MỤC LỤC

LỜI CẢM ƠN

Đầu tiên, tôi xin gửi lời cảm ơn chân thành và sâu sắc nhất tới PGS.TS Trịnh Nhật Tiến, người thầy đã nhiệt tình hướng dẫn và truyền đạt những kiến thức cần thiết, để tôi hoàn thành khóa luận này.

Tôi xin gửi lời cảm ơn tới gia đình, chính là nguồn lực động viên tôi phấn đấu trong học tập và cuộc sống. Tôi cũng xin cảm ơn các thầy, cô giáo của khoa Công nghệ thông tin, Trường Đại học dân lập Hải Phòng đã tận tình dạy dỗ, chỉ bảo tôi trong suốt những năm học ở trường

Tôi xin gửi lời cảm ơn tới các bạn sinh viên trong lớp CT1001, Khoa Công Nghệ Thông Tin, Trường Đại Học Dân Lập Hải Phòng đã cho tôi một môi trường rất tốt để học tập.

Tuy có nhiều cố gắng trong quá trình học tập cũng như thời gian làm khóa luận nhưng không thể tránh khỏi những thiếu sót, tôi rất mong được sự góp ý quý báu của tất cả các thầy cô giáo và các bạn để khóa luận của tôi được hoàn thiện.

Tôi xin chân thành cảm ơn!!

Hải Phòng ,ngày 10 tháng 7 năm 2010

Sinh Viên

NGUYỄN THỊ THÚY

MỞ ĐẦU

Bước vào thời kì kinh tế tri thức, khi tri thức này càng trở lên đắt giá, đồng thời với đó, các tài liệu trong máy tính hay tài liệu truyền qua mạng máy tính được biểu diễn dưới dạng số hóa (chỉ dùng số 0 và số 1), ta có thể gọi tài liệu số, ngày càng nhiều và phổ biến, thì vấn đề bảo vệ bản quyền cho tri thức của con người ngày càng trở lên quan trọng, bởi những đặc trưng tài liệu số:

Dễ dàng sao chép: Chỉ cần một vài thao tác đơn giản như click chuột, một cuốn tiểu thuyết dày hàng nghìn trang, hay một tác phẩm trị giá nhiều triệu đô la của danh họa Picasso có thể được sao chép chỉ trong vài giây. Điều quan trọng hơn nữa là khi sao chép tài liệu số thì chất lượng bản sao chép được giữ nguyên so với bản gốc.

Dễ dàng phát tán: . Ngày nay, chỉ sau vài phút tìm kiếm trên mạng, người sử dụng có thể dễ dàng tìm và tải về những bộ phim mới nhất còn chưa được trình chiếu ở rạp. Cùng với đó, một người sử dụng bình thường có thể trở thành nguồn phát tán tài liệu cũng rất dễ dàng, thông qua các tin nhăn tức thời(IM_Instant Message), email hay các dịch vụ chia sẻ file trực tuyến(online file sharing service).

Dễ dàng lưu trữ: dung lượng ổ cứng ngày càng lớn, giá thành các thiết bị lưu trữ ngày càng rẻ đã khiến cho việc lưu trữ các tà liệu số hóa trở lên đơn giản hơn bao giờ hết.

Vì vậy, khi trao đổi thông tin trên mạng, những tình huống mới nảy sinh:

Người ta nhận được một bản tin trên mạng, thì lấy gì làm đảm bảo rằng nó là của đối tác đã gửi cho họ. Khi nhận được tờ Sec điện tử hay tiền điện tử trên mạng, thì có cách nào để xác nhận rằng nó là của đối tác đã thanh toán cho ta. Tiền đó là tiền thật hay giả?

Thông thường, người gửi văn bản quan trọng phải ký phía dưới. Nhưng khi truyền trên mạng, văn bản hay giấy thanh toán có thể bị trộm cắp và phía dưới nó có thể dán một chữ ký khác

Để giải quyết tình hình trên và để đảm bảo cho nhu cầu giữ bí mật thông tin liên lạc cũng như đảm bảo an toàn dữ liệu, từ lâu con người đã phát minh ra một số công cụ hết sức hiệu quả như:

Mã hóa được hiểu là thay đổi hình dạng thông tin gốc, khiến người khác khó nhận ra, tức là giấu đi ý nghĩa của thông tin gốc. Mã hóa là một công cụ mạnh, và có lịch sử lâu đời, đã có nhiều kết quả nghiên cứu thành công và có ứng dụng rất lớn trong việc đảm bảo an toàn thông tin liên lạc.

Chữ kí số (digital signature) là đoạn dữ liệu ngắn đính kèm với văn bản gốc thực tác giả (người kí văn bản) của văn bản và giúp người nhận kiểm tra tính nội dung văn bản gốc.

Thủy vân (watermarking) là một ứng dụng đã có từ lâu đời để bảo vệ bản quyền cho các cuốn sách. Tuy nhiên, thủy vân số (digital watermarking) lại là một lĩnh vực mới, đang nhận được nhiều sự quan tâm cũng như nghiên cứu của chuyên gia trên thế giới. Sử dụng thủy vân số có thể thay đổi và tác động vào chất lượng của tài liệu số như ý muốn, đồng thời với đó là thủy vân số có thể gắn liền với tài liệu, đảm bảo tài liệu được bảo vệ bản quyền cho tới khi bị hủy hoại.

Hàm băm (hash function) là hàm có nhiệm vụ "lọc" (băm) tài liệu (bản tin) và cho kết quả là một giá trị "băm"có kích thước cố định, còn gọi là "đại diện tài liệu" hay "đại diện bản tin", "đại diện thông điệp đệm". Nhờ đó ta có thể đảm bảo tài liệu được vẹn toàn trên đường truyền.

Trong nội dung khóa luận này, tôi xin tập trung trình bày những kết quả nghiên cứu đã đạt được trong việc ứng dụng các phương pháp bảo vệ bản quyền tài liệu số.

BẢNG CÁC CHỮ VIẾT TẮT, THUẬT NGỮ

Viết tắt	Tiếng anh	Tiếng việt
RSA	Rivest, Shamir, Adleman	Tên riêng
LSB	Least Significant Bit	Bit có trọng số thấp
DCT	Discrete Cosine Transform	Biến đổi cosine rời rạc
FFT	Fast Fourier Transform	Biến đổi Fourier nhanh
PN	Pseu-random Number	Dãy giả ngẫu nhiên
MD	Message Digist	Thông báo Digist
BSCNN		Bội số chung nhỏ nhất
USCLN		Ước số chung lớn nhất
DWT	Discrete Wavelet Transform	Biến đổi sóng rời rạc

Chương 1. MỘT SỐ KHÁI NIỆM TRONG TOÁN HỌC

1.1. TÍNH CHIA HẾT VÀ SỐ NGUYÊN TỐ

1.1.1.Tính chia hết

Xét 2 số nguyên a và b. Ta gọi a chia hết cho b $\Leftrightarrow \exists$ số nguyên n thỏa mãn a=b*n. Khi đó a được gọi là bội số của b, b được gọi là ước số của a. Kí hiệu a/b.

A được gọi là chia cho b dư r $\Leftrightarrow \exists$ số nguyên k và r thỏa mãn a = k.b+r. Khi đó r gọi là số dư của phép chia a cho b.

Xét dãy số (a_1, a_2,…, a_n).

Nếu b là ước số chung của tất cả các số trong dãy số trên, và tất cả các ước số chung khác của dãy đều là ước số của a, thì ta gọi b là ước số chung lớn nhất của dãy.

Kí hiệu b = USCLN (a_1, a_2,..., a_n) = gcd (a=a_1, a_2,..., a_n).

Nếu a là bội số chung của tất cả các số trong dãy số trên, và tất cả các bội số chung khác của dãy đều là bội số của b, thì ta gọi a là bội số chung nhỏ nhất của dãy.

Ki hiệu b = BSCNN (a_1, a_2,..., a_n) = lcm (a_1, a_2,…, a_n).

Ta có: gcd (a, b) = 1 $\Leftrightarrow$ a và b nguyên tố cùng nhau

1.1.2. Số nguyên tố

Số nguyên tố là số tự nhiên lớn hơn 1, chỉ chia hết cho 1 và chính nó.

Các số tự nhiên không phải là số nguyên tố thì gọi là hợp số.

Số nguyên tố đóng vai trò rất quan trọng trong lĩnh vực an toàn thông tin.

Số lượng các số nguyên tố là vô hạn, đồng thời cho đến nay người ta vẫn chưa tìm ra được quy luật của dãy số nguyên tố.

Số nguyên tố đã được nghiên cứu từ trước Công nguyên. Hiện nay, đã có rất nhiều thuật toán được nghiên cứu nhằm xác định một số có phải là số nguyên trong tố hay không.

Gần đây nhất, vào tháng 8 năm 2008, đã tìm ra số nguyên tố có gần 13 triệu chữ số, là số nguyên tố dạng Mersenne.

1.2. KHÔNG GIAN Z_n VÀ CẤU TRÚC NHÓM

1.2.1.Không gian Z_n và các phép tính cơ bản

Z_n được định nghĩa là tập hợp các số tự nhiên nhỏ hơn n

$Z_n = \{1,2,...,n-1\}$.

Z_n^* được định nghĩa là tập hợp các số tự nhiên nhỏ hơn n và nguyên tố cùng nhau với n.

$Z_n^* = \{x/x \in N, x< n, \gcd(x,n)=1\}$.

Trong không gian Z_n, các phép toán đều được thực hiện theo modulo n.

Phép cộng phép trừ và phép nhân được thực hiện bình thường như trong không gian Z, tuy nhiên kết quả cuối cùng phải được tính theo modulo n.

Phép chia trong không gian Z_n liên quan tới khái niệm phần tử nghịch đảo

Phần tử nghịch đảo của $a \in Z_n$ định nghĩa là $b \in Z_n$ thỏa mãn

$a.b = 1(\text{mod } n)$, ký hiệu $b = (\text{mod } n)/a$.

Vì vậy, phép chia a cho b trong không gian Z_n chỉ có nghĩa nếu b có phần tử nghịch đảo, bởi vì $a/b= a.b^{-1}$.

1.2.2. Cấu trúc nhóm

Nhóm là một bộ 2 phần tử (G,*), trong đó G là tập hợp khác rỗng, * là phép toán 2 ngôi thỏa mãn:

Tính kết hợp: (a*b)*c = a*(b*c) mọi a,b,c € G.

- Tồn tại phần tử trung lập $e \in G$ thỏa mãn : $e *x = x * e= e \ \forall \ x \in G$.
- Nhóm con của nhóm (G,*) là nhóm (S, *)thỏa mãn: $S \cap G$.
- Phần tử trung lập e của G nằm trong S.
- S khép kín đối với phép * và lấy nghịch đảo trong G.

Nhóm được gọi là nhóm cyclic nếu nó được sinh ra từ một trong các phần tử của nó. Phần tử đó gọi là phần tử nguyên thủy.

1.2.3. Dãy số giả ngẫu nhiên

Khái niệm "ngẫu nhiên" đóng một vai trò hết sức quan trọng trong đời sống và trong lĩnh vực an toàn thông tin.

Một dãy bit được coi là ngẫu nhiên hoàn toàn, tức là nếu ta biết toàn bộ các bit từ 0 tới bit n, thì ta cũng không có thêm thông tin gì để đoán nhận bit n+1 là 0 hay 1.

Như vậy, ta không có cách nào đoán nhận một dãy bit là ngẫu nhiên hay không, vả lại, trong máy tính, ta buộc phải sinh ra dãy bit theo một số hữu hạn các quy tắc nào đó, thì không thể coi là ngẫu nhiên được nữa. Vì vậy, trong thực tế, chúng ta chỉ có thể sử dụng các dãy số giả ngẫu nhiên (pseu-random number) mà thôi.

Các chuỗi giả ngẫu nhiên được hiểu là, nếu ta biết các bit từ 0 tới n, thì vẫn "khó" đoán được bit n+1.

Một số thuật toán sinh dãy số giả ngẫu nhiên như thuật toán sinh dãy giả ngẫu nhiên RSA, thuật toán Blum Blum Shud,v.v…

1.3. KHÁI NIỆM ĐỘ PHỨC TẠP THUẬT TOÁN

Thuật toán được định nghĩa là một dãy hữu hạn các chỉ thị mô tả một quá trình tính toán nào đó.

Một bài toán được gọi là "giải được" nếu tồn tại một thuật toán giải quyết bài toán đó. Ngược lại bài toán gọi là "không giải được".

Tuy nhiên, không phải bài toán nào thuộc lớp bài toán "giải được" cũng có thể giải được trong thực tế. Do đó, người ta đưa ra khái niệm chi phí để giải một bài toán, chi phí này liên quan mật thiết tới thuật toán giải bài toán đó, phụ thuộc vào bốn tiêu chí sau:

+ Thuật toán có dễ hiểu không.
+ Thuật toán có dễ cài đặt không.
+ Số lượng bộ nhớ cần sử dụng.
+ Thời gian thực hiện chương trình.

Trong các tiêu chí đó, tiêu chí thời gian thực hiện được đánh giá là quan trọng nhất.

Độ phức tạp thời gian cực đại thuật toán, thường được hiểu là số các phép tính cơ bản mà thuật toán phải thực hiện, trong trường hợp xấu nhất. Với cỡ dữ liệu đầu vào là n, thời gian thực hiện bài toán là t(n) được gọi là tiệm cận tới hàm f(n) nếu với n đủ lớn thì tồn tại số c thỏa mãn $t(n) \leq c.f(n)$. Nếu f(n) là một hàm đa thức thì thuật toán được gọi là có độ phức tạp thời gian đa thức.

Hiện nay, hầu hết các bài toán giải được trong thực tế đều là các bài toán có độ phức tạp thời gian đa thức. Các bài toán có độ phức tạp số mũ thực tế là khó thể giải được (có thể mất nhiều triệu tới nhiều tỷ năm).

Từ lý thuyết độ phức tạp tính toán, xuất hiện một khái niệm quan trọng trong lĩnh vực an toàn thông tin: hàm một phía và hàm một phía có cửa sập.

Hàm một phía (one way function): hàm số y=f(x) được gọi là hàm một phía, nếu khi biết giá trị của x thì ta dễ dàng tính được giá trị của y, nhưng ngược lại, nếu biết giá trị của y, ta "khó" tính được giá trị của x.

Hàm một phía có cửa sập (trapdoor one way function): Hàm một phía có cửa sập là hàm một phía, mà nếu biết "cửa sập" thì ta có thể dễ dàng tính ra giá trị của x khi biết giá trị của y.

1.4. HÀM PHI EULER VÀ QUAN HỆ "ĐỒNG DƯ"

1.4.1 Hàm Phi Euler

1.4.1.1. Định nghĩa

Hàm Phi Euler của số nguyên dương n là số các số nguyên tố cùng nhau với n nhỏ hơn n. Kí hiệu θ(n)

Ví dụ : θ(6)= 2, θ(26)= 12

1.4.1.2. Tính chất của hàm Phi Euler

+ Nếu n là số nguyên tố thì θ (n)= n-1 Ví dụ :θ (7)=6
+ Nếu p,q là 2 số nguyên tố cùng thì θ (p*q) = θ(p) * θ(q)
  Ví dụ: θ(26) = θ(2*13) = θ(2)*θ(13) = 1*12 = 12
+ Nếu p là số nguyên tố thì :θ(p) = (p-1)*p

Định lý:

Nếu p là số nguyên tố cùng nhau thì a =1 mod n.

1.4.2. Quan hệ "đồng dư"

1.4.2.1.Khái niệm:

Cho các số nguyên a, b, m (m>0). Ta nói rằng a và b "đồng dư" với nhau theo modulo m, nếu chia cả a và b cho m, ta nhận được cùng một số dư.

Ký hiệu a ≡ b (mod n).

Ví dụ:

17 ≡ 5 (mod 3) vì chia 17 và 5 cho 3, được cùng số dư là 2.

Nhận xét: Các mệnh đề sau đây là tương đương:

1/. a ≡ b (mod m)

2/. m \ (a-b)

3/. Tồn tại số nguyên t sao cho a = b + mt

Chứng minh:

1/.$\Rightarrow$ 2/.

Nếu có 1 thì theo định nghĩa: a,b chia cho m, phải có cùng số dư, do đó :

$a = mq_a + r$; $b = mq_b + r$; Suy ra $(a-b) = (q_a - q_b)$, tức là m\(a-b).

2/.$\Rightarrow$ 3/.

Nếu có 1. tức là m\ (a-b). Nghĩa là có $t \in Z$ sao cho (a-b) = mt hay

a = b + mt.

3/.$\Rightarrow$1/.

Nếu có 1. tức là tồn tại số nguyên t sao cho a = b + mt

Lấy a chia cho m, giả sử thương là q_a và dư r, hay $a = mq_a + r$ ($0 \leq r < m$), do đó: $b + mt = a = mq_a + r$ hay $b = m(q_a - t) + r$ ($0 \leq r < m$). Điều đó chứng tỏ khi chia a và b cho m được cùng số dư r, hay $a \equiv b \pmod m$.

1.4.2.2. Tính chất

1/. Quan hệ "đồng dư" là quan hệ tương đương trong Z:

Với mọi số nguyên dương m ta có:

$a \equiv a \pmod m$ với mọi $a \in Z$; (Tính chất phản xạ)

$a \equiv b \pmod m$ thì $b \equiv a \pmod m$; (Tính chất đối xứng)

$a \equiv b \pmod m$ và $b \equiv c \pmod m$ thì $a \equiv c \pmod m$; (Tính chất bắc cầu)

2/. Tổng hay hiệu các "đồng dư":

$(a+b) \pmod n \equiv [(a \bmod n) + (b \bmod n)] \pmod n$

$(a-b) \pmod n \equiv [(a \bmod n) - (b \bmod n)] \pmod n$

Tổng quát:

Có thể cộng hoặc trừ từng vế nhiều đồng dư thức theo cùng một modulo m, ta được một đồng dư thức theo cùng modulo m tức là:

Nếu $a_i \equiv b_i \pmod m$, i = 1…k, thì $\sum_{i=1}^{k} t_i a_i = \sum_{i=1}^{k} t_i b_i \pmod m$ với $t_i = \pm 1$.

3/. Tích các "đồng dư":

$(a*b) \pmod n \equiv [(a \bmod n) * (b \bmod n)] \pmod n$

Chương2. MỘT SỐ KHÁI NIỆM TRONG MẬT MÃ HỌC

2.1. VẤN ĐỀ MÃ HÓA

2.1.1. Khái niệm mã hóa

* Mã hóa là quá trình chuyển thông tin có thể đọc được (gọi là bản rõ) thành thông tin ”khó” thể đọc được theo cách thông thường (gọi là bản mã).

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

* Thuật toán mã hóa hay giải mã là thủ tục tính toán để thực hiện mã hóa hay giải mã.

* Khoá mã hóa là một giá trị làm cho thuật toán mã hóa thực hiện theo cách riêng biệt và sinh ra bản rõ riêng. Thông thường khóa càng lớn thì bản mã càng an toàn. Phạm vi các giá trị có thể có của khóa được gọi là không gian khóa.

* Hệ mã hóa là tập các thuật toán, các khóa nhằm che giấu thông tin, cũng như làm rõ nó.

Phân loại hệ mã hóa

Hiện có hai loại mã hóa chính: mã hóa khóa đối xứng, và mã hóa khóa bất đối xứng.

Mã hóa khoá đối xứng là hệ mã hóa mà biết được khóa lập mã thì có thể tính được khóa giải mã và ngược lại.

Mã hóa khoá bất đối xứng là hệ mã hóa có khóa lập mã và khóa giải mã khác nhau (ke ≠ kd), biết được khóa này cũng “khó” tính được khóa kia. Vì vậy chỉ cần bí mật khóa giải mã, còn công khai khóa lập mã. Do đó hệ mã hóa loại này còn có tên gọi là hệ mã hóa khóa công khai.

2.1.2. Hệ mã hóa khóa đối xứng

Hệ mã hóa khóa đối xứng có khóa lập mã và khóa giải mã “giống nhau”, theo nghĩa biết được khóa này thì dễ tính được khóa kia. Vì vậy phải giữ bí mật cả hai khóa.

Hệ mã hóa khóa đối xứng còn được gọi hệ mã hóa khóa bí mật, hay hệ mã hóa khóa riêng.

Đặc trưng của hệ mã hóa khóa đối xứng:

- Khóa phải được thỏa thuận và giữ bí mật giữa hai bên truyền tin. Khóa phải được truyền trên kênh an toàn giữa hai bên truyền tin. Điều này làm phức tạp quá trình thiết lập khóa. Hơn nữa, nếu giữa hai bên truyền tin không có kênh an toàn nào thì không thể thiết lập được quá trình truyền tin.
- Nếu bên tấn công biết dược khóa giải mã thì hệ mã hóa sẽ không còn bí mật.
- Tốc độ tính toán nhanh.

Ví dụ: Hệ mã hóa cổ điển

Ta thường đồng nhất Z_{26} với bảng ký tự tiếng Anh, do đó phép hoán vị trên Z_{26} cũng được hiểu là một phép hoán vị trên tập hợp các ký tự tiếng Anh, thí dụ một phép hoán vị π được cho bởi bảng:

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

s	**t**	**u**	**v**	**w**	**x**	**y**	**z**
v	**w**	**u**	**e**	**k**	**j**	**d**	***i***

Với hệ mã hóa hoán vị có khóa π, bản rõ

x = hengapnhauvaochieuthubay

sẽ được chuyển thành bản mã

v = ghsoxlsgxuexfygzhumgunxd

Thuật toán giải mã với khóa π, ngược lại sẽ biến y thành bản rõ x.

2.1.2.1. Đặc điểm của hệ mã hóa khóa đối xứng

Ưu điểm:

Hệ mã hóa khóa đối xứng mã hóa và giải mã nhanh hơn hệ mã hóa khóa bất đối xứng.

Nhược điểm:

+ Mã hóa khóa đối xứng chưa thật an toàn với lý do sau:

Người mã hóa và người giải mã phải có "chung" một khóa. Khóa phải được giữ bí mật tuyệt đối, vì biết khóa này "dễ" xác định được khóa kia và ngược lại.

+ Vấn đề thỏa thuận khóa và quản lý khóa chung là khó khăn và phức tạp. Người gửi và người nhận phải luôn thống nhất với nhau về khóa. Việc thay đổi khóa là rất khó và dễ bị lộ. Khóa chung phải được gửi cho nhau trên kênh an toàn.

Mặt khác khi hai người (lập mã, giải mã) cùng biết "chung" một bí mật, thì càng khó giữ được bí mật.

2.1.2.2. Nơi sử dụng hệ mã hóa khóa đối xứng

Hệ mã hóa khóa đối xứng thường được sử dụng trong môi trường mà khóa chung có thể dễ dàng trao quyền bí mật, chẳng hạn trong cùng một mạng nội bộ. Hệ mã hóa khóa đối xứng thường dùng để mã hóa những bản tin lớn, vì tốc độ mã hóa và giải mã nhanh hơn hệ mã hóa khóa công khai.

2.1.3. Hệ mã hóa khóa bất đối xứng

2.1.3.1. Giới thiệu

Trong mật mã cổ điển mà cho tới ngày nay vẫn còn được sử dụng, Alice (người gửi) và Bob (người nhận) bằng cách chọn một khóa bí mật K. Sau đó Alice dùng khóa K để mã hóa theo luật e_k và Bob dùng khóa K đó để giải mã theo luật giải d_k. Trong hệ mật này, d_k hoặc giống như e_k hoặc dễ dàng nhận được từ nó Nhược điểm lớn của hệ mật này là nếu ta để lộ e_k thì làm cho hệ thống mất an toàn, chính vì vậy chúng ta phải tạo cho các hệ mật này một kênh an toàn mà kinh phí để tạo một kênh an toàn không phải là rẻ.

Người gửi tin bây giờ sẽ mã hóa bằng khóa công khai của bên nhận, và tiến hành truyền tin. Bên nhận sẽ nhận tin, và sử dụng khóa bí mật của mình để giải mã bản tin. Kẻ tấn công trên đường truyền cho dù có được bản mã và khóa công khai cũng không thể tính ra được bản rõ. Vì để tính được bản rõ cần có khóa bí mật của bên nhận.

Đặc trưng của hệ mã hóa công khai:

+ Thuật toán chỉ được viết một lần, công khai cho nhiều người sử dụng.

+ Mỗi người chỉ cần giữ khóa bí mật của riêng mình, do đó khả năng bị lộ khóa sẽ ít hơn.

+ Khi có được các tham số đầu vào của hệ mã hóa, thì việc giải mã phải trong thời gian đa thức.

+ Tốc độ tính toán rất chậm.

+ Cần phải có chứng nhận của bên thứ ba có thẩm quyền (CA), bởi có thể xảy ra tình trạng giả mạo khoá công khai.

2.1.3.2. Ưu điểm của hệ mã hóa khóa bất đối xứng

Ưu điểm:

+ Hệ mã hóa công khai có ưu điểm chủ yếu sau:

Thuật toán được viết một lần công khai cho nhiều lần dùng, cho nhiều lần dùng, họ chỉ cần giữ bí mật khóa riêng của mình.

+ Khi biết các tham số ban đầu của hệ mã hóa, việc tính ra cặp khóa công khai và khóa bí mật phải là "dễ", tức là trong thời gian đa thức.

Người gửi bản rõ P và khóa công khai, thì "dễ" tạo ra bản mã C

Người nhận bản mã C và khóa bí mật, thì "dễ" giải được thành bản rõ P.

+ Người mã hóa dùng khóa công khai, người giải mã giữ khóa bí mật. Khả năng lộ khóa bí mật khó hơn vì chỉ có một người giữ gìn.

Nếu thám mã biết khóa công khai, cố gắng tìm khóa bí mật, thì chúng phải đương đầu với bài toán "khó".

+ Nếu thám mã biết khóa công khai và bản mã C, thì việc tìm ra bản rõ P cũng là bài toán "khó", số phép trừ là vô cùng lớn, không khả thi.

Hạn chế:

Hệ mã hóa khóa công khai: mã hóa và giải mã chậm hơn hệ mã hóa đối xứng.

2.1.3.3. Nơi sử dụng hệ mã hóa khóa bất đối xứng

Hệ mã hóa khóa bất đối xứng thường được sử dụng chủ yếu trên các mạng công khai như internet, khi mà việc trao chuyển khóa bí mật tương đối khó khăn.

Đặc trưng nổi bật của hệ mã hóa bất đối xứng là khóa công khai (public key) và bản mã (ciphertext) đều có thể gửi đi trên một kênh truyền tin không an toàn. Có biết cả khóa công khai và bản mã, thì thám mã cũng không dễ khám phá được bản rõ.

Nhưng vì có tốc độ mã hóa và giải mã chậm, nên hệ mã hóa công khai chỉ dùng để mã hóa những bản tin ngắn thường được sử dụng cho cặp người dùng thỏa thuận khóa bí mật của hệ mã hóa khóa riêng.

2.1.3.4. Hệ mã hóa RSA

Định nghĩa:

Sơ đồ: (Rivest,Shamir,Adleman đề xuất năm 1977)

Tạo cặp khóa (bí mật, công khai) (a,b) :

Chọn bí mật số nguyên tố lớn p,q, tính n = p * q, công khai n, đặt P= C =Z_n

Tính bí mật $\Phi(n) = (p-1)(q-1)$. Chọn khóa công khai $b < \Phi(n)$, nguyên tố với $\Phi(n)$.

Khóa bí mật a là phần tử nghịch đảo của b theo mod $\Phi(n)$: $a* b \equiv 1 \pmod{\Phi(n)}$.

Tập cặp khóa (bí mật, công khai) $K=\{(a,b)/a,b \in Z_n , a*b \equiv 1(\text{mod}\Phi(n))\}$.

Với bản rõ $x \in P$ và bản mã $y \in C$, định nghĩa :

- Hàm mã hoá : $y= e_k(x) = x^b \bmod n$
- Hàm giải mã : $x = d_k (y) = y^a \bmod n$

Ví dụ:

* Bản rõ chữ : RENAISSANCE

* Sinh khóa :

Chọn bí mật số nguyên tố p=53, q= 61, tinh n = p*q = 3233, công khai n.

Đặt P = C = Z_n, tính bí mật $\phi(n) = (p-1)(q-1)= 52 * 60 = 3120$.

+ Chọn khóa công khai b là nguyên tố với $\phi(n)$, tức là ƯCLN(b,$\phi(n)$) =1,

chọn b = 71.

+ Khóa bí mật a là phần tử nghịch đảo của b theo mod $\phi(n)$:

$a*b \equiv 1(\text{mod } \phi(n))$.Từ $a*b \equiv 1(\text{mod } \phi(n))$, ta nhận được khóa bí mật a = 791.

* Bản rõ số:

R E N A I S S A N C E (dấu cách)

17 04 13 00 08 18 18 00 13 02 04 26

m1 m2 m3 m4 m5 m6

* Theo phép lập mã : $c_i = m_i^b \bmod n = m_i^{71} \bmod 3233$, ta nhận được:

* Bản mã số:

c_1	c_2	c_3	c_4	c_5	c_6
3106	0100	0931	2691	1984	2927

* Theo phép giải mã: $m_i = c_i^* \bmod n = c_i^{791} \bmod 3233$, ta nhận lại bản rõ

Độ an toàn:

1). Hệ mã hóa RSA là tất định, tức là với một bản rõ x, và một khóa bí mật a, thì chỉ có một bản mã y.

2). Hệ mã hóa RSA an toàn, khi giữ được bí mật khóa giải mã a, p, q, $\phi(n)$.

Nếu biết được p và q, thì thám mã dễ dàng tính được $\phi(n) = (q - 1)(p - 1)$.

Nếu biết được $\phi(n)$, thì thám mã sẽ tính được a theo thuật toán Euclide mở rộng. Nhưng phân tích n thành tích của p và q là bài toán "khó".

Độ an toàn của hệ mật mã RSA dựa vào khả năng giải bài toán phân tích số nguyên dương n thành tích của 2 số nguyên tố lớn p và q.

2.1.3.5. Hệ mã hóa Elgamal

Hệ mã ElGamal được T.ElGamal đề xuất năm 1985, dựa vào độ phức tạp của bài toán tính lôgarit 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.

Sơ đồ : (Elgamal đề xuất năm 1985)

Tạo cặp khóa (bí mật, công khai) (a, b) :

Chọn số nguyên tố P sao cho bài toán logarith rời rạc trong Z_p là khó giải.

Chọn phần tử nguyên thủy $g \in Z_p^*$. Đặt $P = Z_p^*$, $C = Z_p^* \times Z_p^*$.

Chọn khóa bí mật là $a \in Z_p^*$. Tính khóa công khai $h \equiv g^a \bmod p$

Định nghĩa tập khóa : $K=\{(p, g, a, h): h \equiv g^a \bmod p \}$.

Các giá trị p, g, h được công khai, phải giữ bí mật a.

Với bản rõ $x \in P$ và bản mã $y \in C$, với khóa $k \in K$ định nghĩa:

Lập mã: chọn ngẫu nhiên bí mật $r \in Z_{p-1}$, bản mã là $y=e_k (x, r)= (y_1, y_2)$

Trong đó $y_1= g^r \bmod p$ và $y_2 = x*h^r \bmod p$

Giải mã: $d_k(y_1, y_2) = y_2 (y_1^a)^{-1} \bmod p$.

Ta chú ý rằng trong một mạng truyền thông bảo mật với việc dùng sơ đồ mã hóa Elgamal, mỗi người tham gia tự chọn cho mình các tham số ρ, α, a, rồi tính β, sau đó lập và công bố khóa công khai K' =(ρ, α, β), nhưng phải giữ tuyệt mật khóa bí mật chính là bài toán tính logarit rời rạc, một bài toán khó cho đến nay chưa có một thuật toán nào làm việc trong thời gian đa thức giải được nó.

Thí dụ :

Chọn p=2579, α=2, a=765, ta tính được $\beta = 2^{765} = 949 \bmod 2579$. Ta có khóa công khai (2579, 2, 949) và khóa bí mật 765. Giả sử để lập mã cho x = 1299, ta chọn ngẫu nhiên k = 853, sẽ có

$$e_{k'}(1299, 853) = (2^{853}, 1299.949^{853}) \bmod 2579$$
$$= (453, 2396).$$

Và giải mã ta được lại :

$$d_{k'}(453, 2396) = 2396.(453^{765})^{-1} \bmod 2579 = 1299.$$

Độ an toàn:

+ Hệ mã hóa Elgamal là không tất định, tức là với một bản rõ x và một khóa bí mật a, thì có thể có nhiều hơn một bản mã y, vì trong công thức lập mã còn có thành phần ngẫu nhiên r.

+ Độ an toàn của hệ mật mã Elgamal dựa vào khả năng giải bài toán logarit rời rạc trong Z_p. Theo giả thiết trong sơ đồ, thì bài toán này phải là "khó" giải.

Cụ thể như sau : Theo công thức lập mã $y = e_k (x, r) = (y_1, y_2)$, trong đó $y_1 = g^r \bmod p$ và $y_2 = x^* h^r \bmod p$

Như vậy muốn xác định bản rõ từ công thức y_2, thám mã phải biết được r. Giá trị này có thể tính được từ công thức y_1, nhưng lại gặp bài toán logarit rời rạc.

2.2. VẤN ĐỀ CHỮ KÝ SỐ

2.2.1. Giới thiệu về chữ ký số

Chữ kí viết tay thông thường trên giấy được dùng để xác minh người kí nó. Chữ kí dùng hàng ngày như trên một bức thư nhận tiền từ nhà băng, kí hợp đồng…

Sơ đồ chữ kí số là phương pháp kí một bức điện lưu dưới dạng điện tử. Chẳng hạn một bức điện có kí hiệu được truyền trên mạng máy tính. Dưới đây trình bày một vài sơ đồ chữ kí số.

Trước đây, với những tài liệu giấy truyền thống, để chứng thực tác giả một văn bản, người ta phải kí vào văn bản đó. Chữ kí tay như vậy sẽ gắn vật lý với văn bản, và có đặc điểm là giống nhau(tương đối) giữ các văn bản khác nhau, nếu cùng một người kí. Để xác thực chữ kí đó, người ta sẽ nhờ các chuyên gia giám định, và trong nhiều trường hợp vẫn gây tranh cãi.

Đối với tài liệu số, thì chữ kí điện tử không thể theo mô hình như vậy, do đặc tính dễ sao chép của của các tài liệu số. Nếu chữ ký điện tử giống nhau qua các văn bản, người ta có thể dễ dàng sao chép chữ kí điện tử này và gắn vào các văn bản giả mạo. Do đó, chữ kí điện tử ngoài việc gắn liền với tác giả, còn phải gắn liền với văn bản.

Chữ ký điện tử có tư tưởng gần giống với hệ mã hóa khóa công khai. Để kí lên một tài liệu, người ký sẽ sử dụng khóa bí mật của mình. Để kiểm tra chữ ký, người kiểm tra sẽ dùng khóa công khai của người ký. Như vậy, những ai không biết khóa bí mật thì không thể giả mạo chữ ký.

Định nghĩa

Sơ đồ chữ ký được định nghĩa là một bộ năm phần tử (P,A,K,S,V), trong đó:

P là tập hữu hạn các văn ban có thể.

A là tập hữu hạn các chữ ký có thể.

K là tập hữu hạn các khóa.

S là tập các thuật toán ký.

V là tập các thuật toán kiểm thử.

Với mỗi khóa k thuộc K, có thuật toán ký $sig_k \in S$ và thuật toán kiểm thử $ver_k \in V$.

Ký lên văn bản $x \in P : s = sig_k (x)$.

Kiểm thử : $ver_k(x,s) = true \Leftrightarrow s = sig_k(x)$.

2.2.2. Sơ đồ chữ ký RSA

Định nghĩa

Cho n= p.q, p và q là các số nguyên tố. Cho $P=C=Z_n$

Tính bí mật $\phi(n) = (p-1)(q-1)$. Chọn khóa công khai $b < \phi(n)$. b là nguyên tố cùng $\phi(n)$.

Khóa bí mật a là phần tử nghịch đảo của b theo mod $\phi(n)$: $a*b \equiv 1 \pmod{\Phi(n)}$.

Tập khóa (bí mật, công khai) $K = \{(a, b)/ a, b \in Z_n, a*b \equiv 1 \pmod{\phi(n)}\}$.

Ký số : Chữ ký trên $x \in P$ là $y = Sig_k(x) = x^a \bmod n$, $y \in A$

Kiểm tra chữ kí:

Ver (x,y)= đúng $\Leftrightarrow x \equiv y^b \pmod n$, $(x,y \in Z_n)$

Chú ý:

- So sánh giữa sơ đồ chữ ký RSA, và sơ đồ mã hóa RSA ta thấy có tương ứng.
- Việc ký chẳng qua là mã hóa, việc kiểm thử lại chính là việc giải mã:

Việc "ký số" vào x tương ứng việc mã hóa tài liệu x.

Kiểm thử chữ ký chính là việc giải mã "chữ ký", để kiểm tra tài liệu đã giải mã có đúng là tài liệu trước khi ký không. Thuật toán và khóa kiểm thử "chữ ký" là công khai, ai cũng có thể kiểm thử được chữ ký.

Ví dụ: Chữ ký trên x = 2

Tạo cặp khóa (bí mật, công khai) (a, b):

Chọn bí mật số nguyên tố p = 3 , q = 5, tính n = p * q = 3*5 = 15, công khai n. Đặt $P= C = Z_n$. Tính bí mật $\Phi(n) = (p-1)(q-1) = 2*4 = 8$.

Chọn khóa công khai $b =3 < \Phi(n)$, nguyên tố với $\Phi(n) = 8$.

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

$a* b \equiv 1 \pmod{\Phi(n)}$.

Ký số : Chữ ký trên $x = 2 \in P$ là

$y = Sig_k(x) = x^a \pmod n = 2^3 \pmod{15} = 8$, $y \in A$.

Kiểm tra chữ ký : $Ver_k(x, y) =$ đúng $\Leftrightarrow x \equiv y^b \pmod n$

$\Leftrightarrow 2 \equiv 8^b \pmod{15}$.

Độ an toàn của chữ ký RSA:

1). Người gửi G gửi tài liệu x, cùng chữ ký y đến người nhận N, có 2 cách sử lý:

* Ký trước, mã hóa sau:

G ký trước vào x bằng chữ ký $y = Sig_G(x)$, sau đó mã hóa x và y nhận được $z = e_G (x, y)$. G gửi z cho N.

Nhận được z, N giải mã z để để được x, y.

Tiếp theo kiểm tra chữ ký $Ver_N(x, y) = true$?

* Mã hóa trước, ký sau:

G mã hóa trước x bằng $u = e_G(x)$, sau đó ký vào u bằng chữ ký $v = Sig_G(u)$.

G gửi (u, v) cho N.

Nhận được (u, v), N giải mã u được x.

Tiếp theo kiểm tra chữ ký $Ver_N(u, v) = true$?

2). Giả sử H lấy trộm được thông tin trên đường truyền từ G đến N.

+ Trong trường hợp a, H lấy được z. Trong trường hợp b, H lấy được (u, v).

+ Để tấn công x,trong cả hai trường hợp, H đều phải mã hóa thông tin lấy được.

+Để tấn công vào chữ ký, thay bằng chữ ký (giả mạo), thì xảy ra điều gì?

- Trường hợp a, để tấn công chữ ký, H phải giải mã z, mới nhận được y.
- Trường hợp b, để tấn công chữ ký v, H đã sẵn có v, mới nhận được y.

H thay chữ ký v trên u, bằng chữ ký của H là $v' = Sig_H(u)$, gửi (u, v') đến N.

Khi nhận được v', N kiểm thử thấy sai, gửi phản hồi lại G.

G có thể chứng minh chữ ký đó là giả mạo.

G gửi chữ ký đúng v cho N, nhưng quá trình truyền tin sẽ bị chậm lại.

+ Như vậy trong trường hợp b, H có thể giả mạo chữ ký mà không cần giải mã. Vì thế có lời khuyên: Hãy ký trước sau đó mã hóa cả chữ ký.

2.2.3. Sơ đồ chữ ký Elgamal

Định nghĩa:

Cho p là số nguyên tố sao cho bài toán logarithm sời rạc trên Z_p là khó và giả sử α thuộc Z_n là phần tử nguyên thủy p= Z_p^*, a= Z_p * Z_{p-1} và định nghĩa:

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

Giá trị p, α, β là công khai còn a là bí mật.

Với K = (p, α, a, β) và một số ngẫu nhiên (mật) k ∈ Z . Định nghĩa :

$$Sig_k (x, y) = (\gamma, \delta),$$

Trong đó $\gamma = \alpha^k \bmod p$

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

Với x,γ ∈ Zp và δ ∈ Z, ta định nghĩa :

$$Ver (x, \gamma, \delta) = true \Leftrightarrow \beta^\gamma \gamma^\delta \equiv \alpha^x \pmod p.$$

Bản cải tiến của của sơ đồ này đã được Viện tiêu chuẩn và công nghệ quốc gia Mỹ (NIST) chấp nhận làm chữ kí số.

Sơ đồ chữ ký Elgamal là không tất định, giống như hệ mã hóa khóa công khai Elgamal. Điều này có nghĩa là có nhiều chữ kí hợp lệ trên bức điện cho trước bất kỳ. Thuật toán xác minh phải có khả năng chấp nhận bất kỳ chữ ký hợp lệ khi xác thực.

Nếu chữ kí được thiết lập đúng khi xác minh sẽ thành công vì :

$$\beta^\gamma \gamma^\delta \equiv \alpha^{a\gamma} \alpha^{k\gamma} \pmod p$$
$$\equiv \alpha^x \pmod p$$

Là ở đây ta sử dụng hệ thức:

$$a\gamma + k\delta \equiv x \pmod{p-1}$$

Sơ đồ chữ kí số Elgamal.

Ví dụ:

Giả sử p=467, α=2, a=127. Khi đó $\beta = 2^{127} \bmod 467 = 132$. Cho x=100; ta chọn ngẫu nhiên k=213($\in Z_{466}^*$) và được $k^{-1} \bmod 466 = 431$. Chữ ký trên văn bản x = 100 với số ngẫu nhiên k=213 là (γ, δ), trong đó $\gamma = 2^{213} \bmod 467 = 29$ và δ= (100-127.29).431 mod 466=51.

Để kiểm thử ta tính :

$\beta^\gamma.\gamma^\delta = 132^{29}.29^{51} = 189 \pmod{467}$, $\alpha^x = 2^{100} = 189 \pmod{467}$, hai giá trị đó đồng dư với nhau theo mod 467, chữ ký (γ,δ) = (29,51) được xác nhận là đúng.

Sơ đồ chữ ký ElGamal được xem là an toàn, nếu việc ký trên một văn bản là không thể giả mạo được, nói cách khác, không thể có một người nào ngoài chủ thể hợp pháp có thể giả mạo chữ ký của chủ thể hợp pháp có thể giả mạo chữ ký của chủ thể hợp pháp đó trên một văn bản bất kỳ.

Vì vậy, việc giữ bí mật khóa K'=a dùng để tạo chữ ký là có ý nghĩa quyết định đối với việc bảo đảm tính an toàn của chữ ký.

Độ an toàn :

Trường hợp: Giả mạo chữ ký không cùng với tài liệu được ký.

+ H cố gắng giả mạo chữ ký trên x, mà không biết khoá bí mật a.

Như vậy, H phải tính được γ và δ.

- Nếu chọn trước γ, H phải tính δ qua đẳng thức $h^{\gamma} * \gamma^{\delta} \equiv g^{x} \bmod p$.

Tức là $\gamma^{\delta} \equiv g^{x} h^{-\gamma} \bmod p$ hay $\delta \equiv \log_{\gamma} g^{x} h^{-\gamma} \bmod p$

- Nếu chọn trước δ, H phải tính γ qua chương trình $h^{\gamma} * \gamma^{\delta} \equiv g^{x} \bmod p$.

  Hiện nay chưa có cách hữu hiệu 2 trường hợp trên, nhưng phỏng đoán là khó hơn bài toán logarit rời rạc.

Có thể có cách tính γ, δ đồng thời với (γ, δ) là chữ ký? Chưa có trả lời rõ!

- Nếu chọn trước γ, δ sau đó tính x, H phải đối đầu với bài toán logarit rời rạc.

Ta có $h^{\gamma} * \gamma^{\delta} \equiv g^{x} \bmod p$.

Như vậy :$x \equiv \log_{g} g^{\delta} \equiv \log_{g} h^{\gamma} * \gamma^{\delta}$

2.3. HÀM BĂM

2.3.1. Định nghĩa hàm băm

Hàm băm là thuật toán không dùng khóa để mã hóa (ở đây dùng thuật ngữ "băm" thay cho "mã hóa"), nó có nhiệm vụ "lọc" (băm) tài liệu (bản tin) và cho kết quả là một giá trị "băm" có kích thước cố định, còn gọi là "đại diện tài liệu", hay "đại diện bản tin", "đại diện thông điệp".

Hàm băm là hàm một chiều, theo nghĩa giá trị của hàm băm là duy nhất, và từ giá trị băm này, "khó thể" suy ngược lại nội dung hay ban đầu của tài liệu gốc.

2.3.2 . Đặc tính của hàm băm

Hàm băm h là hàm một chiều (one-way Hash) với các đặc tính sau:

1). Với tài liệu đầu vào (bản tin gốc) x, chỉ thu được giá trị băm duy nhất $z = h(x)$.

2). Nếu dữ liệu trong bản tin x bị thay đổi hay bị xóa để thành bản tin x', thì giá trị băm $h(x') \neq h(x)$.

Cho dù chỉ là một sự thay đổi nhỏ, ví dụ chỉ thay đổi 1 bit dữ liệu của bản tin gốc x, thì giá trị băm h(x) của nó cũng vẫn thay đổi. Điều này có nghĩa là: hai thông điệp khác nhau, thì giá trị băm của chúng cũng khác nhau.

3). Nội dung của bản tin gốc "khó" thể suy ra từ giá trị hàm băm của nó. Nghĩa là: với thông điệp x thì "dễ "tính được $x = h(x)$, nhưng lại khó tính ngược lại được x nếu chỉ biết giá trị băm h(x) (Kể cả khi biết hàm băm h).

2.3.3. Ứng dụng của hàm băm

1). Với bản tin dài x, thì chữ ký trên x cũng sẽ dài, như vậy tốn thời gian "ký", tốn bộ nhớ lưu giữ "chữ ký", tốn thời gian truyền "chữ ký" trên mạng.

Người ta dùng hàm băm h để tạo đại diện bản tin $z = h(x)$, nó có độ dài ngắn (VD 128 bit). Sau đó ký trên z, như vậy chữ ký trên z sẽ nhỏ hơn rất nhiều so với chữ ký trên bản tin gốc x.

2). Hàm băm để xác định tính toàn vẹn dữ liệu.

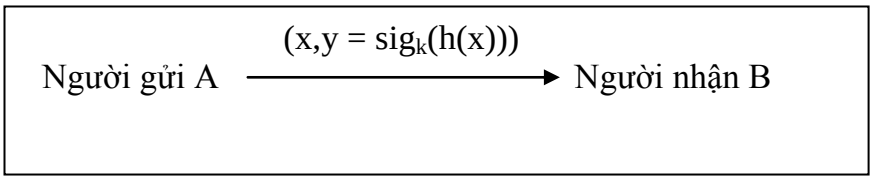
3). Hàm băm dùng để bảo mật một số dữ liệu đặc biệt, ví dụ bảo vệ mật khẩu, bảo vệ khóa mật mã,……

2.3.4. Tính chất của hàm băm

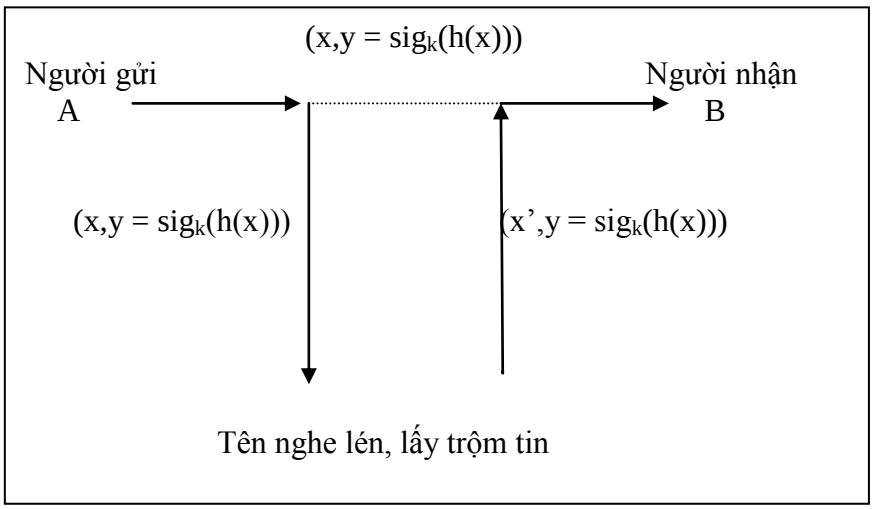
1/. Tính chất 1: Hàm băm h là không va chạm yếu.

Ví dụ: Xét kiểu tấn công sau: Kiểu tấn công theo tính chất 1.

Hình a: Cách đi đúng của thông tin: thông tin được truyền đúng từ A đến B.



Hình b : Thông tin bị lấy trộm và bị thay đổi trên đường truyền.



Kiểu tấn công theo tính chất 1

+ Người A gửi cho B bản tin (x,y) với y = sig$_k$(h(x)). B không nhận được (x,y) vì :

+ Trên đường truyền, tin bị lấy trộm. Tên trộm bằng cách nào đó tìm được một bản tin x' ≠ x nhưng lại có h(x') = h(x). hắn thay thế x bằng x', và chuyển tiếp (x',y) cho B.

+ Người B nhận được (x',y), và vẫn xác thực được thông tin đúng đắn. Do đó, để tránh kiểu tấn công như trên, hàm h phải thỏa mãn tính chất : không va chạm yếu.

Khái niệm: Hàm băm không va chạm yếu.

Hàm băm h được gọi là không va chạm yếu, nếu cho trước bức điện x,"khó" thể tính toán để tìm ra bức điện x' ≠ x mà h(x') = h(x).

2/. Tính chất 2: Hàm băm h là không va chạm mạnh

Ví dụ :

Xét kiểu tấn công như sau: Kiểu tấn công theo tính chất 2.

+ Đầu tiên, tên giả mạo tìm được hai thông điệp khác nhau x' và x (x' ≠ x) mà có h(x') = h(x). (Ta coi bức thông điệp x là hợp lệ, còn x' là giả mạo).

+ Tiếp theo, Hắn thuyết phục ông A kí vào bản tóm lược h(x) để nhận được y. Khi đó (x', y) là bức điện giả mạo nhưng hợp lệ vì h(x') = h(x).

Để tránh kiểu tấn công này, hàm h phải thỏa mãn tính chất : không va chạm mạnh.

Khái niệm: Hàm băm không va chạm mạnh

Hàm băm h được gọi là không va chạm mạnh "khó" thể tính toán để tìm ra hai bức thông điệp khác nhau x' và x (x' ≠ x) mà có h(x') = h(x).

3/. Tính chất 3 : Hàm băm h là hàm một chiều.

Ví dụ : Xét kiểu tấn công như sau : Kiểu tấn công theo tính chất 3.

+ Người A gửi cho người B thông tin (x, z, y) với z = h(x), $y = sig_k(z)$.

+ Giả sử tên giả mạo tìm được bản tin x', được tính ngược từ bản tóm lược z = h(x).

+ Tên trộm thay thế bản tin x hợp lệ, bằng bản tin x' giả mạo, nhưng lại có z= h(x'). Hắn ta ký số trên bản tóm lược z của x' bằng đúng chữ ký hợp lệ. Nếu làm được như vậy, thì (x', z, y) là bức điện giả mạo nhưng hợp lệ.

Để tránh được kiểu tấn công này, hàm băm h cần thỏa mãn tính chất một chiều.

Khái niệm: Hàm băm một chiều.

Hàm băm h được gọi là hàm băm một chiều nếu khi cho trước một bản tóm lược thông báo z thì "khó thể" tính toán để tìm ra thông điệp ban đầu x sao cho h(x) = z.

2.3.5. Hàm băm MD4

2.3.5.1 Khái niệm "thông điệp đệm"

"Thông điệp đệm" (Messege Padding) là sâu bit có độ dài chia hết cho 512.

"Thông điệp đệm" được lưu trong mảng M = M[0] M[1]... M[N-1].

Trong đó M[i] là sâu bit có độ dài 32 bit. Gọi là word.

N≡ 0 mod 16. (32 × 16 = 512).

M được xây dựng từ bản tin gốc a bằng thuật toán:

1. d= 447 –(|a| mod 512).(=512 nếu |a|,mod 512> 447).
2. Giả sử 1 là kí hiệu biểu diễn nhị phân của |a| mod 2^{64}, tl: |1| = 64.
3. M = a || 1|| 0^d || 1.

*). Độ dài của xâu a || 1 || 0^d là |a| + 1 + d = 448 mod 512.

*). Độ dài của "thông điệp đệm" M là

448 mod 512 + |1| = 448 mod 512 + 64 = 512 mod 512.

Chú ý: Vì M = a || 1 || 0^d || 1 nên

d = |M| - (|a| + 1 +1)=

512 – (|a| +1 +64) = 512 –(|a| +65) = 447 – (|a| mod 512).

Ví dụ:

Xâu đầu vào là a = "ABC", xây dựng M như sau :

a: = "ABC" = "01000001 01000010 01000011". (Chú ý: 'A' = 65).

*). Độ dài tính theo bit của xâu a: |a| = 24 bit

=> d = 447 – (|a| mod 512) = 423.

|a| + 1 + d = 24 + 1 + 423 = 448 mod 512.

*). Biểu diễn nhị phân của độ dài xâu a là l:

l = |a| mod 2^{64} = 24 mod 2^{64} = 24 =16 +8 = $(\underbrace{00\ldots00}_{59_{20}}\ 11000)_2$

=> Độ dài của l là |l| = $|\underbrace{00\ldots00}_{59_{20}}\ 11000|$ = 59 + 5 =64.

M = a || 1 || 0^d ||l.

=> M = 01000001 01000010 01000011 || 1 || $\underbrace{00\ldots00}_{423_{20}}$ || $\underbrace{00\ldots00}_{59_{20}}$ 11000

M = M[0]M[1] … M[N-1] , N = 0 mod 16.

M[0] = 01000001 01000010 01000011 10000000

M[1] = M[2] = …. =M[13] = M[14] =$\underbrace{00\ldots..00}_{32_{20}}$

M[15] = 00000000 00000000 00000000 00011000

Trong việc xây dựng M, ta gắn số 1 đơn lẻ vào sau a, sau đó thêm tiếp các số 0 vào đủ để độ dài của M đồng dư với 448 modulo 512. Cuối cùng nối thêm 64 bit (chính la |l|) chứa biểu diễn nhị phân về độ dài ban đầu của x (được rút gọn theo modulo 2^{64} nếu cần).

Xâu kết quả M có độ dài chia hết cho 512. Vì thế khi chặt M thành các word 32 bit, số word nhận được là N sẽ chia hết cho 16.

Mục đích việc tạo ra mảng M _ "thông điệp đệm" _là để các hàm băm xử lý trên từng khối (block) 512 bit, tức là 16 word, cùng một lúc.

2.3.5.2. Thuật toán hàm băm MD4

INPUT: thông điệp là một xâu a có độ dài b bit.

OUTPUT:Bản băm, đại diện cho thông điệp gốc, độ dài cố định 128 bit

1/. Tóm tắt thuật toán

Bước 1: Khởi tạo thanh ghi

Có 4 thanh ghi để tính toán nhằm đưa ra đoạn mã : A, B, C, D. Bản tóm lược của thông điệp được xây dựng như sự kết nối củ các thanh ghi có độ dài 32 bit. Các thanh ghi này được khởi tạo giá trị hecxa.

word A:= 67 45 23 01 word B := ef cd ab 89

word C:= 98 ba dc fe word D := 10 32 54 76

Bước 2: Xử lý thông điệp a trong 16 khối word, có nghĩa là xử lý cùng một lúc 16 word = 512 bit.

Chia mảng M thành các khối 512 bit, đưa từng khối 512 bit vào mảng T[j]. Mỗi lần xử lý một khối 512 bit. Lặp lại N/16 lần .

2/. Thuật toán MD4

A := 67 45 23 01 B := ef cd ab 89

C := 98 ba dc fe D := 10 32 54 76

FOR i := 0 TO N/16-1 DO

for j :=0 to 15 do T[j] = M[16i +j];

AA := A; BB := B;

CC := C; DD := D;

Mỗi lần xử lý 16 từ, mỗi từ 32 bit, tl: 512 bit.

Vòng 1

Vòng 2

Vòng 3

A = A + AA; B = B + BB; C = C + CC; D = D + DD;

Gán giá trị cho 4 biến AA, BB, CC, DD bằng giá trị bốn thanh ghi A, B, C, D tương ứng.

3/. Các phép tính và các hàm dùng trong Thuật toán MD4

* Các phép toán logic được sử dụng trong ba vòng.

$X \wedge Y$ là phép toán AND theo từng bit giữa X và Y

$X \vee Y$ là phép toán OR theo bit giữa X và Y

$X \oplus Y$ là phép toán XOR theo từng bit giữa X và Y

$\neg X$ chỉ phép bù của X

$X + Y$ là phép cộng theo modulo 2^{32}

$X <<< s$ là phép toán vòng trái X đi s vị trí ($0 \leq s \leq 31$)

* Ba hàm F, G, H dùng tưng ứng trong vòng 1,2,2.

Mỗi hàm này là một hàm boolean tính theo bit.

$F(X, Y, Z) = (X \wedge Y) \vee ((\neg X) \wedge Z)$

$G(X, Y, Z) = (X \wedge Y) \vee (X \wedge Z) \vee (Y \wedge Z)$

$H(X, Y, Z) = X \oplus Y \oplus Z$

Ba vòng trong MD4 là hoàn toàn khác nhau. Mỗi vòng gồm một trong 16 word trong T được xử lý. Các phép toán được thực hiện trong ba vòng tạo ra các giá trị mới trong bốn thanh ghi. Cuối cùng, bốn thanh ghi được cập nhật ở 3.4 bằng cách cộng ngược các giá trị lưu trước đó. Phép cộng này được xác định là cộng các số nguyên dương, được rút gọn theo modulo 2^{32}.

4/. Ba vòng "băm"

Vòng 1

1. A = (A +F(B, C, D) + T[0])<<<3
2. D = (D +F(A, B, C) + T[1])<<<7
3. C = (C +F(D, A, B) + T[2])<<<11
4. B = (B +F(C, D, A) + T[3])<<<19
5. A = (A +F(B, C, D) + T[4])<<<3
6. D = (D +F(A, B, C) + T[5])<<<7
7. C = (C +F(D, A, B) + T[6])<<<11
8. B = (B +F(C, D, A) + T[7])<<<19
9. A = (A +F(B, C, D) + T[8])<<<3
10. D = (D +F(A, B, C) + T[9])<<<7
11. C = (C +F(D, A, B) + T[10])<<<11
12. B = (B +F(C, D, A) + T[11])<<<19
13. A = (A +F(B, C, D) + T[12])<<<3
14. D = (D +F(A, B, C) + T[13])<<<7
15. C = (C +F(D, A, B) + T[14])<<<11
16. B = (B +F(C, D, A) + T[15])<<<19

Kết quả của VD a sau khi được xử lý qua vòng 1

1. 64B3DA82	5. 3D5E5934	9. 59798D5E	13. 7551AAC6
2. 34D8EB03	6. 489D5140	10. D206302D	14. 789B984F
3. B7BCB118	7. CCD14D6C	11. 753D6134	15. F55A1F31
4. 6D91B115	8. 454D0E92	12. F52AED08	16. ABA71E22

Vòng 2

1. A = (A + G(B, C, D) + T[0] + 5A827999) <<< 3
2. D = (D + G(A, B, C) + T[4] + 5A827999) <<< 5
3. C = (C + G(D, A, B) + T[8] + 5A827999) <<< 9
4. B = (B + G(C, D, A) + T[12] + 5A827999) <<< 13
5. A = (A + G(B, C, D) + T[1] + 5A827999) <<< 3
6. D = (D + G(A, B, C) + T[5] + 5A827999) <<< 5
7. C = (C + G(D, A, B) + T[9] + 5A827999) <<< 9
8. B = (B + G(C, D, A) + T[13] + 5A827999) <<< 13
9. A = (A + G(B, C, D) + T[2] + 5A827999) <<< 3
10. D = (D + G(A, B, C) + T[6] + 5A827999) <<< 5
11. C = (C + G(D, A, B) + T[10] + 5A827999) <<< 9
12. B = (B + G(C, D, A) + T[14] + 5A827999) <<< 13
13. A = (A + G(B, C, D) + T[13] + 5A827999) <<< 3
14. D = (D + G(A, B, C) + T[7] + 5A827999) <<< 5
15. C = (C + G(D, A, B) + T[11] + 5A827999) <<< 9
16. B = (B + G(C, D, A) + T[15] + 5A827999) <<< 13

Giá trị 5A827999 là một hằng số ở dạng hecxa có độ dài 32 bit

Kết quả của VD a sau khi được xử lý qua vòng 2

1. 558C2E28	5. 558C2E28	9. 31E9FE4A	13. B60A11E6
2. 5A0E08F9	6. 5A0E08F9	10. 6F68E462	14. 2DED6D8E
3. F6A9B390	7. F6A9B390	11. D745F88A	15. A2870B31
4. 7876BC8F	8. 7876BC8F	12. 7050BC10	16. 4384D178

Vòng 3

1. A = (A + H(B, C, D) + T[0] + 6ED9EBA1) <<< 3
2. D = (D + H(A, B, C) + T[8] + 6ED9EBA1) <<< 9
3. C = (C + H(D, A, B) + T[4] + 6ED9EBA1) <<< 11
4. B = (B + H(C, D, A) + T[12] + 6ED9EBA1) <<< 15
5. A = (A + H(B, C, D) + T[2] + 6ED9EBA1) <<< 3
6. D = (D + H(A, B, C) + T[10] + 6ED9EBA1) <<< 9
7. C = (C + H(D, A, B) + T[6] + 6ED9EBA1) <<< 11
8. B = (B + H(C, D, A) + T[14] + 6ED9EBA1) <<< 15
9. A = (A + H(B, C, D) + T[1] + 6ED9EBA1) <<< 3
10. D = (D + H(A, B, C) + T[9] + 6ED9EBA1) <<< 9
11. C = (C + H(D, A, B) + T[5] + 6ED9EBA1) <<< 11
12. B = (B + H(C, D, A) + T[13] + 6ED9EBA1) <<< 15
13. A = (A + H(B, C, D) + T[3] + 6ED9EBA1) <<< 3
14. D = (D + H(A, B, C) + T[11] + 6ED9EBA1) <<< 9
15. C = (C + H(D, A, B) + T[7] + 6ED9EBA1) <<< 11
16. B = (B + H(C, D, A) + T[15] + 6ED9EBA1) <<< 15

Giá trị 6ED9EBA1 là một hằng số ở dạng hecxa có độ dài 32 bit.

Kết quả của VD a sau khi được xử lý qua vòng 3

1. 98A7C489	5. F3031C80	9. C02E826B	13. 03477E5E
2. E70B031C	6. 7D7A371B	10. F38DC78B	14. 77509F0A
3. A96B2FFA	7. 1C2487DE	11. E3C7F63B	15. FB3D792D
4. 58BE9F94	8. F7767709	12. 812AB00F	16. 23D73C06

4). Kết quả "băm"

Kết quả ra là đoạn mã có độ dài 128 bit, được thu gọn từ thông điệp a có độ dài b bit. Đoạn mã này thu được từ 4 thanh ghi A, B, C, D: bắt đầu từ byte thấp của thanh ghi A cho đến byte cao của thanh ghi D.

Với VD a = "ABC", kết quả cuối cùng là đại diện văn bản:

A = 6A8CA15F C = 93F85626

B = 671E4A D = 3409907C

Chú ý : A = A + AA = 03477E5E

67452301

= 6A8CA15F

2.4.VẤN ĐỀ THỦY KÝ

2.4.1 Khái niệm

Khái niệm thủy vân đã ra đời từ lâu. Năm 1282, thủy vân đã có hoa văn trên đó. Điều này giúp các xưởng sản xuất giấy đánh dấu bản quyền trên tờ giấy của họ làm ra. Đến thế kỷ 18, thủy vân đã có nhiều ứng dụng ở Châu Âu và Mỹ trong việc xác thực bản quyền hay chống tiền giả. Thuật ngữ thủy vân bắt nguồn từ một loại mực vô hình và chỉ hiện lên khi nhúng vào nước.

Thủy vân số (digital watermarking) là một công cụ giúp đánh dấu bản quyền hay những thông tin cần thiết vào tài liệu điện tử.

Lịch sử thủy vân số:

Thuật ngữ thủy vân số được cộng đồng thế giới chấp nhận rộng rãi vào đầu thập niên 1990. Khoảng năm 1995, sự quan tâm đến thủy vân bắt đầu phát triển nhanh. Năm 1996, hội thảo về che dấu thông tin lần đầu tiên đưa thủy vân vào nội dung chính. Đến năm 1999, SPIE đã tổ chức hội nghị đặc biệt về bảo mật và thủy vân trên các nội dung đa phương tiện. Cũng trong khoảng thời gian, một số tổ chức đã quan tâm đến kỹ thuật watermarking với những mức độ khác nhau. Chẳng hạn CPTWG thử nghiệm hệ thống thủy vân bảo vệ phim trên DVD. SDMI sử dụng thủy vân trong việc bảo vệ các đoạn nhạc. Hai dự án khác được liên minh Châu Âu ủng hộ, VIVA và Talisman đã thử nghiệm sử dụng thủy vân để theo dõi phát sóng.

Vào cuối thập niên 1990, một số công ty đưa thủy vân vào thương trường, chẳng hạn các nhà phân phối nhạc trên Internet sử dụng Liqid Audio áp dụng công nghệ của Verance Corporation. Trong lĩnh vực thủy vân ảnh, photoshop đã tích hợp một bộ nhúng và bộ dò thủy vân tên là Digimarc.

2.4.2. Quá trình nghiên cứu thủy vân số

Thủy vân số được coi là ra đời từ năm 1954, với bằng sáng chế của Emile Hembrooke. Tuy nhiên, nghiên cứu thủy vân vẫn chưa được đặt ra như một lĩnh vực nghiên cứu độc lập cho tới những năm 1980. Tuy nhiên khái niệm thủy vân chỉ được hoàn thiện vào giữa những năm 90 của thế kỷ 20.

Những nghiên cứu đầu tiên về thủy vân đều tập trung vào nghiên cứu "thủy vân mù" (blind watermark). Thủy vân mù là thủy vân được nhúng mà không cần quan tâm tới nội dung của môi trường nhúng. Tương tự như vậy, các thuật toán tách thủy vân mù đều độc lập với những thành phần dữ liệu không chứa thủy vân. Có thể ví thủy vân mù như chữ ký tay, nội dung của thủy vân không thay đổi với các môi trường nhúng khác nhau.

Vào năm 1999, đã có một sự thay đổi lớn diễn ra. Trong một bài báo đăng trên IEEE, Cox và các đồng nghiệp đã nhận ra, chất lượng thủy vân sẽ tốt hơn rất nhiều nếu như thủy vân có quan tâm đến môi trường nhúng. Các thủy vân này được gọi là thủy vân giàu (informed watermark) , khi đó nội dung của thủy vân được hiểu là một hàm của nội dung môi trường nhúng. Có thể so sánh ý tưởng này với ý tưởng về chữ ký điện tử.

Đi xa hơn nữa, vào năm 2000, hai nhóm tác giả B.Chen, G.W.Wornell và J.Chou, Pradhan, Ramchandran đã phát triển từ bài báo của M.Costa năm 1983 "Writing on diry paper" để phát triển một hướng nghiên cứu rất mới. Ý tưởng chính của Costa là, có hai loại nhiễu sẽ tác động lên nội dung của bản tin truyền đi. Loại nhiễu thứ nhất, là loại nhiễu xảy ra tại bên gửi, do các vụ biến đổi và xử lý tài liệu. Loại nhiễu này có thể kiểm soát. Loại nhiễu thứ hai là loại nhiễu xảy ra trên đường truyền, và chúng ta không thể kiểm soát được chúng. Costa lý luận rằng, các thuật toán thủy vân trước đây chỉ cố gắng nhúng thủy vân vào loại nhiễu thứ nhất, cho nên dung lượng tin giấu được là rất nhỏ. Costa cũng đã chỉ ra dung lượng tin cần giấu là độc lập với loại nhiễu thứ nhất. Do đó, nếu ta coi toàn bộ tài liệu số là nhiễu thứ nhất, chúng ta sẽ có một phương pháp để nhúng một lượng thông tin rất lớn vào tài liệu.

Thủy vân có một ứng dụng rất quan trọng là bảo vệ sự toàn vẹn của tài liệu và chống xuyên tạc. Để thỏa mãn yêu cầu này của thủy vân, các nghiên cứu trước kia đều cố gắng áp dụng một mô hình tổng quát lên toàn bộ tài liệu. Tuy nhiên, vào năm 1995, Cox và các đồng nghiệp đã nhận ra, họ có thể sử dụng mô hình tri giác (perceptual model) để giảm dung lượng cần giấu. Thay vì cố gắng áp dụng một mô hình tổng quát lên toàn bộ tài liệu, thực ra chỉ cần áp dụng thủy vân lên một số phần quan trọng của tài liệu mà thôi. Đây có thể coi là một dạng đặc biệt của thủy vân giàu, vì nội dung thủy vân cũng bị phụ thuộc vào tài liệu.

Như một chân lý của cuộc sống, luôn tồn tại sự thống nhất và đấu tranh giữa các mặt đối lập. Với sự ra đời của thủy vân, thì khoảng từ năm 1990 trở về sau, đã có nhiều nghiên cứu về tấn công cũng như chống tấn công đối với thủy vân. Những nghiên cứu này đã thúc đẩy quá trình nghiên cứu thủy vân đạt được nhiều kết quả mới.

Thủy vân sử dụng công nghệ trải phổ (spread spectrum) được giới thiệu cùng thời điểm với mô hình tri giác, là một lỗ lực nhằm cân bằng giữa tính bền vững (robustness) và tính tin cậy (fidelity) của thủy vân số. Công nghệ trải phổ sẽ trải một băng tần hẹp vào một băng tần rộng hơn, do đó tỷ lệ nhiễu trên mỗi tần số trở lên rất nhỏ. Phía bên người gửi sẽ tổng hợp lại các tín hiệu này, và lúc này nhiễu trở nên lớn. Công nghệ trải phổ là một hướng đi có nhiều triển vọng của kỹ thuật thủy vân.

Chất lượng tài liệu điện tử sau khi giấu tin phải không được thay đổi nhiều để cho con người khó có thể nhận ra bằng các giác quan thông thường.

Thủy vân số là một lĩnh vực nghiên cứu mới, có nhiều triển vọng. Những năm gần đây lĩnh vực này có được sự quan tâm đáng kể của các nhà nghiên cứu.

2.4.3. Các đặc tính và phân loại thủy vân

2.4.3.1.Các đặc tính thủy vân

Tính ẩn: tính ẩn là khả năng khó bị nhận ra của thủy vân sau khi đã nhúng vào tài liệu điện tử, mà chủ yếu là các giác quan của con người. Nói cách khác, các tài liệu điện tử phải chịu ít sự thay đổi về mặt chất lượng khi nhúng vân.

Tính bền vững: Tính bền vững được hiểu tùy vào mục đích của từng lại thủy vân, ví dụ với thủy vân dùng để bảo vệ bản quyền, thì thủy vân phải bền với các phép tấn công hay biến đổi, trong khi với thủy vân dùng để chống xuyên tạc hoặc đảm bảo toàn vẹn dữ liệu, thì thủy vân phải bị phá hủy ngay khi có sự tác động hoặc tấn công.

Tính bảo mật: Sau khi thủy vân số đã được nhúng vào tài liệu, thì yêu cầu chỉ cho những người có quyền mới có thể chỉnh sửa và phát hiện thủy vân .

Tính hiệu quả: yêu cầu thuật toán thủy vân phải làm việc được một vùng lớn các ảnh có thể.

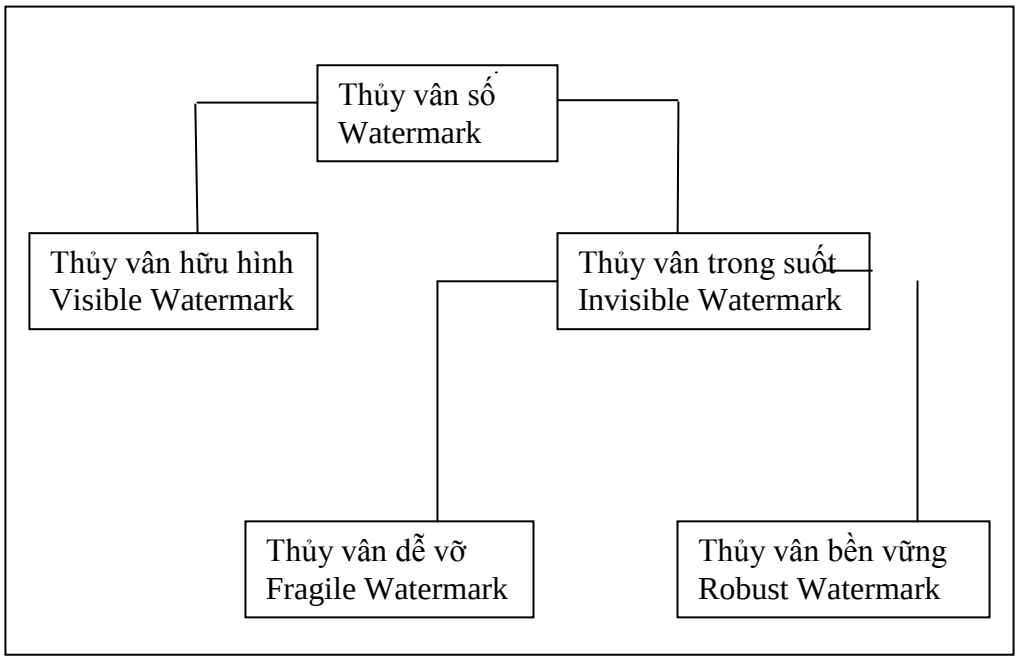
Dung lượng giấu: Thuật toán thủy vân cho phép giấu càng nhiều thông tin càng tốt.

Tuy nhiên, các yêu cầu trên thường là trái ngược nhau, và người ta phải cân đối giữa các yêu cầu để phù hợp với từng bài toán cụ thể.

2.4.3.2. Phân loại thủy vân

Có nhiều phương pháp để phân loại thủy vân, dưới đây trình bày phương pháp phân loại phổ biến nhất:

Dựa vào miền tác động, chúng ta có thể phân loại thủy vân thành tác động lên miền không gian ảnh (spatial domain) và tác động lên miền tần số ảnh (frequency domain).

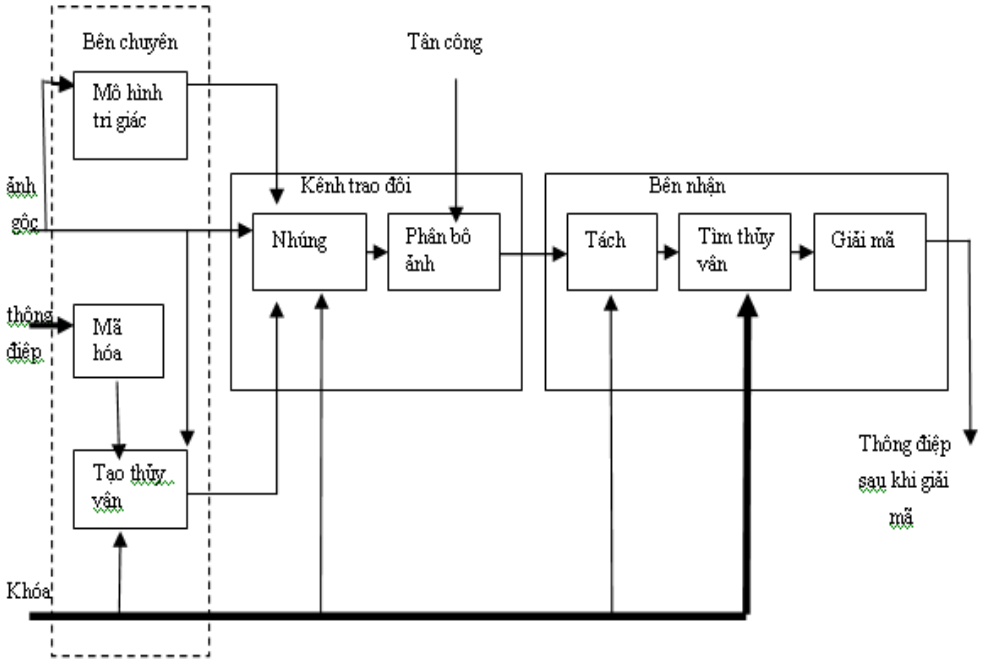


Phân loại thủy vân

Dựa vào tác động tới thị giác con người, chúng ta có thủy vân hiện (visible watermarking) hoặc thủy vân ẩn (invisible watermark). Thủy vân ẩn lại chia thành thủy vân bền (robust watermarking) và thủy vân dễ vỡ (fraglie watermark).

Thủy vân hiện có ưu điểm là nhìn được bằng mắt thường, khiến cho tất cả người sử dụng đều biết được bản quyền của ảnh. Tuy nhiên, nó sẽ tác động tới chất lượng ảnh và gây mất thẩm mỹ.

2.4.4. Qui trình thực hiện thủy vân



Quy trình thực hiện thủy vân

Quy trình thực hiện thủy vân được trải qua bốn bước như sau :

Bước 1: Tạo thủy vân

Thủy vân có thể là một logo hoặc một dãy nhị phân với độ dài cho trước. Thủy vân có thể được biến đổi trước khi đem giấu vào ảnh bằng cách mã hóa, hoặc chuyển đổi định dạng.

Bước 2: Nhúng thủy vân

Thủy vân có thể được nhúng trực tiếp vào ảnh hoặc vào dạng biến đổi của nó. Đối với các ứng dụng bảo vệ bản quyền thì việc nhúng thủy vân vào dạng biến đổi của ảnh là điều cần thiết để đảm bảo tính bền vững của thủy vân trước các biến đổi như nén ảnh. Để đảm bảo sự thay đổi ít nhất về chất lượng ảnh, thủy vân nên được nhúng vào thành phần tần số "giữa" của ảnh sau khi biến đổi ảnh. Đó là vì các thành phần tần số "thấp" rất nhạy cảm đối với các thay đổi và vì vậy sẽ tạo ra sự biến đổi đáng kể chất lượng ảnh, còn thành phần tần số cao thường bị loại trong quá trình nén ảnh mà không làm giảm chất lượng ảnh, do đó thủy vân sẽ dễ dàng bị mất.

Bước 3: Tách thủy vân

Để tách thủy vân ra khỏi ảnh, ta sẽ dùng khóa k trong quá trình nhúng, và ảnh cần tách thủy vân. Thuật toán tách thủy vân có các bước ngược với thuật toán nhúng.

Bước 4: Kiểm tra thủy vân

Đối với thủy vân là một logo thì sau khi tách thủy vân, việc xác định thủy vân có tồn tại hay không là đơn giản. Nếu thủy vân là một dãy số có phân bố Gauss thì có thể dựa vào kiểu tương quan, kiểu phân bố của dãy số thu được để đánh giá sự tồn tại thủy vân.

Đối với ứng dụng nhằm xác thực ảnh thì cần phải xem là có thủy vân hay không. Điều này dẫn đến mô hình kiểm chứng giả thiết và hiệu quả của hệ thống thủy vân có thể được đánh giá theo thuật ngữ lỗi loại I và lỗi loại II. Lỗi loại I ứng với trường hợp thủy vân được tìm thấy mặc dù nó không tồn tại. Còn lỗi loại II ứng với trường hợp thủy vân tồn tại nhưng không tìm thấy.

2.4.5. Các thuật toán thủy vân trên ảnh

Các thuật toán thực hiện thủy vân hiện trên ảnh là tương đối dễ dàng, và đã được nghiên cứu nhiều trong môn xử lý ảnh số. Yêu cầu về tính thẩm mỹ đề cao. Tất nhiên cũng cần có yêu cầu khó sử dụng công cụ xử lý ảnh để loại bỏ thủy vân.

2.4.5.1. Thuật toán giấu thủy vân vào các bit có trọng số thấp

Ý tưởng tự nhiên của thủy vân với ảnh số, cũng như giấu tin, đó sẽ là sử dụng các bit có trọng số thấp (Least Significant Bit - LSB) để giấu thủy vân.

Các bit có trọng số thấp được hiểu là các bit mà nếu thay đổi giá trị của sẽ ít làm thay đổi đến chất lượng ảnh.

Ví dụ, với ảnh bitmap 256 màu, màu của mỗi điểm ảnh được biểu diễn bằng 8 bit, nếu ta thay đổi bit thứ tám của mã màu, thì mã màu cũng chỉ thay đổi giá trị có 1 đơn vị, nên nhìn chung thì cả bức ảnh không bị ảnh hưởng nhiều.

Ta có thể minh họa thuật toán như sau:

Xét thủy vân là chuỗi bit : 0111.

Xét bức ảnh là chuỗi bit :

11001101 11000001 11110000 11110010

Để nhúng thủy vân vào bức ảnh, ta sẽ chia bức ảnh thành các khối 8 bit, và đặt giá trị bit cuối cùng của khối bằng giá trị của bit thủy vân tương ứng

Với minh họa trên, chúng ta có bức ảnh sau khi nhúng thủy vân là:

11001100 11000001 11110001 11110011

Để tách thủy vân, đơn giản ta chỉ làm ngược lại quy trình trên, tức là tách ra các bit cuối cùng của từng khối 8 bit, ta sẽ thu được thủy vân ban đầu.

Muốn tăng tính an toàn của hệ thống, có thể nhúng liên tiếp thủy vân vào các khối 8 bit liền nhau, bởi thường thì dung lượng bức ảnh sẽ lớn hơn nhiều lần so với độ dài của thủy vân.

Ưu điểm của thuật toán trên là đơn giản, và dung lượng giấu cao.

Tuy nhiên, nhược điểm là do quá đơn giản nên rất dễ bị tấn công. Kẻ tấn công chỉ cần thay đổi ngẫu nhiên giá trị của các bit có trọng số thấp là thủy vân đã bị phá hủy.

2.4.5.2. Thuật toán thủy vân ghép nối

Thuật toán được trình bày bởi Bender và đồng nghiệp năm 1996.

Xét một bức ảnh, ta sẽ chia bức ảnh thành hai tập con có trọng lượng phần tử bằng N, gọi là hai tập con A và B. Mỗi phần tử trong tập con A được cộng thêm một lượng d, ngược lại mỗi phần tử trong tập B bị trừ đi một lượng d.

Gọi E(A) và E(B) là các giá trị trung bình của tập A và tập B. Ta sẽ có $E(A) \approx E(B) \approx E(A \cup B)$ và $E(A) - E(B) \approx 0$.

Gọi a và b là hai tập có n phần tử, lấy ngẫu nhiên trong A và B.

$$S = \frac{1}{N}(a[i]-b[i])$$

Theo luật thống kê ta sẽ có:

E(S) = 2d nếu dữ liệu có thủy vân.

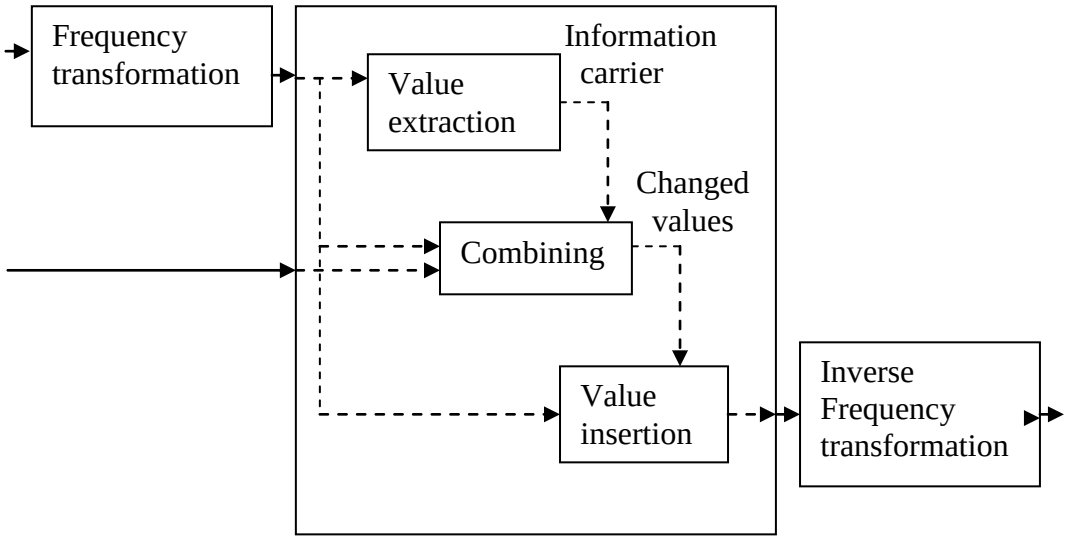
E(S) = 0 nếu dữ liệu không có thủy vân.

Như vậy, để kiểm tra xem có thủy vân hay không, ta sẽ sử dụng luật thống kê. Nếu E(S) lớn hơn một ngưỡng nào đó thì có thể coi là dữ liệu có thủy vân.

2.4.5.3. Thuật toán thủy vân trên miền DCT

1/. Phương pháp Cox

Khác với các phép thủy vân dựa trên biến đổi không gian ảnh, tương đối dễ bị tấn công và phát hiện, năm 1995, Cox đã đưa ra một mô hình khác, đó là nhúng thủy vân vào miền tần số.



Mô hình nhúng thủy vân của Cox

Trong mô hình của Cox, một chuỗi các giá trị $c_0 = c_0[1],c_0[2],\ldots,c_0[n]$ được trích xuất từ ảnh. Các giá trị này được gọi là các giá trị mang, và chúng sẽ chứa thủy vân. Thủy vân là một chuỗi số thực w=w[1], w[2],…, w[n].

Theo Cox đề xuất, thủy vân có thể được nhúng theo một trong ba công thức:

$$c_w[i] = c[i] + \alpha w[i]$$

$$c_w[i] = c[i] (1+\alpha w[i])$$

$$c_w[i] = c[i] \exp(\alpha w[i])$$

trong đó tham số α đặc trưng cho tính bền vững của thủy vân, và được thay đổi tùy từng bài toán cụ thể.

Để kiểm tra thủy vân, người ta cũng phải dùng phương pháp thống kê.

2/. Phương pháp Burgett

Dựa trên mô hình của Cox, năm 1998, Burgett đã đề xuất một thuật toán thủy vân ảnh dựa trên biến đổi DCT với định dạng ảnh JPEG.

Thuật toán của Burgett sẽ chia ảnh JPEG thành các khối (block) có kích thước 8×8 điểm ảnh. Mỗi khối sẽ được biến đổi DCT. Thuật toán sẽ chọn ngẫu nhiên các khối để nhúng thủy vân. Thủy vân được nhúng trên mỗi khối bằng cách đổi chỗ một cặp hệ số của biến đổi DCT.

Năm 2002, GS.TSKH Nguyễn Xuân Huy có đề xuất một thuật toán thủy vân ảnh trên miền DCT như sau:

a/. Quá trình nhúng thủy vân

Chia ảnh có kích thước m x n thành (m × n)/ 64 khối, mỗi khối có kích thước 8×8. Biến đổi DCT cho từng khối.

Xét một khối bất kỳ B sau khi biến đổi DCT thu được khối B', ta chọn hai hệ số bất kỳ trong miền tần số giữa của B', gọi hai hệ số là b'(i, j) và b'(p, q).

Gọi a là tham số của thuật toán, thỏa mãn a = 2 (2t+1) với t là số nguyên dương.

Tính d = ||b'(i, j)|-|b'(p, q)|| mod a.

Bit s_i sẽ được nhúng sao cho thỏa mãn điều kiện sau:

d >= 2t+1 nếu s_i = 1.

d < 2t+1 nếu s_i =0.

Nếu d < 2t+1 và s_i =1, thì một trong hai hệ số DCT có giá trị tuyệt đối lớn hơn sẽ bị thay đổi theo công thức sau để thỏa mãn d >= 2t+1:

Max (|b'(i, j)|, |b'(p, q)|) = Max(|b'(i, j)|, |b'(p,q)|)+ ([0,75*a]-d)

với phép toán [] là phép toán lấy phần nguyên.

Hoặc cũng có thể thay đổi theo công thức sau:

Min(|b'(i, j)|,|b'(p, q)|) = Min(|b'(i, j)|,|b'(p, q)|)- ([0,25*a]+d)

Tương tự , nếu d >= 2t+1 và s_i =0, thì ta áp dụng hai công thức sau để thay đổi hệ số DCT:

Max(|b'(i, j)|,|b'(p, q)|)= Max(|b'(i, j)|,|b'(p, q)|)-(d-[0,25*a])

Hoặc

Min(|b'(i, j)|,|b'(p, q)|)= Min(|b'(i, j)|,|b'(p, q)|)+[1,25*a]-d

b/.Quá trình tách thủy vân

Đọc khối DCT từ ảnh chứa thủy vân và vị trí hai hệ số đã biến đổi, sau đó tính: d = ||b'(i, j)|-|b'(p, q)||mod a

Nếu d>= 2t+1 thì gán s_i =1, ngược lai gán s_i =0.

2.4.5.4. Thuật toán thủy vân ảnh trên miền DWT

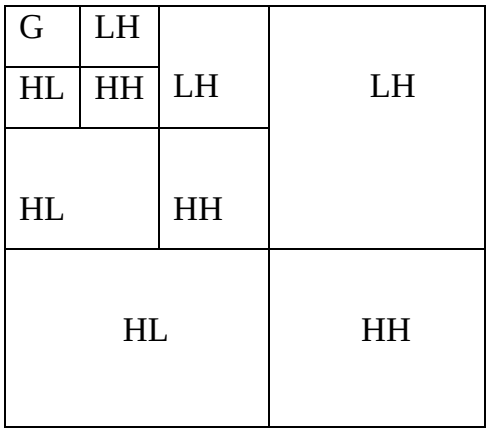
Tương tự như thuật toán thủy vân ảnh trên miền DCT, trong biến đổi DWT, thủy vân được nhúng vào các dải tần số cao nhất, theo công thức sau:

$C_w^{LH}[i,j] = C_o^{LH}[i,j] + \alpha\lambda^{LH}[i,j]\ W[iN + j]$

$C_w^{LH}[i,j] = C_o^{HL}[i,j] + \alpha\lambda^{HL}[i,j]\ W[MN + iN + j]$

$C_w^{LH}[i,j] = C_o^{HH}[i,j] + \alpha\lambda^{HH}[i,j]\ W[2MN + iN + j]$

Mô hình của biến đổi DWT được cho trong hình sau:



Năm 2004, hai tác giả Lê Tiến Thường và Nguyễn Thanh tuấn tại Đại Học Bách Khoa Hồ Chí Minh có đề xuất một giải pháp sử dụng DWT để nhúng thủy vân vào ảnh.

Thuật toán được thực hiện DWT cho ảnh.

Một tập các hệ thống lớn nhất có chiều dài bằng chiều dài watermark trong băng tần thích hợp được trích ra và cộng với watermark theo công thức:

$$C_w = C + \alpha W$$

Quá trình tách thủy vân được thực hiện ngược lại :

$$W = (C_w' - C)/\alpha$$

Trong đó C_w' là các hệ số lớn nhất của ảnh.

Do ảnh có thể bị tấn công nên có thể $C_w \neq C_w'$.

Khi tách được thủy vân,ta so sánh nó với thủy vân gốc S bằng hệ số tương quan d:

$$d = \sum_{i=1}^{N}\sum_{j=1}^{N}(\overline{W}_i * W_j) \Big/ \sqrt{\sum_{i=1}^{N} \overline{Wi}^{\,2} \sum_{j=1}^{N} W_j^{\,2}}$$

Giá trị của d nằm trong khoảng từ -1 tới 1 , nếu d càng gần 1 thì càng có cơ sở xác nhận là ảnh có được nhúng thủy vân.

Thuật toán cho kết quả tốt hơn so với thuật toán nhúng thủy vân trên miền DCT của Cox, đồng thời thủy vân cũng bền với các phép tấn công nén JPEG 2000, lọc, và co rãn ảnh.

2.4.5.5. Thuật toán thủy vân ghép sử dụng biến đổi Karbunen_Loeve

Giáo sư Wang Shuozhong (Vương Thừa Trung) tại Đại học Thượng Hải có đề xuất thuật toán thủy vân dựa trên biến đổi Karbunen_Loeve, hay còn gọi là phân tích các thành phần quan trọng (PCA).

Thuật toán:

Xét một chuỗi các vector f_k, với k = 1,2,…..K là các mẫu được lấy từ một quá trình ngẫu nhiên, f_k có kích thước $R\times1$.

Biến đổi các thành phần quan trọng, hay biến đổi Karbunen_Loeve, được định nghĩa

$$g_k = Af_k$$

Trong đó A là ma trận chuyển kích thước $R\times R$, với mỗi cột là vector đặc trưng (eigenvector) của ma trận thống kê C_F được lấy từ quá trình biến đổi ảnh F. Các cột trong A được sắp sếp theo thứ tự giảm dần của giá trị riêng (eigenvalue).

Xét một ảnh có kích thước MxN, được chia thành K = I.J miền, một miền sẽ là một ma trận 2 chiều có kích thước PxQ, với P = M/I và Q = N/J. Các miền cũng có thể được tổ chức như một mảng một chiều có kích thước PxQ. Có rất nhiều phương pháp chia ảnh thành các miền như vậy. Giáo sư Vương đề xuất một phương pháp đơn giản như sau:

$$f_{i,j}(p,q) = S[\,P(i-1)+p,\ Q(j-1)+q\,], \quad \begin{matrix} i=1,2,\ldots,I \\ j=1,2,\ldots,J \\ p=1,2,\ldots,P \\ q=1,2,\ldots,Q \end{matrix}$$

Như vậy, bức ảnh ban đầu của chúng ta bây giờ có thể xem như K mẫu được lấy từ một quá trình ngẫu nhiên R chiều. Như vậy, kỹ thuật PCA được giới thiệu ở trên đã có thể sử dụng được.

Xét ảnh G , theo như quá trình phân tách ảnh thành các miền đã trình bày ở trên, ta có thể viết lại ảnh G dưới dạng sau:

$$G = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1K} \\ g_{21} & g_{22} & \cdots & g_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ g_{R1} & g_{R2} & \cdots & g_{RK} \end{bmatrix} = [g_1 \quad g_2 \quad \ldots \quad g_k]$$

Sắp xếp lại G theo thứ tự giảm dần của trị riêng, trở thành $q_1, q_2, \ldots, q_R$.

Trị riêng của từng khối q_i chính là năng lượng của khối. Như vậy, khối q_1 có nhiều năng lượng nhất, và sẽ ảnh hưởng nhiều nhất tới ảnh. Như vậy, để tăng tính bền vững của thủy vân, ta nhúng thủy vân vào các khối có hệ số nhỏ, trong khi để tăng tính ẩn thì ta nhúng thủy vân vào các khối có hệ số lớn.

2.4.5.6. Thuỷ vân dễ vỡ và thủy vân giòn

1/. Thủy vân dễ vỡ

Thủy vân dễ vỡ (fragile watermark) là một dạng thủy vân đặc biệt, được sử dụng đảm bảo tính toàn vẹn thông tin của ảnh số. Đặc điểm của thủy vân dễ vỡ là, chỉ cần ảnh bị thay đổi thì thủy vân sẽ bị phá hủy, do đó chống lại sự xuyên tạc nội dung của ảnh.

Thuật toán thủy vân dễ vỡ đầu tiên được Yeung và Mintzer đưa ra vào năm 1997. Thuật toán chỉ hoạt động trên các ảnh xám, với thủy vân là một chuỗi bit. Phía bên người nhận sẽ đọc từng điểm ảnh, trích xuất thủy vân và so nó với thủy vân được công bố. Nếu có sự khác biệt, thì độ xám của điểm ảnh sẽ được thay đổi tới khi hai thủy vân thu được là giống nhau. Thuật toán của Yeung và Mintzer làm việc và bảo toàn tính toàn vẹn dữ liệu cho từng điểm ảnh, cũng như hỗ trợ khả năng khôi phục ảnh gốc.

Một thuật toán khác, được Wong Ping Wah đưa ra năm 1998, kết hợp giữa thủy vân số và mật mã khóa công khai. Trong thuật toán của Wong, mức xám của LSB trong ảnh gốc sẽ được đặt bằng 0. Sau đó, ảnh gốc sẽ được chia thành các khối (block) có kích thước bằng kích thước thủy vân. Kích thước của ảnh cùng với mỗi khối đó sẽ được băm, kết quả thu được sẽ được XOR với thủy vân. Kết quả của phép XOR sẽ được mã hóa bằng hệ mã hóa RSA, sau đó nhúng vào LSB của ảnh gốc.

Phía bên nhận, sẽ làm công việc ngược lại. Đầu tiên, ảnh cũng sẽ lại được chia thành các khi và thông tin nhúng trong LSB sẽ được thu hồi và giải mã cũng bằng hệ mã hóa RSA. Cùng với đó, bên nhận cũng sẽ băm các khối của ảnh thu được cùng kích thước ảnh. Hai kết quả đó được XOR với nhau để thu được thủy vân và so sánh thủy vân thu được với thủy vân gốc được lưu trong cơ sở dữ liệu của bên gửi.

Cả hai thuật toán trên đều có nhược điểm, là không tận dụng sự tương quan giữa các khối ảnh liền nhau, và dung lượng giấu tin thấp.

Để cải tiến nhược điểm này, năm 2003, hai tác giả Li Chang Tsun và Yang Fong man ở Đại Học Havard đã đề xuất một thuật toán thủy vân mới. Thuật toán này sẽ thay đổi mức xám của mỗi điểm ảnh đi một lượng nhất định, lượng này phụ thuộc bản thân thủy vân và một điểm ảnh lân cận của điểm ảnh đang xét. Do điểm ảnh lân cận này là bí mật, cho nên sẽ làm tăng độ an toàn của thủy vân. Thuật toán của Li và Yang là thuật toán thủy vân miền không gian ảnh.

Cũng trong năm 2003, ba tác giả Hsieh Tsung Han, Li Chang Tsun và Wang Shuo đã đề xuất một thuật toán thủy vân trên miền tần số.

Ảnh gốc X được DCT và chia thành các khối 8x8. Một ảnh nhị phân A có kích thước bằng X được tạo ta từ khóa bí mật. Ảnh nhị phân B được tạo ra theo luật : tất cả các điểm ảnh tương ứng với các hệ số khác 0 thì có giá trị 1, ngược lại có giá trị 0. Thủy vân W được tạo ra bằng cách X= A XOR B. W cũng được chia thành các khối 8x8. Sau khi DCT mỗi khối Xi, bốn hệ số Xi(h), Xi(h-1), Xi(h-2) và Xi (h-3) thỏa điều kiện là tần số thấp hơn hoặc bằng tần số giữa h sẽ được chọn là các hệ số thủy vân. Bốn hệ số này được điều chỉnh sao cho phương trình sau được thỏa mãn:

$$P(S_i(j) * X_i(j)) = W_i(j)$$

Trong đó:

P() là hàm số, trả về giá trị 1 nếu số lượng bit 1 trong tham số là lẻ, và bằng 0 nếu ngược lại.

* là phép toán nối $X_i(j)$ và $S_i(j)$.

$S_i(j)$ là tổng của các hệ số thủy vân không âm và $W_i(j)$ thuộc tập $N_i(j)$ – tập chứa DCT của X_i và 8 khối xung quanh.

$$Si(j) = \sum_{m \in N_1\ j} \sum_{n \in\ h-3,\ h} (W_m\ n \oplus W_i\ j) . X_m\ n$$

Quá trình được lặp lại cho tới khi tất cả các khối được thủy vân. Quá trình trích xuất thủy vân được thực hiện theo qui trình ngược lại.

2/.Thủy vân giòn

Thủy vân dễ vỡ có đặc điểm là rất nhạy cảm với thay đổi ảnh, tức chỉ cần có một thay đổi nhỏ trên ảnh là thủy vân đã bị phá hủy. Điều này không thích hợp với thực tế, khi ảnh truyền trên Internet có thể bị sử lý như nén, xoay chiều mà không làm thay đổi nội dung. Thủy vân giòn (semi-fragile watermark) ra đời nhằm cung cấp một khả năng tùy biến tốt hơn cho thủy vân ảnh. Thủy vân giòn chỉ bị hủy khi ảnh có sự thay đổi rất lớn, làm biến dạng ảnh.

Năm 1999, Kundur và Hatzinakos đã đưa ra một thuật toán thủy vân giòn trên ảnh sử dụng DWT.

Đầu tiên, hai tác giả định nghĩa hàm lượng tử hóa Q(t) như sau:

$Q(t) = 0$, nếu $[f/a.2^{-1}]$ chẵn.

$Q(t) = 1$ trong trường hợp ngược lại.

Với hàm [] là hàm lấy phần nguyên của một số. f là một hệ số, 1 là cấp DWT

Thực hiện DWTL cấp đối với ảnh. Trừ hệ số của băng tần có tần số rất thấp, mỗi hệ số f(i) phải thỏa mãn phương trình sau, với qkey(i) là một biến boolean, phụ thuộc vào các lân cận của điểm ảnh i :

$$q(f(i)) = w(i) \text{ XOR } qkey(i)$$

Nếu phương trình trên không được thỏa, cần biến đổi f(i) như sau:

$$Ff(i) = f(i) - a.2^1, \text{ nếu } f(i) > 0$$

$$Ff(i) = f(i) + a.2^1, \text{ nếu } f(i) \leq 0$$

Sau khi quá trình nhúng thủy vân kết thúc, thực hiện IDWT để thu lại ảnh đã nhúng thủy vân.

Ở phía nhận, sẽ thu lại thủy vân bằng công thức sau:

$$W_e(i) = Q(f(i)) \text{ XOR } qkey(i)$$

Hệ số thay đổi ảnh (Tamper Assesment Function- TAF) được định nghĩa là :

$$TAF(w_e, w) = 1/N_w.\sum(w(i) \text{ XOR } w_e(i))$$

TAF đặc trưng cho sự biến đối nhiều hay ít của ảnh, và quyết định có chấp nhận sự thay đổi đó hay không là tùy vào người dùng.

2.4.6. Thủy vân bảo vệ bản quyền audio

Tình hình vi phạm bản quyền âm nhạc ở Việt Nam hiện nay là hết sức trầm trọng. Các đĩa CD lậu với giá chỉ vài nghìn đồng được bày bán công khai và rộng rãi trên nhiều thành phố lớn. Các công ty, tổ chức tự do sử dụng các tác phẩm âm nhạc mà không cần biết đến sự cho phép của tác giả. Theo thống kê, tỷ lệ vi phạm bản quyền âm nhạc của Việt Nam năm 2008 đạt tới 85%, đứng thứ ba ở khu vực Châu Á – Thái Bình Dương.

Quá trình thủy vân đối với audio có nhiều lợi nhuận, bởi tai người không thể nghe được các miền tần số quá cao hoặc quá thấp (trung bình ngưỡng nghe được của tai người chỉ từ 20Hz đến 20kHz). Nếu tận dụng để nhúng thủy vân vào những miền tần số đó thì con người không thể phát hiện được bằng các giác quan thông thường.

2.4.6.1. Giới thiệu audio số

Về bản chất, âm thanh là một sóng cơ học. Âm thanh lan truyền trong môi trường chất rắn là nhanh nhất. Tuy nhiên, để có thể truyền âm thanh đi xa với tốc độ cao, người ta sẽ phải điều chế sóng âm thành sóng điện tử, truyền trên các phương tiện truyền dẫn tới bên nhận. Tại bên nhận sẽ tiến hành giải điều chế để thu được sóng âm ban đầu.

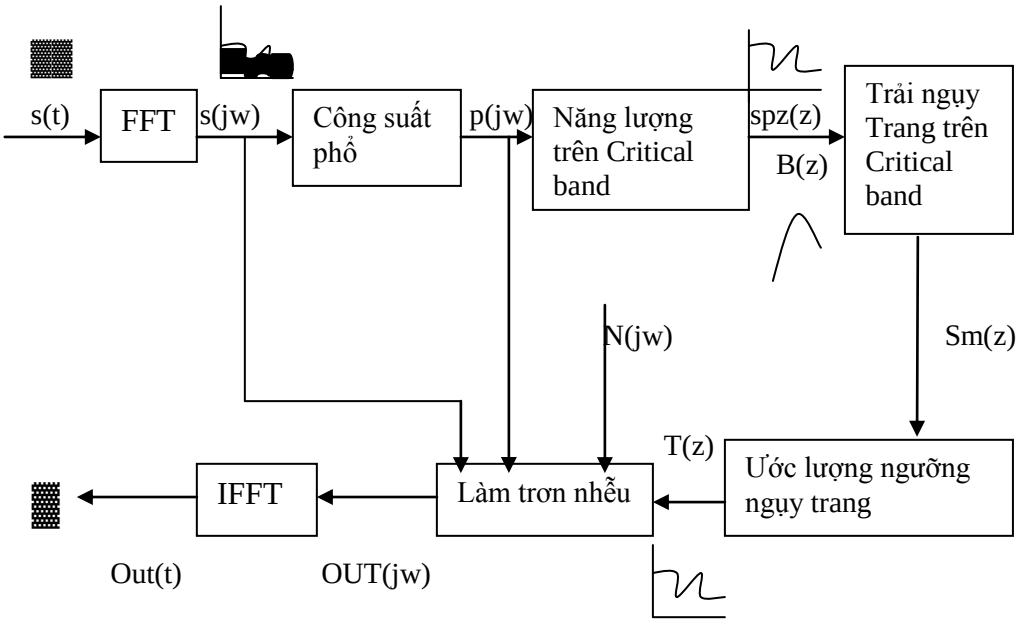
Do âm thanh là một tín hiệu liên tục, nên để có thể số hóa và lưu trữ trong máy tính, chúng ta cần có quá trình lấy mẫu. Để tín hiệu thu được trung thực so với âm thanh gốc, cần tăng tần số lấy mẫu, băng thông bộ lọc và số bit trên mẫu.

Các file nhạc chất lượng cao hiện nay thường có bit lên tới 320Kbps, trung bình là từ 128 Kbps tới 192 Kbps.

2.4.6.2. Thuật toán thủy vân trên audio sử dụng kỹ thuật trải phổ

1/. Mô hình giả lập hệ thính giác

Mô hình giả lập hệ thính giác là thuật toán mô phỏng cơ chế cảm nhận âm thanh của tai người.



Mô hình giả lập hệ thính giác

Ta chia tín hiệu âm thanh thành nhiều đoạn ngắn, có chồng lấn nhỏ lên nhau, gọi là frame.

Gọi s(t) là tín hiệu âm thanh trên miền thời gian.

Sử dụng DFT để chuyển tín hiệu này sang miền tần số. Từng đại lượng trong mô hình được mô tả ở hình trên được tính như sau:

$s(jw) = FFT(s(t))$

$Sp(jw) = |Sw(jw)|^2$

$Spz(z) = \sum Sp(jw)$

Spz được biểu diễn bằng đơn vị Bark. Công thức chuyển đổi từ đơn vị Hz sang Bark như sau:

$$Z = 13\tan^{-1}(\frac{0.76*f}{1000}) + 3.5\tan^{-1}((\frac{f}{5000})^2) = \frac{26.81*f}{1960+f} - 0.53$$

HBZ và LBZ là các tần số trên và tần số dưới của critical band z.

$$B(z)=15.91 + 7.5(z+0.474)-17{,}5\sqrt{1+ z + 0.474^2}$$

$Sm(z) = Spz(z) * B(z)$

T(z) là ngưỡng nghe sau cùng, được tính bằng công thức :

$T(z)=max(Tnorm(z),TH)$

Với Tnorm(z) là ngưỡng nghe sau khi chuẩn hóa ngưỡng nghe thô.

TH là ngưỡng nghe của con người.

$Tnorm(z) = T_{raw}(z)/P_z$.

Trong đó $T_{raw}(z)$ là ngưỡng năng lượng thô, P_z là tổng số điểm trong băng tần z.

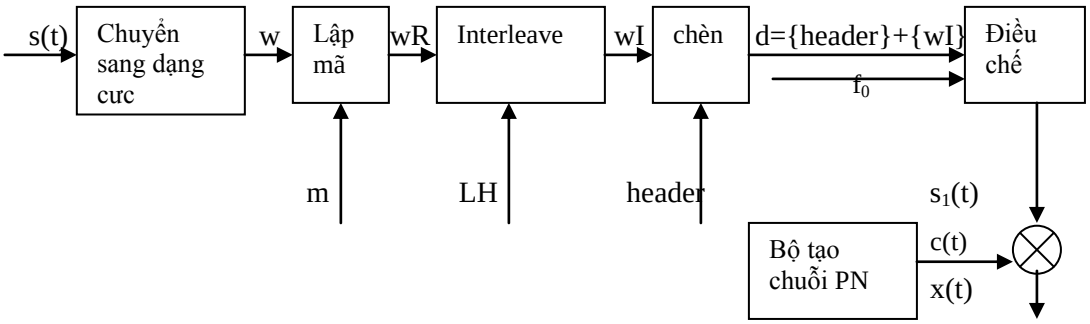
2/.Thuật toán thủy vân

Thuật toán sẽ làm âm thanh giả nhiễu Jammer, là nhiễu có cường độ và năng lượng lớn hơn nhiều so với tín hiệu, trong khi thủy vân lại được giả làm âm thanh gốc.

Như vậy, thủy vân sẽ được âm thanh che chắn, và bền với các phép giảm nhiễu cũng như các phép chuyển đổi định dạng âm thanh.

Tạo thủy vân:

Sơ đồ tạo thủy vân của thuật toán được cho ở hình dưới đây:



Sơ đồ tạo thủy vân

Bước 1: Chuyển chuỗi tín hiệu thủy vân sang dạng cực và lặp dãy bit w m lần. Dãy bit là dãy bit thủy vân ở dạng cực.

Bước 2: Cho chuỗi bit wR đi qua ma trận Interleave H dòng và I cột.

Đầu vào lấy theo dòng, đầu ra lấy theo cột.

Bước 3: thêm head.

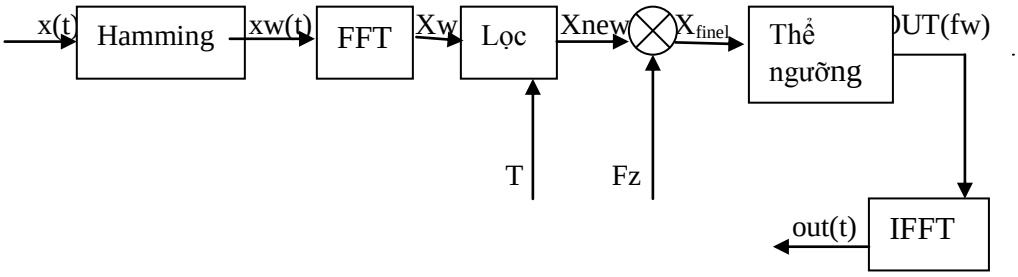
Bước 4: chuyển biểu diễn dãy bit d sang miền thời gian.

Bước 5: Điều chế và trải chuỗi tín hiệu.

Bộ tạo chuỗi PN: là bộ tạo ra một dãy giả ngẫu nhiên (pseu-random number).

Nhúng thủy vân:

Mô hình nhúng thủy vân được cho ở hình dưới:



Sơ đồ nhúng thủy vân

Bước 1: Chia dãy tín hiệu thành N frame. Mỗi frame có n block.

Bước 2: Áp dụng phép biến đổi Fourier nhanh (Fast Fourier transform) cho từng frame.

Xw= FFT (frame[i]*hamming[nBlock])

Bước 3: Chuyển từ miền tần số sang miền Bark.

Bước 4: Tìm trong dãy tín hiệu âm thanh chứa các thành phần nằm trên ngưỡng T, lưu lại vị trí các điểm đó vào dãy above.

X_{new}[above] = Xw[above]

Bước 5: trải tín hiệu, và kết hợp tín hiệu âm thanh và tín hiệu thủy vân:

X_{new} =X_{new} *Fz.

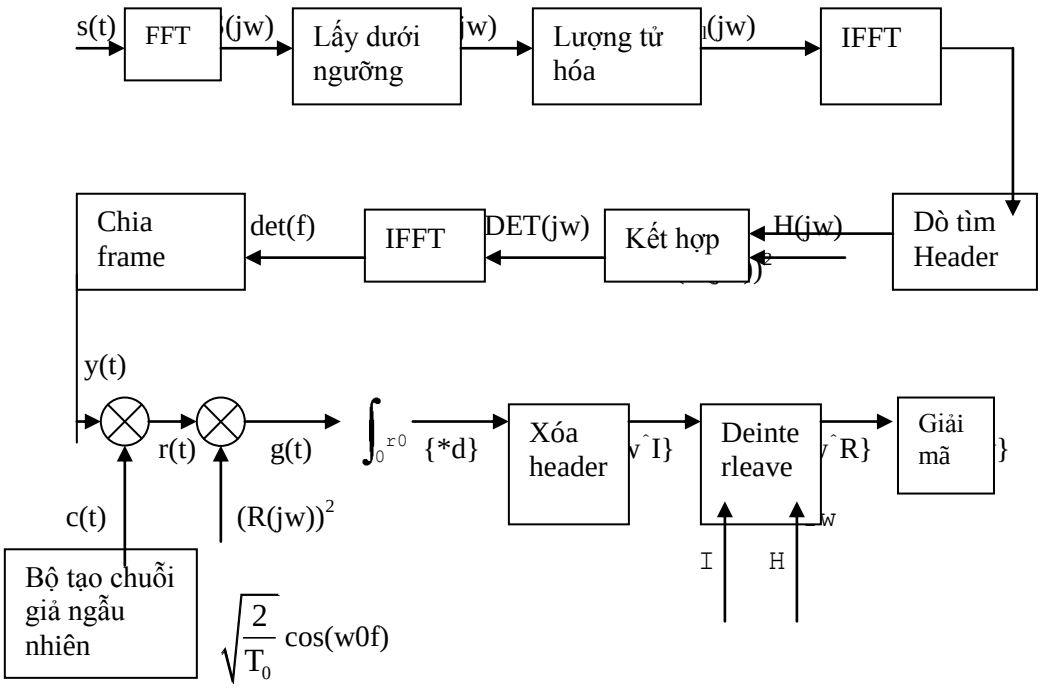
OUT = X_{new} + Snew.

Bước 6 : biến đổi ngược về miền thời gian.

out = IFFT (OUT)

Tách thủy vân:

Quá trình tách thủy vân được chia thành ba giai đoạn



Sơ đồ tách thủy vân

Giai đoạn 1: Lọc bỏ tín hiệu âm thanh và tạo tín hiệu R_{final}.

Bước 1: lọc lấy các thành phần nằm dưới ngưỡng T. R(below) = Sw(below)

Bước 2: lượng tử hóa. Fz[i]= 1/max |R(i)|. R(i) = R(i) * Fz[i]

Bước 3 : chuyển R về miền thời gian thực.

Giai đoạn 2: Dò tìm header.

Xây dựng bộ lọc phân giải cao và áp dụng để tìm ra vị trí đầu tiên của thủy vân.

Giai đoạn 3: Tổng hợp thủy vân.

Xóa bỏ header của chuỗi tìm được và cho tín hiệu thu được đi qua ma trận Interleave để thu lại thủy vân.

Chương 3. BẢO VỆ BẢN QUYỀN TÀI LIỆU SỐ VÀ THỬ NGHIỆM CHƯƠNG TRÌNH

3.1. MỘT SỐ PHƯƠNG PHÁP BẢO VỆ BẢN QUYỀN TÀI LIỆU SỐ

3.1.1. Bảo vệ bản quyền bằng mã hóa

Để bảo vệ bản quyền một tài liệu số người ta mã hóa tài liệu đó. Kẻ gian không nhận biết được nội dung tài liệu này. Do đó không giám nhận tài liệu là của mình.

Ví dụ:

A gửi cho B một bức tranh. Trước khi gửi, A mã hóa để cho C không thể biết đó là bức tranh. Nên C không biết gì để nhận đó là của mình. Khi B nhận được bản mã, B sẽ dùng khóa của mình để giải mã bức tranh.

3.1.2. Bảo vệ bản quyền bằng chữ ký số

Một tài liệu số muốn được bảo vệ bản quyền thì người ta sẽ ký điện tử lên tài liệu đó. Khi kẻ gian muốn nhận là của mình cũng không được vì đã có chữ kí điện tử của chủ sở hữu ở trên tài liệu đó. Nếu kẻ gian muốn nhận là của mình thì phải giả mạo chữ ký số.

Ví dụ:

A gửi cho B một bức tranh, để bảo vệ sở hữu bức tranh thì A kí vào bức tranh. Kẻ gian muốn nhận bức tranh của mình, thì người A có thể minh chứng bức tranh là của mình vì có chữ ký trên bức tranh đó. Kẻ gian không thể giả mạo chữ kí trên bức tranh, nên không giám nhận bức tranh đó thuộc quyền sở hữu của mình. Khi B nhận được bức tranh đó, B sẽ kiểm tra xem có khớp chữ kí có hay không.Nếu không khớp là không đúng.

3.1.3. Bảo vệ bản quyền bằng hàm băm

Muốn bảo vệ bản quyền trước khi gửi tài liệu số cho người khác, người ta băm tài liệu đó ra để tạo ra đại diện. Kẻ gian sẽ không biết người gửi đã dùng hàm băm gì để băm tài liệu đó.

Ví dụ:

A gửi một bức tranh cho B.

Trước khi gửi A sẽ băm bức tranh đó ra, tạo được đại diện Y và gửi đại diện cùng bức tranh X cho B. Khi B nhận được cặp (X, Y), băm bức tranh X ra, tạo đại diện Y', so sánh với đại diện Y. Nếu Y' = Y thì tức là bức tranh vẫn còn nguyên vẹn không bị sửa đổi trên đường truyền. Nếu không khớp tức là bức tranh đã bị vào tay kẻ gian trên đường truyền.

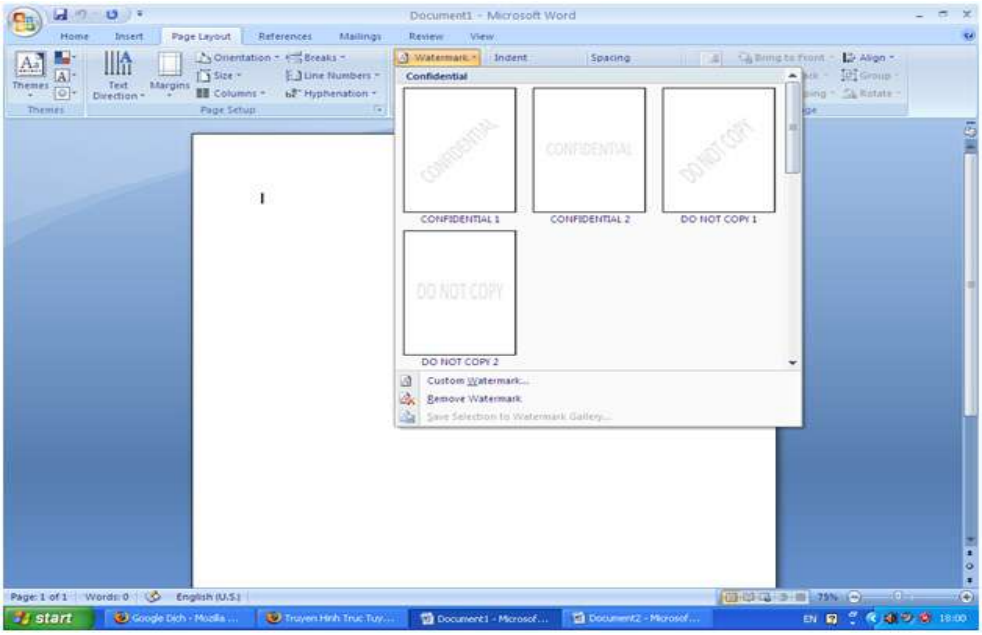
Người A đăng kí sở hữu bức tranh X bằng minh chứng bản quyền Y

3.1.4. Bảo vệ bản quyền bằng thủy vân ký

Mục đích của thủy vân với bảo vệ bản quyền là gắn một "dấu hiệu" vào tài liệu số cần giữ bản quyền. Ví dụ gắn dấu vân tay để xác định người dùng của sản phẩm. Dấu hiệu có thể là một dãy số như mã hàng hóa quốc tế, một tin nhắn hoặc một logo,…

Ví dụ:

A gửi cho B bức tranh, thì A kèm tên của mình thủy vân vào bức tranh gửi cho B. Giống như việc sử dụng watermark trong word 2007. Đây là thủy vân hiện.



Thủy vân ẩn: Trong bức tranh, A bí mật giấu tên của mình trong các bít ít quan trọng trong bức tranh. Điều này cho phép trao đổi thông tin mà không gây chú ý đối với kẻ gian.

3.2. CHƯƠNG TRÌNH THỬ NGHIỆM NHÚNG THỦY VÂN TRONG MIỀN LSB CỦA ẢNH

Bít LSB là bit có ảnh hưởng ít nhất tới việc quyết định tới màu sắc của mỗi điểm ảnh, vì vậy khi ta thay đổi bit ít quan trọng của một điểm ảnh thì màu sắc của mỗi điểm ảnh mới sẽ tương đối gần với điểm ảnh cũ.

3.2.1. Giới thiệu bài toán

1/. Xét bài toán có

- Input: +Một file ảnh màu gốc S.

+Thông điệp cần thủy vân (S): một chuỗi ký tự hoặc một ảnh nhỏ.

-Output: + Một file ảnh đã nhúng thủy vân S'.

2/. Ý tưởng

Tách ra các bit ít quan trọng LSB của mỗi điểm ảnh, sau đó ta sẽ giấu thông điệp tại mỗi bit không quan trọng đó.

3/. Các bước thực hiện thủy vân

- Chuyển thủy vân cần giấu sang dạng nhị phân.
- Đọc dữ liệu của ảnh (sử dụng kỹ thuật Lockbit để tăng tốc độ xử lý).
- Tại mỗi điểm ảnh, xét thành phần Blue (thành phần mắt người khó phân biệt), và nhúng bit thủy vân. Cụ thể:

+Biến đổi giá trị của Blue của điểm ảnh sang dạng nhị phân 8 bit (B) và tính tổng số bít 1 (T).

+Thực hiện nhúng thủy vân:

Nhúng bit 1:

*Khi T = 1 (T lẻ), ta không thay đổi giá trị bit cuối cùng của B;

*Khi T = 0 (T chẵn), ta đổi bit cuối cùng của B (nếu đang là 1 thì chuyển thành 0 và ngược lại)

Nhúng bit 0:

*Khi T = 1, ta đổi bít cuối của B (nếu đang là 1 thì chuyển thành 0 và ngược lại)

*Khi T =0, ta không thay đổi giá trị bít cuối cùng của B:

+Trả lại giá trị mới cho thành phần Blue của điểm ảnh đang xét.

- Quá trình được thực hiện cho đến khi giấu hết các bít thủy vân vào ảnh.
- Như vậy đảm bảo được khi xét mỗi thành phần Blue của mỗi điểm ảnh đã biến đổi. Nếu :

+Tổng số bít 1 là chẵn→ bít ta giấu là 0

+Tổng số bit 1 là lẻ→ bít ta giấu là 1

4/. Các bước tách thủy vân

- Xét thành phần Blue của những điểm ảnh đã thực hiện biến đổi theo trình tự đã giấu. Căn cứ theo quy tắc:

+Tổng số bít 1 là chẵn→ tức bít ta giấu là 0

+Tổng số bit 1 là lẻ→ tức bít ta giấu là 1

- Lưu các giá trị đó lại ta sẽ được thủy vân đã giấu.

3.2.2. Kết quả thực hiện

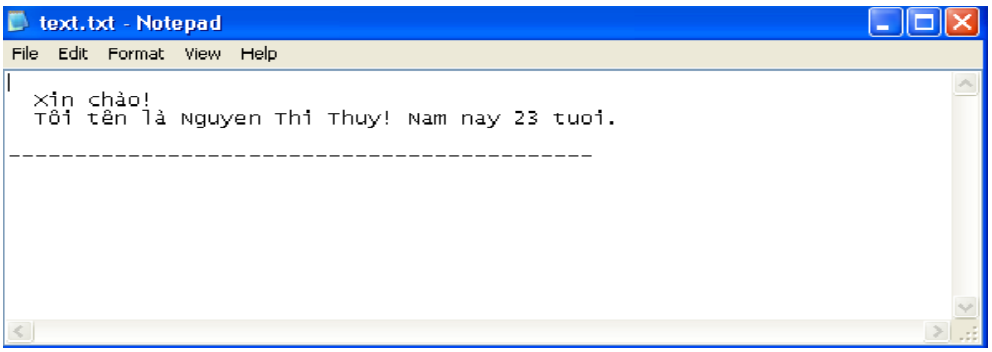
1/. Giao diện và chức năng của chương trình

Giao diện chính của chương trình

Các chức năng chính của chương trình

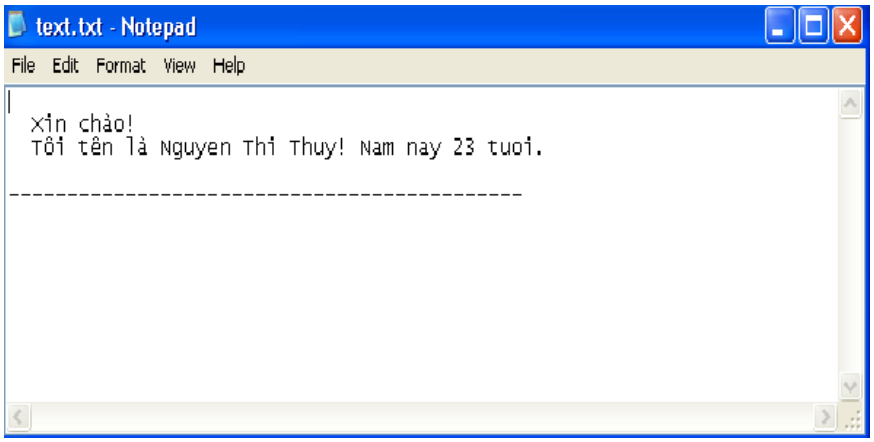
2/. Thử nghiệm thủy vân là dữ liệu văn bản

ảnh gốc



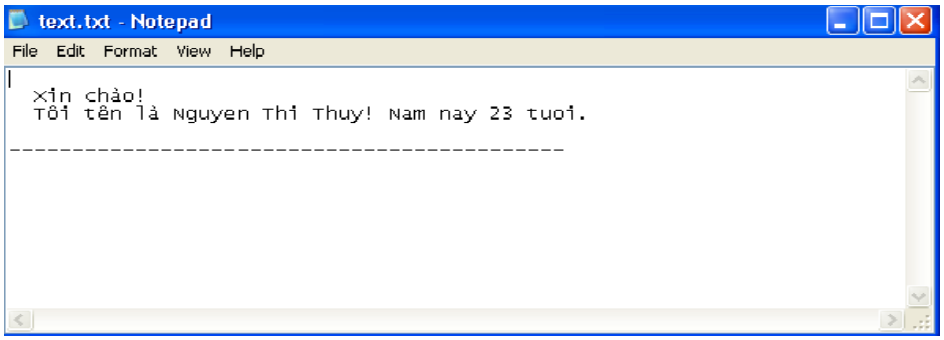
Nội dung thủy vân dạng văn bản

Ảnh sau khi nhúng thủy vân dạng văn bản



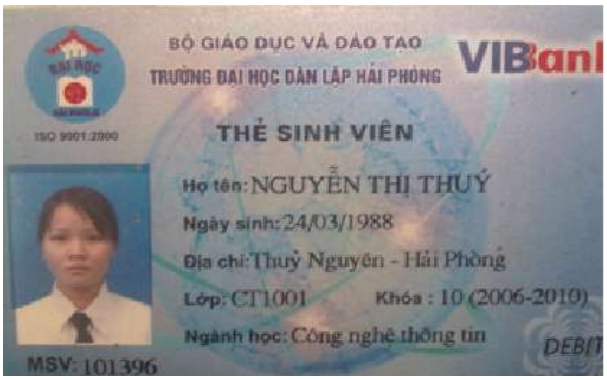
Kết quả sau khi giải mã

Ảnh sau khi nhúng thủy vân dạng văn bản bị vẽ ngẫu nhiên

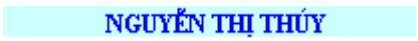


Kết quả sau khi giải mã

3/. Thủy vân là dữ liệu hình ảnh



Ảnh gốc



Nội dung thủy vân dạng hình ảnh

Ảnh sau khi nhúng thủy vân dạng hình ảnh

Nhận xét: Thủy vân dạng ảnh có khả năng chống chịu lại các phép xử lý tốt hơn nhiều so với thủy vân dạng ký tự. Do tính chất bền vững được đảm bảo hơn nên thủy vân dạng ảnh được sử dụng nhiều hơn.

KẾT LUẬN

Nội dung khóa luận đã trình bày các phương pháp bảo vệ bản quyền tài liệu số. Lĩnh vực bảo vệ bản quyền là một lĩnh vực rất rộng lớn, bao gồm nhiều mặt, nhiều lĩnh vực của đời sống. Bảo vệ bản quyền phải có sự góp sức của khung hành lang pháp lý, ý thức của cộng đồng sử dụng cũng như của chính tác giả, và cuối cùng là các kỹ thuật bảo vệ tài liệu số.

Kết quả chính của khóa luận:

1/. Tìm hiểu và nghiên cứu lý thuyết

- Các khái niệm, của mã hóa, chữ ký số, thủy vân ký…
- Một số phương pháp bảo vệ bản quyền tài liệu số

2/. Thử nghiệm chương trình thủy vân

- Nhúng được dữ liệu của một đoạn văn hoặc một hình ảnh vào trong ảnh gốc. Tương ứng với mỗi lần thực hiện thủy ký lên đối tượng gốc, đều đưa ra một khóa giải mã.
- Với mỗi ảnh bất kỳ đã nhúng thủy ký của chương trình (nhúng văn bản hoặc hình ảnh). Hệ thống tách được dữ liệu nhúng phù hợp.
- Kết quả của ảnh gốc và ảnh nhúng thủy ký gần như không có sự khác biệt nào đáng chú ý
- Giao diện chương trình dễ sử dụng.

Trong thời gian tới, em sẽ tiếp tục hoàn thiện đề tài hơn nữa nhằm mục tiêu hướng đến việc triển khai sử dụng trong thực tế.

Cuối cùng em xin chân thành cảm ơn thầy PGS TS. Trịnh Nhật Tiến đã tận tình hướng dẫn, giúp đỡ em hoàn thành tốt đề tài khóa luận.

PHỤ LỤC

Vi phạm bản quyền tác giả có thể bị phạt tới 500 triệu đồng

Nghị định số 47/2009/NĐ-CP ngày 13/5/2009 qui định sử phạt vi phạm hành chính (VPHC) về quyền tác giả, quyền liên quan với mức phạt tiền cao nhất là 500 triệu đồng sẽ có hiệu lực thi hành từ ngày 30/6/2009

Mức phạt cao nhất tới 500 triệu đồng:

Nghị định qui định chi tiết từng hành vi vi phạm, hình thức và mức phạt. Bao gồm: Vi phạm qui định về đăng ký; vi phạm về hoạt động của tổ chức đại diện tập thể; vi phạm qui định trong giám định về quyền tác giả, quyền liên quan; vi phạm qui định về tổ chức tư vấn, dịch vụ; cản trở bất hợp pháp hoạt động quản lý nhà nước, thanh tra, kiểm tra về quyền tác giả, quyền liên quan…

Đặc biệt, mức phạt nặng nhất 500 triệu đồng áp dụng đối với các hành vi vi phạm như sau: Sao chép tác phẩm, sao chép trực tiếp hoặc gián tiếp cuộc biểu diễn, sao chép bản định hình chương trình phát sóng mà không được phép của chủ sở hữu quyền (trong trường hợp hàng hóa vi phạm có giá trị trên 500 triệu đồng); chiếm đoạt quyền biểu diễn tác phẩm trước công chúng, quyền sao chép tác phẩm, quyền phân phối, nhập khẩu bản gốc hoặc bản sao tác phẩm, quyền cho thuê bản gốc hoặc bản sao tác phẩm điện ảnh, chương trình máy tính, quyền phát sóng, tái phát sóng chương trình phát sóng…

Theo quy định cũ tại Nghị định 56/2006/NĐ-CP, mức phạt tối đa đối với các hành vi VPHC về quyền tác giả, quyền liên quan là 70 triệu đồng.

Bên cạnh đó, một điểm mới của nghị định là ngoài các hình thức xử phạt chính và bổ sung, tổ chức, cá nhân vi phạm còn có thể bị buộc áp dụng một hoặc nhiều biện pháp khắc phục hậu quả như: Buộc khôi phục lại quyền đứng tên, đặt tên, bảo vệ sự toàn vẹn của tác phẩm; buộc tiêu hủy hàng hóa vi phạm; buộc dỡ bỏ bản gốc, bản sao tác phẩm, cuộc biểu diễn, bản ghi âm, ghi hình, chương trình phát sóng đã truyền đạt trái phép trên mạng kỹ thuật số hay dưới hình thức điện tử.

Thẩm quyền xử phạt:

Theo Nghị định, thanh tra viên chuyên nghành thuộc Bộ hoặc sở Văn Hóa, Thể thao và Du Lịch (VHTTDL)đang thi hành công vụ có quyền phạt đến 0,5 triệu đồng; Chánh Thanh tra sở VHTTDL có quyền phạt đến 30 triêu đồng và Chánh Thanh tra Bộ VHTTDL có quyền phạt đến mức tối đa của khung hình phạt.

Chủ tịch UBND cấp xã có quyền phạt đến 2 triệu đồng, cấp huyện phạt đến 30 triệu đồng và cấp tỉnh có quyền phạt đến mức tối đa của khung hình phạt.

Ngoài ra, các lực lượng Hải quan, Quản lý thị trường, Công an nhân dân, Bộ Đội Biên Phòng, Cảnh sát biển cũng có thẩm quyền xử phạt một số hành vi vi phạm.

Bên cạnh các văn bản luật, dưới luật điều chỉnh các quan hệ liên quan đến quyền tác giả và quyền liên quan như Bộ luật dân sự, Luật sở hữu trí tuệ, Luật xuất bản…, cùng với quá trình mở cửa và hội nhập, Việt Nam đã tham gia một số điều ước liên quan đến bảo hộ quyền tác giả như: Công ước Berne về bảo hộ các tác phẩm văn học, nghệ thuật; Hiệp định TRIPs về những khía cạnh liên quan tới quyền sở hữu trí tuệ; Hiệp ước WIPO về quyền tác giả và về biểu diễn, ghi âm…Đồng thời, Việt Nam cũng đã ký một số hiệp định song phương với một số quốc gia về bản quyền và các vấn đề liên quan tới quyền tác giả như: Hiệp định Việt Nam – Hoa Kỳ về thiết lập quan hệ quyền tác giả; Hiệp Định Việt Nam – Thụy Sĩ về bảo hộ sở hữu trí tuệ và hợp tác trong lĩnh vực sở hữu trí tuệ.

(Theo chinhphu.vn)

TÀI LIỆU THAM KHẢO

[1]. PGS TS. Trịnh Nhật Tiến, "Giáo trình an toàn dữ liệu", Trường Đại Học Công Nghệ, Đại Học Quốc Gia Hà Nội.

[2]. Chu Văn Huy, "Nghiên cứu kỹ thuật thủy vân số trong việc bảo vệ bản quyền ảnh số", tiểu luận, Trường Đại Học Công Nghệ, Đại Học Quốc Gia Hà Nội.

[3]. http://www.google.com.vn

[4]. http://wikipedia.org

[5]. http://chinhphu.vn