HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

NGUYỄN HUY THẮNG

NGHIÊN CỨU VỀ CHỮ KÝ SỐ VÀ ỨNG DỤNG TRONG HÓA ĐƠN ĐIỆN TỬ TẠI VNPT HÀ NỘI

Chuyên ngành: Hệ thống thông tin

Mã số: 60.48.01.04

TÓM TẮT LUẬN VĂN THẠC SĨ

Người hướng dẫn khoa học: PGS. TS TRẦN ĐÌNH QUẾ

HÀ NỘI – 2013

MỞ ĐẦU

Thời gian gần đây, các nhà cung cấp dịch vụ mạng Viễn thông ở nước ta đang ngày càng chú ý đến chất lượng dịch vụ cũng như đẩy mạnh việc chăm sóc khách hàng của mình ngày một tốt hơn. Cùng với điều đó, Internet cũng đang ngày càng phát triển ở nước ta, mọi dữ liệu đều có thể được số hóa và truyền tải và lưu trữ dễ dàng trên mạng.

Trước đây việc thể hiện các loại cước trên ấn phẩm in chi tiết của Viễn thông Hà Nội là rất nhiều, công in và các chi phí dành cho in ấn là rất lớn. Nhằm tiết giảm chi phí in, bản kê chi tiết theo định dạng HTM đã được đưa vào hệ thống portal của Viễn thông Hà Nội và giúp cho khách hàng chủ động được việc tra cứu cũng như lưu trữ được các bản kê hàng tháng. Lợi ích này được thể hiện rõ rệt khi giảm được rất nhiều kinh phí dành cho in ấn, giảm được đến hơn 2/3 chi phí dành cho in ấn phẩm. Việc triển khai hóa đơn điện tử tại VNPT Hà Nội sẽ giúp giảm chi phí quản lý, lưu trữ, tìm kiếm hóa đơn đã sử dụng và giảm chi phí phát hành bộ chứng từ thu cước: hóa đơn in, thông báo cước, bảng kê chi tiết .... Đề tài luận văn "Nghiên cứu về chữ kí số và ứng dụng trong hóa đơn điện tử tại VNPT Hà Nội tập trung nghiên cứu kỹ thuật kí số, sau đó xây dựng một chương trình có áp dụng kĩ thuật kí số qua TOKEN để kí lên file hóa đơn PDF của Viễn thông Hà Nội.

Chương 1: **Tổng quan về chữ ký số**

Chương này tập trung nghiên cứu khái quát lý thuyết chữ kí số. Trong đó sẽ có cái nhìn tổng quan về chữ kí số và các thuật toán mã hoá khóa công khai. Sau khi có được lý thuyết cơ bản về chữ kí số, luận văn sẽ nêu ra phương pháp tiếp cận và thực hiện.

Chương 2: **Cơ sở chữ ký số**

Chương này nghiên cứu cụ thể về cơ sở hạ tầng cơ bản để tạo chữ ký số. Trong đó nghiên cứu cụ thể về mã hoá khóa công khai dùng trong tạo chữ ký số - từ đó đánh giá để chọn ra giải thuật tối ưu hơn, hạ tầng khoá công khai PKI, hàm băm, kỹ thuật tạo chữ ký số, PDF với chữ ký số.

Chương 3: **Áp dụng chữ ký số cho bài toán in hóa đơn VT01 tại Viễn thông Hà Nội**

Chương này sẽ xây dựng một ứng dụng cụ thể, cài đặt một ứng dụng chữ ký số cho tài liệu PDF hóa đơn VT01 tại Viễn thông Hà Nội.

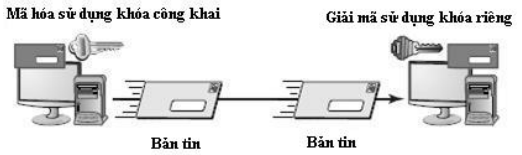
CHƯƠNG 1: KHÁI QUÁT VỀ CHỮ KÍ SỐ

Chương này tập trung nghiên cứu khái quát lý thuyết chữ kí số. Trong đó sẽ có cái nhìn tổng quan về chữ kí số và các thuật toán mã hoá khóa công khai. Sau khi có được lý thuyết cơ bản về chữ kí số, luận văn sẽ nêu ra phương pháp tiếp cận và thực hiện.

1.1 Tổng quan chữ kí số

1.1.1 Thuật toán khóa công khai

Mã hóa bất đối xứng thường được hiểu là mã hoá sử dụng khóa công khai [1]. Mã hóa bất đối xứng sử dụng một cặp khóa: khóa bí mật và khóa công khai, được miêu tả như hình 1.1. Mỗi quá trình truyền tin sử dụng một cặp khóa duy nhất và có thể sử dụng linh hoạt. Khóa bí mật cần phải lưu trữ riêng và đảm bảo tính bảo mật, không được truyền trên mạng. Khóa công khai có thể được cung cấp miễn phí và công bố tới mọi người.



Hình 1.1 Hệ thống sử dụng mã hóa khóa công khai [1]

Tương tự như mã hoá khoá bí mật, phương pháp này cũng có các thành phần chính như sau:

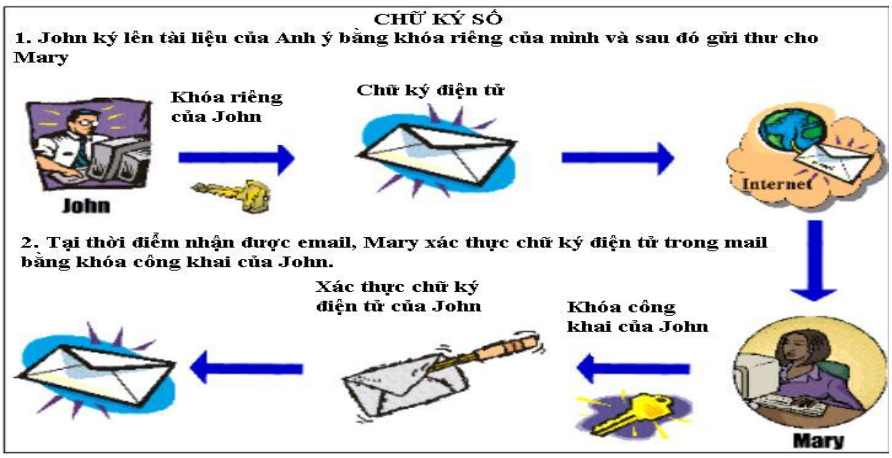
- Plaintext: bản tin gốc.
- Encryption Algorithm: phép biến đổi xuôi, thực hiện biến đổi bản tin gốc.
- Public/Private keys: cặp khóa công khai/bí mật.
- Ciphertext: bản tin đã biến đổi.
- Decryption Algorithm: phép biến đổi ngược, khôi phục bản tin gốc.

Quá trình sử dụng mã hóa khóa công khai:

- Bên nhận sinh cặp khóa.
- Khóa công khai thường được chứng thực bởi một bên thứ ba tin cậy và chuyển cho người gửi theo các phương thức truyền thông thông thường.
- Bên gửi nhận được khóa công khai, kiểm tra các thông tin chứng thực khoá và dùng khóa này để mã hóa thông điệp và gửi cho bên nhận.
- Bên nhận sử dụng khoá bí mật để giải mã thông điệp. D(Kd(E(Ke,M)) = M
- Thông điệp có thể bị bên thứ ba lấy trộm, nhưng không thể đọc được nội dung.

1.1.2 Chữ ký số

Chữ ký số : Là một thể chứng thực được mã hóa bởi khoá bí mật của người gửi. Chữ ký số là thông tin đi kèm theo dữ liệu (văn bản, hình ảnh, video...) nhằm mục đích xác định chủ thể của dữ liệu đó. Quá trình tạo và xác thực chữ ký số được mô tả như Hình 1.2.



Hình 1.2 Chữ ký số

1.1.2.1 Vai trò của chữ ký số

Chữ ký số dùng cho các văn bản số, cho biết toàn bộ văn bản đã được ký bởi người ký. Và người khác có thể xác minh điều này. Chữ ký số tương tự như chữ ký thông thường, đảm bảo nội dung tài liệu là đáng tin cậy, chính xác, không hề thay đổi trên đường truyền và cho biết người tạo ra tài liệu là ai. Tuy nhiên, chữ ký số khác chữ ký thường, vì nó tuỳ thuộc vào văn bản. Chữ ký số sẽ thay đổi theo văn bản còn chữ ký thường thì không hề thay đổi.

Chữ ký số được sử dụng để cung cấp chứng thực chủ sở hữu, tính toàn vẹn dữ liệu và chống chối bỏ nguồn gốc trong rất nhiều các lĩnh vực.

1.1.2.2 Ứng dụng của chữ ký số

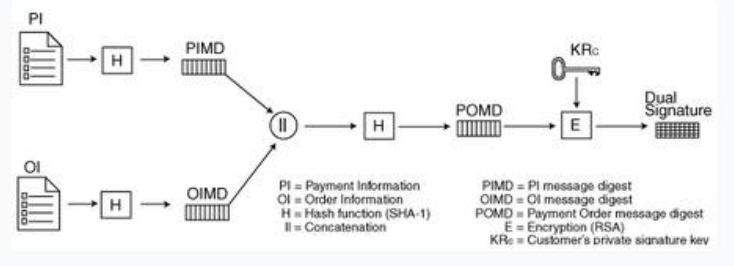
Giải pháp dùng chữ ký số là tối ưu vì nó có hiệu lực pháp luật, do đó không cần in ấn tài liệu mà vẫn có thể xác nhận được tài liệu, đảm bảo tính toàn vẹn và không chối bỏ. Chữ ký số được phát hành bởi bên thứ ba là cơ quan chứng thực có thẩm quyền cấp phát, thu hồi, quản lý chứng chỉ số cho các thực thể thực hiện các giao dịch an toàn (Certificate Authority hoặc CA) nên đảm bảo tính khách quan. Như vậy, quá trình tạo chữ ký số, xác nhận các yêu cầu pháp lý, bao gồm xác thực người ký, xác thực tin nhắn, là thành công và hiệu quả.

Chính vì những ưu điểm của chữ ký số, nó được dùng trong nhiều ứng dụng: Đảm bảo an ninh truyền thông, ngân hàng trực tuyến, thương mại điện tử, đảm bảo an ninh cho thư điện tử, …

1.2 Chữ ký kép

Chữ ký kép được sử dụng để xác minh khi dữ liệu được tạo thành từ các phần khác nhau của các đơn vị logic đơn lẻ. Chữ ký kép kết nối hai bản tin một cách an toàn, nhưng mỗi bên chỉ có thể đọc được thông tin dành riêng cho mình [9].

Trong thương mại điện tử, trường hợp khách hàng muốn gửi một thông tin đặt hàng (OI – Order Information) tới nhà cung cấp và thông tin thanh toán (Payment Information - PI) tới ngân hàng. Nhà cung cấp không cần biết mã số thẻ tín dụng của khách hàng và ngân hàng cũng không cần thiết chi tiết đặt hàng của khách hàng. Khách hàng được cung cấp sự bảo vệ tính riêng tư bằng việc giữ hai mục tách rời nhau. Tuy nhiên, hai mục phải được liên kết với nhau theo cách mà có thể được sử dụng để giải quyết các vấn đề tranh cãi khi cần. Liên kết được yêu cầu để khách hàng có thể chứng minh rằng thanh toán này dành cho đặt hàng này mà không phải là cho các mặt hàng hoặc dịch vụ khác. Có thể tham khảo hình 1.3 sau:



Hình 1.3 Cấu trúc chữ ký kép [9]

1.3 Hiện trạng thực tế đối với vấn đề in ấn hóa đơn tại Viễn thông Hà Nội

1.3.1 Hiện trạng in hóa đơn VT01

Hiện nay tại Viễn thông Hà Nội, việc in hóa đơn VT01 theo mẫu của Bộ Tài Chính vẫn được in theo quy trình như sau:

- Dữ liệu cước hàng tháng sau khi được tính sẽ được tổng hợp lại theo từng mã khách hàng trên cơ sở dữ liệu Oracle.
- In hóa đơn cho các mã khách hàng theo từng khu vực và được sắp xếp theo thứ tự nhất định: Đơn vị, Mã đường thư, Số hóa đơn
- Tạo file text hóa đơn (*.txt) có cấu trúc theo từng khu vực (ví dụ: VT1THK04.TXT; VT2CHK04.TXT; VT3PCG04.TXT; …
- Bàn giao dữ liệu hóa đơn VT01 dạng text cho nhà máy in

- Sau khi in, hóa đơn sẽ được trả về cho từng đơn vị quản lý bán hàng và được chuyển đến tay các đại lý thu thuê để đi thu tiền của khách hàng.

1.3.2 Một số vấn đề sai sót có thể mắc phải

Do số lượng hóa đơn hàng tháng in là lớn, lên đến gần một triệu khách hàng trong một tháng, việc phát sinh sai sót là khó tránh khỏi:

- Tính cước sai cho khách hàng
- Tổng hợp cước sai cho khách hàng
- Bàn giao dữ liệu in hóa đơn VT01 sai tháng cần in

Những sai sót này lại thường chỉ được phát hiện khi hóa đơn VT01 đã đến được tay khách hàng. Điều này sẽ làm tốn rất nhiều chi phí công in ấn, giấy mực cũng như nhân công để đi thu hồi lại các ấn phẩm đã in sai

1.5 Tổng kết chương

Chữ ký số có vai trò quan trọng trong các giao dịch điện tử. Chữ ký số tạo ra một bước tiến lớn trong các giao dịch thương mại điện tử, đảm bảo tính an toàn và tin cậy trong truyền thông trên mạng. Các thông tin khi truyền trên mạng trở nên nhanh chóng, tin cậy hơn. Chữ ký số cũng giúp cho quá trình làm việc trên mạng nhanh chóng và hiệu quả hơn, giảm thiểu các chi phí liên quan như khi dùng chữ ký thông thường.

CHƯƠNG 2: CƠ SỞ CHỮ KÝ SỐ

Chương này nghiên cứu cụ thể về cơ sở hạ tầng cơ bản để tạo chữ ký số. Trong đó nghiên cứu cụ thể về mã hoá khóa công khai dùng trong tạo chữ ký số - từ đó đánh giá để chọn ra giải thuật tối ưu hơn, hạ tầng khoá công khai PKI, hàm băm, kỹ thuật tạo chữ ký số, PDF với chữ ký số.

2.1 Thuật toán mã hoá khóa công khai RSA

Thuật toán RSA được phát minh năm 1978, sử dụng chế độ mã hóa khối. RSA là một thuật toán mã hóa khóa công khai [5]. Đây là thuật toán đầu tiên phù hợp với việc tạo ra chữ ký điện tử đồng thời với việc mã hóa. Nó đánh dấu một sự tiến bộ vượt bậc của lĩnh vực mật mã học trong việc sử dụng khóa công cộng. Thuật toán RSA có hai khóa: khóa công khai (hay khóa công cộng) và khóa bí mật (hay khóa cá nhân). Mỗi khóa là những số cố định sử dụng trong quá trình mã hóa và giải mã. Khóa công khai được công bố rộng rãi cho mọi người và được dùng để mã hóa. Những thông tin được mã hóa bằng khóa công khai chỉ có thể được giải mã bằng khóa bí mật tương ứng. Nói cách khác, mọi người đều có thể mã hóa nhưng chỉ có người biết khóa cá nhân (bí mật) mới có thể giải mã được.

2.1.1 Sinh cặp khóa

Các tham số

1. Chọn hai số nguyên tố lớn p và q. Tính n = p x q và m = φ(n) = (p = 1) x (q-1).

2. Chọn e, $1 \leq e \leq m - 1$, sao cho gcd (e, m) = 1.

3. Tìm d sao cho e x d = 1 (mod m), tức là tính d = e-1 (mod m.

Khóa công khai (Public key) là (e, n)

Khoá bí mật (Private key) là d, p, q).

2.1.2 Mã hóa và giãi mã

Với M, C là một số nguyên $\in$ (0, n) và là biểu diễn dạng số nguyên của bản rõ và bản mã tương ứng. Ta có:

$C = E_{PU}(M)$: mã hóa bản rõ với khóa PU

$M = D_{PR}(E_{PU}(M))$: giải mã bản mã với khóa PR (ko cho phép tính được PR từ PU)

Dạng mã hóa / giải mã:

$C = M^e \bmod n$

$M = c^d \bmod n = M^{ed} \bmod n$

PU = {e, u} -> Public

PR = {d, n} -> Private

2.1.3 Bảo mật của RSA

Độ an toàn của hệ thống RSA dựa trên vấn đề của toán học: bài toán phân tích ra thừa số nguyên tố các số nguyên lớn. Nếu bài toán trên là khó (không tìm được thuật toán hiệu quả để giải chúng) thì không thể thực hiện được việc phá mã toàn bộ đối với RSA. Phá mã một phần phải được ngăn chặn bằng các phương pháp chuyển đổi bản rõ an toàn.

Bài toán RSA là bài toán tính căn bậc e môđun n (với n là hợp số): tìm số m sao cho $c = m^e \bmod n$, trong đó (e, n) chính là khóa công khai và c là bản mã. Hiện nay phương pháp triển vọng nhất giải bài toán này là phân tích n ra thừa số nguyên tố. Khi thực hiện được điều này, kẻ tấn công sẽ tìm ra số mũ bí mật d từ khóa công khai và có thể giải mã theo đúng quy trình của thuật toán. Nếu kẻ tấn công tìm được 2 số nguyên tố p và q sao cho: n = pq thì có thể dễ dàng tìm được giá trị (p-1)(q-1) và qua đó xác định d từ e. Chưa có một phương pháp nào được tìm ra trên máy tính để giải bài toán này trong thời gian đa thức (polynomial-time). Tuy nhiên người ta cũng chưa chứng minh được điều ngược lại (sự không tồn tại của thuật toán). Có thể tham chiếu bảng sau để thấy số thao tác và thời gian thực hiện phân tích số n thành số nguyên tố theo phương pháp General Number Field Sieve (GNFS):

$$O\left(exp\left(\left(\frac{64}{9}\log n\right)^{\frac{1}{3}}(\log\log n)^{\frac{2}{3}}\right)\right)$$

Bảng 2.1 Thử nghiệm độ bảo mật của RSA

Số bit của n	Số thao tác	Thời gian
100	$9{,}6\times 10^{8}$	16 phút
200	$3{,}3\times 10^{12}$	38 ngày
300	$1{,}3\times 10^{15}$	41 năm
400	$1{,}7\times 10^{17}$	5313 năm
500	$1{,}1\times 10^{19}$	$3{,}5\times 10^{5}$ năm
1024	$1{,}3\times 10^{26}$	$4{,}2\times 10^{12}$ năm
2048	$1{,}5\times 10^{35}$	$4{,}9\times 10^{21}$ năm

2.2 Cơ sở hạ tầng của khóa công khai

PKI hạ tầng cơ sở khóa công khai là một cơ chế để cho một bên thứ 3 (thường là nhà cung cấp chứng thực số) cung cấp và xác thực định danh các bên tham gia vào quá trình trao đổi thông tin. Cơ chế này cũng cho phép gán cho mỗi người sử dụng trong hệ thống một cặp khóa công khai/khóa bí mật.

Dựa trên cách sử dụng của khóa công khai và chữ ký điện tử, PKI chính là bộ khung của các chính sách, dịch vụ và phần mềm mã hóa, đáp ứng nhu cầu bảo mật của người sử dụng. PKI cung cấp một cặp khóa, trong đó có một khóa là khóa công khai (Public key), khóa còn lại là khóa bí mật (Private key) mà người sử dụng phải giữ bí mật. Hai khóa này có liên quan mật thiết đến nhau, sao cho một thông điệp được mã hóa bởi một khóa công khai thì chỉ giải mã được bởi một khóa bí mật tương ứng.

2.2.1 Mô hình PKI

2.2.1.1 Mô hình phân tầng của hạ tầng cơ sở khóa công khai

Mô hình này tương ứng với cấu trúc phân cấp với CA gốc và các CA cấp dưới. CA gốc xác nhận các CA cấp dưới, các CA này lại xác nhận các CA cấp thấp hơn. Các CA cấp dưới không cần xác nhận các CA cấp trên.

Trong mô hình này, mỗi thực thể sẽ giữ bản sao khoá công khai của root CA và kiểm tra đường dẫn của chứng thư bắt đầu từ chữ ký của CA gốc.

Mặc dù có những nhược điểm, song mô hình này vẫn thích hợp với yêu cầu của các tổ chức chính phủ vì cấu trúc phân cấp tự nhiên sẵn có.

2.2.1.2 Mô hình CA cầu

Mô hình này hoạt động quanh một CA trung tâm với nhiều CA khác. Trong mô hình này, các CA có thể cộng tác với nhau. Đây là một mô hình kết hợp hai mô hình CA-gốc và Cross-CA. Điều này cung cấp cách đơn giản để quản lý CA gốc, bởi vì nó chỉ yêu cầu một cặp chứng chỉ chéo cho mỗi CA, so sánh với n2 chứng chỉ trong hệ thống hoàn thành.

2.2.2 Chứng chỉ số X.509

X.509 là một chuẩn cho chứng chỉ số quốc tế được sử dụng để chứng thực cho thông tin chủ thể và khóa công khai của các tổ chức, cá nhân. Khuôn dạng chứng chỉ X509 có các thành phần cơ bản như sau:

- Phiên bản (Version): Chỉ ra dạng phiên bản.
- Số hiệu (Serial Number): Số hiệu nhận dạng duy nhất của chứng chỉ này. Nó được CA phát hành gán cho.

- Tên thuật toán ký (Signature): Tên thuật toán ký được CA sử dụng để ký chứng chỉ.
- Người phát hành (Issuer): Tên theo chuẩn X.509 của CA phát hành (được trình bày chi tiết hơn trong mục "Tên trong X.509").
  - Tên tổ chức CA phát hành giấy chứng nhận: Tên phân biệt theo chuẩn X.500 (X.500 Distinguised Name – X.500 DN)
  - Hai CA không được sử dụng cùng một tên phát hành.
- Thời gian hợp lệ (Validity): Ngày/ giờ có hiệu lực và hết hạn của 1 chứng chỉ.
  - Not – before: Thời gian chứng nhận bắt đầu có hiệu lực.
  - Not – after: Thời gian chứng nhận hết hiệu lực.
  - Các giá trị thời gian này được đo theo chuẩn thời gian quốc tế, chính xác đến từng giây.
- Chủ thể (Subject): Tên X.509 của đối tượng nắm giữ khoá riêng (Tương ứng với khoá công khai được chứng thực).
- Thông tin về khoá công khai của chủ thể (Subject Public-key Information): Gồm có khoá công khai của chủ thể cùng với một tên thuật toán sử dụng khóa công khai này.
- Tên duy nhất của người phát hành (Issuer unique identifier): Là một chuỗi bit tuỳ chọn, được sử dụng để chỉ ra tên rõ ràng của CA phát hành, trong trường hợp cùng một tên được gán cho thực thể khác nhau trong cùng thời gian.
- Tên duy nhất của chủ thể (Subject unique indentifier): Là một chuỗi bit tuỳ chọn, được sử dụng để chỉ ra tên rõ ràng của chủ thể, trong trường hợp cùng một tên được gán cho các thực thể khác nhau trong cùng thời gian.
- Extensions: Chứa các thông tin bổ sung cần thiết mà người thao tác CA muốn đặt vào chứng nhận. Được đưa ra trong X.509 phiên bản ba.
- Signature:
  - Chữ ký điện tử được tổ chức CA áp dụng.
  - Tổ chức CA sử dụng khóa bí mật có kiểu quy định trong trường thuật toán chữ ký.
  - Chữ ký bao gồm tất cả các phần khác trong giấy chứng nhận.

➔ CA chứng nhận cho tất cả các thông tin khác trong giấy chứng nhận chứ không chỉ cho tên chủ thể và khóa công cộng.

2.3 Các hàm băm (Hash Functions)

Hàm băm tiếp nhận các đầu vào với kích thước bất kỳ, và đầu ra là một khối dữ liệu có kích thước cố định [5]. Từ văn bản M, ta có thể dễ dàng tính ra bản băm của M là H(M), nhưng từ H(M) không thể tìm ra M. Và một tính chất quan trọng nhất của hàm băm là với những văn bản khác biệt nhau dù là rất nhỏ, thì sau khi qua hàm băm kết quả nhận được cũng phải khác nhau, ta có thể gọi là độ nhạy cảm của hàm băm với sự thay đổi của văn bản.

2.3.1 Cơ sở hàm băm

Định nghĩa: Một hàm băm H sẽ lấy ở đầu vào một thông tin X có kích thước biến thiên và sinh kết quả có độ dài cố định, được gọi là cốt của thông điệp.

2.3.2 Một số đặc tính của băm

Tính chất 1:

Một hàm băm h có tính phi đụng độ cao khi với một bức điện x cho trước, không thể tìm ra một bức điện x' # x mà h(x') = h (x). [10]

Tính chất 2:

Một hàm băm h có tính đụng độ cao khi không thể tìm ra những bức điện x và x' sao cho x # x' và h(x') = h(x) . [10]

Tính chất 3:

Một hàm băm h có tính một chiều khi với cốt của một bức điện z cho trước không thể tìm được bức điện x sao cho h(x) == z . [10]

Một cách tổng quát, giả sử một hàm băm có n giá trị băm khác nhau, nếu chúng ta có k giá trị băm từ k thông tin khác nhau được chọn ngẫu nhiên, thì xác suất để không xảy ra đụng độ là:

$$\left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right)\dots\left(1-\frac{k-1}{n}\right)=\prod_{i=1}^{k-1}\left(1-\frac{i}{n}\right)$$

$$Với\ \frac{i}{n}\ll 1, thì\ \prod_{i=1}^{k-1}\left(1-\frac{i}{n}\right)\approx\prod_{i=1}^{k-1}e^{\frac{-i}{n}}=e^{\frac{-k(k-1)}{2n}}$$

Do đó, xác suất để xảy ra đụng độ ít nhất là 1 - $e^{\frac{-k(k-1)}{2n}}$. Giả sử gọi xác suất trên là epsilo, ta có:

$$1-e^{\frac{-k(k-1)}{2n}}\approx\varepsilon(*)$$

Suy ra: $k^2 - k \approx 2n log \frac{1}{1-\varepsilon}$. Suy ra $k \approx \sqrt{2nlog \frac{1}{1-\varepsilon}}$

Theo công thức này khi giá trị e gần với 1 thì $log \frac{1}{1-\varepsilon}$ vẫn khá nhỏ nên k là tỉ lệ với $\sqrt{n}$. Với ε = 0.5 ta có $k = 1.1774\sqrt{n}$.

2.4 Chữ ký số

2.4.1 Quá trình tạo và kiểm tra chữ ký số

Quá trình sử dụng chữ ký số được thực hiện theo 2 giai đoạn: Tạo chữ ký và kiểm tra chữ ký số. Có thể dùng khoá công khai hoặc khoá bí mật để thực hiện các khâu trên. Hai quá trình tạo và kiểm tra sẽ được trình bày cụ thể tiếp sau đây.

Các bước tạo chữ ký:

- Dùng giải thuật băm để tính message digest của thông điệp cần truyền đi. Kết quả ta được một message digest.
- Sử dụng khóa bí mật của người gửi để mã hóa message digest thu được ở bước 1. Thông thường ở bước này ta dùng giải thuật RSA. Kết quả thu được gọi là digital signature của thông điệp ban đầu. Công việc này gọi là "ký" vào thông điệp. Sau khi đã ký vào thông điệp, mọi sự thay đổi trên thông điệp sẽ bị phát hiện trong giai đoạn kiểm tra. Ngoài ra, việc ký này đảm bảo người nhận tin tưởng thông điệp này xuất phát từ người gửi chứ không phải là ai khác.
- Gộp digital signature vào thông điệp ban đầu và gửi đến người nhận.

Các bước kiểm tra:

- Tách message ban đầu và chữ ký số.
- Dùng khóa công khai của người gửi (khóa này được thông báo đến mọi người) để giải mã chữ ký số của thông điệp.
- Dùng giải thuật (MD5 hoặc SHA) băm thông điệp ban đầu.
- So sánh 2 chuỗi băm kết quả thu được ở 2 bước trên. Nếu trùng nhau, ta kết luận thông điệp này không bị thay đổi trong quá trình truyền và thông điệp này là của người gửi.

2.4.2 Các loại hệ chữ ký

2.4.2.1 Hệ chữ ký RSA

Dựa vào ưu điểm của hệ mã RSA, nếu thiết lập được sơ đồ chữ ký dựa trên bài toán phân tích ra thừa số nguyên tố thì độ an toàn của chữ ký sẽ rất cao. Việc thiết lập sơ đồ xác

thực chữ ký RSA rất đơn giản, ta chỉ cần đảo ngược hàm mã hóa và giải mã. Sau đây là sơ dồ chữ ký RSA:

*Cho n = p*q trong đó p,q là các số nguyên tố. Đặt p = A = Z_n và định nghĩa:*

*K = {(n,p,q,a,b) : n = p*q, p là các số nguyên tố, ab ≡ 1 (mod Ø (n))}.*

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

Với K = (n, p, q, a, b) ta xác định:

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

Và

$Ver_k(x,y) = TRUE \Leftrightarrow x \equiv y^b \pmod n$ với x, y € Z_n . [10]

2.4.2.2 Hệ chữ ký DSA

Hệ chữ ký DSA dựa vào thuật toán chữ ký số (DSA) - là chuẩn của chính phủ liên bang hoa kỳ hoặc FIPS đề ra cho chữ ký số [7].

Sinh khóa

Việc tạo khóa gồm hai bước. Bước thứ nhất là lựa chọn các tham số cho thuật toán được chia sẻ giữa các người sử dụng khác nhau trong cùng hệ thống:

- Chọn một hàm băm mã hóa H.
- Chọn kích thước khóa L. Đây là thước đo chính quyết định sức mạnh mã hóa của khóa. DSS chuẩn ràng buộc L là bội số của 64 và 512≤L≤1024. Sau đó, FIPS 186-2 xác định L luôn là 1024. Không lâu sau, NIST 800-57 đề nghị độ dài khóa là 2048 (hoặc 3072) để thời gian an toàn đến năm 2010 (hoặc 2030), sử dụng tương ứng với các giá trị băm và q dài hơn.
- Chọn một số nguyên tố q cùng số bit với đầu ra của H.
- Chọn một số nguyên tố p độ dài L bit sao cho p–1 là bội của q. Tức là p=qz–1 với số nguyên z nào đó.
- Chọn g = h(p-1)/q mod p với h bất kỳ (1<h<p–1), và chọn lại nếu kết quả là 1. Hầu hết cách chọn h đều nhận được g có thể sử dụng, thông thường chọn h=2.

Các tham số thuật toán (p,q,g) có thể chia sẻ giữa những người khác nhau trong hệ thống. Bước thứ hai tính các khóa bí mật và khóa công khai của từng người :

- Chọn x ngẫu nhiên sao cho 0<x<q.
- Tính $y=g^x \bmod p$

- Khóa công khai là (p,q,g,y), khóa bí mật là x

Ký và kiểm tra chữ ký

Để ký một thông điệp m, người ký thực hiện các bước sau:

- Phát sinh một số ngẫu nhiêu k (0<k<q) cho mỗi thông điệp.
- Tính r=(g^k mod p) mod q.
- Tính s = k^{-1}(h(m) + xr) mod q.
- Tính toán lại chữ ký trong trường hợp không chắc chắn r=0 hoặc s=0.
- Chữ ký là (r,s)

Để kiểm tra chữ ký, người nhận thực hiện các bước sau:

- Loại bỏ chữ ký nếu 0<r<q hoặc 0<s<q không thỏa mãn.
- Tính w=s^{-1} mod q.
- Tính u_1=($\mathcal{H}$(m)×w) mod q.
- Tính u_2=(r×w) mod q.
- Tính v=(($g^{u_1} \times y^{u_2}$)mod p) mod q
- Chữ ký có hiệu lực nếu v=r.

2.4.2.3 Đánh giá các thuật toán tạo chữ ký RSA và DSA

Để so sánh tốc độ của hai thuật toán chữ ký số RSA và DSA. Thử nghiệm dưới đây đã được tiến hành và ghi nhận : Độ dài khóa được thử nghiệm cho cả RSA và DSA là 576, 640, 704, 768, 832, 896, 960, 1024, 2048, 3072 (bit). Ứng với mỗi độ dài khóa, lần lượt cho cả RSA và DSA phát sinh khóa, ký văn bản ngẫu nhiên (kích thước 2 MB) và kiểm tra chữ ký tạo được. Hàm băm mã hóa SHA-1 được chọn để sử dụng cho cả RSA và DSA. Thử nghiệm được lặp lại 50.000 lần. Kết quả nhận được như sau:

Bảng 2.3 So sánh thời gian tạo khóa, tạo chữ ký và xác nhận chữ ký của RSA với DSA

Kích thước (bit)	**Tạo khóa (giây)**	**Tạo chữ ký (giây)**	**Xác nhận chữ ký (giây)**

	RSA	DSA	DSA/ RSA	RSA	DSA	RSA/ DSA	RSA	DSA	RSA/ DSA
512	0,0408	0,5676	13,93	0,0351	0,0011	32,60	0,0320	0,0017	19,32
576	0,0568	0,8030	14,14	0,0361	0,0013	27,24	0,0321	0,0022	14,60
640	0,0757	1,2464	16,47	0,0371	0,0015	24,53	0,0319	0,0025	12,57
704	0,0994	1,7948	18,06	0,0387	0,0019	20,25	0,0320	0,0031	10,16
768	0,1278	2,3668	18,52	0,0408	0,0016	25,29	0,0321	0,0040	7,94
832	0,1609	3,0526	18,97	0,0428	0,0021	20,31	0,0322	0,0044	7,34
896	0,2026	4,2369	20,92	0,0454	0,0027	16,58	0,0321	0,0050	6,36
960	0,2446	5,4622	22,33	0,0480	0,0026	18,45	0,0321	0,0061	5,29
1024	0,2734	7,1210	26,05	0,0515	0,0035	14,86	0,0318	0,0068	4,69
2048	2,4876	103,1124	41,45	0,1749	0,0124	14,16	0,0325	0,0240	1,35
3072	11,1882	508,2395	45,43	0,5056	0,0278	18,19	0,0341	0,0539	0,63

Thử nghiệm trên môi trường Windows XP, bộ xử lý Pentium 4 3.00 GHz, bộ nhớ 512 MB.

Kết quả thử nghiệm cũng cho thấy tốc độ xác nhận chữ ký của RSA không thay đổi đáng kể khi kích thước khóa tăng do số mũ công khai e được sử dụng luôn là một số đủ lớn (giá trị phổ biến hiện nay là 65537) và tốc độ thực hiện phép lũy thừa modulo (phép toán chính trong quy trình xác nhận chữ ký) sẽ tăng không nhiều. Ngược lại, tốc độ xác nhận chữ ký của DSA mặc dù thấp hơn RSA nhưng sẽ ngày càng tăng khi kích thước khóa tăng lên. Nguyên nhân là do quy trình xác nhận chữ ký của DSA gồm rất nhiều phép tính tốn chi phí cao (phép lũy thừa modulo và phép nhân) nên khi kích thước khóa tăng dần thì điều này sẽ trở thành gánh nặng. Mặc khác, nếu kích thước L được chọn lớn hơn thì tốc độ xác nhận chữ ký sẽ chậm hơn nữa.

2.4.2.4 Hệ chữ ký ElGammal

Hệ chữ ký ElGammal được thiết kế riêng biệt cho mục đích chữ ký, trái ngược với RSA thường được sử dụng cho cả mã hóa công khai và chữ ký. Hệ chữ ký ElGammal là không xác định, nghĩa là có rất nhiều giá trị chữ ký cho cùng một thông điệp cho trước [7]. Thuật toán xác minh phải có khả năng nhận bất kỳ giá trị chữ ký nào như là việc xác thực. Sơ đồ chữ ký ElGammal được miêu tả như sau:

Cho p là một số nguyên tố như bài toàn logarit rời rạc trong z_p, $\alpha \in Z_p^*$ *là một phần tử nguyên tử và* $p = Z_p^*$, $a = (Z_p^*)^* Z_{p-1}$ *và định nghĩa:*

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

Trong đó giá trị p, α, β *là công khai còn a là bí mật*

Với $K = \{(p, \alpha, a, \beta)$ *và chọn một số ngẫu nhiên* $k \in Z^*_{p-1}$, *định nghĩa:*

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

Trong đó $\gamma = \alpha^k \bmod p$

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

Với $x, \gamma \in Z^*_p$ *và* $\delta \in Z_{p-1}$, *định nghĩa:*

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

Nếu chữ ký là đúng thì việc xác nhận thành công khi:

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

Trong đó $a\gamma + k\delta \equiv x \pmod{p-1}$. [10]

B sẽ tính toán chữ ký bằng việc sử dụng cả giá trị bí mật a (một phần của khóa) và số bí mật ngẫu nhiên k (giá trị để ký bức điện). Việc xác minh có thể thực hiện được chỉ với các thông tin được công khai.

2.4.2.5 Chuẩn chữ ký số

Chuẩn chữ ký điện tử (DSS) được sửa đổi từ hệ chữ ký ElGammal. DSS sử dụng một khóa công khai để kiểm tra tính toàn vẹn của dữ liệu nhận được và đồng nhất với dữ liệu của người gửi. DSS cũng có thể sử dụng bởi người thứ ba để xác định tính xác thực của chữ ký và dữ liệu của nó.

DSS đã sửa đổi hệ chữ ký ElGamma cho phù hợp theo cách này một khác khéo léo, để mỗi 160 bit bức điện được ký sử dụng một chữ ký 320 bit, nhưng việc tính toán được thực hiện với 512 bit modulo p. Cách này được thực hiện nhờ việc chia nhỏ Z_p^* thành các trườn có kích thước 2^{160}. Việc thay đổi này sẽ làm thay đổi giá trị δ:

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

Điều này cũng làm cho giá trị kiểm tra cũng thay đổi:

$A^x\beta^\gamma \equiv \gamma^\delta \pmod p$

Nếu UCLN $(x + \alpha\gamma, p -1) = 1$ thì sẽ tồn tại $\delta^{-1} \bmod (p-1)$, do đó ta có

$A^{x\delta\hat{}(-1)}\beta^{\gamma\delta\hat{}(-1)} \equiv \gamma \pmod p$

Đây chính là sự đổi mới của DSS. Chúng ta cho q là một số nguyên tố 160 bit sao cho q | (p – 1) và α là một số thứ q của 1 mod p, thì β và γ cũng là số thứ q của 1 mod p.

Do đó, α, β và γ có thể được tối giản trong modulo p mà không ảnh hưởng gì đến việc xác minh chữ ký. Sơ đồ thuật toán:

Cho p là một số nguyên tố 512 – bit trong trường logarit rời rạc Z_p; q là một số nguyên tố 160 – bit và q chia hết (p-1). Cho α thuộc Z_p^*; $p = Z_p^*$, $A = Z_q * Z_q$ và định nghĩa:

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

Trong đó giá trị p, q, α, β là công khai, còn a là bí mật.

Với K = (p, α, a, β) và chọn một số ngẫu nhiên k (1 <= k <= q-1) và định nghĩa:

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

Trong đó $\gamma = (\alpha^k \bmod p) \bmod q$

$\delta = (x + a^*\gamma)k^{-1} \bmod q$

Với x thuộc Z^*_p *và γ, δ thuộc* Z_q *, việc xác minh được thực hiện bằng cách tính:*

$E_1 = x\delta^{-1} \bmod q$

$E_2 = \gamma\delta^{-1} \bmod q$

$Ver(x, \gamma, \delta) = TRUE \Leftrightarrow (\alpha^{E1}\beta^{E2} \bmod p) \bmod q = \gamma$ [10]

Chú ý rằng, với DSS thì $\delta \neq 0 \pmod q$ vì giá trị $\delta^{-1} \bmod q$ cần cho việc xác minh chữ ký . Khi B tình một giá trị $\delta \equiv 0 \pmod q$ trong thuật toán ký, anh ta nên bỏ nó đi và chọn một số ngẫu nhiên k mới.

2.5 PDF và chữ kí số

PDF (viết tắt từ tên tiếng Anh Portable Document Format, Định dạng Tài liệu Di động) là một định dạng tập tin văn bản khá phổ biến của hãng Adobe System. Tương tự như định dạng Word (.doc), PDF hỗ trợ văn bản thô (text) cùng với phông chữ, hình ảnh đồ họa, âm thanh và nhiều hiệu ứng khác. Tuy nhiên, việc hiển thị văn bản PDF không phụ thuộc vào môi trường làm việc của người sử dụng (cấu hình máy, phần mềm và hệ điều hành).

Không như văn bản Word, một văn bản PDF, trong hầu hết các trường hợp sẽ được hiển thị giống nhau trên những môi trường làm việc khác nhau. Chính vì ưu điểm này, định dạng PDF đã trở nên phổ biến cho việc phát hành sách, báo hay các tài liệu khác qua mạng Internet.

Để đọc được tập tin PDF trên máy vi tính, người dùng phải có một phần mềm hỗ trợ định dạng này. Phần mềm phổ biến hiện nay là Adobe Reader hay Foxit Reader. [6]

2.6 Tổng kết chương

Chương này đã cung cấp một cái nhìn toàn vẹn và tổng quát nhất về thuật toán RSA, hàm băm, chữ ký số, cơ sở hạ tầng tạo khóa và định dạng PDF. Chương này cũng cung cấp một cái nhìn tổng quát nhất về các hệ chữ ký số, đánh giá giữa các hệ chữ ký số. Những thuật toán này chính là chìa khóa để tạo ra một chữ ký số hợp lệ và tin cậy dùng cho ký và xác thực văn bản.

CHƯƠNG 3: ÁP DỤNG CHỮ KÝ SỐ CHO BÀI TOÁN IN HÓA ĐƠN VT01 TẠI VIỄN THÔNG HÀ NỘI

Chương này sẽ xây dựng ứng dụng cụ thể, cài đặt một ứng dụng chữ ký số cho tài liệu PDF hóa đơn VT01 tại Viễn thông Hà Nội.

3.1 Yêu cầu chung của bài toán

Với một lượng khách hàng lớn, hàng tháng dữ liệu in hóa đơn VT01 của VNPT Hà Nội lên tới gần một triệu khách hàng, những rủi ro sai sót trong khâu in cũng như khâu cung cấp dữ liệu dùng cho in ấn là khó tránh khỏi. Yêu cầu cấp thiết là dựa trên file text vẫn dùng để in hóa đơn VT01 hàng tháng (file text có cấu trúc), thông qua chương trình sẽ chuyển đổi tạo thành file VT01 của khách hàng theo định dạng PDF, giúp việc lưu giữ, in ấn và tra cứu dễ dàng hơn.

- Tạo ra file hóa đơn VT01 của từng mã khách hàng, thuận tiện cho việc lưu trữ và tìm kiếm.
- Lấy số hóa đơn cho từng khách hàng thông qua Web Service của Trung Tâm Tin Học
- Giảm thiểu việc in ấn, tránh được các sai sót không đáng có.
- Sử dụng USB TOKEN để kí
- File PDF sau khi tạo xong được lưu giữ và là đầu vào cho bài toán Thanh Toán của VNPH Hà Nội.

Bài toán là tạo ra file hóa đơn VT01 theo định dạng PDF dựa trên file text VT01 có cấu trúc. Chức năng cơ bản của ứng dụng là: Người sử dụng ký tài liệu dùng chứng chỉ của họ hoặc chứng chỉ được cho phép kí (USB TOKEN do VNPT Hà Nội cung cấp) để tạo ra file hóa đơn PDF dựa trên file text có cấu trúc.

3.2 Cài đặt - Thử nghiệm

3.2.1 Mô tả chi tiết File đầu vào

Tên file: VTxxxxmm.TXT

Bảng 3.1 Mô tả độ rộng các trường trong file text hóa đơn VT01

STT	Trường	Độ rộng (Ký tự)	Mô tả
1	Mã cơ quan	8	Theo yêu cầu của DVKH từ tháng 042009.
	Mã đường thư	6	

	Mã đường thư phụ	2	
	Số hoá đơn	4	
2	Tên đơn vị nhận tiền	60	
3	MST ĐV nhận tiền	120	
4	Mã Vạch	14	xxxxxxxmmyyVT (8 ký tự đầu là mã KH)
5	Tên KH1	30	Tên khách hàng (phần thứ nhất)
6	Tên KH2	90	(Tên khách hàng đầy đủ = Tên KH1 + Tên KH2)
7	Mã số thuế 2	15	Mã số thuế của khách hàng
8	Địa chỉ 1	70	Địa chỉ khách hàng (phần thứ nhất)
9	Địa chỉ 2	50	Địa chỉ khách hàng đầy đủ = Địa chỉ 1 + Địa chỉ 2
10	Số máy	25	Số máy đại diện của khách hàng
11	Mã bưu điện	30	Là một chuỗi gồm mã khách hàng, mã đường thư, đường thư phụ và thứ tự hóa đơn của khách hàng.
12	Tháng cước	20	Cước tháng MM/YYYY
13	Các khoản chịu thuế	25	Chuỗi text khoản mục
14	Khoản tiền chịu thuế	20	Khoản tiền chịu thuế
15	Các khoản không chịu thuế và thu khác	45	Chuỗi text khoản mục
16	Khoản tiền không chịu thuế và thu khác	20	Khoản tiền không chịu thuế và thu khác
17	Khuyến mại (không thu tiền)	30	Chuỗi text khoản mục
18	Tiền khuyến mại	20	Tiền khuyến mại
19	Các khoản truy thu, giảm trừ	30	Chuỗi text khoản mục
20	Tiền truy thu, giảm trừ	20	Tiền truy thu, giảm trừ
21	Cộng tiền dịch vụ	20	Cộng tiền dịch vụ (total)
22	Tiền thuế GTGT	20	Tiền thuế GTGT
23	Tổng tiền	20	Tổng tiền
24	Tiền chữ 1	70	Số tiền bằng chữ phần thứ nhất
25	Tiền chữ 2	90	Số tiền bằng chữ phần thứ hai
26	Số thứ tự để đếm	7	Số thứ tự để đếm số khách hàng thực tế in ra

3.2.2 Một số tiến trình chính của chương trình

Chương trình được xây dựng trên môi trường Visual C# trong bộ ứng dụng Microsoft Visual Studio 2010.

- Đọc thông tin của USB TOKEN, lưu vào biến nhớ.
- Đọc file Text có cấu trúc (mỗi dòng là một mã thanh toán), chuyển thành đối tượng thanh toán, đưa đối tượng vào hàng đợi.
- Gọi các tiến trình (luồng) in đồng thời.
- Các tiến trình này vào hàng đợi lấy dữ liệu thanh toán rồi mang ra tạo VT01.
- Lấy mã số hóa đơn từ Trung Tâm Tin Học thông qua Web Service.
- Thực hiện ký hóa đơn.
- Ghi dữ liệu hóa đơn ra đĩa cứng.

3.3 Một số yêu cầu về phần cứng và phần mềm

Đề xuất:

CPU: Intel® Core ™ 2 Duo E7400

RAM: 2GB

HDD: 500 GB

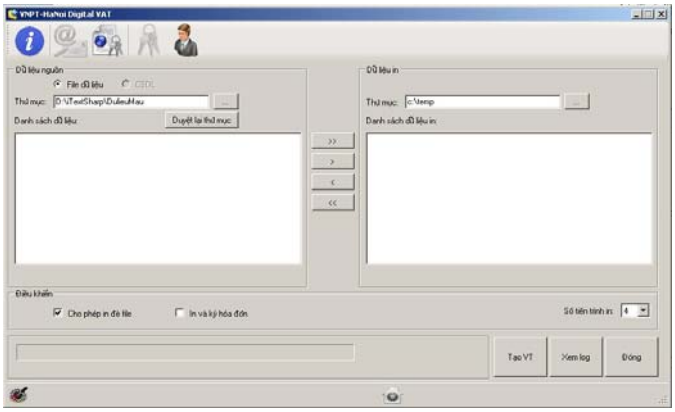
OS: Microsoft Windows XP

Mouse, Key Board, Moniter

Máy cài đặt Microsoft .NET Framework 4 Maintenance, dung lượng đĩa cứng còn trống ít nhất 20MB.

3.4 Một số hình ảnh của chương trình

- Màn hình chính của chương trình.



Hình 3.1 Màn hình chính của chương trình

- Duyệt thư mục chứa file text và đường dẫn file đích

Hình 3.2 Duyệt thư mục file

- Chọn người ký
- Nhập PIN TOKEN
- Chương trình chạy
- File kết quả nhận được

Hình 3.6 Kết quả

3.5 Tổng kết chương

Chương này trình bày bài toán in hóa đơn VT01 tại Viễn thông Hà Nội. Với đầu vào là dữ liệu file text VT01 có cấu trúc, người dùng thông qua chương trình sẽ tạo ra những file đầu ra là hóa đơn PDF đã được ký bằng USB TOKEN do VNPT Hà Nội cung cấp.

KẾT LUẬN

Kết quả đạt được:

Luận văn tập trung xem xét một số vấn đề về kỹ thuật tạo chữ ký số, với một số kỹ thuật đi kèm như RSA, các hàm băm, … Một số kết quả luận văn đã đạt được:

- Những khái niệm, đặc điểm cơ bản của một hệ thống PKI. Ứng dụng của hạ tầng khóa công khai trong thương mại điện tử. Tư tưởng của thuật toán cấp phát khóa, sinh và kiểm tra chữ ký số (RSA, DSA,…), về ưu nhược điểm của từng thuật toán, định nghĩa và đặc tính của hàm băm.
- Xây dựng một ứng dụng chuyển đổi file dữ liệu text có cấu trúc hiện đang dùng để in trên giấy hóa đơn VT01 của Viễn thông Hà Nội sang file PDF. Nhằm phục vụ cho nhu cầu lưu trữ, tra cứu, tiết giảm chi phí in ấn. Điều này đã đẩy nhanh thời gian đại lý thu thuê phải đi thu hồi và phát lại hóa đơn VT01 trong trường hợp nếu có xảy ra sai sót trong khâu in ấn và rất thiết thực trong việc bảo vệ môi trường.

Hướng phát triển:

- Chương trình hiện nay mới chỉ chạy trên dữ liệu text hóa đơn có sẵn của Viễn thông Hà Nội. Hướng phát triển tiếp theo là cải tiến chương trình để có thể chạy trực tiếp luôn trên cơ sở dữ liệu Oracle mà không phải qua khâu trung gian là file text có cấu trúc.
- Chương trình mới chỉ có phần tạo, kí hóa đơn mà chưa có phần xác nhận từ phía người nhận và bên cơ quan chứng thực (CA) để kiểm tra chứng chỉ số có hợp lệ hay không.