ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Vũ Quang Hòa

PHƯƠNG PHÁP CHỨNG MINH KHÔNG TIẾT LỘ THÔNG TIN VÀ ỨNG DỤNG TRONG GIAO DỊCH TRÊN MẠNG MÁY TÍNH

KHOÁ LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ thông tin

Cán bộ hướng dẫn : PGS.TS Trịnh Nhật Tiến

Cán bộ đồng hướng dẫn : ThS. Đặng Thu Hiền

HÀ NỘI - 2010

LỜI CẢM ƠN

Trước hết em xin gửi lời cảm ơn đến PGS.TS Trịnh Nhật Tiến, người thầy đã hướng dẫn em phát triển khóa luận này từ lý thuyết đến ứng dụng. Sự hướng dẫn của thầy đã giúp em có thêm được những hiểu biết sâu rộng về một số vấn đề liên quan đến bảo mật thông tin. Qua đó, những lý thuyết bảo mật cũng lôi cuốn em và sẽ trở thành hướng nghiên cứu tiếp của em sau khi tốt nghiệp.

Em xin gửi lời cảm ơn đến cô Đặng Thu Hiền đã giúp em hoàn thành luận văn một cách tốt nhất. Từ đó, em có được những hiểu biết mới cũng như hoàn thành khóa luận một cách tốt nhất.

Đồng thời em cũng xin chân thành cảm ơn các thầy cô trong bộ môn nói riêng cũng như các thầy cô trong khoa Công Nghệ nói chung. Nếu không có các thầy, các cô và khoa thì em không thể hoàn thành tốt luận văn này được.

Em xin gửi lời cảm ơn đến các thành viên lớp K51CA, những người đã tìm hiểu và cùng em phát triển cơ sở công nghệ để xây dựng nên ứng dụng nêu trong khóa luận này.

Sau cùng, em xin gửi lời cảm ơn đến gia đình, bạn bè đã tạo mọi điều kiện để em xây dựng thành công luận văn này.

Hà Nội, tháng 5 năm 2010

Sinh viên thực hiện

VŨ QUANG HÒA

MỤC LỤC

MỤC LỤC CÁC HÌNH VẼ

MỤC LỤC CÁC BẢNG

DANH MỤC TỪ VIẾT TẮT

Ký hiệu viết tắt	Giải thích
CT	Cử tri
gcd(m, n)	Ước chung lớn nhất
KP	Kiểm phiếu
Prover	Người chứng minh
TT	Người trung thực
Verifier	Người xác minh

LỜI NÓI ĐẦU

Ngày nay Internet đã trở thành một phần không thể thiếu trong mỗi người dân Việt Nam nói riêng cũng như mỗi người dân trên thế giới nói riêng. Thông tin không ngừng được trao đổi, mua bán,…trên mạng Internet, cũng chính vì lý do này mà việc bảo mật, đảm bảo an toàn thông tin đang là nhu cầu cấp thiết. Trước các yêu cầu cần thiết đó, lý thuyết về mật mã thông tin đã ra đời nhằm đảm bảo tính an toàn dữ liệu tại nơi lưu trữ cũng như khi dữ liệu được truyền trên mạng.

Khoá luận này tập trung vào nghiên cứu các khái niệm cơ bản, cơ sở lý thuyết toán học modulo sử dụng trong bảo mật thông tin, các phương pháp "chứng minh không tiết lộ thông tin" và đặc biệt là ứng dụng của "chứng minh không tiết lộ thông tin" trong bỏ phiếu thăm dò từ xa.

Chứng minh không tiết lộ thông tin đã được nghiên cứu từ những năm 80, là phương pháp chứng minh không có nghĩa là "không để lộ thông tin" mà "để lộ thông tin ở mức ít nhất" về sự vật, sự việc cần chứng minh. Với việc "không để lộ" người xác minh sẽ không có nhiều hiểu biết về sự vật sự việc, họ chỉ thu được chút ít thông tin (coi như là không) về đặc điểm tính chất của nó.

Ngành mật mã học luôn phát triển không ngừng, trong phạm vi khóa luận này, chúng tôi chỉ trình bày về một vấn đề nhỏ là phương pháp "chứng minh không tiết lộ thông tin" đồng thời tìm hiểu một số ứng dụng thực tế của cơ sở lý thuyết này.

Chương 1 : CÁC KHÁI NIỆM VÀ THUẬT TOÁN CƠ BẢN

Chương này trình bày các vấn đề cơ bản trong toán học được ứng dụng nhiều trong các bài toán an toàn thông tin. Đó là các vấn đề về lý thuyết toán học sử dụng trong bảo mật và mã hóa thông tin như : Mã hóa đồng cấu, chữ ký mù, chia sẻ bí mật ngưỡng Shamir và mã hóa Elgamal. Thông qua đó hình thành cơ sở lý thuyết cho an toàn truyền tin trên mạng máy tính.

1.1 LÝ THUYẾT MODULO

1.1.1 Hàm phi Euler

1/ Định nghĩa

Cho $n >= 1$, $\Phi(n)$ được định nghĩa là số các số nguyên trong khoảng từ [1, n] nguyên tố cùng nhau với n. Hàm $\Phi(n)$ được gọi là hàm Euler phi.

2/ Tính chất của hàm Euler

- Nếu p là số nguyên tố thì $\Phi(n) = p - 1$.
- Hàm phi Euler là hàm có tính nhân : Nếu $gcd(m, n) = 1$ thì $\Phi(mn) = \Phi(m)\,\Phi(n)$ (trong đó $gcd(m, n)$ là ký hiệu ước số chung lớn nhất của m và n)
- Nếu $n = p_1^{e1}p_2^{e2}...p_k^{ek}$ trong đó $p_1, p_2, ..., p_k$ là các thừa số nguyên tố của n thì:

$$\Phi(n) = n(1 - \frac{1}{p1})(1 - \frac{1}{p2})\ldots(1 - \frac{1}{pk})$$

1.1.2 Đồng dư thức

1/ Định nghĩa

Cho a và b là các số nguyên, a được gọi là đồng dư với b theo modulo n, ký hiệu: $a \equiv b \pmod n$ nếu $(a - b)$ chia hết cho n. Số nguyên n được gọi là modulus đồng dư.

2/ Ví dụ

$10 \equiv 3 \pmod 7$ vì $10 - 3 = 7$ chia hết cho 7

$7 \equiv -4 \pmod{11}$ vì $7 - (-4) = 11$ chia hết cho 11

3/ Tính chất của đồng dư

Cho $a, a_1, b, b_1, c \in Z$. Ta có các tính chất sau:

* $a \equiv b \ (mod\ n)$ nếu và chỉ nếu a và b cùng có số dư khi chia cho n
* $a \equiv a \ (mod\ n)$ – Tính phản xạ
* $a \equiv b \ (mod\ n)$ thì $b \equiv a \ (mod\ n)$ – Tính đối xứng
* $a \equiv b \ (mod\ n)$ và $b \equiv c \ (mod\ n)$ thì $a \equiv c \ (mod\ n)$ – Tính bắc cầu
* nếu $a \equiv a_1 \ (mod\ n)$ và $b \equiv b_1 \ (mod\ n)$ thì :

  $a + b \equiv a_1 + b_1 \ (mod\ n)$

  $a.b \equiv a_1.b_1 \ (mod\ n)$

Quan hệ "đồng dư" theo modulo n trên tập Z (tập các số nguyên) là một quan hệ tương đương (vì có tính chất phản xạ, đối xứng, bắc cầu), do đó nó tạo ra trên tập một phân hoạch gồm các lớp tương đương : hai số nguyên thuộc cùng một lớp tương đương khi và chỉ khi chúng có cùng một số dư khi chi cho n.

Mỗi lớp tương đương đại diện bởi một số duy nhất trong tập $Z_n = \{0, 1, 2, \dots , n-1\}$ là số dư khi chia các số trong lớp cho n, ký hiệu một lớp được đại diện bởi số a là $[a]_n$:

Như vậy : $[a]_n = [b]_n$ tương đương với $a \equiv b \ (mod\ n)$

Vì vậy ta có thể đồng nhất Z_n với tập các lớp tương đương theo modulo n.

$Z_n = \{0, 1, 2, \dots , n-1\}$ được gọi là tập các "thặng dư đầy đủ" theo modulo n. Mọi số nguyên bất kỳ đều có thể tìm được trong Z_n một số đồng dư với mình theo modulo n.

1.1.3 Không gian Z_n

1/ Các định nghĩa trong không gian Z_n

Các số nguyên theo modul n ký hiệu Z_n là tập hợp các số nguyên $\{0,1,2,\dots, n-1\}$. Các phép toán cộng, trừ, nhân trong Z_n được thực hiện theo modulo n.

2/ Ví dụ

$Z_{25} = \{0,1,2,\dots,24\}$. Trong Z_{25} : $13 + 16 = 4$, bởi vì: $13 + 16 = 29 \equiv 4 \ (mod\ 25)$. Tương tự, $13*16 = 8$ trong Z_{25}

- Cho $a \in Z_n$. Nghịch đảo nhân của a theo modulo n là một số nguyên $x \in Z_n$ sao cho $a*x \equiv 1 \ (mod\ n)$. Nếu x 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}.

- Cho $a, b \in Z_n$. Phép chia của a cho b theo modulo n là tích của a và b^{-1} theo modulo n, và chỉ dược xác định khi b có nghịch đảo theo modulo n.

3/ Các tính chất trong không gian Z_n

- Cho $a \in Z_n$, a có nghịch đảo khi và chỉ khi $gcd(a, n) = 1$ trong đó :

  $gcd(a, n)$ (greatest common divisor) là ký hiệu ước số chung lớn nhất của a và n

Ví dụ:

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

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

Tiếp theo là sự tổng quát hoá của tính chất 1.6

- Giả sử $d = gcd(a, n)$. Phương trình đồng dư $ax \equiv b \pmod n$ 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.

4/ Định lý phần dư Trung Hoa CRT

Nếu các số nguyên $n_1, n_2, ..., n_k$ là các số nguyên tố cùng nhau từng đôi một thì hệ phương trình đồng dư :

$x \equiv a_1 \pmod{n_1}$

$x \equiv a_2 \pmod{n_2}$

.....

$x \equiv a_k \pmod{n_k}$

có nghiệm duy nhất theo modulo $n = n_1 n_2 \ldots n_k$

5/ Thuật toán của Gausse

Nghiệm x trong hệ phương trình đồng dư (định lý phần dư Trung Hoa) được tính như sau :

$$x = \sum_{i=1}^{k} a_i N_i M_i \bmod n$$

trong đó: $N_i = n/n_i$, $M_i = N_i^{-1} \bmod n_i$

Ví dụ:

Cặp đồng dư: $x \equiv 3 \pmod 7$ và $x \equiv 7 \pmod{13}$ có nghiệm duy nhất $x \equiv 59 \pmod{91}$

Tính chất :

Nếu $gcd(n_1, n_2) = 1$ thì cặp đồng dư $x \equiv a \pmod{n_1}$ và $x \equiv a \pmod{n_2}$ có nghiệm duy nhất $x \equiv a \pmod{n_1 n_2}$

1.1.4 Nhóm nhân Z_n^*

1/ Các định nghĩa trong nhóm nhân Z^*_n

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

Đặc biệt, nếu n là số nguyên tố thì $Z^*_n = \{a \in Z_n \mid 1 \le a \le n-1\}$

Cho $a \in Z_n^*$. Bậc của a, ký hiệu là $ord(a)$ là số nguyên dương t nhỏ nhất sao cho

$$a^t \equiv 1 \pmod n.$$

2/ Các tính chất trong Z_n^*

- Cho $n \ge 2$ là số nguyên :
  - (*Định lý Euler*) Nếu $a \in Z_n^*$ thì $a^{\Phi(n)} \equiv 1 \pmod n$.
  - Nếu n là tích của các số nguyên tố phân biệt và nếu $r \equiv s \pmod{\Phi(n)}$ $a^r \equiv a^s \pmod n$ 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 $\Phi(n)$
- 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 $a^r \equiv a^s \pmod p$ 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 $p-1$
  - Đặc biệt $a^p \equiv a \pmod p$ với mọi số nguyên a.

1.1.5 Thặng dư

1/ Định nghĩa thặng dư

Cho $a \in Z_n^*$. a được gọi là thặng dư bậc 2 theo modulo n hoặc bình phương theo modulo n nếu tồn tại $x \in Z_n^*$ sao cho $x^2 \equiv a \ (mod\ n)$. Nếu không tồn tại x thì a được gọi là thặng dư không bậc 2 theo modulo n. Tập hợp các thặng dư bậc 2 theo modulo n ký hiệu là Q_n và tập hợp các thặng dư không bậc 2 theo modulo n ký hiệu là $\overline{Q}_n$.

Chú ý vì định nghĩa $0 \notin Z_n^*$ nên $0 \notin Q_n$ và $0 \notin \overline{Q}_n$

2/ Tính chất của thặng dư

Cho n là tích của 2 số nguyên tố p và q. Khi đó $a \in Z_n^*$ là một thặng dư bậc 2 theo modulo n khi và chỉ khi $a \notin Q_n$ và $a \notin \overline{Q}_n$. 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 đó: $Q_{21} = \{1, 4, 16\}$ và $\overline{Q}_{21} = \{2, 5, 8, 10, 11, 13, 17, 19, 20\}$

1.1.6 Căn bậc Modulo

1/ Định nghĩa

Cho $a \in Q_n$. Nếu $a \in Z_n^*$ thoả mãn $x^2 \equiv a \ (mod\ n)$ thì x được gọi là căn bậc *2* của a theo modulo n.

2/ Tính chất (Số căn bậc 2)

- Nếu p là một số nguyên tố lẻ thì $a \in Q_n$ thì a có chính xác 2 căn bậc 2 theo modulo p
- Tổng quát hơn: cho $n = p_1^{e1} p_2^{e2} ... p_k^{ek}$ trong đó p_i là các số nguyên tố lẻ phân biệt và $e_i \geq 1$. Nếu $a \in Q_n$ thì a có chính xác 2^k căn bậc 2 theo modulo n.

3/ Ví dụ

Căn bậc 2 của 13 theo modulo 37 là 7 và 30. căn bậc 2 của 121 modulo 315 là 11, 74, 101, 151, 164, 214, 241 và 304.

1.1.7 Các thuật thoán trong Z_n^*

1/ Định nghĩa

Cho *n* là số nguyên dương. Như đã nói ở trước, các phần tử trong Z_n sẽ được thể hiện bởi các số nguyên {0, 1, 2,…, *n-1*}. Ta thấy rằng: nếu $a, b \in Z_n$ thì:

$$(a + b) \bmod n = \begin{cases} a + b \text{ nếu } a + b < n \\ a + b - n \text{ nếu } a + b \geq n \end{cases}$$

Vì vậy, phép cộng modulo (và phép trừ modulo) có thể được thực hiện mà không cần thực hiện các phép chia . Phép nhân modulo của *a* và *b* có thể 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 Euclidean mở rộng như mô tả sau:

2/Thuật toán tính nghịch đảo nhân trong Z_n

INPUT: $a \in Z_n$

OUTPUT: $a^{-1} \bmod n$ nếu tồn tại.

1. Sử dụng thuật toán Euclidean mở rộng sau để tìm các số nguyên *x* và *y* sao cho: $ax + ny = d$ *với* $d = gcd(a, n)$.
2. Nếu $d > 1$ thì $a^{-1} \bmod n$ không tồn tại. Ngược lại, return (*x*).

3/ Thuật toán Euclidean mở rộng:

INPUT: 2 số nguyên dương *a* và *b* với $a \geq b$.

OUTPUT: $d = gcd(a, b)$ và các số nguyên *x, y* thoả mãn: $ax + by = d$

1. Nếu $b = 0$ thì đặt $d \leftarrow a, x \leftarrow 1, y \leftarrow 0$ và return (*d, x, y*)
2. Đặt $x_2 \leftarrow 1, x_1 \leftarrow 0, y_2 \leftarrow 0, y_1 \leftarrow 1$
3. Khi $b > 0$ thực hiện:
   - 3.1. $q \leftarrow [a/b], r = a - qb, x \leftarrow x_2 - qx_1, y \leftarrow y_2 - qy_1$
   - 3.2. $a \leftarrow b, r \leftarrow b, x_2 \leftarrow x_1, x_1 \leftarrow x, y_2 \leftarrow y_1, y_1 \leftarrow y$
4. Đặt $d \leftarrow a, x \leftarrow x_2, y \leftarrow y_2$ và return (*d, x, y*)

Số mũ modulo có thể được tính một các hiệu quả bằng thuật toán bình phương và nhân liên tiếp, nó được sử dụng chủ yếu trong nhiều giao thức mã hoá. Một phiên bản của thuật toán này như sau: Giả sử biểu diễn nhị phân của k là $\sum_{i=0}^{t} k_i 2^i$ với $k_i \in \{0,1\}$.

4/ Thuật toán bình phương liên tiếp để tính số mũ modulo trong Z_n.

INPUT: $a \in Z_n$ và số nguyên dương $0 \leq k < n$ trong đó k có biểu diễn nhị phân là: $k = \sum_{i=0}^{t} k_i 2^i$

OUTPUT: $a^k \bmod n$

1. Đặt $b \leftarrow 1$. Nếu $k = 0$ return (b)
2. Đặt $A \leftarrow a$
3. Nếu $k_0 = 1$ thì đặt $b \leftarrow a$.
4. For $i = 1$ to t do

   Đặt $A \leftarrow A^2 \bmod n$

   Nếu $k_i = 1$ thì $b \leftarrow A \,.\, b \bmod n$
5. Return (b).

Ví dụ: (Tính số mũ modulo)

Bảng 1 : Mô tả các bước tính : $5^{596} \bmod 1234 = 1013$

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	779	947	925
B	1	1	625	625	67	67	1059	1059	1059	1013

Độ phức tạp theo bit của các phép toán cơ bản trong Z_n được trình bày trong bảng sau:

Bảng 2 : Độ phức tạp theo bit của các phép toán cơ bản trong Z

Phép toán	Độ phức tạp về bit
Cộng modulo $(a + b) \bmod n$	$O(\lg n)$
Trừ modulo $(a - b) \bmod n$	$O(\lg n)$
Nhân modulo $(a\ b) \bmod n$	$O((\lg n)^2)$
Nghịch đảo theo modulo $a^{-1} \bmod n$	$O((\lg n)^2)$
Số mũ modulo $a^k \bmod n, k < n$	$O((\lg n)^3)$

1.1.8 Tính căn bậc bất kỳ trong Z_n^*

Sử dụng tính chất trong Z_n^* : Nếu n là tích của các số nguyên tố phân biệt và nếu $r \equiv s \pmod{\Phi(n)}$ $a^r \equiv a^s \pmod n$ 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 $\Phi(n)$ để tính căn bậc k trong Z_n :

Giả sử tính $\sqrt[x]{y}$ trong không gian Z_n. Áp dụng công thức $\sqrt[x]{y} = y^{1/x} \equiv y^z \pmod n$. Theo tính chất trên thì ta phải có $1/x \equiv z \pmod{\Phi(n)}$. Sử dụng thuật toán Euclidean mở rộng để tính nghịch đảo nhân $z = 1/x$ trong $Z_{\Phi(n)}$. Do đó: $\sqrt[x]{y} \equiv y^z \pmod n$. Sử dụng thuật toán bình phương liên tiếp để tính số mũ modulo y^z trong Z_n.

Ví dụ:

Tính $\sqrt[7]{5}$ trong Z_{13}

$\sqrt[7]{5} \pmod{13} \equiv 5^{1/7 \pmod{12}} \pmod{13} = 5^7 \pmod{13} = 8 \rightarrow \sqrt[7]{5} = 8$

1.2 VẤN ĐỀ MÃ HÓA

Mặc dù mã hoá đã được sử dụng từ rất lâu trong các hoạt động ngoại giao và quân sự nhưng chỉ sau khi bài báo "Lý thuyết truyền tin trong các hệ thống bảo mật" của Claude Shannon [10] ra đời nó mới trở thành một môn khoa học. Trước đó các vấn đề về mã hoá, mật mã gần như là một môn "nghệ thuật".

Mã hoá là phần rất quan trọng trong vấn đề bảo mật. Mã hoá ngoài nhiệm vụ chính là làm cho tài liệu an toàn hơn, nó còn có một lợi ích quan trọng là : thay vì truyền đi tài liệu thô (không được mã hoá) trên một đường truyền đặc biệt, được canh phòng cẩn mật không cho người nào có thể "xâm nhập" vào lấy dữ liệu, người ta có thể truyền một tài liệu đã được mã hoá trên bất cứ đường truyền nào mà không lo dữ liệu bị đánh cắp vì nếu dữ liêu có bị đánh cắp đi nữa thì dữ liệu đó cũng không dùng được.

Một số khái niệm liên quan :

- **Thuật toán mã hoá/ giải mã** : là thuật toán dùng để chuyển thông tin thành dữ liệu mã hoá hoặc ngược lại.
- **Khoá** : là thông tin mà thuật toán mã/ giải mã sử dụng để mã hóa/ giải mã thông tin. Mỗi khi một thông tin đã được mã hoá thì chỉ có những người có khoá thích hợp mới có thể giải mã. Nếu không thì dù dùng cùng một thuật toán giải mã nhưng cũng không thể phục hồi lại thông tin ban đầu. Đây là đặc điểm quan trọng của khoá : mã hoá chỉ phụ thuộc vào khoá mà không phụ thuộc vào thuật toán mã/ giải mã. Điều này giúp cho một thuật toán mã/ giải mã có thể được sử dụng rộng rãi.

Với hình thức khá phổ biến hiện nay là truyền tin qua thư điện tử và không sử dụng các công cụ mã hoá, bảo mật cũng như chữ ký điện tử thì các tình huống sau có thể xảy ra :

- Không chỉ người nhận mà người khác có thể đọc được thông tin.
- Thông tin mà ta nhận được có thể không phải là của người gửi đúng đắn.
- Thông tin nhận được bị người thứ ba sửa đổi.
- Bị nghe trộm: thông tin được truyền đi trên đường truyền có thể bị ai đó "xâm nhập" vào lấy ra, tuy nhiên vẫn đến được người nhận mà không bị thay đổi.

- Bị thay đổi : thông tin bị chặn lại ở một nơi nào đó trên đường truyền và bị thay đổi. Sau đó thông tin đã bị thay đổi này được truyền tới cho người nhận như không có chuyện gì xảy ra.
- Bị lấy cắp : thông tin bị lấy ra nhưng hoàn toàn không đến được người nhận.

Khi đó thì khỏi nói đến thương mại điện tử, chính phủ điện tử với nền quản lý hành chính điện tử, vv và vv. Để giải quyết vấn đề này, thông tin trước khi truyền đi sẽ được mã hoá và khi tới người nhận, nó sẽ được giải mã trở lại.

Để đảm bảo rằng chỉ người cần nhận có thể đọc được thông tin mà ta gửi khi biết rằng trên đường đi, nội dung thông tin có thể bị theo dõi và đọc trộm, người ta sử dụng các thuật toán đặc biệt để mã hoá thông tin. Trong trường hợp này, trước khi thông tin được gửi đi, chúng sẽ được mã hoá lại và kết quả là ta nhận được một nội dung thông tin "không có ý nghĩa". Khi thông điệp bị theo dõi hoặc bị bắt giữ trên đường đi, để hiểu được thông tin của bạn, kẻ tấn công phải làm một việc là giải mã nó. Thuật toán mã hoá càng tốt thì chi phí cho giải mã đối với kẻ tấn công càng cao. Khi chi phí giải mã cao hơn giá trị thông tin thì coi như bạn đã thành công trong vấn đề bảo mật.

Các thuật toán mã hoá thông tin khá đa dạng nhưng có thể chia ra làm hai hướng:

1.2.1 Mã hoá đối xứng

Là loại mã hoá chỉ dùng 1 khoá cho cả việc mã hoá và giải mã.

1/ Ưu điểm

- Tốc độ mã/ giải mã nhanh. Đây là ưu điểm nổi bật của mã đối xứng.
- Sử dụng đơn giản : chỉ cần dùng một khoá cho cả 2 bước mã và giải mã.

2/ Nhược điểm

- Đòi hỏi khoá phải được 2 bên gửi/ nhận trao tận tay nhau vì không thể truyền khoá này trên đường truyền mà không được bảo vệ. Điều này làm cho việc sử dụng khoá trở nên không thực tế.
- Không an toàn : càng nhiều người biết khoá thì độ rủi ro càng cao.
- Trong trường hợp khoá mã hoá thay đổi, cần thay đổi đồng thời ở cả người gửi và người nhận, khi đó rất khó có thể đảm bảo được là chính bản thân khoá không bị đánh cắp trên đường đi
- Không cho phép ta tạo ra chữ ký điện tử.

3/ Một số thuật toán mã hoá đối xứng

- DES : 56 bit, không an toàn. Có thể dễ dàng bị bẻ khoá trong khoảng vài phút.
- Triple DES, DESX, GDES, RDES: mở rộng độ dài của khoá ở mã DES lên tới 168 bit.
- RC2, RC4, RC5: độ dài khoá có thể lên tới 2048 bit.
- IDEA (International Data Encryption Algorithm) : 128 bit, thường dùng trong các chương trình email.

1.2.2 Mã hoá không đối xứng

Là loại mã hoá dùng một khoá để mã hoá (thường gọi là khoá công khai - public key) và dùng một khoá khác để giải mã (thường gọi là khoá riêng - private key).

1/ Ưu điểm

Đây là loại mã hoá được sử dụng chủ yếu trên Internet. Một người muốn sử dụng loại mã hoá này cần tạo ra một cặp khoá công khai/ bí mật. Anh ta có thể truyền khoá công khai của mình tới bất cứ ai muốn giao tiếp với anh ta mà không sợ người khác lấy khoá này. Cô ta sẽ mã hoá thông điệp của mình bằng khoá công khai đó và gửi tới cho anh ta. Dĩ nhiên là chỉ mình anh ta với khoá bí mật của mình mới có thể thấy được thông điệp của cô. Như vậy kẻ tấn công, cho dù có biết nội dung của khoá công khai và nội dung của thông tin đã bị mã hoá vẫn không thể giải mã được thông tin. Lý do là tính ngược khoá bí mật từ khoá công khai hoặc là rất khó, nếu không nói là không thể. Điều này đạt được trên nguyên tắc sử dụng các hàm một chiều trong toán học khi tính hàm y=f(x) là đơn giản nhưng ngược lại việc tính giá trị y khi đã biết x là rất khó khăn.

2/ Nhược điểm

Tốc độ mã hoá chậm : tốc độ mã hoá nhanh nhất của loại mật mã không đối xứng vẫn chậm hơn nhiều lần so với mật mã đối xứng. Do đó người ta thường kết hợp 2 loại mã hoá để nâng tốc độ mã hoá lên.

3/ Một số thuật toán mã hoá không đối xứng

- RSA : Hệ mã này được dùng nhiều nhất cho web và chương trình email. Độ dài khoá thông thường là từ 512 đến 1024 bit. [8]
- Elgamal : 512 đến 1024 bit.

1.3 VẤN ĐỀ KÝ ĐIỆN TỬ (DIGITAL SIGNATURE)

1.3.1 Khái niệm

Nếu việc sử dụng mật mã đã trở nên phổ biến, không chỉ trong quân đội mà còn trong thương mại và những mục đích cá nhân thì những đoạn tin và tài liệu điện tử sẽ cần những chữ ký giống như các tài liệu giấy.

Cũng giống như trong thực tế, chữ ký để xác nhận cho người nhận rằng bức thư đó do người này gửi mà không phải ai khác. Chữ ký điện tử sử dụng thuật toán mã không đối xứng để định danh người gửi. Thông thường, để bảo vệ các văn bản mã hoá người ta dùng chữ ký điện tử. Việc ứng dụng chữ ký điện tử cũng như công nhận giá trị pháp lý của nó là điều kiện tiên quyết cho thương mại điện tử. Nếu như việc giả mạo chữ ký viết tay hoặc con dấu là không đơn giản thì việc làm giả một đoạn thông tin nào đó là rất dễ dàng. Vì lý do đó, bạn không thể quét chữ ký của mình cũng như con dấu tròn của công ty để chứng tỏ rằng tài liệu mà bạn truyền đi đúng là của bạn.

Khi bạn cần "ký " một văn bản hoặc một tài liệu nào đó, thủ tục đầu tiên là tạo ra chữ ký và thêm nó vào trong thông điệp. Có thể hình dung thủ tục này như sau. Phần mềm mã hoá mà bạn sử dụng sẽ đọc nội dung văn bản và tạo ra một chuỗi thông tin đảm bảo chỉ đặc trưng cho văn bản đó mà thôi. Bất kỳ một thay đổi nào trong văn bản sẽ kéo theo sự thay đổi của chuỗi thông tin này. Sau đó phần mềm đó sẽ sử dụng khoá bí mật của bạn để mã hoá chuỗi thông tin này và thêm nó vào cuối văn bản như một động tác ký (Bạn có thể thấy là chúng ta hoàn toàn không mã hoá nội dung văn bản, chỉ làm động tác ký mà thôi). Khi nhận được văn bản, người nhận lặp lại động tác tạo ra chuỗi thông tin đặc trưng, sau đó sử dụng khoá công khai mà bạn đã gửi để kiểm tra chữ ký điện tử có đúng là của bạn không và nội dung thông điệp có bị thay đổi hay không.

Thuật toán mã hoá không đối xứng đầu tiên và nổi tiếng hơn cả có tên gọi là RSA (được ghép từ chữ cái đầu tiên của tên ba tác giả là Rivest, Shamir, Adleman). Thuật toán RSA cũng được áp dụng để tạo ra chữ ký RSA.

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 Công Nghệ”
2. Mã hoá nó bằng khoá bí mật của mình tạo ra chữ ký điện tử.
3. Gắn chữ ký này vào thông điệp cần gửi rồi và mã hoá toàn bộ bằng khoá công khai của người nhận.

4. Gửi thông điệp đi.

Người nhận sẽ dùng khoá bí mật của mình để giải mã thông điệp và lấy chữ ký ra. Sau đó họ sẽ giải mã chữ ký này bằng khoá công khai của người gửi. Chỉ người gửi nào có khoá 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.

Tuy nhiên chữ ký điện tử tạo ra theo cách này vẫn chưa dùng được. Nó có thể bị cắt và dán vào thông điệp khác mà không cần phải biết khoá bí mật.

1.3.3 Hàm băm sử dụng trong ký điện tử

Một 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 thông điệp đi qua một hàm băm chỉ cho duy nhất một "đại diện" và ngược lại : rất khó có thể tìm được hai thông điệp khác nhau mà có cùng "đại diện" khi đi qua cùng 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 dùng để kiểm tra tính toàn vẹn của thông điệp. Các bước để tao ra chữ ký điện tử như vậy được trình bày như sau :

1. Đưa thông điệp cần gửi qua hàm băm tạo ra đại diện cho thông điệp đó .
2. Mã hoá đại diện bằng khoá bí mật của người gửi để tạo ra chữ ký điện tử.
3. Mã hoá toàn bộ thông điệp và chữ ký bằng khoá công khai của người nhận và gửi đi

Người nhận sẽ giải mã thông điệp bằng khoá bí mật của mình, giải mã chữ ký bằng khoá 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.

1.3.4 Một số hàm băm thường gặp

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

1.4 CHỮ KÝ MÙ

1.4.1 Khái niệm

Theo phương thức bỏ phiếu truyền thống, cử tri mang chứng minh thư và lá phiếu chưa có nội dung gì đến bàn đóng dấu. Ở đó người ta kiểm tra giấy tờ để xác định quyền bỏ phiếu, sau đó họ đóng dấu xác thực trên lá phiếu. Cử tri cất chứng minh thư vào phòng bỏ phiếu, như vậy lá phiếu hoàn toàn không có **thông tin định danh.** Quá trình bỏ phiếu này được gọi là "nặc danh".

1.4.2 Kỹ thuật chữ ký mù RSA

- Giả sử Ban kiểm phiếu (KP) dùng sơ đồ chữ ký RSA *(n, p, q, b, a).*

Nếu cử tri (CT) chuyển ngay Số định danh *x* của mình cho Ban KP thì sẽ nhận được chữ ký là $E(x) = \boldsymbol{x^a} \pmod n$. CT không làm như vậy !

- Cử tri CT che dấu Số định danh *x* bằng bí danh *y*:

  $y = Blind(x) = x * r^b \pmod n$.

  (Trong đó *r* được chọn sao cho tồn tại phần tử nghịch đảo $r^{-1} \pmod n$).

  - Cử tri CT gửi bí danh *y* cho Ban KP.
  - Ban KP ký trên bí danh *y* được chữ ký *z*:
  - $z = E(y) = E(Blind(x)) = E(x * r^b) = (x * r^b)^a = x^a * (r^b)^a = x^a * r$.
  - Ban KP gửi chữ ký *z* cho cử tri CT.

- Cử tri CT "xoá mù" trên z sẽ nhận được chữ ký trên Số định danh *x*:

  $Unblind(z)=Unblind(E(blind(x)))=Unblind(x^a * r)= (x^a * r) * r^{-1} = \boldsymbol{x^a} \pmod n$.

  Cử tri CT đã có được chữ kí của Ban KP trên x, đó là $\boldsymbol{x^a} \pmod n$.

Chương 2 : PHƯƠNG PHÁP CHỨNG MINH KHÔNG TIẾT LỘ THÔNG TIN

2.1 KHÁI NIỆM PHÉP CHỨNG MINH KHÔNG TIẾT LỘ THÔNG TIN

2.1.1 Khái niệm phép chứng minh

Trong toán học và cuộc sống, chúng ta thường muốn chứng minh một vấn đề gì đó cho người khác hiểu. Điển hình, nếu như tôi biết X đúng, và tôi muốn chứng minh điều này cho bạn, tôi cố gắng hết sức để sử dụng những điều đã có và những điều liên quan để chứng minh rằng X đúng.

Ví dụ : Tôi biết rằng 26781 không là số nguyên tố bởi nó gấp 113 237 lần, để chứng minh cho bạn thấy điều đó, tôi chỉ ra rằng thực sự 26781 = 113 * 237.

2.1.2 Hệ thống chứng minh tương tác

Theo lý thuyết tính toán phức tạp, một hệ thống chứng minh tương tác là một máy trừu tượng mà các mô hình tính toán như là việc trao đổi tin nhắn giữa hai bên. Các bên, có Verifier và Prover (người xác minh và người chứng minh), tương tác với nhau bằng cách trao đổi thông điệp để xác định một chuỗi thuộc về một ngôn ngữ hay không? Prover là toàn năng và sở hữu không giới hạn nguồn lực tính toán, nhưng không thể tin tưởng được, trong khi người xác minh đã bị chặn sức mạnh tính toán. Thông điệp được gửi giữa hai bên Verifier và Prover cho đến khi có một câu trả lời cho vấn đề này và đã "thuyết phục" chính nó nếu nó đúng.

Tất cả các hệ thống chứng minh tương tác gồm có hai yêu cầu :

Đầy đủ : Nếu phát biểu là đúng, việc xác minh trung thực (có nghĩa là, một trong các giao thức sau đây là đúng) sẽ được thuyết phục thực tế bởi Prover trung thực.

Hoàn thiện : Nếu phát biểu là sai, không có Prover, ngay cả khi không theo giao thức, có thể thuyết phục người xác minh trung thực rằng nó là đúng, trừ với một số xác suất rất nhỏ.

Chú ý rằng chúng ta không quan tâm tới những gì xảy ra nếu người xác minh không trung thực, chúng ta tin vào người xác minh.

Bản chất cụ thể của hệ thống, và do đó các lớp phực tạp của ngôn ngữ nó có thể nhận ra, phụ thuộc vào những gì sắp xếp giới hạn được đặt trên Verifier, cũng như những gì mà khả năng của nó mang lại – ví dụ, hầu hết các hệ thống chứng minh tương tác phụ thuộc vào rất nhiều vào khả năng của Verifier để đưa ra lựa chọn ngẫu

nhiên. Nó cũng phụ thuộc vào bản chất của tin nhắn trao đổi – có bao nhiêu và những gì chứa bên trong nó. Hệ thống chứng minh tương tác đã tìm thấy một số ý nghĩa sâu sắc đáng ngạc nhiên cho các lớp truyền thống phức tạp được xác minh bằng cách sử dụng chỉ một máy. Các lớp phức tạp của hệ thống chính được miêu tả bằng hệ thống chứng minh tương tác là AM và IP. (Arthur – Merlin protocol và Interactive Proof System)

2.1.3 Phương pháp chứng minh không tiết lộ thông tin

1/ Khái niệm

Một hệ quả tiêu biểu của một phép chứng minh là bạn đã học được một số hiểu biết, khác hơn là bạn đang tin rằng phát biểu là đúng sự thật. Trong ví dụ trước chỉ có bạn biết 26781 không phải là số nguyên tố, nhưng bạn cũng chỉ ra phân tích nhân của số đó.

Chứng minh không tiết lộ thông tin cố gắng tránh khỏi điều này. Trong phương pháp này, Alice muốn chứng minh cho Bob thấy rằng X đúng, Bob sẽ thực sự bị thuyết phục rằng X là đúng, nhưng sẽ không học được bất kỳ điều gì như là hệ quả của quá trình này. Rằng Bob không có thêm hiểu biết.

Ta lại xét thêm một ví dụ đơn giản như thế này :

Giả sử P và V cùng tham gia trò chơi với các quân bài. P đưa ra 2 quân bài úp và nói đó là "**át**" và "**2**". P yêu cầu V chọn quân "át".

Trước khi chọn quân "át", V muốn kiểm tra chắc chắn rằng 2 quân bài đó đích thực là "**át**" và "**2**". V yêu cầu P chứng minh điều này. Nếu P lật 2 quần bài đó lên thì coi như là một cách chứng minh thì trò chơi kết thúc vì V đã nhìn thấy chúng và dĩ nhiên là anh ta có thể chọn ngay ra được quân bài "**át**".

Có một cách khách để P chứng minh rằng quân bài đó là "**át**" và "**2**" mà không phải lật 2 quân bài đó lên, tức là không làm lộ thông tin về 2 quân bài trên tay P. Rất đơn giản, anh ta đưa 50 quân bài còn lại cho V. Nếu V kiểm tra thấy thiếu một quân bài "**át**" và 1 quân bài "**2**" thì có thể coi quân bài trên tay P đưa ra đúng như anh ta nói.

Qua hai ví dụ trên có thể tạm hiểu "Chứng minh không tiết lộ thông tin" không có nghĩa là "không để lộ thông tin" mà nghĩa là "để lộ thông tin ở mức ít nhất" về sự vật sự việc cần chứng minh. Với những "thông tin để lộ", người xác minh không có nhiều hiểu biết (knowledge) về sự vật sự việc, họ chỉ thu được chút ít thông tin (coi như "zero knowledge") về đặc điểm tính chất của nó.

Giao thức $\sum$ là giao thức "Hỏi - Đáp" 3 bước để P chứng minh cho V một vấn đề nào đó.

- P gửi cho V - một giá trị ngẫu nhiên.
- V gửi lại P - một giá trị ngẫu nhiên như là giá trị dùng để kiểm thử.
- P gửi đáp lại V một giá trị.

Kết quả V thừa nhận hoặc bác bỏ vấn đề P chứng minh.

"Chứng minh không tiết lộ thông tin" được phát minh bởi Goldwasser, Micali và Rackoff năm 1981 (được viết tắt là GMR). Chứng minh không tiết lộ thông tin (và chứng minh tương tác nói chung) hóa ra là một trong những lý thuyết hay nhất và có ảnh hưởng lớn nhất trong khoa học máy tính, với ứng dụng ngày càng tăng trong dự án chữ ký thực để chứng minh rất nhiều vấn đề NP-complete là khó ngay cả khi xấp xỉ.

2/ Các thành phần bên trong phép chứng minh không tiết lộ thông tin

Có hai nhân vật mà chúng ta thường xuyên phải để nhắc đến trong vấn đề này :

- **Peggy**, các Prover(người chứng minh) – Peggy có một số thông tin muốn chứng minh cho Victor thấy, nhưng cô ấy lại không muốn nói thẳng bí mật đó cho Victor.
- **Victor**, các Verifier(người xác minh) – Victor hỏi Peggy một loạt các câu hỏi, cố gắng tìm ra được là Peggy có thực sự biết được bí mật đó hay không. Victor không tiếp thu được bất cứ điều gì khác từ bí mật đó, ngay cả khi anh ta gian lận hay không tuân theo chỉ dẫn của giao thức.

3/ Tính chất của giao thức chứng minh không tiết lộ thông tin

Giao thức chứng minh không tiết lộ thông tin có thể được mô tả như là các giao thức mật mã khác có tính năng đặc biệt được mô tả trong [10] – H. Aronsson. Zero knowledge protocols and small systems :

- Người xác minh (verifier) không thể tiếp thu được bất cứ một điều gì từ giao thức này : Verifier không học thêm được bất cứ điều gì từ giao thức này, bởi anh ta không thể tự mình tìm hiểu mà không có người chứng minh (prover). Đây chính là nội dung chính của giao thức chứng minh không tiết lộ thông tin (giống như không có tri thức nào được trao đổi ở đây). Không có thuộc tính này, giao thức này sẽ được gọi là giao thức tiết lộ tối thiểu, tức là nó yêu cầu hoàn toàn không có thông tin nào có thể để lộ trong trường hợp này.

- Prover sẽ không gian lận Verifier : Nếu Peggy không biết bí mật đó, rõ ràng xác suất thành công của cô ấy là rất nhỏ. Sau số vòng lặp tương đối lớn của giao thức này, tỉ lệ Prover gian lận sẽ được làm nhỏ nhất khi cần thiết. Giao thức này cũng được cắt và chọn lựa, tức là chỉ cần lần đầu tiên Prover thất bại, Victor có thể biết được ngay rằng Peggy gian lận. Như vậy, với mỗi vòng lặp của giao thức này, xác suất thành công sẽ cao hơn rất nhiều. Giao thức này có thể làm việc tốt ngay cả khi xác suất may mắn của Prover gian lận cao vì ta có thể tăng số lần vòng lặp của giao thức. Nói cách khác, khả năng nắm bắt của Verifier rất cao, có thể bảo vệ bản thân tránh khỏi bị thuyết phục bởi những vấn đề sai (không có vấn đề gì mà Prover có thể đánh lừa được Verifier).
- Verifier không thể gian lận Prover được : Victor không có thêm được bất cứ thông tin nào từ giao thức, thậm chí nếu anh ta không tuân theo những chỉ dẫn đó. Điều duy nhất Victor có thể làm để thuyết phục chính anh ta rằng Peggy biết bí mật đó. Điều mà Prover tiết lộ chỉ là một giải pháp của trong rất nhiều giải pháp cho một vấn đề, không bao giờ tất cả những điều ấy kết hợp lại có thể tìm ra được bí mật đó.
- Verifier không thể giả làm Prover để chứng minh cho người thứ ba được. Bởi vì không có thông tin rò rỉ từ Peggy cho Victor, Victor không thể thử để thay thể Peggy cho bên thứ 3 bất kỳ bên ngoài. Với một số giao thức khác, người trung gian có thể tấn công là điều có thể, tuy nhiên, điều đó có nghĩa là có ai đó có thể chuyển tiếp lưu lượng truy cập từ Peggy thật và cố gắng thuyết phục một Victor khác, thủ phạm, là Peggy. Ngoài ra nếu các bản ghi Verifier khác ghi lại cuộc đàm thoại giữa anh ta và Prover, thì bản ghi đó cũng không thể được dùng để thuyết phục bên thứ ba nào cả. Nó trông giống như một cuộc trò chuyện giả (ví dụ: trong đó Verifier và Prover cùng đồng ý bắt tay trước khi yêu cầu Verifier chọn)
- Victor có thể chứng minh bất cứ một luận điểm đúng nào : Thuộc tính này được gọi là hoàn thiện và nó là khả năng nắm bắt cơ bản của một số Prover để thuyết phục Verifier về luận điểm đúng đó (thuộc về một số định trước của một tập hợp các luận điểm đúng)

Giả sử, Peggy có gắng thuyết phục Victor là cố ấy biết một bí mật có thể chấp nhận được của một công thức toán logic φ. Cô ấy có thể làm điều này bằng cách gửi đi sự chuyển nhượng chân lý đáp ứng cho φ tới Victor. Thay vào đó, GMR [8] [6] chuẩn bị một kịch bản xác suất mà Victor bị thuyết phục bởi Peggy thực sự có một chuyển nhượng cho φ bằng cách trao đổi thật nhiều tin nhắn với anh ta và khẳng định rằng cô ta biết bí mật. Vì tính hợp lệ và tính đây đủ của phép chứng minh không tiết lộ thông tin, nếu như có một chân lý tồn tại, thì Peggy có thể làm như vậy mà Victor hầu như luôn luôn chấp nhận, nhưng nếu một chân lý như vậy không tồn tại, thì không có vấn đề gì mà Peggy làm, Victor chắc chắn sẽ từ chối.

Một ý niệm khác của xác suất đã đề xuất một cách độc lập bởi Babai [3] được gọi là giao thức Arthur – Merlin. Nó đã được hạn chế so với mô hình GMR bởi trong mô hình này, Verifier được yêu cầu phải tiết lộ tất cả các bit ngẫu nhiên ngay sau khi kết thúc giao thức.

2.2 PHÂN LOẠI ỨNG DỤNG XUẤT PHÁT TỪ THỰC TIỄN

Chứng minh không tiết lộ thông tin có rất nhiều ứng dụng. Các ứng dụng thực tiễn nhất được chia làm hai loại :

2.2.1 Thiết kế giao thức

Một giao thức là một thuật toán cho các bên tương tác để đạt được một số mục tiêu. Ví dụ, chúng ta thấy giao thức trao đổi khóa Diffie-Hellman. Trong giao thức này, chúng ta giả định rằng cả hai bên đều làm theo hướng dẫn của giao thức, và điều duy nhất ta lo lắng là ta đã trở thành một đối thủ thụ động.

Tuy nhiên, trong mật mã học, chúng ta muốn thiết kế các giao thức rằng cần phải đạt được bảo mật thậm chí khi một trong các bên "gian lận" và không theo hướng dẫn. Đây là một vấn đề khó khăn kể từ khi chúng ta không có cách để biết chính xác các bên sẽ "gian lận".

Tuy nhiên, một cách để tránh gian lận là như sau : Nếu Alice chạy một giao thức với Bob, để cho Bob biết rằng Alice không gian lận, cô ấy sẽ gửi toàn bộ dữ liệu đầu vào cho Bob và sau đó Bob có thể chứng thực điều này.

Tuy nhiên, cách này sẽ không được chấp nhận nhất là với Alice : lý do duy nhất là họ đang cùng chạy giao thức này và họ không thể thực sự tin tưởng đối phương, và những dữ liệu đầu vào mà cô ấy có là bí mật, và cô ấy không muốn chia sẽ chúng.

Chứng minh không tiết lộ thông tin cung cấp một giải pháp cho vấn đề này. Thay vì gửi hết các dữ liệu đầu vào của mình, Alice sẽ chứng minh không tiết lộ thông tin rằng cô ấy đã theo đúng hướng dẫn. Bob sẽ bị thuyết phục, nhưng cũng sẽ không biết được bất cứ điều gì về những dữ liệu đầu vào của Alice ngoài những cái anh ấy đã biết trước kia.

Thực tế, chúng ta sẽ thấy rằng có thể làm điều này một cách chung, áp dung cơ bản cho tất cả các giao thức mã hóa. Vì thế, một kỹ thuật tổng hợp (phát minh bởi Goldreich, Micali và Wigderson , GMW) [10] là để thiết kế ra một giao thức mật mã đầu tiên giả định tất cả mọi người sẽ phải làm theo hướng dẫn và sau đó "yêu cầu" sẽ bắt họ phải làm theo chỉ dẫn sử dụng hệ thống chứng minh không tiết lộ thông tin.

2.2.2 Đề án nhận dạng

Một phần đơn giản hơn và ứng dụng trực tiếp là đề án nhận dạng. Giả sử rằng chúng ta muốn kiểm soát truy cập vào các bộ phận khoa học máy tính. Một cách để

làm điều đó là cho người được ủy quyền (có thẩm quyền) một số bí mật là mã PIN, và có một ô trên cửa nơi gõ số PIN trên hộp đó. (Một cách thuận tiện hơn, nhưng về cơ bản cũng giống như cách người được ủy quyền có thẻ mà truyền số PIN cho hộp).

Một nhược điểm của phương pháp này là hộp vẫn còn ở bên ngoài suốt thời gian ấy, và nếu có ai đó có thể kiểm tra chiếc hộp, có lẽ họ sẽ có thể xem bộ nhớ của nó và trích xuất bí mật chìa khóa của tất cả mọi người. Như vậy, từ một quan điểm bảo mật, nó là tốt hơn nếu hộp không chứa thông tin bí mật nào cả, và thậm chí là nếu có ai đã cài đặt một hộp để giả mạo họ cũng sẽ không biết gì về những bí mật mã PIN.

Chứng minh không tiết lộ thông tin giúp ta theo các cách sau :

* Có hộp chứa một thể hiện của vấn đề khó khăn. Ví dụ, hộp có thể chứa n hợp số mà không cần phân tích nhân của nó.
* Cung cấp cho những người có thẩm quyền giải pháp cho vấn đề này. Ví dụ, họ có thể nhận phân tích nhân của n để n = p.q
* Những người có thẩm quyền sẽ chứng minh cho họ biết hộp phân tích nhân không tiết lộ thông tin.

Sau đây chúng tôi sẽ tập trung nghiên cứu một loại ứng dụng là “thiết kế giao thức” bằng việc đi sâu vào phân tích hai ví dụ nổi bật của loại ứng dụng này.

2.3 ỨNG DỤNG TRONG THĂM DÒ TỪ XA

Chúng ta đã biết một số kỹ thuật thăm dò ý kiến từ xa (các kỹ thuật này có trong bỏ phiếu điện tử - Electronic Voting). Cử tri giữ bí mật lá phiếu khi truyền từ xa tới ban kiểm phiếu bằng cách mã hoá nội dung lá phiếu. Theo kỹ thuật "mã hoá đồng cấu", ban kiểm phiếu có thể tính được kết quả thăm dò từ xa mà không cần phải giải mã nội dung lá phiếu. Vấn đề nảy sinh là cử tri phải chứng minh được với ban kiểm phiếu rằng **lá phiếu của mình là hợp lệ** nhưng **nội dung lá phiếu thì không được tiết lộ** với họ. Để thực hiện điều này, hiện nay người ta dùng kỹ thuật "Chứng minh không tiết lộ thông tin" (Zero-knowledge proof). Chúng tôi trình bày ý tưởng trên để thực hiện bỏ phiếu loại "Chọn 1 trong k".

Ở đây chúng ta coi "người được thăm dò" là cử tri để dễ xác định hoạt động.

2.3.1 Các khái niệm

1/ Vấn đề bỏ phiếu thăm dò từ xa (Electronic Voting) :

Nghiên cứu về "Bỏ phiếu thăm dò từ xa" là một chủ đề quan trọng đóng góp cho sự tiến bộ của xã hội dân chủ. Nếu một hệ thống bỏ phiếu thăm dò an toàn và tin cậy, nó sẽ được sử dụng thường xuyên để thu thập ý kiến của mọi người cho nhiều quyết định về chính trị và xã hội thông qua hệ thống tự động hóa. "Bỏ phiếu thăm dò từ xa'" cũng phải đạt được các tính chất như "bỏ phiếu truyền thống" [1]. Một qui trình bỏ phiếu gồm một số giai đoạn (công đoạn). Hiện nay có nhiều kỹ thuật mật mã để thực hiện hợp lý trong từng giai đoạn.

Trong luận văn này tôi xin trao đổi về giai đoạn ***Cử tri (CT) chuyển lá phiếu thăm dò tới Ban kiểm phiếu (Ban KP)*** cho sơ đồ bỏ phiếu loại "**Chọn 1 trong k**". Trong giai đoạn này người ta sử dụng kỹ thuật "Mã hóa đồng cấu - Chia sẻ bí mật" (Homomorphic Encryption – Secret Sharing) [1], kỹ thuật "Chứng minh không tiết lộ thông tin" (Zero-knowledge proof).

2/ Giai đoạn cử tri chuyển là phiếu đến ban kiểm phiếu :

Theo suy nghĩ thông thường, khi Cử tri (CT) chuyển lá phiếu tới Ban kiểm phiếu (Ban KP) thì họ chỉ cần mã hóa nội dung lá phiếu là đủ. Vì tiếp theo Ban KP chỉ cần giải mã nội dung lá phiếu là tính được kết quả (kiểm phiếu).

Nhưng trên thực tế có thể xảy ra các tình huống sau:

- Ban KP hay một nhóm thành viên Ban KP không trung thực đã gian lận phiếu thăm dò, ví dụ sửa lại nội dung lá phiếu sau khi giải mã (trước khi kiểm phiếu). Để khắc phục tình hình này, người ta dùng kỹ thuật "Mã hóa đồng cấu - Chia sẻ bí mật". Với giải pháp này Ban KP không phải giải mã từng lá phiếu nhưng vẫn tính được kết quả.
- Để bảo đảm công khai kiểm phiếu, lá phiếu đã mã hóa khi tới Ban KP phải được niêm yết công khai. Như vậy nhìn trên bảng niêm yết này, CT sẽ nhận ra lá phiếu của mình và họ có thể "***bán***" phiếu thăm dò . Để khắc phục tình trạng này, người ta dùng một "Người xác minh trung thực" (TT - honest verifier) làm trung gian giữa CT và Ban KP. Cử tri gửi lá phiếu từ xa tới Ban KP thông qua người trung gian TT. Sau khi xác minh lá phiếu hợp lệ, anh ta làm "mù " lá phiếu (mã hóa lá phiếu lần thứ 2), tiếp đó gửi nó về Ban KP. Trên bảng niêm yết công khai, CT không thể nhận ra lá phiếu của mình để có thể "***bán***" phiếu thăm dò".

Khi giải quyết 2 tình huống trên lại xuất hiện hai vấn đề khác:

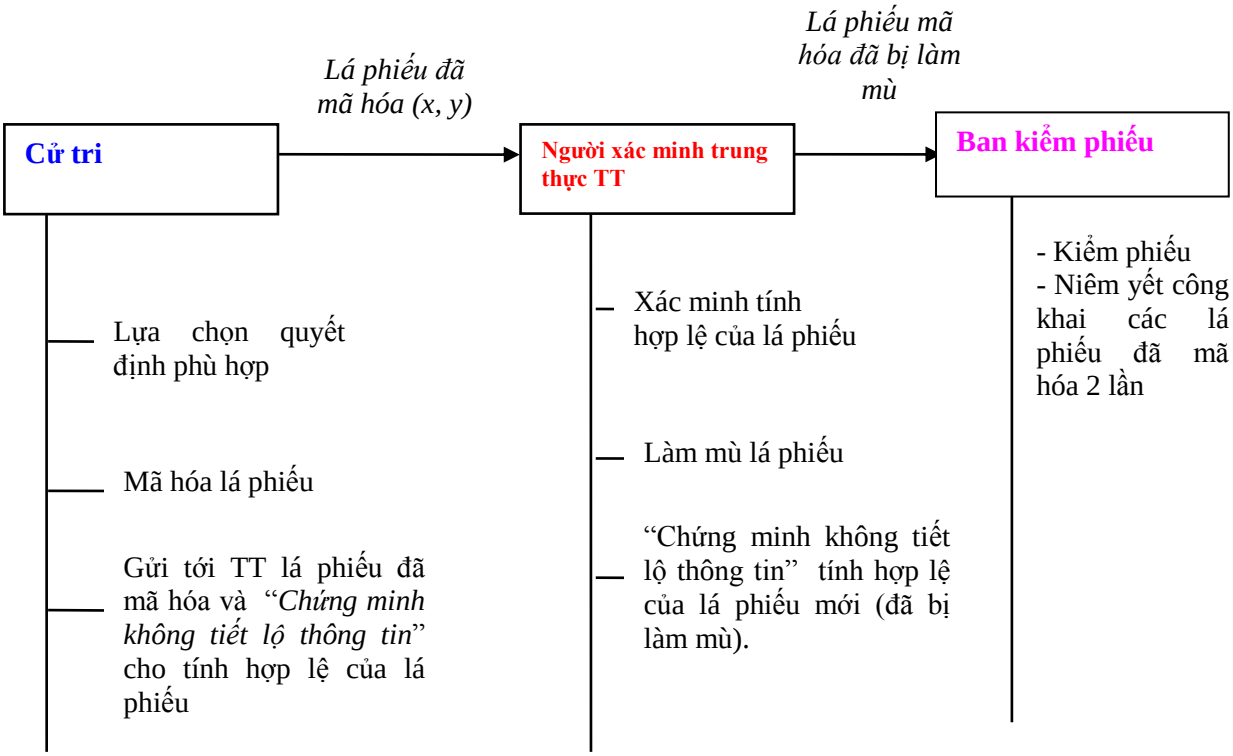
- Một là CT phải chứng minh cho TT biết lá phiếu của họ là hợp lệ, tức là nội dung lá phiếu chỉ ghi một trong số **k** lựa chọn (loại lựa chọn "chọn 1 trong **k**"), không cần phải chỉ rõ lá phiếu ghi rõ lựa chọn nào. Cách chứng minh như vậy gọi là "Chứng minh không tiết lộ thông tin". Với cách chứng minh này, nội dung lá phiếu không bị tiết lộ, trong khi mọi người đủ bằng chứng tin được rằng lá phiếu này là hợp lệ.
- Hai là TT phải chứng minh cho CT, Ban KP,…biết rằng lá phiếu bị làm "mù" vẫn hợp lệ (theo nghĩa trên) bằng cách chỉ ra rằng anh ta sở hữu giá trị β để là "mù" lá phiếu. TT chứng minh điều này cũng bằng phương pháp "Chứng minh không tiết lộ thông tin", tức là không cần phải tiết lộ chính giá trị β.

Sau đây là sơ đồ giai đoạn Cử tri (CT) chuyển lá phiếu tới Ban kiểm phiếu:

Giao thức 1: CT mã hóa lá phiếu bằng hệ mã hóa Elgamal, CT gửi nó tới TT kèm theo "Chứng minh không tiết lộ thông tin" cho tính hợp lệ của lá phiếu đó.

Giao thức 2: Sau khi xác minh lá phiếu hợp lệ, TT làm "mù" lá phiếu và gửi nó về Ban KP kèm theo "Chứng minh không tiết lộ thông tin" cho tính hợp lệ của lá

phiếu đã bị làm "mù". Cụ thể chứng minh quyền sở hữu giá trị bí mật β dùng để làm "mù" lá phiếu.



Hình 1 : Sơ đồ cử chi chuyển lá phiếu đến ban kiểm phiếu

2.3.2 Chứng minh tính hợp lệ của lá phiếu (x, y) (giao thức 1)

Theo sơ đồ giai đoạn Cử tri (CT) chuyển lá phiếu tới Ban kiểm phiếu, phải thực hiện *Giao thức 1*. Tức là CT mã hóa lá phiếu bằng hệ mã hóa Elgamal, lá phiếu đã mã hoá được gửi tới người xác minh trung thực (TT) kèm theo "Chứng minh không tiết lộ thông tin" cho tính hợp lệ của lá phiếu đó.

Trong cuộc thăm dò từ xa "Chọn 1 trong **k**", nếu cử tri nào đó chọn G_i là lựa chọn thứ i trong danh sách, thì ***lá phiếu hợp lệ*** phải ghi G_i với i là một trong các giá trị *1, 2, ..., k*. Bằng mã hóa Elgamal, lựa chọn G_i được mã hóa thành :

$$(x, y) = (g^{\alpha}, h^{\alpha} G_i).$$

Như vậy cử tri muốn chứng minh với người xác minh trung thực TT rằng lá phiếu (x,y) là hợp lệ thì anh ta phải chỉ ra một trong số k đẳng thức sau là đúng.

$$(\log_g x = \log_h (y / G_1)) \vee ... \vee ((\log_g x = \log_h (y / G_k)). \quad (1)$$

Để chứng minh (1) mà không bị lộ G_i, CT và TT thống nhất dùng giao thức "Chứng minh không tiết lộ thông tin" như sau:

Bảng 3 : Giai đoạn 1 cử tri chứng minh lá phiếu hợp lệ

Cử tri CT		**Người xác minh TT**
- Mã hóa lá phiếu $[(x,y)=(g^{\alpha},\ h^{\alpha}G_i)]$ - Chọn ngẫu nhiên $w \in Z_p$ Tính $a_i = g^w$, $b_i = h^w$		
- Với $j = 1,..., i-1, i+1,..., k$ chọn d_j, $r_j \in Zp$. (Chưa chọn d_i , r_i) Tính $a_j = g^{r_j} x^{d_j}$, $b_j = h^{r_j}(y/G_j)^{d_j}$ - Đặt $(A,B)=(a_1,b_1),...,(a_k,b_k)$ (Sử dụng a_i, b_i đã tính ở trên)	$\xrightarrow{(x,y),(A,B)}$	
- CT tính: (Trước đó chưa chọn d_i , r_i) $d_i = c - \sum_{j \neq i} d_j$ $r_i = w - \alpha d_i$ $(D,R)=(d_1,r_1),...,(d_k,r_k)$	$\xleftarrow{c}$ $\xrightarrow{(D,R)}$	- TT chọn ngẫu nhiên $c \in Z_p$ - TT kiểm tra: $c \overset{?}{=} d_1 + ... + d_k$ $cho\_j = 1,...,k$ $a_j \overset{?}{=} g^{r_j} x^{d_j}$ $b_j \overset{?}{=} h^{r_j}(y/G_j)^{d_j}$ Nếu đều đúng TT kết luận: Lá phiếu hợp lệ

Ví dụ 1: Chứng minh tính hợp lệ của lá phiếu đã mã hóa $(x, y) = (g^{\alpha}, h^{\alpha} G_i)$.

Giả sử cuộc thăm dò từ xa "chọn 1 trong 3". Các lựa chọn là 1 hoặc 2 hoặc 3.

Ký hiệu lựa chọn *i* là G_i . Để chứng minh tính hợp lệ của lá phiếu, cử tri phải chứng minh :

$$(\log_g x = \log_h (y / G_1)) \vee ... \vee ((\log_g x = \log_h (y / G_k)). \qquad (1)$$

Để chứng minh (1), CT và TT thống nhất dùng giao thức "Chứng minh không tiết lộ thông tin" như sau:

Chọn phần tử sinh $g=3$, $\alpha=5$, khóa bí mật $\boldsymbol{s=2}$, khóa công khai $\boldsymbol{h}=g^s=3^2$.

Ký hiệu 3 lựa chọn $G_1=1$, $G_2=2$, $G_3=3$. Giả sử cử tri CT chọn $G_i=\boldsymbol{2}$.

Cử tri CT		Người xác minh TT
- CT mã hóa lá phiếu $[(x, y) = (3^5, (3^2)^5.2]$ - CT chọn ngẫu nhiên w=2 Tính $a_2 = 3^2, b_2 = (3^2)^2$ - Với j = 1, 3 Chọn d_1=8, r_1=9 và tính: $a_1 = 3^9 . (3^5)^8$ $b_1 = (3^2)^9(\frac{(3^2)^5.2}{1})^8$ Chọn d_3=10, r_3=11 và tính: $a_3 = 3^{11} . (3^5)^{10}$ $b_3 = (3^2)^{11}(\frac{(3^2)^5.2}{3})^{10}$ $(A, B) = (3^9. (3^5)^8, (3^2)^5(\frac{(3^2)^5.2}{1})^8), (3^2, (3^7)^2),$ $(3^{11}. (3^5)^{10}, (3^2)^{11}(\frac{(3^2)^5.2}{3})^{10})$		
	$\xrightarrow{(x,y),(A,B)}$	TT chọn ngẫu nhiên c=13
	$\xleftarrow{c}$	
- CT tính $d_2 = c - \sum_{j \neq i} d_j$ $= c - (d_1+d_3) = 13-(8+10) = -5$ - CT tính $r_2 = w - \alpha d_i$ $= 2 - 5 d_2 = 2 -5.(-5) = 2 + 25 = 27$ - CT đặt (D,R)=(8, 9),(-5, 27),(10, 11)		
	$\xrightarrow{(D,R)}$	TT kiểm tra: thấy đều đúng $c=d_1+d_2+d_3=8+(-)+10=13$ $a_j = g^{r_j} x^{d_j}$ $b_j = h^{r_j} (y / G_j)^{d_j}$ j=1,2,3. =>Kết luận: lá phiếu hợp lệ

2.3.3 Chứng minh quyền sở hữu giá trị bí mật β (giao thức 2)

Theo sơ đồ giai đoạn Cử tri (CT) chuyển lá phiếu tới Ban kiểm phiếu (Ban KP), phải thực hiện *Giao thức 2*. Tức là sau khi xác minh lá phiếu của CT là hợp lệ, người xác minh trung thực (TT) làm "mù" lá phiếu và gửi nó về Ban KP kèm theo "Chứng minh không tiết lộ thông tin" cho tính hợp lệ của lá phiếu đã bị làm "mù".

TT làm "mù" lá phiếu thông qua cặp (*u, v*) dựa trên giá trị ***bí mật*** β. Như vậy để chứng minh lá phiếu đã bị làm "mù" vẫn hợp lệ, TT phải chứng minh rằng anh ta sở hữu giá trị bí mật β thõa mãn $u = g^{\beta}$, $v = h^{\beta}$. Nhưng mặt khác TT không muốn để lộ β. Có một giao thức hiệu quả để anh ta làm việc này: giao thức $\sum$ (đã trình bày ở mục trên).

Trong sơ đồ dưới đây, TT là người chứng minh (P), người kiểm tra (V) là CT, Ban KP…

Bảng 4 : Giai đoạn 2, TT chứng minh lá phiếu làm mù là hợp lệ

Người chứng minh TT (P)		**Người kiểm tra (V)**
- P có $[(u,v) = (g^{\beta}, h^{\beta})]$		
- P chọn $w \in Zp$		
Tính $(a,b) := (g^w, h^w)$	$\xrightarrow{(a,b)}$ P gửi V giá trị ngẫu nhiên w thông qua (a, b)	
	$\xleftarrow{c}$ V gửi lại P giá trị ngẫu nhiên c	- V chọn $c \in Z_p$
- P tính $r := w + \beta c$	$\xrightarrow{r}$ P đáp lại V bằng r	- Kiểm tra: $g^r \stackrel{?}{=} au^c$ $h^r \stackrel{?}{=} bv^c$ Nếu đều đúng ➔ V thừa nhận P sở hữu giá trị β

Chú ý :

Nếu không biết β, người chứng minh P không thể tạo ra: $r := w + \beta c$ để kiểm tra.

$$g^r = g^{w+\beta c} = g^w . g^{\beta c} = a u^c$$
$$h^r = h^{w+\beta c} = h^w h^{\beta c} = b v^c$$

Ví dụ 2:

Người chứng minh P chọn $g=3$, $s=2$, $h=g^s=3^2$. Anh ta có β=**5** sử dụng trong $(u,v) = (g^\beta, h^\beta)$=($3^5$, $(3^2)^5$), cặp số này dùng để làm "mù" lá phiếu đã mã hoá của cử tri.

P muốn chứng minh với V rằng anh ta sở hữu β mà không muốn để lộ giá trị β. P thực hiện giao thức $\sum$ với người xác minh V như sau:

Người chứng minh TT (P)		**Người kiểm tra (V)**
- P có $[(u,v)=(3^5,(3^2)^5)]$ - P chọn $w \in Z_p = 2$. Tính $(a, b) = (g^w, h^w) = (3^2, (3^2)^2)$	$\xrightarrow{(a,b)}$	
	$\xleftarrow{c}$	- V chọn c = 2
- P tính $r = w + \beta c = 2 + \mathbf{5} * 2 =$ 12	$\xrightarrow{r}$	- V kiểm tra các đẳng thức đều đúng: $g^r = 3^{12} = 3^2 (3^5)^2 = a u^c$ $h^r = (3^2)^{12} = (3^2)^2 ((3^2)^2)^5 = b v^c$ có ➔ V thừa nhận P sở hữu β = 5.

Nếu người nào đó giả mạo rằng đã biết β để tạo $(u,v)=(g^\beta, h^\beta)$ thì "*khó*" có thể tính được $r=w+\beta c$, tức là bước kiểm thử $g^r \stackrel{?}{=} a u^c$, $h^r \stackrel{?}{=} b v^c$ "khó" có thể thực hiện được.

Vì a, b, c, r, g, h, u, v đều công khai nên ai cũng có thể xác minh được $r = w+\beta c$.

Nhờ giao thức trên mọi người tin rằng người xác minh TT đã dùng β để làm "mù" lá phiếu.

2.3.4 Giai đoạn cử tri chuyển lá phiếu đến ban kiểm phiếu (phương án 2)

Trong mục 2.3.3 khóa luận đã trình bày giai đoạn Cử tri (CT) chuyển lá phiếu tới Ban kiểm phiếu (Ban KP). Nó được thực hiện bằng ***Giao thức 1*** và ***Giao thức 2***, ta,k gọi là phương án 1. Có phương án khác (tạm gọi là 2) cũng để thực hiện giai đoạn này bằng 2 giao thức. Giao thức 1 giống trong phương án 1. Giao thức 2 có thay đổi như sau:

Sau khi TT xác minh lá phiếu của CT là hợp lệ, sau khi CT xác minh TT sở hữu giá trị β thì chính CT làm "mù" lá phiếu và gửi nó về Ban KP (thay vì TT làm "mù" lá phiếu và gửi nó về Ban KP như theo giao thức 2 của phương án 1). Trong phương án này chúng tôi đề nghị: mỗi lần xử lý một lá phiếu, tại mỗi bước thử điều kiện, nếu không thoả mãn, công việc xử lý dừng lại với lá phiếu này để chuyển ngay sang lá phiếu tiếp theo.

Bảng 5 : Phương án 1 gồm 2 giai đoạn một và hai

Cử tri (CT)		**Người xác minh TT**
- CT mã hoá lá phiếu $[(x,y)=(g^{\alpha},\ h^{\alpha}G_i)]$ - CT chọn ngẫu nhiên $w_1 \in Z_p$ Tính $a_i = g^{w_1}$ $\quad b_i = h^{w_1}$		
- Tính với $j = 1,..,i-1,i+1,...,k$ chọn d_j, $r_j \in Z_p$, tính $a_j = g^{r_j} x^{d_j}$, $\quad b_j = h^{r_j}(y/G_j)^{d_j}$ - Đặt $(A,B)=(a_1,b_1),...,(a_k,b_k)$	$\xrightarrow{(x,y),(A,B)}$	
	$\xleftarrow{c_1}$	TT chọn ngẫu nhiên $c_1 \in Z_p$
- CT tính $d_i = c_1 - \sum_{j \neq i} d_j$ $r_i = w_1 - \alpha d_i$ $(D,R)=(d_i,r_i),...,(d_k,r_k)$	$\xrightarrow{(D,R)}$	

		- TT kiểm tra: $c_1 \overset{?}{=} d_1 + ... + d_k$ $cho\_j = 1,...,k$ $a_j \overset{?}{=} g^{r_j} x^{d_j}$ $b_j \overset{?}{=} h^{r_j} (y/G_j)^{d_j}$
		- Nếu điều kiện sai thì dừng giao thức và hủy bỏ lá phiếu. - Nếu đúng thì tiếp tục GT.
		- TT chọn β ngẫu nhiên bí mật và tính: $[(u,v) = (g^{\beta}, h^{\beta})]$
	$\xleftarrow{(a,b)}$ TT gửi CT giá trị w_2 thông qua (a, b)	- TT Chọn $w_2 \in Zp$, tính: $(a,b) := (g^{w_2}, h^{w_2})$
- CT chọn $c_2 \in Z_q$	$\xrightarrow{c_2}$ CT gửi lại TT giá trị c_2	
	$\xleftarrow{r}$ TT đáp lại CT bằng r	- TT tính $r := w_2 + \beta c_2$
- CT kiểm tra: $g^r \overset{?}{=} a u^{c_2}$ $h^r \overset{?}{=} b v^{c_2}$		
- Nếu điều kiện sai thì thực hiện lại giao thức vì có thể CT đang giao dịch với người mạo danh TT. - Nếu điều kiện đúng thì CT mã hóa lại (làm "mù") lá phiếu nhờ cặp (u, v) của TT: $(x',y')=(xu, yv)$		

2.4 ỨNG DỤNG TRONG SỬ DỤNG TIỀN ĐIỆN TỬ VÀ LƯỢC ĐỒ BRAND

Lược đồ Brand được xây dựng dựa trên chữ ký số Schnorr và bài toán đại diện trong nhóm cấp nguyên tố.

G_q là nhóm con cấp q của Z_p^*, trong đó p, q là số nguyên tố thoả mãn $q|(p-1)$

Ngân hàng khởi tạo 5 thành phần: (g, h, g_1, g_2, d).

* $(g, h) \in G_q$ (generator – tuple) : khoá công khai của ngân hàng được dùng trong sơ đồ ký ở giao thức rút tiền, x là khoá bí mật của ngân hàng.

$$x = \log_g h \quad (h = g^x)$$

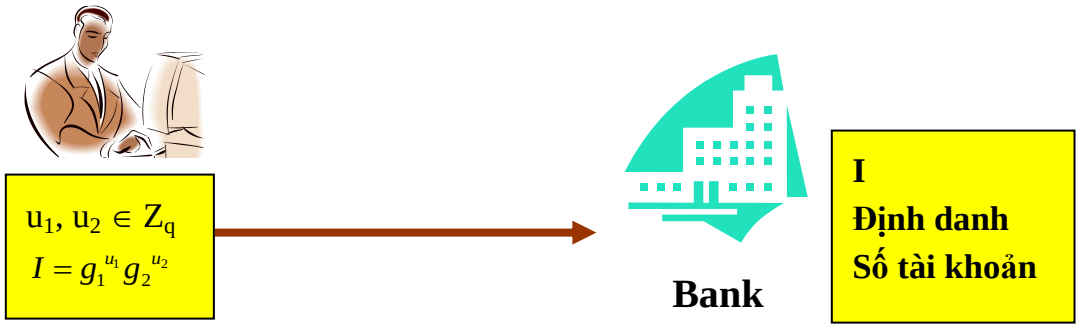
* (g_1, g_2): bộ phần tử sinh của G_q.
* Phần tử sinh giả d (khác g_1 và g_2), đảm bảo rằng định danh của người dùng sẽ không bị phát hiện trong giao thức thanh toán.

2.4.1 Khởi tạo tài khoản

b1) Alice tạo ngẫu nhiên $u_1, u_2 \in Z_q$, tính $I = g_1^{u_1} g_2^{u_2}$, chuyển I đến Ngân hàng.

b2) Ngân hàng lưu $I = g_1^{u_1} g_2^{u_2}$ cùng định danh của Alice và số tài khoản, nhưng ngân hàng không biết u_1 và u_2.

Trường hợp Alice tiêu đồng tiền hai lần, ngân hàng có thể tìm ra (u_1, u_2) và tính được I, từ I tìm ra định danh kẻ gian lận.



Hình 2 : Quá trình khởi tạo tài khoản

2.4.2 Chứng minh đại diện tài khoản

Khi Alice rút tiền, đầu tiên phải xưng danh với ngân hàng, bằng cách chứng minh với ngân hàng là sẽ rút tiền trong tài khoản mà Alice sở hữu.

Phương pháp được dùng ở đây là "**chứng minh không tiết lộ thông tin**".

Alice phải chứng minh cho Ngân hàng rằng: Alice biết u_1 và u_2 (vì Alice là chủ sở hữu tài khoản), nhưng không tiết lộ giá trị u_1, u_2 cho ngân hàng.

Quá trình xác thực được tiến hành như sau:

b1) Alice chọn ngẫu nhiên $w_1, w_2 \in Z_q$ và gửi $y = g_1^{w_1} g_2^{w_2}$ đến Ngân hàng.

b2) Ngân hàng thử thách để kiểm tra có đúng Alice sở hữu tài khoản không, bằng cách chọn ngẫu nhiên $C_r \in Z_q$ và gửi đến Alice.

b3) Alice tính $r_1 = w_1 + C_r u_1 \bmod q$, $r_2 = w_2 + C_r u_2 \bmod q$, gửi đến Ngân hàng.

b4) Ngân hàng chấp nhận xác thực là đúng khi và chỉ khi:

$$yI^{C_r} = g_1^{r_1} g_2^{r_2} \text{ trong đó } I = g_1^{u_1} g_2^{u_2}$$

Bởi vì, nếu Alice thực sự là chủ sở hữu tài khoản, thì phải biết u_1, u_2 (là 2 giá trị khởi tạo tài khoản lúc ban đầu) và nếu biết được chúng thì:

$$yI^{C_r} \equiv g_1^{w_1} g_2^{w_2} (g_1^{u_1} g_2^{u_2})^{C_r} \equiv g_1^{w_1 + u_1 C_r} g_2^{w_2 + u_2 C_r} \equiv g_1^{r_1} g_2^{r_2}$$

Bảng 6 : Quá trình chứng minh đại diện

Alice (người chứng minh)	***Ngân hàng (người kiểm tra)***
Biết u_1, u_2 là đại diện của $I = g_1^{u_1} g_2^{u_2}$	Chỉ biết I, g_1, g_2; không biết u_1, u_2
Tạo 2 số ngẫu nhiên $w_1, w_2 \in Z_q$	
Tính $y = g_1^{w_1} g_2^{w_2}$ gửi đến ngân hàng	Nhận y, chọn ngẫu nhiên $C_r \in Z_q$
	Gửi thử thách C_r đến Alice
Nhận C_r, tính: $r_1 = w_1 + C_r u_1 \bmod q$, $r_2 = w_2 + C_r u_2 \bmod q$ Và gửi chúng đến Ngân hàng	Nhận r_1, r_2. Kiểm tra: $yI^{C_r} = g_1^{r_1} g_2^{r_2}$ Nếu thoả mãn, Ngân hàng chấp nhận Alice biết đại diện của I (có nghĩa là biết u_1, u_2)

2.4.3 Giao thức rút tiền.

Nếu xác thực được chấp nhận, thì quá trình rút tiền được tiến hành như sau :

b1) Ngân hàng trừ một lượng tiền tương ứng từ tài khoản Alice. Ngân hàng và Allice cùng tính được $m = Id$ (d là phần tử sinh và công khai).

a. Ngân hàng gửi Alice: $\boldsymbol{z} = m^x$, $a = g^w$, $b = m^w$

b. (w được chọn ngẫu nhiên từ Z_q, x là khoá bí mật của ngân hàng).

b2) Alice chọn 3 số ngẫu nhiên $s \in Z_q^*$; $u, v \in Z_q$ để làm "mù" m, z, a, b

c. $m' = m^s = (Id)^s = g_1^{u_1 s} g_1^{u_2 s} d^s$

d. $z' = z^s$; $a' = a^u g^v$; $b' = b^{su} m^{sv}$

e. Tách ngẫu nhiên:

f. $u_1 s = (x_1 + x_2) \bmod q$, $u_2 s = (y_1 + y_2) \bmod q$

g. với $s = z_1 + z_2 \bmod q$

h. Tính $A = g_1^{x_1} g_2^{y_1} d^{z_1}$; $B = m'/A = g_1^{x_2} g_2^{y_2} d^{z_2}$

b3) Alice dùng hàm băm H tính $c' = H(m', z', a', b', A)$.

Làm "mù" c' bằng $c = \frac{c'}{u} \bmod q$, gửi c đến ngân hàng.

b4) Ngân hàng ký trên c được $r = xc + w \bmod q$, gửi r cho Alice, ghi có vào tài khoản của Alice.

Alice chấp nhận nếu kiểm tra thấy $g^r = h^c a$ và $m^r = z^c b$ và tính $r' = ru + v \bmod q$.

Lúc này, Alice có đồng tiền điện tử thật sự được đại diện bởi: $A, B, Sign(A, B)$ với $Sign(A, B) = (z', a', b', r')$ là chữ ký của Ngân hàng.

Nhưng làm thế nào chúng ta có thể biết được giá trị của từng đồng tiền.

Có hai cách khác nhau để giải quyết vấn đề này:

Cách 1: Ngân hàng sử dụng một khoá công khai cho mỗi loại tiền. Nghĩa là, nếu có k đồng tiền khác biệt thì ngân hàng phải công khai k khoá công khai sau: $(g_1, h_1) \ldots (g_k, h_k)$.

Cách 2: Chọn k phần tử sinh giả (dummy generator) khác nhau được công khai $d_1, ..., d_k$. Mỗi phần tử sinh được dùng để biểu hiện giá trị của mỗi đồng tiền.

Bảng 7 : Giao thức rút tiền

Alice		**Ngân hàng**
$I = g_1^{u_1} g_2^{u_2}$		x : khoá bí mật *(g, h):* khoá công khai ($h = g^x$)
		$w \in Z_q$
	z, a, b ←	$m = Id$, $\mathbf{z} = m^x$, $a = g^w$, $b = m^w$
$m = Id = g_1^{u_1} g_2^{u_2} d$ $s \in Z_q^*$ $m' = m^s = (Id)^s = g_1^{u_1 s} g_1^{u_2 s} d^s$ $z' = z^s$ Tách ngẫu nhiên $u, v \in Z_q$ $a' = a^u g^v$ $b' = b^{su} m^{sv}$ *c' = H (m', z', a', b', A).* $c = \frac{c'}{u} \bmod q$	c →	
	r ←	$r = x\mathbf{c} + w \bmod q$
Kiểm tra: $g^r = h^c a$ và $m^r = z^c b$ Tính r' = $ru + v \bmod q$ Đồng tiền: *(A, B, Sign (A, B))* với *Sign (A, B) = (z', a', b', r')*		

2.4.4 Giao thức thanh toán

Khi Alice muốn mua hàng hay sử dụng dịch vụ của Bob, trước tiên Alice cần phải gửi tiền cho Bob, quá trình thanh toán được thực hiện theo những bước sau:

b1) Alice gửi tiền *(A, B, Sign (A, B))* đến Bob.

$A = g_1^{x_1} g_2^{y_1} d^{z_1}$; $B = m' / A = g_1^{x_2} g_2^{y_2} d^{z_2}$

Sign (A, B) = (z', a', b', r')

b2) Đầu tiên, Bob kiểm tra xem $AB \neq 1$ hay không.

Nếu $AB = 1$, có nghĩa:

$$(g_1^{x_1} g_2^{y_1} d^{z_1})(g_1^{x_2} g_2^{y_2} d^{z_2}) = 1$$
$$\Rightarrow g_1^{x_1+x_2} g_2^{y_1+y_2} d^{z_1+z_2} = g_1^{u_1 s} g_2^{u_2 s} d^{s} = 1$$
$$\Rightarrow s = 0$$

Vậy, ngân hàng không xác định được u_1, u_2 trong trường hợp "double-spending".

Sau đó, Bob kiểm tra chữ ký của ngân hàng *sign(A,B)* có hợp lệ không.

Nếu đúng, Bob thử thách Alice bằng cách gửi $c \in Z_q^*$, c không cần thiết là số ngẫu nhiên, nhưng phải đảm bảo là duy nhất trong mỗi lần thanh toán.

Bob tính c như sau:

$c = H_0$ *(A, B,* I_b*, date/time),* với I là định danh của Bob, date/time là nhãn thời gian của giao dịch, H_0 là hàm băm.

b3) Alice phản hồi với:

$$r_1 = x_1 + cx_2 \ mod\ q$$

$$r_2 = x_2 + cy_2 \ mod\ q$$

$$r_3 = x_3 + cz_2 \ mod\ q$$

b4) Bob kiểm tra, nếu $g_1^{r_1} g_2^{r_2} g_2^{r_3} = AB^C$ thì chấp nhận thanh toán vì:

$$g_1^{r_1} g_2^{r_2} g_2^{r_3} = g_1^{x_1+cx_2} g_2^{y_1+cy_2} g_3^{z_1+cz_2} = (g_1^{x_1} g_2^{y_1} g_3^{z_1})(g_1^{cx_2} g_2^{cy_2} g_3^{cz_2}) = AB^C$$

Bảng 8 : Giao thức thanh toán

Alice		**Bob**
	$A, B, sign(A, B)$ →	
		$AB \neq 1?$ Kiểm tra $sign(A, B)$ $c \in Z_q^*$
	← c	
$r_1 = x_1 + cx_2 \mod q$ $r_2 = x_2 + cy_2 \mod q$ $r_3 = x_3 + cz_2 \mod q$		
	r_1, r_2, r_3 →	
		$g_1^{r_1} g_2^{r_2} g_2^{r_3} = AB^C$? Nếu đúng Bob chấp nhận thanh toán.

2.4.5 Giao thức gửi

b1) Bob gửi thông tin thanh toán *(A, B, Sign (A, B))*, c, r_1, r_2 và r_3 đến ngân hàng.

b2) Ngân kiểm tra chữ ký có chính xác không và đồng tiền không được tiêu xài trước đó.

Bob thử thách Alice bằng giá trị $c = H_0 (A, B, I_b, date/time)$

Alice trả lời lại giá trị r_1, r_2, r_3.

Nếu tất cả đều thoả mãn, Ngân hàng gửi tiền vào tài khoản của Bob.

Chương 3 : THỬ NGHIỆM CHƯƠNG TRÌNH VỚI ỨNG DỤNG TRONG THĂM DÒ TỪ XA

3.1 MÔ TẢ CHƯƠNG TRÌNH

3.1.1 Giới thiệu

Chương trình mô phỏng hai giao thức trên được thiết kế trên nền WEB cho các ứng dụng trên Internet được viết bằng ngôn ngữ PHP và có cài đặt chương trình Xampp để hỗ trợ và Notepad++ cho việc lập trình. Do được ứng dụng trong thăm dò ý kiến từ xa nên việc chương trình được chạy trên nền WEB là một lựa chọn đúng đắn.

1/Cấu hình của hệ thống :

Phần cứng(cấu hình tối thiểu) :

Bộ nhớ ổ cứng : 20gb

Bộ nhớ ram : 128mb

Tốc độ máy tối thiểu : 1GHz

Phần mềm :

Hệ điều hành : Linux, Window,…

Ngôn ngữ lập trình : PHP, HTML, CSS

2/ Các thành phần của chương trình :

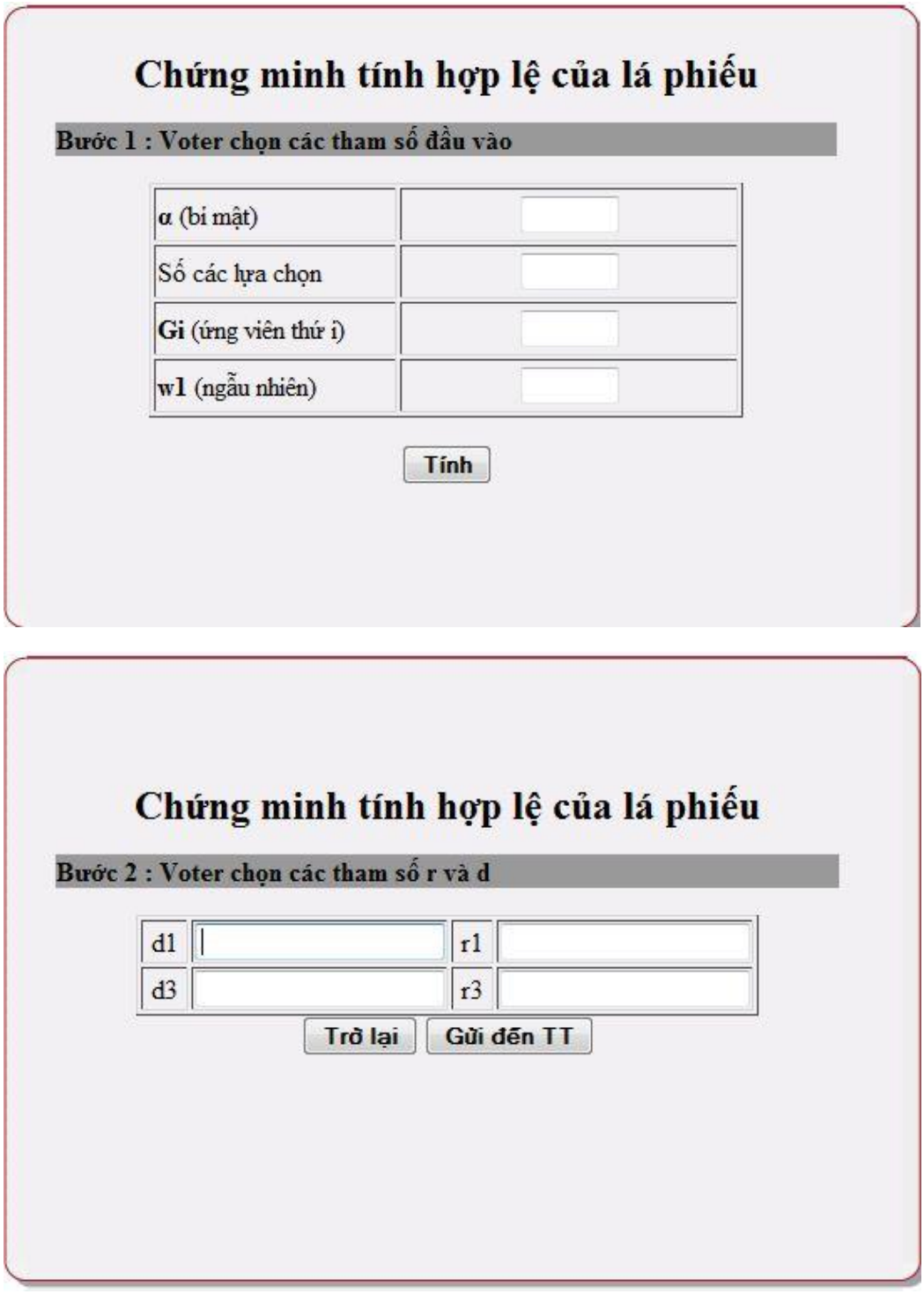
Chương trình thực hiện trên nền Web nên các thành phần của nó gồm các trang web và các mẫu điền thông tin đầu vào của người dùng.

3.1.2 Mô tả các chức năng chính

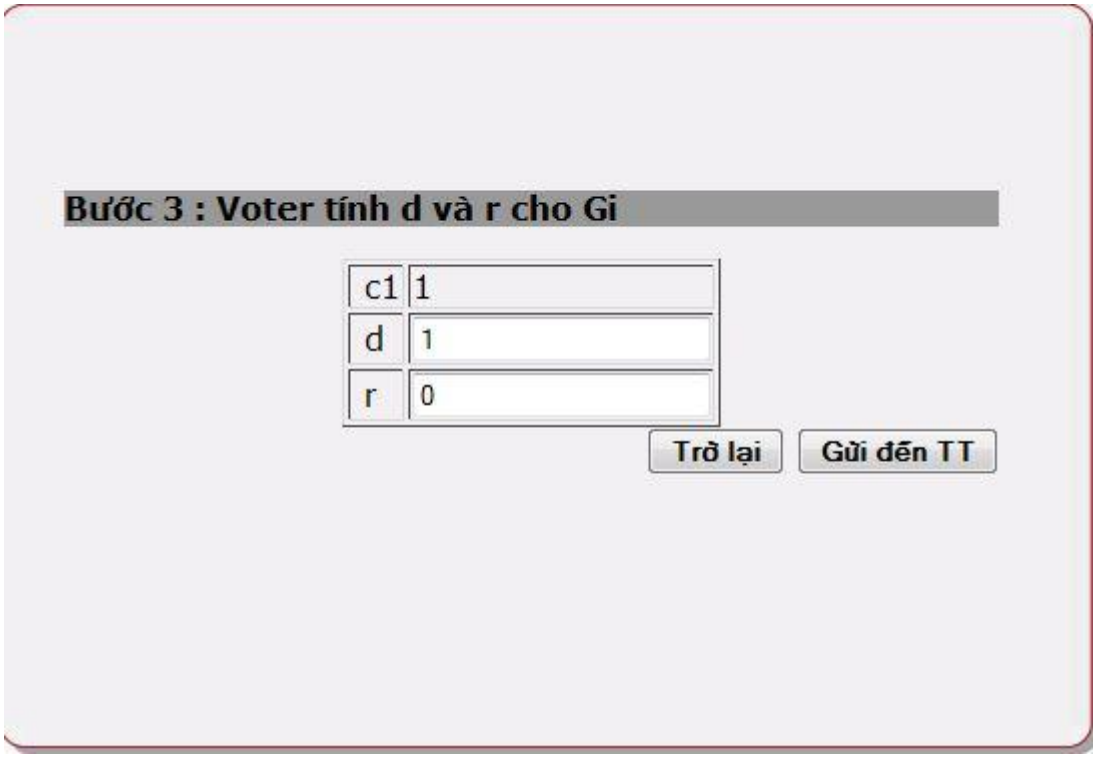
1/ Giao thức 1 :

CT chứng minh tính hợp lệ của lá phiếu sau khi đã được mã hóa và gửi đến TT :

Bước 1 : Với việc Cử tri điền các thông tin cần thiết để có thể mã hóa lá phiếu :

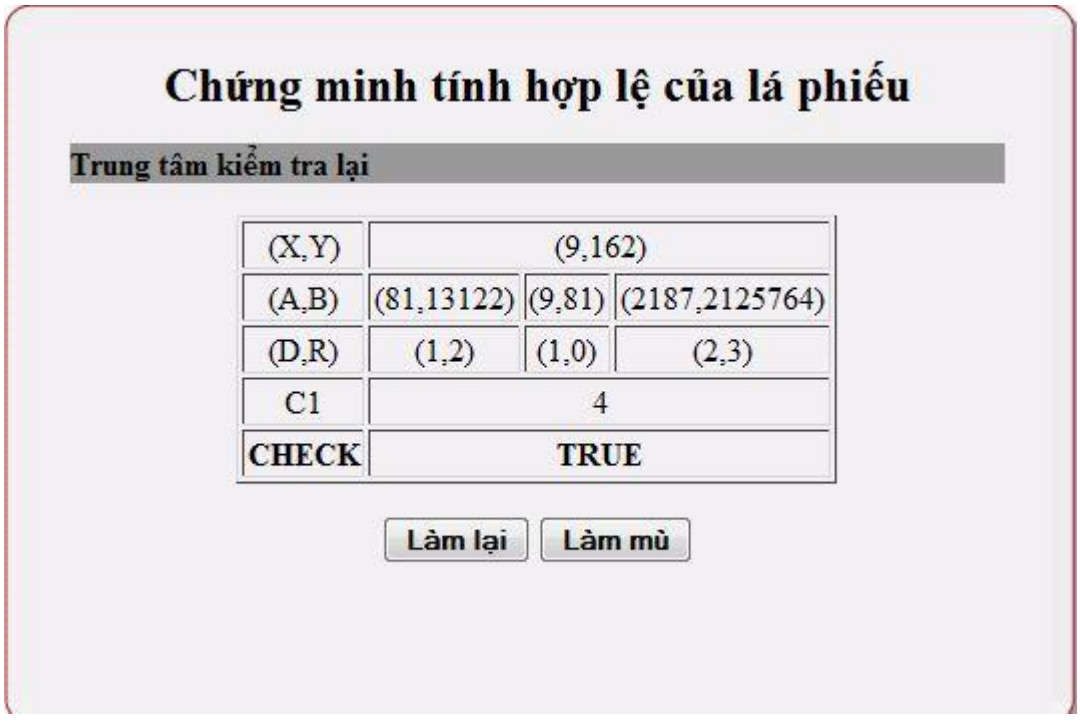


Hình 3 : CT điền các thông tin cần thiết để mã hóa lá phiếu thăm dò



Hình 4 : Các thông số trả về từ TT và các tính toán của CT

Bước 2 : Sau khi đã tính toán hết các tham số còn lại, cử tri sẽ gửi lại cho TT, TT kiểm tra, nếu các tham số không thỏa mãn thì sẽ loại lá phiếu, nếu đúng sẽ chấp nhận và tiếp tục mã hóa lá phiếu lần 2 và gửi cho ban KP:

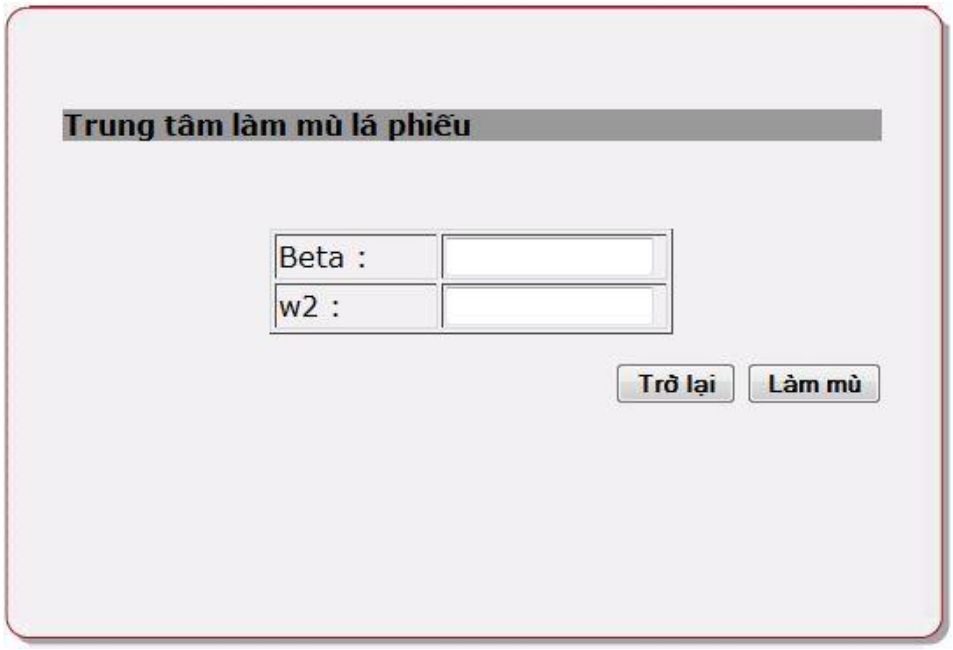


Hình 5 : Lá phiếu khi đã được TT kiểm tra lại

2/ Giao thức 2

TT chứng minh lá phiếu làm mù gửi tới ban KP cũng hoàn toàn hợp lệ :

Bước 1 : TT sẽ điền các tham số đầu vào và tính toán, sau đó gửi cho CT:

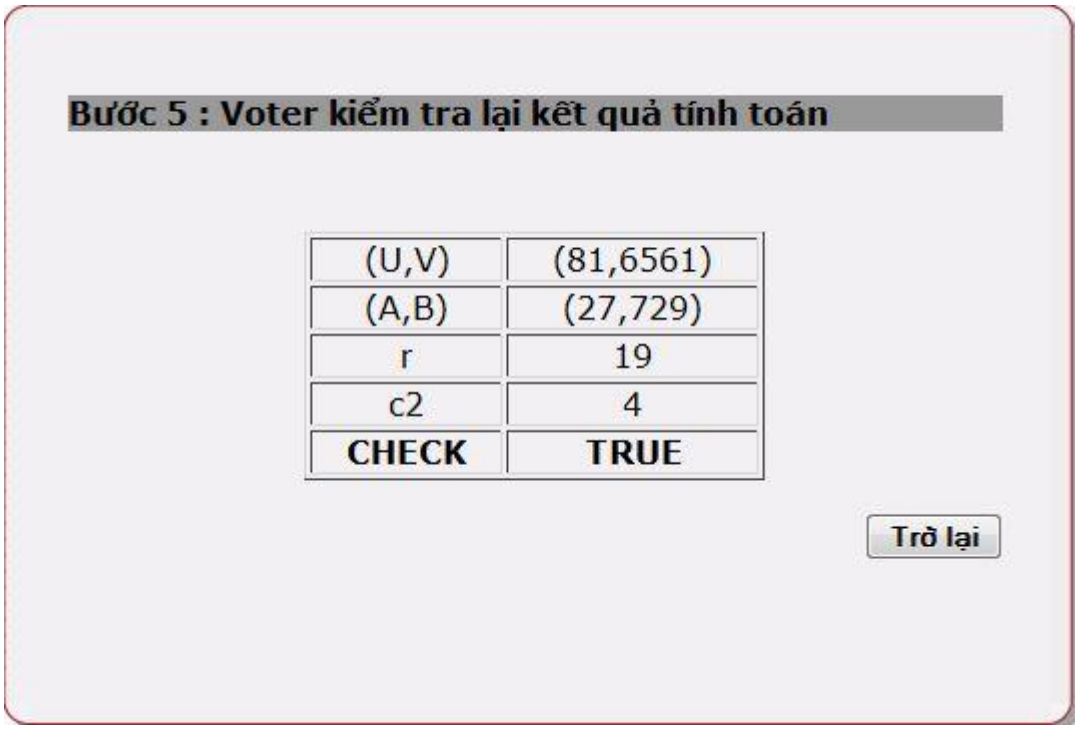


Hình 6 : TT tính Beta và w_2

Bước 2 : Sau đó TT sẽ gửi luôn Beta và w_2 cho CT, CT trả lại giá trị c_2 và TT sẽ tính toán r.

Hình 7 : TT tính r

Bước 3 : Cử tri kiểm tra lại kết quả nhận được, nếu đúng thì lá phiếu làm mù lần 2 hoàn toàn hợp lệ, nếu không đúng, CT sẽ không chấp nhận.



Hình 8 : CT kiểm tra lại kết quả

3.2 THÀNH PHẦN CHÍNH CỦA CHƯƠNG TRÌNH

3.2.1 Cử tri chứng minh tính hợp lệ của lá phiếu

```
//Lay Mang A
function getArrayAj($d, $r, $gi, $x, $numOfCans, $w1) {
  global $g;
  $a = array();

  for ($i=1; $i<$gi; $i++) {
        $a[$i] = luythua($g, $r[$i]) * luythua($x, $d[$i]);
  }
  $a[$gi] = getAi($w1);
  for ($i=$gi+1; $i<=$numOfCans; $i++) {
        $a[$i] = luythua($g, $r[$i]) * luythua($x, $d[$i]);
  }
  return $a;
}

//Lay mang B
function getArrayBj($d, $r, $gi, $y, $numOfCans, $w1) {
  global $h;
  $b = array();

  for ($i=1; $i<$gi; $i++) {
        $b[$i] = luythua($h, $r[$i]) * luythua(($y/$i), $d[$i]);
  }
  $b[$gi] = getBi($w1);
  for ($i=$gi+1; $i<=$numOfCans; $i++) {
```

```
        $b[$i] = luythua($h, $r[$i]) * luythua(($y/$i), $d[$i]);
    }
    return $b;
}
//Nguoi trung thuc kiem tra la phieu hop le
function check($g, $h, $a, $b, $d, $r, $x, $y, $c1) {
    //global $g, $h, $a, $b, $d, $r, $x, $y, $c1;
    $sum = 0;
    for ($i=1; $i<=count($d); $i++) {
        $sum = $sum+$d[$i];
    }
    if($sum != $c1) return false;
    for($i=1; $i<=count($a); $i++) {
        if($a[$i] != luythua($g, $r[$i])*luythua($x, $d[$i])) return false;
        if($b[$i] != luythua($h, $r[$i])*luythua(($y/$i), $d[$i])) return false;
    }
    return true;
}
```

3.2.2 Người trung thực chứng minh có giữ tham số bí mật β

```
//Lay gia tri Di voi i la vi tri nguoi duoc chon
function getDi($d, $c1, $gi, $numOfCans) {
    $di = $c1;
    for ($i=1; $i<=$numOfCans; $i++) {
        $di = $di-$d[$i];
    }
    return $di;
```

```
}

//Lay gia tri Ri voi i la vi tri nguoi duoc chon
function getRi($w1, $alpha, $di) {
  return $w1-$alpha*$di;
}

//Them D[gi] vao mang D
function addDgiToD($c1, $gi, $numOfCans) {
  global $d;
  $d[$gi] = getDi($d, $c1, $gi, $numOfCans);
}

//Them R[gi] vao mang R
function addRgiToR($c1, $gi, $numOfCans, $w1, $alpha) {
  global $r, $d;
  $r[$gi] = getRi($w1, $alpha, getDi($d, $c1, $gi, $numOfCans));

}
```

//Voter kiem tra lai ket qua chung minh nguoi trung thuc giu tham so bi mat β

```
function voterCheck($r, $c2, $a, $b, $u, $v) {
  global $g, $h;
  if (luythua($g, $r) != luythua($u, $c2)*$a) return false;
  if (luythua($h, $r) != luythua($v, $c2)*$b) return false;
  return true;
}
```

KẾT LUẬN

"Chứng minh không tiết lộ thông tin" không có nghĩa là "không để lộ thông tin" mà nghĩa là "để lộ thông tin ở mức ít nhất" về sự vật sự việc cần chứng minh. Với những "thông tin để lộ", người xác minh không có nhiều hiểu biết (knowledge) về sự vật sự việc, họ chỉ thu được chút ít thông tin (coi như "zero knowledge") về đặc điểm tính chất của nó.

Kết quả chính của khóa luận gồm có :

1. Tìm hiểu và nghiên cứu qua tài liệu để hệ thống lại các vấn đề sau :
   - Các khái niệm và thuật toán cơ bản
   - Vấn đề "chứng minh không tiết lộ thông tin"
   - "Chứng minh không tiết lộ thông tin" trong thăm dò từ xa
   - "Chứng minh không tiết lộ thông tin" trong tiền điện tử
2. Thử nghiệm chương trình trong việc ứng dụng "chứng minh không tiết lộ thông tin" trong thăm dò từ xa.

TÀI LIỆU THAM KHẢO

[1] Andrew Neff, *"Conducting a Universally Verifiable Electronic Election Using Homomorphic Encryption "*, VoteHere.net, November 2000

[2] Berry Schoenmakers, *"A brief Comparision of Cryptographic Schemes for Electronic Voting"*, Tartu, Estonia, May 17, 2004

[3] Byoungcheon Lee, Kwangjo Kim, *"Receipt-free Electronic Voting through Collaboration of Voter and honest Verifier"*

[4] C. E. Shannon "*Communication Theory of Secrecy Systems*", Bell Systems Tech. Jr. Vol 28, pages 656-715, 1949

[5] Goldreich, Micali and Wigderson "Zero-Knowledge and Secure Computation", July 1991

[6] Helger Lipmaa, *"Zero knowledge and some applications"*, Nordic Research Training course, Bergen, June 15, 2004

[7] Information Security Research Centre, Faculty of Information Technology, Queensland University of Technology, *"Electronic Voting and Cryptography"*, May 2002

[8] Ivan Damgard, Jens Groth and Gorm Salomonsen, *"The Theory and Implementation of an Electronic Voting System"*, July 31,2002

[9] Trịnh Nhật Tiến, Nguyễn Đình Nam, Trương Thị Thu Hiền, "*Một số kỹ thuật Bỏ phiếu từ xa",* Hội thảo Một số vấn đề chọn lọc của Công nghệ thông tin, Thái Nguyên, tháng 8 năm 2003

[10] Trịnh Nhật Tiến, Trương Thị Thu Hiền, "*Mã hóa đồng cấu và ứng dụng",* Hội nghị khoa học cơ bản và ứng dụng CNTT toàn quốc lần thứ 1, Đại học Quốc Gia Hà Nội, tháng 10 năm 2003