề tính ngược trở nên dễ dàng. Cửa sập ở đây là một điều kiện nào đó giúp chúng ta dễ dàng tính ngược.

Ví dụ: $Y=f(x)=x^b \bmod n$ tính xuôi thì dễ nhưng tính ngược $x = y^a \bmod n$ thì khó vì phải biết a với $a * b =1 \pmod{\emptyset(n)}$ trong đó $\emptyset(n) =(p-1)*(q-1)$. Nhưng nếu biết

Cửa sập p, q thì việc tính n= p*q và tính a trở nên dễ dàng.

Hộp thư là một ví dụ khác về hàm một phía có cửa sâp. Bất kỳ ai cũng có thể bỏ thư vào thùng. Bỏ thư vào thùng là một hành động công cộng. Mở thùng thư không phải hành động công cộng. Nó là khó khăn, bạn sẽ cần đến mỏ hàn để phá hoặc những công cụ khác. Tuy nhiên, nếu bạn có " cửa sập" (trong trường hợp này là chìa khóa của hòm thư) thì công việc mở hòm thư thật dễ dàng.

1.1.3. Một số khái niệm và Độ phức tạp của thuật toán

1.1.3.1. Khái niệm thuật toán

1. Khái niệm bài toán

Bài toán được diễn đạt bằng hai phần:

- Input: Các dữ liệu vào của bài toán.
- Output: Các dữ liệu ra của bài toán (kết quả).

2. Khái niệm thuật toán

- "Thuật toán" được hiểu đơn giản là cách thức để giải một bài toán. Cũng có thể được hiểu bằng hai quan niệm: Trực giác hay Hình thức như sau:

+ Quan niệm trực giác về "Thuật toán"

Một cách trực giác, thuật toán được hiểu là một dãy hữu hạn các qui tắc (chỉ thị, mệnh lệnh) mô tả một quá trình tính toán, để từ dữ liệu đã cho (Input) ta nhận được kết quả (Output) của bài toán.

+ Quan niệm toán học về "Thuật toán"

Một cách hình thức, người ta quan niệm thuật toán là một máy Turing.

- Thuật toán được chia thành hai loại: Đơn định và không đơn định.

+ Thuật toán đơn định (Deterministic): Là thuật toán mà kết quả của mọi phép toán đều được xác định duy nhất.

+ Thuật toán không đơn định (Non - deterministic): Là thuật toán có ít nhất một phép toán mà kết quả của nó là không duy nhất.

1.1.3.2. Khái niệm độ phức tạp của thuật toán

1/. Chi phí của thuật toán (tính theo một bộ dữ liệu vào)

Chi phí phải trả cho một quá trình tính toán gồm chi phí về thời gian và bộ nhớ.

- **Chi phí thời gian** của một quá trình tính toán là thời gian cần thiết để thực hiện một quá trình tính toán.

Với thuật toán tựa Algol: Chi phí thời gian là số các phép tính cơ bản thực hiện trong quá trình tính toán .

- **Chi phí bộ nhớ** của một quá trình tính toán là số ô nhớ cần thiết để thực hiện một quá trình tính toán.

Gọi A là một thuật toán, e là dữ liệu vào của bài toán đã được mã hoá bằng cách nào đó. Thuật toán A tính trên dữ liệu vào e phải trả một giá nhất định.

Ta ký hiệu: tA (e) là giá thời gian và lA(e) là giá bộ nhớ.

2/. Độ phức tạp về bộ nhớ (trong trường hợp xấu nhất)

$l_A(n) = \max\{l_A(e), \text{với } |e| \leq n\}$. (n là kích thước đầu vào của thuật toán)

3/. Độ phức tạp thời gian (trường hợp xấu nhất) $t_A(n) = \max\{t_A(e), \text{với } |e| \leq n\}$

4/. Độ phức tạp tiệm cận

Độ phức tạp PT(n) được gọi là *tiệm cận tới hàm f(n)*, ký hiệu *O(f(n))* nếu ∃ các số n_0 , c mà PT(*n*) ư c.f(*n*) , $\forall n \geq n_o$.

5/. Độ phức tạp đa thức

Độ phức tạp PT(*n*) được gọi *đa thức*, nếu nó tiệm cận tới *đa thức p(n).*

6/. Thuật toán đa thức

- Thuật toán được gọi là *đa thức* nếu độ phức tạp về thời gian trong trường hợp xấu nhất của nó là *đa thức*.

- Nói cách khác:

+ Thuật toán thời gian đa thức: là thuật toán có độ phức tạp là $O(n^t)$ trong đó t là hằng số.

+ Thuật toán thời gian hàm mũ: là thuật toán có độ phức tạp $O(t^{f(n)})$ trong đó t là hằng số và f(n) là hàm đa thức của n.

- Thời gian chạy của các lớp thuật toán khác nhau:

Độ phức tạp	Số phép tính($n=10^6$)	Thời gian(10^6phép tính/s)
O(1)	1	1micro giây
O(n)	10^6	1 giây
$O(n^2)$	10^{12}	11,6 ngày
$O(n^3)$	10^{18}	32 000 năm
$O(2^n)$	10^{301030}	10^{301006} tuổi của vũ trụ

- Có người cho rằng ngày nay máy tính với tốc độ rất lớn, không cần quan tâm nhiều tới thuật toán nhanh, chúng tôi $_{xi}$n dẫn một ví dụ đã được kiểm chứng. Bài toán xử lý n đối tượng, có ba thuật toán với 3 mức phức tạp khác nhau sẽ chịu 3 hậu quả như sau: Sau 1 giờ:

+ Thuật toán A có độ phức tạp O(n): 3,6 triệu đối tượng.

+ Thuật toán B có độ phức tạp O(nlog n): 0,2 triệu đối tượng.

+ Thuật toán C có độ phức tạp O(2n): 21 đối tượng.

1.2. HỆ MÃ HÓA

1.2.1. Khái niệm mã hóa

Chúng ta biết rằng thông tin truyền đi trên mạng rất dễ bị trộm cắp. Để đảm bảo việc truyền tin an toàn, người ta thường mã hóa thông tin trước khi truyền đi. Việc mã hóa cần theo quy tắc nhất định

Mã hóa là kỹ thuật đã được dung lâu đời để đảm bảo an toàn thông tin. Hiện nay có nhiều phương pháp mã hóa khác nhau, mỗi phương pháp có những ưu điểm và nhược điểm riêng. Tùy theo yêu cầu cụ thể để lựa chọn phương pháp mã hóa. Sau đây là một số khái niệm dùng trong mật. mã

-Mã hóa: là quá trình chuyển đổi thông tin từ dạng đọc được gọi là **bản rõ**, thành thông tin không thể đọc được(đối với những người không có quyền) theo cách thông thường được gọi là **bản mã**,

-Giải mã: là quá trình chuyển thông tin ngược lại từ **bản mã** sang **bản rõ**.

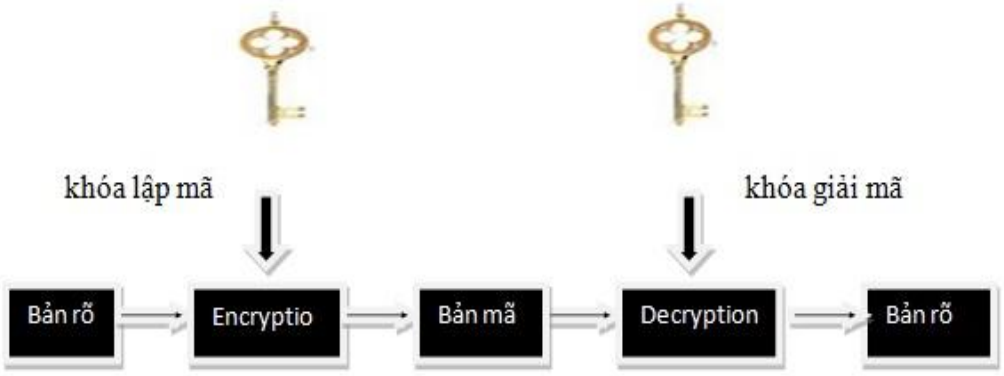
-Thuật toán mã hóa: là các thuật toán, các công thức tính toán để **mã hóa** và giải mã thông tin. Thuật toán càng phức tạp thì độ an toàn của bản mã càng cao.

-Khóa mã hóa: là các giá trị làm cho các thuật toán mã hóa chạy theo cách riêng để **mã hóa** và **giải mã**, khóa bí mật bao gồm có khóa lập mã và khóa giải mã. Phạm vi các giá trị có thể của khóa được gọi là không gian khóa. Không gian khóa càng cao thì độ an toàn của bản mã càng cao.

-Hệ mã hóa: là tập hợp các thuật toán, các khóa nhằm mã hóa, giải mã thông tin.

-Mật mã học: là ngành nghiên cứu mật mã: tạo mã và phân tích mã.

-Phân tích mã: là các kỹ thuật phân tích mã, kiểm tra tính toàn vẹn hoặc phá vỡ sự bí mật của bản mã. Phân tích còn được gọi là **thám mã**.



Hình 1: sơ đồ mã hóa

Hiện nay có hai loại mật mã: hệ mật mã khóa bí mật và hệ mật mã khóa công khai. **Hệ mật mã khóa bí mật** (còn gọi là hệ mã khóa đối xứng) dễ hiểu, dễ thực hiện thi nhưng độ an toàn không cao. Với các hệ mã khóa bí mật, nếu biết khóa lập mã hay thuật toán lập mã, người ta còn có thể tìm thấy ngay được bản rõ. Ngược lại, các **hệ mật mã khóa công khai** (còn gọi là hệ mật mã phi đối xứng) cho biết khóa lập mã K và hàm lập mã e_k, thì cũng rất khó tìm được cách giải mã. Và việc thám mã là rất khó khăn do độ phức tạp tính toán lớn.

Hệ mật mã được định nghĩa là bộ năm (P, C, K , E, D) trong đó:
P là một tập hợp hữu hạn các bản rõ có thể.
C là một tập hữu hạn các bản mã có thể.
K là một tập hữu hạn các khóa có thể.
E là tập các hàm lập mã.
D là tập các hàm giải mã.
Với mỗi k có một hàm lập mã e_k, e_k: P, và một hàm giải mã $d_k \in D$,
d_k: C sao cho: d_k (e_k(x)) =x, $\forall x \in P$.

1.2.2. Hệ mật mã khóa bí mật

Hệ mật mã khóa bí mật là hệ mật mã mà khóa mã hóa có thể dễ dàng tìm được từ khóa giải mã và ngược lại. Hệ mật mã khóa bí mật yêu cầu người gửi và người nhận phải thỏa thuận một khóa trước khi tin tức được gửi đi, khóa này phải được cất giữ bí mật. Độ an toàn của hệ này phụ thuộc vào khóa. Nếu để lộ khóa, thì bất kì người nào cũng có thể mã hóa và giải mã thông báo.

Trong mã hóa khóa bí mật, quá trình mã hóa và quá trình giải mã sử dụng cùng một thuật toán và khóa.

Độ an toàn của mã hóa khóa bí mật phụ thuộc vào một vài yếu tố như thuật toán mã hóa phải đủ mạnh (sao cho việc giải mã thông báo chỉ dựa vào bản mã là không khả thi), sự bí mật của khóa (không phải là sự bí mật của thuật toán).

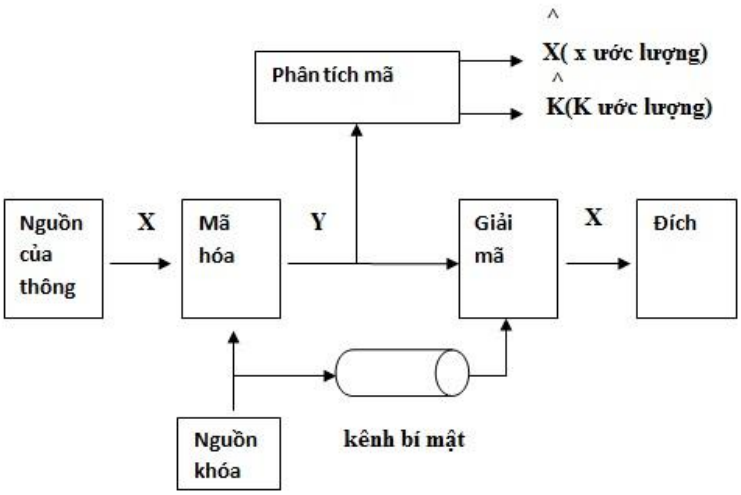
Xem lược đồ mã hóa trong hình 2. Nguồn A tạo ra một thông báo ở bảng rõ, X= {X_1, X_2, ….X_M}. Khóa được dùng khi mã hóa có dạng K ={ K_1, K_2……, K_l}. Nếu khóa do nguồn sinh ra, khóa phải được chuyển cho đích theo một kênh an toàn nào đó. Có thể dùng một thành viên thứ ba a để sinh khóa và phân phối khóa an toàn cho cả nguồn và đích.

Với đầu vào là thông báo X và khóa mã K, đầu ra của thuật toán mã hóa là một bản mã Y = { Y_1, Y_2,….,Y_n}. Chúng ta có thể viết như sau:

$$\mathbf{Y= E_k(x)}$$

Khi nhận được bản mã, người nhận có thể giải mã bằng các dùng cùng một khóa và thuật toán (dùng khi giải mã) như sau:

$$\mathbf{X=D_k(Y)}$$



Hình 2: mô hình khóa đối xứng

Việc mã hóa và giải mã thông báo nhanh và hiệu quả. Tuy nhiên, khóa phải được giữ cẩn thận. Nếu bị lộ khóa, tất cả các thông báo trước đó đều bị lộ và cả người gửi và người nhận phảo dùng khóa mới cho các cuộc truyền thông tiếp theo.

Quá trình phân phối khóa mới cho các thành viên rất khó khăn. Một vấn đề nảy sinh đối với mã hóa khóa đối xứng là chúng không thích hợp trong các môi trường lớn, chẳng hạn internet. Do mỗi cặp thành viên truyền thông trên internet phải có khóa bí mật khi họ muốn trao đổi thông tin với nhau một cách an toàn, dẫn đến số lượng khóa sẽ rất lớn, giống như hệ thống đường dây điện thoại riêng không có các trạm chuyển mạch. Với N thành viên tham gia truyền thông, chúng ta cần C^2_N khóa bí mật, chẳng hạn với 12 người muốn truyền thông, chúng ta cần 66 khóa bí mật.

1.2.3. Một số hệ mật mã cổ điển

1.2.3.1. Mã dịch chuyển:

Định nghĩa: Mã dịch chuyển: (P, C, K, E, D)

P = C = K = Z_{20} với k ϵ K

định nghĩa: $e_k(x) = (x + k) \bmod 26$ $d_k(y) = (y – k) \bmod 26$

(x, y ϵ Z_{26})

***Ví dụ*:** Dùng khoá k = 9 để mã hoá dòng thư: "toinaydichoi" dòng thư đó tương ứng với dòng số

t	o	i	n	a	y	d	i	c	h	o	i
19	14	8	12	0	24	3	8	2	7	14	8

qua phép mã hoá e_9 sẽ được:

2	23	17	22	9	7	12	17	11	16	23	17
c	x	r	w	j	h	m	r	l	q	x	r

bản mã sẽ là:

"qnwcxrcqdkjh"

Nhận được bản mã đó, dùng d_9 để nhận được bản rõ.

Cách đây 2000 năm mã dịch chuyển đã được Julius Ceasar sử dụng

Với khoá k=3 mãđịch chuyển được gọi là mã Ceasar.

Tập khoá phụ thuộc vào Z_m với m là số khoá có thể

Trong tiếng anh tập khóa chỉ có 26 khóa có thể, việc thám mã có thể được thực hiện bằng cách duyệt tuần tự 26 khóa đó, vì vậy độ an toàn của mã dịch chuyển rất thấp.

1.2.3.2. Mã thay thế:

Định nghĩa: Mã thay thế: (P, C, K, E, D)

P = C = Z_{26} , K = S (Z) Với mỗi π є K, tức là một hoán vị trên Z_{26}, ta xác định

e π(x) = π (x)

dπ(y) = π^{-1} (y)

với x, y є Z_{26} , π^{-1} là nghịch đảo của л

Ví dụ: π được cho bởi (ở đây ta viết chữ cái thay cho các con số thuộc Z_{26}):

a	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*	*l*	*m*	*n*
x	*n*	*y*	*a*	*h*	*p*	*o*	*g*	*z*	*q*	*w*	*b*	*t*	*s*

o	*p*	*q*	*r*	*s*	*t*	*u*	*v*	*w*	*x*	*y*	*z*
f	*l*	*r*	*c*	*v*	*m*	*u*	*e*	*k*	*j*	*d*	*i*

bản rõ:

"toinaydichoi"

sẽ được mã hoá thành bản mã (với khoá π):

"mfzsxdazygfz"

Dễ xác định được π^{-1}, và do đó từ bản mã ta tìm được bản rõ.

Mã thay thế có tập hợp khoá khá lớn bằng số các hoán vị trên bảng chữ cái, tức số các hoán vị trên Z_{26}, hay là 26! > 4.10^{26}. Việc duyệt toàn bộ các hoán vị để thám mã là rất khó, ngau cả đối với máy tính. Tuy nhiên, bằng phương pháp thống kê, ta có thể dễ dàng thám được các bản mã loại này, và do đó mã thay thế cũng không

dễ dàng thám được các bản mã loại này, và do đó mã thay thế cũng không thể được xem là an toàn.

1.2.3.3. Mã Anffine:

Định nghĩa: Mã Anffine: (P, C, K, E, D)

$P = C = Z_{26}$, $K = \{ (a, b) \epsilon Z_{26} \times Z_{26} : (a, 26) = 1 \}$

với mỗi $k = (a, b) \epsilon K$ ta định nghĩa:

$$e_k(x) = ax + b \bmod 26$$

$$d_k(y) = a\ (y - b) \bmod 26$$

Trong đó $x,y \epsilon Z_{26}$.

Ví dụ: Lấy k = (5, 6).

Bản rõ:

"toinaydichoi"

	t	o	i	n	a	y	d	i	c	h	o	i
x	19	14	8	13	0	14	3	8	2	7	14	8

y=5x + 6 mod 26

y	23	24	20	19	6	24	21	20	16	15	24	20
	x	y	u	t	g	y	v	u	q	p	y	u

Bản mã:

"xyutgyvuqpyu"

Thuật toán giải mã trong trường hợp này có dạng:
d k(y) = 21(y − 6) mod 26
Với mã Apphin, số các khoá có thể có bằng (số các số ≤ 26 và nguyên tố với 26) × 26tức là 12 × 26 = 312.
Việc thử tất cả các khoá để thám mã trong trường hợp này tuy khá mất thì giờ nếu tính bằng tay nhưng không khó khăn gì nếu dùng máy tính.
Do vậy, mã Apphin cũng không phải là mã an toàn.

1.2.3.4. Mã Vigenere

Định nghĩa : Mã Vigenere: (P, C, K, E, D)

Cho m là số nguyên dương.

$$P = C = K = Z26^m$$

với mỗi khoá $k = (k_1, k_2,\ldots,k_m)$ є K có:

$e_k (x_1, x_2 ,\ldots, x_m) = (x_1 + k_1 , x_2 + k_2 ,\ldots, x_m + k_m)$

$$d (y_1 , y_2 ,\ldots, y_m) = (y_1 - k_1 , y_2 - k_2 ,\ldots, y_m - k_m)$$

các phép cộng phép trừ đều lấy theo modulo 26

Ví dụ: Giả sử m = 6 và khoá k là từ CIPHER - tức k=(2, 8, 15, 7, 4, 17).

Bản rõ:

"toinaydichoi"

	t	o	i	n	a	y	d	i	c	h	o	i
x	19	14	8	13	0	24	3	8	2	7	14	8
k	2	8	15	7	4	17	2	8	15	7	4	17
y	21	22	23	20	4	15	5	16	17	14	18	25
	v	w	x	u	e	p	f	q	r	o	s	z

Bản mã:

"vwxuepfqrosz"

Từ bản mã đó, dùng phép giải mã d_k tương ứng, ta lại thu được bản rõ.

Chú ý: Mã Vigenere với m = 1 sẽ trở thành mã Dịch chuyển.

Tập hợp các khoá trong mã Vigenere mới $m \geq 1$ có tất cả là 26^m khoá có thể có.

Với m = 6, số khoá đó là 308.915.776, duyệt toàn bộ chừng ấy khóa để thám mã bằng tính thì khó, nhưng với máy tính thì vẫn là điều dễ dàng.

1.2.3.5. Mã Hill:

Định nghĩa Mã Hill: (P, C, K, E, D)

Cho m là số nguyên dương.

P = C = *Z26m*

K = { k ϵ *Z26mxm* : (det(k), 26) = 1 }

với mỗi k ϵ K định nghĩa:

e $(x_1 , x_2 ,\ldots, x_m) = (x_1 , x_2 ,\ldots, x_m).k$

$d_k (y_1 , y_2 ,\ldots, y_m) = (y_1 , y_2 ,\ldots,y_m).k^{-1}$

Ví dụ: Lấy m = 2, và k = $\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$

Với bộ 2 ký tự (x_1, x_2), ta có mã là $(y_1, y_2) = (x_1 , x_2).k$ được tính bởi

$$Y_1=11.x_1+3.x_2$$

$$Y_2=8.x_1+7.x_2$$

Giả sử ta có bản rõ: "**tudo**", tách thành từng bộ 2 ký tự, và viết dưới dạng số ta được 19 20 | 03 14.

Lập bản mã theo quy tắc trên, ta được bản mã dưới dạng số là:

09 06 | 23 18, và dưới dạng chữ là **" fgxs".**

Chú ý:

Để đơn giản cho việc tính toán, thông thường chọn ma trận vuông 2x2. Khi đó có thể tính ma trận nghịch đảo theo cách sau:

Giả sử ta có $k = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

Ta có ma trận nghịch đảo

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} \frac{d}{ad-bc} & \frac{-b}{ad-bc} \\ \frac{-c}{ad-bc} & \frac{a}{ad-bc} \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Và được tính như sau

$$e_k(x_1, x_2, \ldots, x_m) = (x_{\pi(1)}, x_{\pi(2)}, \ldots, x_{\pi(m)})$$

$$d_k(y_1, y_2, \ldots, y_m) = (y_{\pi^{-1}(1)}, y_{\pi^{-1}(2)}, \ldots, y_{\pi^{-1}(m)})$$

Một chú ý là để phép chia luôn thực hiện được trên tập Z_{26} thì nhất thiết định thức của k: det(k) = (ad-bc) phải có phần tử nghịch đảo trên Z_{26}, nghĩa là (ad-bc) phải là một trong các giá trị: 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23 hoặc 25. Đây cũng là điều kiện để ma trận k tồn tại ma trận nghịch đảo.

Khi đó: k^{-1} .k = I là ma trận đơn vị (đường chéo chính bằng 1)

Định thức của $\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$

Là 11*7 – 8*3 = 1 ≡ 1 mod 26

Khi đó

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \begin{pmatrix} \frac{d}{ad-bc} & \frac{-b}{ad-bc} \\ \frac{-c}{ad-bc} & \frac{a}{ad-bc} \end{pmatrix}$$

1.2.3.6. Mã hoán vị:

Định nghĩa: Mã hoán vị: (P, C, K, E, D) :Cho m là số nguyên dương.

$$P=C=Z_{26} \quad ,K=S_m$$

với mỗi $k = \pi \epsilon S_m$, ta có

$$\begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}^{-1} = \begin{pmatrix} 7 & -8 \\ -3 & 11 \end{pmatrix} \equiv \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix} \ mod\ 26$$

trong đó π^{-1} là hoán vị nghịch đảo của π

Ví dụ: Giả sử m = 6, và khoá k được cho bởi phép hoán vị π

1	2	3	4	5	6
3	5	1	6	4	2

Khi đó phép hoán vị nghịch đảo π^{-1} là:

1	2	3	4	5	6
3	6	1	5	2	4

Bản rõ:

"toinaydichoi"

	t	o	i	n	a	y	d	i	c	h	o	i
vt	1	2	3	4	5	6	1	2	3	4	5	6
π	1->3	2->5	3->1	4->6	5->4	6->2	1->3	2->5	3->1	4->6	5->4	6->2
vt	3	5	1	6	4	2	3	5	1	6	4	2
	i	a	t	y	n	o	c	o	d	i	h	i

Bản mã: "iatynocodihi"

Dùng hoán vị nghịch đảo, từ bản mật mã ta lại thu được bản rõ.

Chú ý:

Mã hóa vị là một trường hợp riêng của mã Hill. Thực vậy, cho phép hoán vị π của {1, 2,….., m} ta có thể xác định ma trận $K_\pi = (k_{ij})$, với

$$k_{ij} = \begin{cases} 1 & \text{nếu} \quad i = \pi(j) \\ 0 & \text{nếu ngược lại} \end{cases}$$

Thì dễ thấy rằng mã Hill với khoá K_π trùng với mã hoán vị với khoá π.

Với m cho trước, số các khoá có thể có của mã hoán vị là m!

Dễ nhận thấy với m = 26 ta có số khóa 26! (mã Thay thế).

1.2.4. Hệ mã hóa khóa công khai

Mã hóa công khai có ưu điểm là số lượng khóa không lớn, nếu có N người muốn trao đổi thông tin với người khác an toàn thì chỉ cần duy nhất N cặp khóa, ít hơn rất nhiều so với mã hóa đối xứng. Ưu điểm thứ hai của việc phân phối khóa không phải là một vấn đề khó. Khóa công khai của mỗi người có thể được gửi đi theo kênh an toàn nếu cần thiết và không yêu cầu bất kỳ sự kiểm soát đặc biệt nào khi phân phối. Mã hóa công khai có khả năng thực thi chữ ký số, có nghĩa là một tài liệu có thể được ký và gửi cho người nhận bất kỳ cùng với chống chối bỏ. thực tế khó có một người nào đó khác ngoài người ký- sinh ra chữ ký điện tử , thêm vào đó, người ký không thể chối bỏ việc ký tài liệu sau khi đã ký.

Mã hóa công khai có một số khó khăn là: Quá trình mã hóa và giải mã chậm so với mã hóa đối xứng. khoảng thời gian này tăng lên một cách nhanh chóng nếu bạn và khách hàng của bạn tiến hành thương mại trên internet. Người ta không có ý định thay thế mã hóa đối xứng bằng mã hóa công khai. Chúng bổ xung lẫn nhau.

Các thuật toán công khai dùng một khóa để mã hóa và khóa khác để giải mã. Chúng có tính chất quan trọng là không thể xác định được khóa giải mã nếu chỉ nếu căn cứ vào thuật toán và khóa mã hóa.

Quá trình mã hóa công khai gồm các bước cơ bản sau:

Mỗi thành viên có một cặp khóa, khóa này dùng để mã hóa và giải mã.

Mỗi thành viên công bố khóa mã hóa của mình bằng cách đặt khóa này vào một dịa chỉ được công bố công khai. Đây chính là khóa công khai. Khóa cùng cặp được giữ bí mật và đó là khóa riêng.

Nếu A muốn gửi cho B một thông báo, B giải mã thông báo bằng khóa riêng của B. không một người nhận nào khác có thể giải mã thông báo, bởi chỉ B mới biết khóa riêng của mình.

Với cách giải quyết này, tất cả thành viên tham gia truyền thông đều có thể có được các khóa công khai. Khóa riêng của mỗi thành viên được giữ bí mật. quá trình liên lạc chỉ an toàn chừng nào khóa riêng còn được giữ bí mật. mỗi thành viên có thể thay đổi các khóa riêng của mình bất cứ lúc nào, đồng thời công bố khóa công khai cùng cặp để thay thế khóa cũ.

Một đặc điểm cũa mã hóa khóa công khai là khóa công khai được công bố công khai và khoá riêng cùng cặp được chủ sở hữu giữ bí mật. Do được công bố công khai, bất cứ người dùng nào cũng có thể lấy được khóa công khai khi muốn dùng nó, nhưng cần phải dùng một cơ chế nào đó để xác thực rằng, khóa công khai đó chính là người gửi thông báo hoặc người nhận thông báo chủ định, và khóa công khai này cùng cặp với khóa riêng của họ.

Vấn đề phân phối khóa công khai được giả quyết qua nhiề ngươi kỹ thuật phân phối khóa công khai như khai báo công khai, thư mục công khai, trung tâm quản lý khóa công và chứng chỉ khóa công khai.

Hiện nay người ta chủ yếu dùng hệ thống chứng chỉ khóa công khai để phân phối khóa công khai. Mỗi chứng chỉ có chứa một khóa công khai và các thông tin khác. Nó chuyển thông tin khóa công khai của mình cho các thành viên khác thông qua chứng chỉ. Các thành viên khác có thể kiểm tra chứng chỉ do cơ quan quản lý tạo ra.

1.2.4.1. Mã RSA:

Hệ mật này sử dụng tính toán trong Z_n, trong đó n là tích của 2 số nguyên tố p hân biệt p và q. Ta thấy rằng $\varphi(n) = (p - 1).(q - 1)$.

Định nghĩa:

Cho n = p.q trong đó p và q là các số nguyên tố. Đặt P = C = Zn và định nghĩa:

K = {(n, p, q, a, b): n = p.q; p, q là các số nguyên tố,

$a.b \equiv 1 \bmod \varphi(n)$}

Với K = (n, p, q, a, b) ta xác định: $eK = x^b \bmod n$

Và $dK = y^a \bmod n$

(x, y ∈ Zn) Các giá trị n và b được công khai và các gia trị p, q, a được giữ kín

Ví dụ:

Chọn p = 2, q = 5. Tính n = p.q = 2*5 = 10

$\varphi(n)= (p - 1).(q - 1) = 1*4 = 4$

Do UCLN($\varphi(n)$, b) = 1 nên chọn b = 3

$a.b \equiv 1 \bmod \varphi(n)$ nên chọn a = 7

Giả sử G muốn gửi bản rõ x = 3 tới N, G phải tính:

$y = eK = x^b \bmod n = 3^3 \bmod 10 = 7$

Khi N nhận được bản mã y = 1, anh ta sử dụng số mũ a mật để tính:

$x = dK = y^a \bmod n = 7^7 \bmod 10 = 3$

Đó chính là bản rõ mà G đã mã hoá.

Độ mật của hệ RSA được dựa trên giả thiết là hàm mã $eK = x^b \bmod n$ là hàm một chiều. Bởi vậy thám mã sẽ khó có khả năng về mặt tính toán để giải mã một bản mã. Cửa sập cho phép N chính là thông tin về phép phân tích thừa số n (n = p.q). Vì N biết phép phân tích này nên anh ta có thể tính $\varphi(n) = (p - 1).(q - 1)$ và rồi tính số mũ giải mã a bằng cách sử dụng thuật toán Eculide mở rộng.

1.2.4.2. Mã Elgamal:

Hệ mật mã ELGamal được T. ElGamal đề xuất năm 1985, dựa vào bài toán tính looogarit rời rạc và sau đó đã nhanh chóng được sử dụng rộng rãi không những trong vấn đề bảo mật truyền tin mà còn trong các vấn đề xác nhận và chữ ký điện tử.

Bài toán logarithm rời rạc trong Zp là đối tượng trong nhiều công trình nghiên cứu và được xem là bài toán khó nếu p được chọn cẩn thận. Cụ thể là không có một thuật toán thời gian đa thức nào cho bài toán logarithm rời rạc. Để gây khó khăn cho các phương pháp tấn công đã biết, p phải có ít nhất 150 chữ số và (p-1)phải có ít nhất một thừa số nguyên tố lớn.

Bài toán logarithm rời rạc trong Zp:

Đặc trưng của bài toán: I = (p, α, β) trong đó p là số nguyên tố, α ∈ Zp là

phần tử nguyên thuỷ (hay phần tử sinh), β ∈ Zp*

Mục tiêu: Hãy tìm một số nguyên duy nhất a, $0 \leq a \leq p - 2$ sao cho:

$\alpha^a \equiv \beta \pmod p$

Ta sẽ xác định số nguyên a bằng log α β.

Định nghĩa mã khóa công khai Elgamal trong Zp*:

Cho p là số nguyên tố sao cho bài toán logarithm rời rạc trong Zp là khó giải

Cho α ∈ Zp* là phần tử nguyên thuỷ. Giả sử P = Zp*, C = Zp* x Zp*.

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

Các giá trị p, α, β được công khai, còn a giữ kín.

Với K =(p, α, a, β) và một số ngẫu nhiên bí mật k ∈ Zp – 1, ta xác định:

eK(x, k) = (y1, y2).

Trong đó: $y1 = \alpha^k \bmod p$

$y2 = x. \beta^k \bmod p$

Với y1, y2 ∈ Zp* ta xác định: $dK(y1, y2) = y2(y1a)^{-1} \bmod p$

***Ví dụ*:**

Chọn p = 7

$\alpha \in Zp^*$ là phần tử nguyên thuỷ nên $\alpha = 3$

Chọn a sao cho $0 \leq a \leq p - 2$ nên a = 2

Khi đó : $\beta = \alpha^a \bmod p = 3^2 \bmod 7 = 2$

Chọn một số ngẫu nhiên bí mật $k \in Zp - 1$, chọn k =3

Giả sử G muốn gửi thông báo x = 3 cho N, G phải tính:

eK(x, k) = (y1, y2)

trong đó:

$y1 = \alpha^k \bmod p = 3^3 \bmod 7 = 6$

$y2 = x.\ \beta^k \bmod p = 3*2^3 \bmod 7 = 3$

Khi N thu được bản mã (y1, y2) = (6, 3), anh ta sẽ tính:

$x = dK(y1, y2) = y2(y1a)^{-1} \bmod p = 3*(6^2)^{-1} \bmod 7 = 3$

Đó chính là bàn rõ mà G đã mã hoá.

1.3. CHỮ KÝ SỐ

1.3.1. Khái niệm chữ ký số

Chữ ký điện tử (Digital signature) là thông tin đi kèm theo dữ liệu (văn bản, hình ảnh video….) nhằm mục đích xác định người chủ của dữ liệu đó. Chữ ký số là nền tảng bảo đảm an toàn an ninh cho nền kinh tế tri thức, đồng thời là cơ sở pháp lý để giải quyết tranh chấp.

Lược đồ chữ ký số là phương pháp ký một thông điệp lưu dưới dạng điện tử. Và thông điệp được ký này có thể truyền trên mạng.

Với chữ ký truyền thống, khi ký lên một tài liệu thì chữ ký là bộ phận vật ký của tài liệu được ký. Tuy nhiên, chữ ký số không gắn một cách vật lý với thông điệp được ký.

Để kiểm chữ ký đối với chữ ký truyền thống việc kiểm tra bằng cách so sánh nó với những chữ ký gốc đã đăng ký. Tất nhiên, phương pháp này không an toàn lắm vì nó tương đối dễ đánh lừa bởi chữ ký người khác. Trong khi chữ ký số thì được kiểm tra bằng cách dùng thuật toán để kiểm tra đã biết. Như vậy " người bắt kì" có thể kiểm tra chữ kí số. việc sử dụng lược đồ an toàn sẽ ngăn chặn khả năng đánh lừa (giả mạo chữ ký).

Chữ ký điện tử phải đáp ứng yêu cầu:

+ Chứng thực: chữ ký thuyết phục được người nhận rằng văn bản chứa nó là do người ký gửi đến.

+ Chống giả mạo: chữ ký là bằng chứng cho việc người ký đã ký lên, bởi không ai có thể giả mạo chữ ký cỉa người ký.

+Chống tái sử dụng: chữ ký không chỉ đặc trưng cho người ký mà còn cả văn bản chứa nó, người ta không thể di chuyển chữ ký vào một tài liệu khác với vai trò như chữ ký hợp pháp của văn bản ấy.

+Chống thay đổi văn bản: sau khi văn bản được ký, nó không thể bị sửa đổi vì mọi sự sửa đổi đều dẫn đến chữ ký không hợp lệ.

+Chống phủ nhận: người ký không thể phủ nhận chữ ký của mình trên văn bản.

Một sơ đồ chữ ký số là bộ 5 (P,A, K,S,V) thõa mãn các điều kiện sau đây:

P: tập hữu hạn các thông điệp
A: tập hữu hạn các chữ ký
K: tập hữu hạn các khóa
với k thuộc K tồn tại một thuật toán kí: $sig_k \in B$ và thuật toán xác minh $ver_k \in V$.
Mỗi sig_k: P-> A và ver_k: PxA-> {true, false} là những hàm sao cho mỗi bức điện x $\in P$ và mỗi chữ ký y $\in A$ thõa mãn phương trình dưới đây:

$$\mathbf{Ver(x,y)}=\begin{cases} \boldsymbol{True} : \boldsymbol{nếu\ y = sig(x)} \\ \boldsymbol{False} : \boldsymbol{nếu\ y \neq sig(x)} \end{cases}$$

1.3.2. Quá trình tạo ra chữ ký điện tử

1) Tạo một câu ngắn gọn để nhận dạng – ví dụ như “Tôi là sinh viên”.

2) Mã hóa nó bằng khóa bí mật của mình tạo ra chữ ký điện tử.

3) Gắn chữ ký điện tử vào thông điệp cần gửi rồi mã hóa toàn bộ bằng khóa công khai của người nhận.

4) Gửi thông điệp đi.

5) Người nhận sẽ dùng khóa bí mật của mình để giải mã thông điệp và lấy chữ ký ra sau đó họ sẽ giải mã chữ ký này bằng khóa công khai của người gửi.Chỉ người nào gửi có khóa bí mật phù hợp mới có thể tạo ra chữ ký mà người nhận giải mã thành công. Do đó người nhận có thể định danh người gửi.

1.3.3. Hàm băm sử dụng trong chữ ký điện tử

Môt thông điệp được đưa qua hàm băm sẽ tạo ra một giá trị có độ dài cố định và ngắn hơn được gọi là "đại diện" hay "bản tóm tắt".Mỗi một thông điệp đi qua một hàm băm chỉ có duy nhất một đại diện và ngược lại. Rất khó để tìm được hai thông điệp khác nhau nào có cùng một đại diện khi đi qua một hàm băm.

Hàm băm thường kết hợp với chữ ký điện tử ở trên để tạo ra một loại chữ ký điện tử vừa an toàn hơn (không thể cắt / dán) vừa có thể kiểm tra tính toàn vẹn của thông điệp.Các bước để tạo ra chữ ký điện tử như sau:

1) Đưa thông điệp cần gửi qua hàm băm tạo ra đại diện cho thông điệp đó.

2) Mã hóa đại diện bằng khóa bí mật của người gửi để tạo ra chữ ký điện tử.

3) Mã hóa toàn bộ thông điệp và chữ ký bằng khóa công khai của người nhận và gửi đi.

Người nhận sẽ giải mã thông điệp bằng khóa bí mật của mình, gải mã chữ ký bằng khóa công khai của người gửi để lấy đại diện ra. Sau đó cho thông điệp qua hàm băm để tạo lại đại diện của thông điệp rồi so sánh với đại diện nhận được: Nếu giống nhau thì người nhận có thể vừa định danh người gửi vừa kiểm tra tính toàn vẹn của thông điệp.

**Một số hàm băm thường gặp:*

- MD5(Message Digest): 128 bit, nhanh, được sử dụng rộng rãi.
- SHA (Secure Hash Algorithm): 160 bit.

1.3.4. Chữ ký RSA

Sơ đồ chữ ký RSA (đề xuất 1987)

Chọn p,q là số nguyên tố lớn.

Tính n= p.q; $\emptyset(n) = (p-1)(q-1)$.

Chọn b là số nguyên tố cùng nhau với $\emptyset(n)$.

Chọn a nghịch đảo với b; a= b^{-1}mod $\emptyset(n)$.

Ký trên x: Sig(x): x^a mod n

Kiểm tra chữ ký: Ver(x,y) =True ⇔ x ≡ yb mod n.

Ví dụ:

Chữ ký: y =x^a mod n = 2^3 mod 15=8; chọn b=3; a=3;

Ký x=2:

Chữ ký: y = x^a mod n = 8^3 mod 15= 2 (chữ ký đúng).

1.3.5. Chữ ký ElGamal

Chọn p là số nguyên tố sao cho bài toán lagarit rời rạc trong Z_p là khó.

Chọn g là phần tử sinh ngẫu nhiên $\in Z_p^*$. Tính $\beta \equiv g^a \bmod p$.

Chọn r ngẫu nhiên $\in Z_{p-1}^*$.

Ký trên x: Sig(x)= (u, v).

Trong đó: $u = g^r \bmod p$, $\delta = (x - a\,v)\ r^{-1} \bmod (p-1)$.

Kiểm tra chữ ký: Ver (x, u, v)= True $\Leftrightarrow \beta^u\ u^v \equiv g^x \bmod p$.

Ví du:

Chọn p =463; g= 2; a =211; $\beta \equiv 2^{211} \bmod 463 = 249$; chọn r= 235; $r^{-1} = 289$

Ký trên x: x =112

Sig(x,r)= sig(112, 235)= (u,v) = (16,108).

u $= 2^{235} \bmod 463 = 16$

v = (112-221*16)*289 mod (463-1)=108.

Kiểm tra chữ ký:

Ver (x, u, v)= True $\Leftrightarrow \beta^u\ u^v \equiv g^x \bmod p$.

$\beta^u\ u^v \equiv 249^{16} * 16^{108} \bmod 463 = 132$.

$g^x \bmod p = 2^{112} \bmod 463 = 132$.

1.3.6. Chữ ký Schnorr (chữ ký một lần)

Sơ đồ chữ ký dùng một lần (one-time signature) là một khái niệm vẫn còn khá mới mẻ song rất quan trọng, đặc biệt là trong một số mô hình về tiền điện tử.

Với sơ đồ chữ ký dùng một lần Schnorr, những người dùng trong cùng hệ thống có thể chia sẻ một số ngẫu nhiên *g* và hai số nguyên tố p và q sao cho:

$q|(p-1)|$, $q \neq 1$ và $g^q \equiv 1 \bmod q$. Sơ đồ chữ ký như sau:

-Lấy G là nhóm con cấp q của Z_n^* với q là số nguyên tố.

-Chọn phần tử sinh $g \in G$ sao cho bài toán gogarit trên G là khó giải.

-Chọn $x \neq 0$ làm khóa bí mật.

-Tính $y = g^x$ làm khóa công khai.

-Lấy H là hàm băm không va chạm.

-Ký:

Chọn r ngẫu nhiên thuộc Z_q.

Tính $c = H(m,g^r)$

Tính $s = (r- c.x) \bmod q$

Chữ ký Shnorr là cặp (c, s)

-Kiểm tra chữ ký:

Với một văn bản m cho trước, một cặp (c, s) được gọi là một chữ ký Schnorr hợp lệ nếu thõa mãn phương trình:

c= H(m,g^s*y^c).

-Để ý rằng ở đây, c xuất hiện cả 2 vế phương trình.

1.3.7. Các loại chữ ký khác

1.3.7.1. Chữ ký đồng thời

Ở đây, chữ ký không phải là của một người mà là của một nhóm người. Muốn tạo được chữ ký, tất cả những người này phải tham gia vào một giao thức

(protocol). Tuy nhiên chữ ký có thể được kiểm định bởi bất cứ ai. Đây là trường hợp dành cho thực tế của việc đưa ra những quyết định của nhiều người.

1.3.7.2. Chữ ký ủy nhiệm

Hệ chữ ký này dành cho các trường hợp mà người chủ chữ ký bị ốm không có khả năng làm việc hay đã đi vắng đến một nơi không có phương tiện máy tính cần thiết để ký. Vì vậy chữ ký ủy nhiệm được tạo ra để người ký có thể ủy nhiệm cho một người nào đó ký thay. Tất nhiên chữ ký ủy nhiệm phải có thuộc tính riêng thêm vào:

Chữ ký ủy nhiệm là phần phân biệt với chữ ký thường, và người được ủy nhiệm không thể tạo được chữ ký chủ (chữ ký của người chủ).

Chữ ký ủy nhiệm cũng có chức năng chứng thực như chữ ký chủ, chỉ có người chủ và người được ủy nhiệm mới có thể tạo được chữ ký này. Người nhận được văn bản có thể hoàn toàn tin tưởng vào chữ ký đó như chữ ký chủ.

Người chủ có thể xác định được danh tính người ký từ một chữ ký ủy nhiệm.

Người được ủy nhiệm không thể chối cãi được nếu đã ký một văn bản ủy nhiệm hợp lệ (tức là anh ta không thể chối bỏ đổ cho ai khác hay chính người chủ đã ký mà lại nói anh ta ký).

1.3.7.3. Chữ ký không thể phủ nhận (chống chối bỏ)

Chữ ký không thể phủ nhận do David Chaum và Hán Van Antwerpen phát minh năm 1989. Ở đây, thuật toán kiểm định đòi hỏi phải có sự tham gia của người ký.

Thực chất đây là chữ ký có tính chất không thể chuyển giao được (Untranferable): chỉ có ý nghĩa với người nhận là người trao đổi làm ăn với người ký, khi chuyển nó cho một người khác thì không có tác dụng nữa (không thể kiểm định được chữ ký nữa). Các văn bản có chữ ký này không nhằm vào mục đích đem đi công bố ở nơi khác mà chỉ có tính chất giấy phép. Vì thế nếu sao chép là mất ý nghĩa.

Chữ ký không thể phủ nhận được dùng trong việc bán sản phẩm mềm: các hàng mềm sẽ bán các sản phẩm của mình có chữ ký chứng tỏ bản quyền. Việc kiểm định đòi hỏi phải liên lạc với hàng này. Nếu như có một ai đó bán phần mềm sao chép thì lúc đó người mua đòi kiểm định sẽ bị lộ ngay vì không thực hiện được.

Chương 2. CHỮ KÝ "MÙ" VÀ ỨNG DỤNG

2.1. CHỮ KÝ " MÙ"

2.1.1. Khái niệm

Chữ ký mù được Chaum giới thiệu vào năm 1983. Chữ ký mù là để người ký tạo ra chữ ký trên một văn bản mà chính người ký cũng không biết nội dung – không biết nội dung nhưng vẫn tạo được chữ ký hợp lệ. Đặc trưng của nó là: Chỉ có duy nhất người chủ của chữ ký mới có khả năng tạo ra chữ ký hợp lệ cho một văn bản và chữ ký cho một văn bản đó có thể được kiểm tra tính đúng đắn bởi bất cứ ai. Chữ ký mù được áp dụng trong kỹ thuật bỏ phiếu từ xa…

Giả sử Alice muốn mua quyển sách Q với giá 50$ từ Bob. Giả sử hai người cùng dùng dịch vụ của một ngân hàng. Giao thức giao dịch gồm ba giai đoạn sau:

-Rút tiền:

+Alice tạo tiền điền tử C (với thông tin: số serial, giá trị của C, ví dụ 50$).

Alice yêu cầu ngân hàng ký "mù" lên C.

+Giao thức ký thành công, thì ngân hàng sẽ trừ 50$ trong tài khoản Alice.

-Tiêu Tiền (Spending):

+Alcie đưa C đã ký của ngân hàng cho Bob và yêu cầu quyển sách Q.

+Bob kiểm tra chữ ký C, nếu chữ ký không hợp lệ thì Bob kết thúc giao thức.

-Gửi tiền (Deposit):

+Bob lấy C từ Alice và gửi cho ngân hàng.

+Ngân hàng xác thực chữ ký trên C.

Nếu chữ ký là hợp lệ, ngân hàng sữ kiểm tra xem C đã được tiêu trước đó chưa.

Nếu C chưa được tiêu, thì ngân hàng sẽ cộng thêm tiền vào tài khoản của Bob.

Nếu việc gửi tiền thành công, Bob gửi C vào tài khoản của mình, ngân hàng cũng khóa thể biết đồng tiền đó nhận từ Alice vì nó đã được ký "mù". Như vậy tiền điện tử C không lưu vết của những ai đã "tiêu" nó.

Khi ký mù lên văn bản x, các bước được tiến hành như sau:

1. **Làm mù x:** A làm mù x bằng một hàm: z=X(x) và gửi z cho B.
2. **Ký:** B ký z trên z bằng hàm y = Sign (z) = Sign (Blind(x)) và gửi lại y cho A.
3. **Xóa mù**: A tiến hành xóa mù trên Y bằng hàm

2.1.2. CHỮ KÝ "MÙ" DỰA TRÊN CHỮ KÝ RSA

2.1.2.1. Chữ ký RSA

Sơ đồ:

Bài toán đặt ra giả sử A muốn lấy chữ ký của B trên x nhưng không muốn cho B biết x. Quá trình thực hiện được tiến hành như sau:

Lấy p,q là các số nguyên tố lớn, n= p.q; $\emptyset(n) = (p-1)(q-1)$, ab = 1mod $\emptyset(n)$, r là một số ngẫu nhiên $\in Z_n$.

1/ **làm mù x**: A làm mù x bằng một hàm: **Blind(x)** = $x*r^b \bmod n$ =**z** và gửi z cho B. r được chọn sao cho tồn tại phần tử nghịch đảo $r^{-1} \pmod n$

2/ **ký:** B ký trên z bằng hàm **Sign(z)** = **Sign(Blind(x))** = $z^a \bmod n$ = **y** và gửi lại y cho A.

Ví Dụ:

Sơ đồ chữ ký mù RSA với K = (n, p, q, r, b, a) trong đó:

p = 3, q = 5, n = p * q = 15

$\emptyset(n)$ = (p -1) * (q – 1) = 8

Chọn khóa công khai b = 3 < $\emptyset(n)$, nguyên tố cùng nhau với $\emptyset(n)$.

Khóa bí mật a là phần tử nghịch đảo của b theo mod $\emptyset(n)$:a = 3.

Giả sử thông điệp cần ký là x = 2.

- Ông A chọn số ngẫu nhiên r = 4 nguyên tố cùng nhau với n. A che dấu định danh x bằng bí danh u:

u = Blind(x) = $x.r^b \pmod n = 2 * 4^3 \pmod{15} = 8$.

- A gửi bí danh u cho B, nhận được chữ ký v:

v = Sig(u) = $u^a \bmod n = 8^3 \pmod{15} = 2$

- Sau khi nhận được v, A xóa mù trên v sẽ nhận được chữ ký trên định danh x:

Unblind(v) = $v * r^{-1} \bmod n = 2 * 4^{-1} \pmod{15} = 8$.

2.2. ỨNG DỤNG CỦA CHỮ KÝ "MÙ"

2.2.1. Ứng dụng trong bỏ phiếu trực tuyến

2.2.1.1. Bài toán

Bài toán bỏ phiếu điện tử cho công dân nước Việt Nam bỏ phiếu về việc đồng ý hay không đồng ý về một dự án sắp được ban hành, hay cuộc bỏ phiếu lựa chọn " 1 trong số n người" vào một vị trí nào đó

Trước tiên ban bầu cử phải giới thiệu , đưa được ra các thông tin về cuộc bỏ phiếu để cho các cử tri đọc và tìm hiểu. Sau khi tìm hiểu xong thì cử tri mới tiến hành bỏ phiếu. Công việc bỏ phiếu gồm 3 giai đoạn chính: Giai đoạn cư tri (CT) đi đăng kí để có quyền bỏ phiếu, giai đoạn 2 là bỏ phiếu và giai đoạn ba là kiểm phiếu.

Giai đoạn 1: Cử tri đăng kí để có quyền bỏ phiếu.

Khi cử tri đến đăng kí, thì cử tri phải gửi chứng minh thư nhân dân (CMT) cho ban bầu cử để kiểm tra xem có đủ điều kiện để bỏ phiếu không.

Để đăng kí quyền bỏ phiếu thì CT phải chọn một định danh cho lá phiếu của mình (vì mỗi lá phiếu phải có thông tin định danh như: họ tên , số CMT…) , nhưng để không bị lộ danh tính, lá phiếu cần ẩn danh để không bị người ngoài theo dõi, thì CT sẽ làm mù (mã hóa) định danh đó đi, sau đó sẽ gửi bí danh (định danh đã được làm mù) đến cho ban bầu cử .

Ban bầu cử tiếp tục kiểm tra bí danh xem có hợp lệ hay không (hợp lệ là: định danh có bị trùng so với cử tri trước đó), nếu hợp lệ thì ban bầu cử lưu các thông tin vào sổ đăng kí đồng thời kí lên bí danh, và gửi lại cho cử tri .Việc lưu lại bí danh ,CMT vào sổ đăng kí để kiểm tra những lần đăng kí sau để tránh tình trạng một cử tri bỏ phiếu 2 lần, hay hai cử tri có định danh trùng nhau.

Nếu bí danh của cử tri bị trùng nhau (có thể do định danh bị trùng, cũng có thể do quá trình làm mù khiến cho bí danh bị trùng) thì ban bầu cử sẽ yêu cầu cử tri chọn lại định danh của mình.

Khi cử tri nhận được chữ ký của ban bầu cử trên bí danh thì cử tri sẽ tiến hành xóa mù trên bí danh đó và nhận được chữ ký của ban bầu cử trên định danh thật của mình. Chữ ký này sẽ được cử tri sử dụng trong quá trình bỏ phiếu.

Giai đoạn 2: Cử tri lựa chọn và ghi thông tin vào lá phiếu của mình. Để không bị lộ thông tin về bỏ phiếu. Cử tri mã hóa nội dung lá phiếu, sau đó gửi kèm theo với định danh thật, và chũ ký của ban bầu cử (đã được xóa mù) đến ban kiểm phiếu.

Giai đoạn 3: Ban bỏ phiếu kiểm tra từng lá phiếu xem có hợp lệ hay không (kiểm tra cặp chữ ký và định danh xem có tương ứng với nhau không) nếu lá phiếu không hợp lệ thì trả lại cho cử tri.

2.2.1.2. Ứng dụng chữ ký mù trong bỏ phiếu điện tử

Trên mỗi lá phiếu của cử tri (trong bỏ phiếu điện tử) phải có một số làm định danh. Để thõa mãn tính nặc danh như trong bỏ phiếu truyền thống, thì trong bỏ phiếu điện tử, chúng ta phải làm mù định danh của cử tri.

Cử tri x chọn một số ngẫu nhiên x_i đủ lớn làm định danh của mình. Vì x_i tạo ngẫu nhiên nên nó sẽ không liên quan đến gì với cử tri x. Khi cử tri x trình giấy tờ hợp lệ thì ban bầu cử sẽ ký lên bí danh x_i của anh ta (định danh đã được làm mù để tránh tiết lộ thông tin bằng cách biến đổi x_i thành $z_i = blind(x_i)$) trước khi đưa cho ban bầu cử ký.

Ban bầu cử sẽ ký và trao chữ ký $y = Sig\ (blind(x_i))$ cho cử tri x. Lúc này x sẽ xóa mù chữ ký trên y được sig(x) là chữ ký cử tri muốn có.

Cơ quan cung cấp chữ ký z_i cho x, nhưng hoàn toàn không biết giá trị x_i

2.2.2. Ứng dụng chữ ký mù trong tiền điện tử

2.2.2.1. Bài toán

Ví dụ: Alice muốn mua một quyển sách với giá 100$ từ một người bán hàng trực tuyến. Alice và người bán sách cùng sử dụng một dịch vụ của một ngân hàng. Giao dịch được thực hiện qua 3 giai đoạn như sau:

1/. Rút tiền

+ Alice tạo đồng tiền điện tử C bao gồm một chuỗi các bit để xác định một vài thông tin như số seri và giá trị của C (trong trường hợp này là 100$). Đồng tiền cần phải ẩn danh để không bị người ngoài theo dõi.

+ Ngân hàng ký mù lên đồng tiền C.

+ Ngân hàng trừ 100$ trong tài khoản của Alice.

2/. Tiêu tiền

+ Alice yêu cầu cuốn sách cần mua, Alice chuyển đồng tiền C (đã có chữ ký của ngân hàng) cho người bán hàng.

+ Người bán hàng kiểm tra sự hợp lệ của đồng tiền C bằng cách xác thực chữ ký (sử dụng khóa công khai của ngân hàng). Nếu chữ ký không hợp lệ thì người bán hàng kết thúc giao thức.

3/. Gửi tiền

+ Người bán hàng lấy đồng tiền C (đã nhận được từ Alice) gửi cho ngân hàng.

+ Ngân hàng xác thực chữ ký trên đồng tiền C. Nếu chữ ký hợp lệ thì ngân hàng sẽ kiểm tra xem đồng tiền C đã được tiêu chưa. Nếu C chưa được tiêu trước đó thì ngân hàng sẽ cộng thêm vào tài khoản của người bán 100$.

+ Sau khi nhận tiền, người bán hàng gửi sách cho Alice.

Trong trường hợp này người bán hàng khó thể biết đồng tiền C đó từ tài khoản nào, hơn thế nữa, khi người bán hàng gửi tiền C vào tài khoản của mình, ngân hàng cũng khó thể biết đồng tiền đó nhận từ Alice vì nó được ký mù.

2.2.2.2. Vấn đề phát sinh khi dùng chữ ký mù

Vấn đề có thể xảy ra đối với việc ký mù vào đồng tiền điện tử. Ví dụ Alice gửi cho ngân hàng một đồng tiền không trung thực và yêu cầu ký. Rất có thể Alice làm tờ tiền $1000 nhưng lại khai báo với ngân hàng là $100 nhưng khi ký mù thì không biết được số lượng tiền trong đồng tiền do nó đã bị làm mù. Ta có thể giải quyết vấn đề bằng 2 cách sau:

1/. Cách 1

Ngân hàng sử dụng các khóa ký (bí mật) khác nhau với các lượng tiền khác nhau. Theo đó, nếu Alice muốn lấy $1000 nhưng khai báo với ngân hàng là $100, vì thế ngân hàng sẽ dùng khóa ký trên $100. ⇒ Khi kiểm tra chữ ký đồng tiền $1000 là không hợp lệ.

2/. Cách 2

Alice và ngân hàng có thể thực hiện một giao thức dựa vào xác suất. Đầu tiên Alice làm 10 tờ tiền (c1, c2, …, c10), các tờ tiền này có mệnh giá giống nhau, chỉ khác nhau về số seri. Sau đó Alice làm mù tất cả các đồng tiền và gửi về cho ngân hàng. Ngân hàng chọn ngẫu nhiên 9 trong số 10 đồng tiền đó để yêu cầu Alice tiết lộ cácthông tin xóa mù chúng. Ngân hàng xóa mù 9 đồng tiền này, nếu tất cả đều hợp lệ thì ngân hàng sẽ ký mù lên đồng tiền còn lại và gửi cho Alice.

Chương 3: Chương trình thử nghiệm

3.1. Yêu cầu hệ thống

Hệ điều hành	: Tất cả hệ điều hành Windows hỗ trợ .NET Framework 2.0 trở lên
CPU	: Pentium 233-megahertz (MHz) trở lên.
RAM	: 256MB trở lên.
Card màn hình	: Không yêu cầu.
Card âm thanh	: Không yêu cầu.
.NET Framework	: Phiển bản 2.0 trở lên.
Internet	: Khuyến cáo nên dùng để trao đổi khóa.

3.2. Các thành phần của chương trình

ngôn ngữ C#

C# được xây dựng từ những ngôn ngữ tiền đặc biệt là C và C++ cho nên những đặc điểm ngôn ngữ của C# rất giống với ngôn ngữ C, C++. Trong phần này,em sẽ trình bày một vài đặc điểm của C#.

Các toán tử

Trong C# có các toán tử thông thường sau:

Các toán tử một toán hạng: ++,- -, !,~
Các toán tử hai toán hạng: *, /, %, +, -
Các toán tử gán: =, *=, /=, %=, +=, -=, <<=, >>=, &=, ^=, !=
Các toán tử quan hệ: <, >, <=, >=, is, as, ==, !=
Các toán tử lô- gíc: &, ^, !, &, |, <<, >>
Các toán tử điều kiện: &&, ||, ?:
Toán tử size of xác định kích thước một kiểu dữ liệu.

Trong C# cũng cho phép chồng toán tử và định nghĩa các toán tử mới theo các qui tắc sau:

Toán tử một toán hạng: type_of_x operation op(x)
Toán tử hai toán hạng: type_of_x,y operation op(x,y)

Trong C# không cho phép định nghĩa lại toán tử gán.

Các kiểu dữ liệu:

C# hỗ trợ hai loại kiểu dữ liệu là kiểu tham trị và kiểu tham biến. Kiểu tham trị bao gồm các kiểu đơn giản như char, int, float. Kiểu tham biến gồm các kiểu lớp, kiểu Interface, kiểu mảng hay nói cách khác tất cả các đối tượng đều là tham biến.

Kiểu tham trị khác kiểu tham biến ở chỗ: những biến tham trị lưa trữ trực tiếp dữ liệu của nó, trái lại biến tham biến lưa trữ con trỏ trỏ tới đối tượng.

C# cung cấp một tập các kiểu được định nghĩa trước hầu hết đã có trong C và C++. Ngoài ra C# lại đưa thêm vào kiểu boolean, string giống như trong Pascal.

C# cho phép chuyển kiểu giống như C nà C++.

Các câu lệnh

C# kế thừa hầu hết các câu lệnh từ C và C++, tuy nhiên cũng có một vài bổ xung và thay đổi đánh chú ý. Chúng ta sẽ điểm qua các câu lệnh sau:

Các lệnh được gán nhãn và lệnh goto: các lệnh được gắn nhẫn có một nhãn đứng đằng trước. Các lệnh goto sẽ nhảy đến các nhãn này và thực thi câu lệnh được gán nhãn

Lệnh if: lệnh if sẽ chọn một biểu thức để làm việc dựa trên giá trị một biểu thức lô-gic. Một lệnh if có thể có thêm lệnh else để thực thi câu lệnh khác khi giá trị biểu thức là sai.

Lệnh swich: lệnh switch thực thi một những lệnh phụ thuộc vào giá trị một biểu thức cho trước.

Các lệnh lặp: các lệnh lặp trong C# bao gồm các lệnh lặp while, do – while, for như trong C

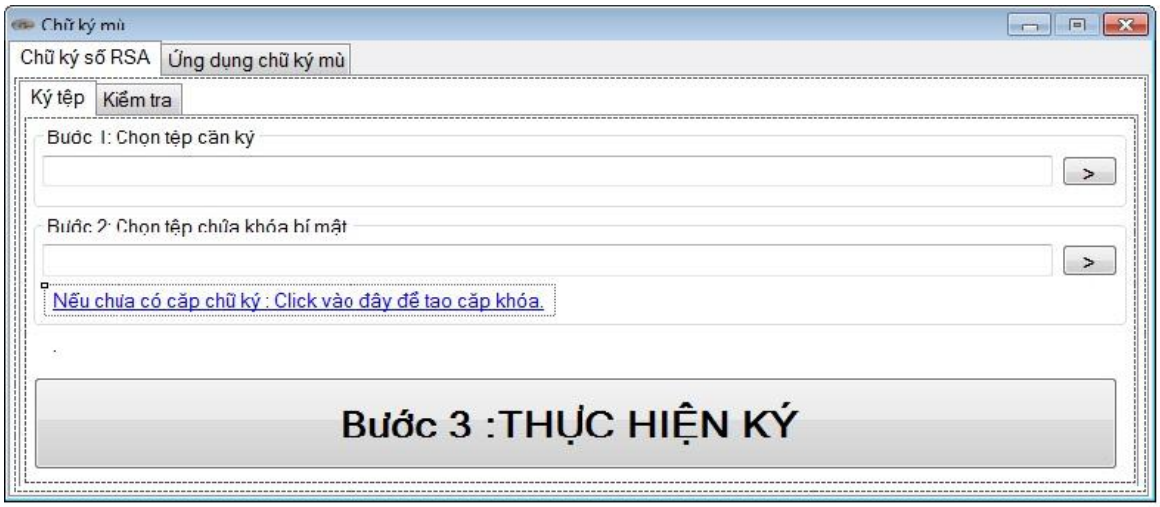
Lệnh lặp foreach (giống như trong VB): một lệnh lặp foreach liệt kê các thành phần trong một tập hợp, thực thi một câu lệnh cho mỗi thành phần của tập hợp đó.

Các lệnh throw, try, catch: các lệnh phục vụ cho quá trình quản lí lỗi trong thời gian chạy (runtime – error) gồm có phát ra một lỗi (throw), cặp lệnh try – catch đón nhận một lỗi và đưa ra hành động xử lí lỗi.

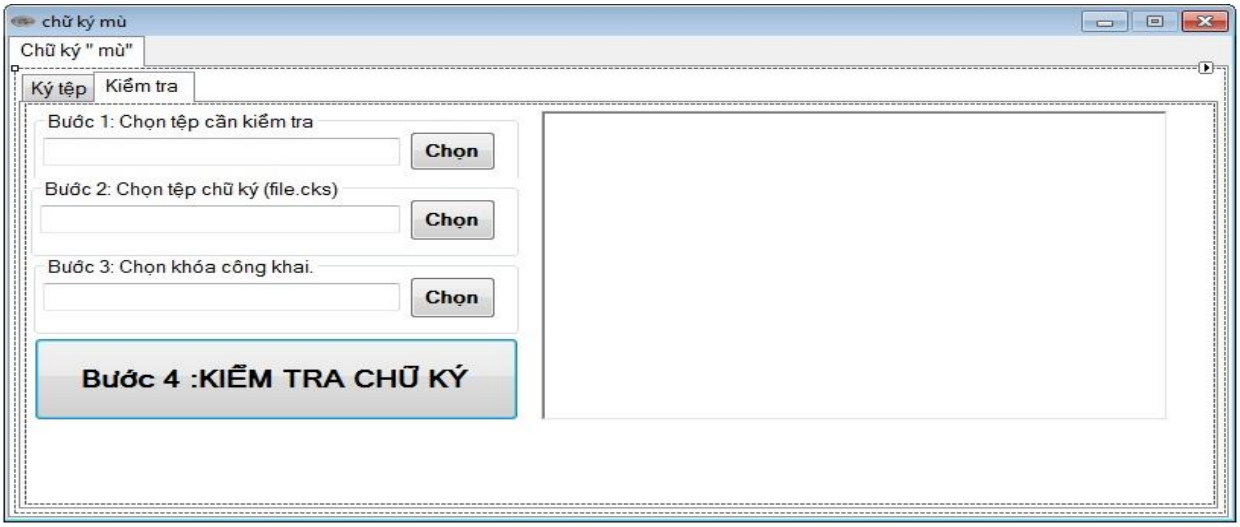
4.1. Giao diện chương trình

4.1.1. Chữ ký RSA

4.1.1.1. Giao thức ký

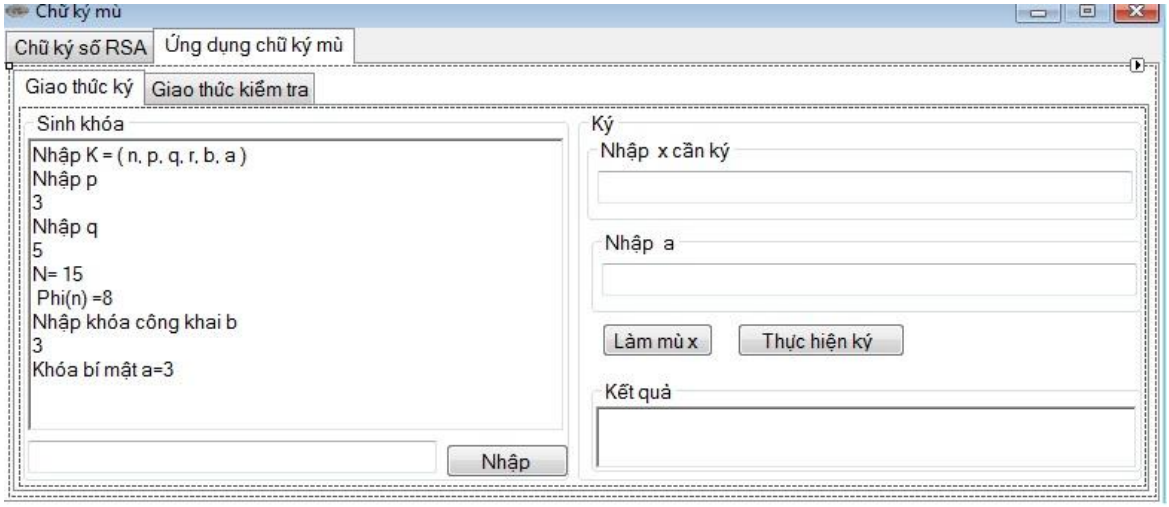


4.1.1.2. Giao thức kiểm tra:

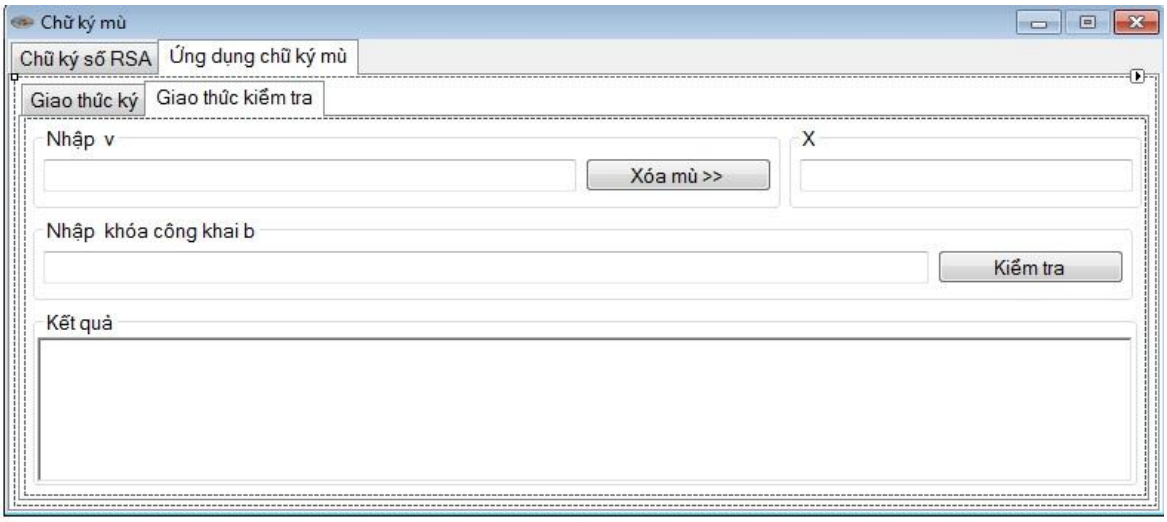


4.1.2. Ứng dụng chữ ký "mù"

4.1.2.1. Giao thức ký



4.1.2.2. Giao thức kiểm tra



KẾT LUẬN

Ngày nay, cùng với sự phát triển của khoa học công nghệ hiện đại và Công nghệ thông tin, ngành mật mã đã có những bước phát triển mạnh mẽ, đạt được nhiều kết quả lý thuyết sâu sắc và tạo cơ sở cho việc phát triển các giải pháp bảo mật, an toàn thông tin trong mọi lĩnh vực hoạt động của con người. Đặc biệt là những ưu điểm của chữ ký số.

Chữ ký số được biết đến khi sự trao đổi thông tin ngày càng phổ biến trên các mạng truyền thông ở nơi mà chữ ký tay không thể phát huy tác dụng. Nhưng bên cạnh những ưu điểm của chữ ký số mang lại nó còn bộc lộ những hạn chế nhất là đối với các chữ ký tự xác thực (RSA, Elgamal…), đó là khả năng bảo vệ chữ ký, độ an toàn và xác thực chữ ký…

Trong đồ án này, em đã đi sâu tìm hiểu về lược đồ chữ ký số chống chối bỏ và ứng dụng.

Với lược đồ chữ ký mù đã giải quyết được yêu cầu của chữ ký số đó là khả năng bảo vệ chữ ký chống sự theo dõi không hợp pháp. Vì chữ ký mù có thể làm mù thông tin của người dùng

Luận văn tập chung vào nghiên cứu cơ sở lý thuyết và xây dựng chương trình về chữ ký số.Tuy còn nhiều điểm cần phải nghiên cứu và hoàn thiện nhưng do thời gian và trình độ còn hạn chế nên không thể tránh khỏi những nhược điểm, rất mong được sự góp ý của các Thầy, Cô và các bạn.

Cuối cùng em xin cảm ơn nhà trường và các thầy cô trong khoa CNTT trường ĐH Dân Lập Hải Phòng, đặc biệt là PGS. TS. Trịnh Nhật Tiến đã tạo điều kiện và tận tình giúp đỡ em hoàn thành đồ án này.

TÀI LIỆU THAM KHẢO

1. An toàn thông tin – PGS. TS. Trịnh Nhật Tiến (NXB ĐHQGHN).

2. TS. Nguyễn Ngọc Cương – "Bài giảng An toàn thông tin".

3. Nguyễn Vân Anh – "Luận văn tốt nghiệp"

4.II D.R Stinson – "Cryptography Theory and Practice", CRC press – 1995.

5. http://google.com , http://v1.wikipedia.org/

Nguồn internet :

(*1) : http://www.vatgia.com/hoidap/4115/77237/lich-su-phat-trien-may-tinh.html