UBND TỈNH QUẢNG NAM
TRƯỜNG ĐẠI HỌC QUẢNG NAM
KHOA: CÔNG NGHỆ THÔNG TIN

NGUYỄN THỊ HOÀNG VY

TÊN ĐỀ TÀI

NGHIÊN CỨU MỘT SỐ THUẬT TOÁN MÃ HÓA DỮ LIỆU VÀ ỨNG DỤNG MÃ HÓA EMAIL

KHOÁ LUẬN TỐT NGHIỆP ĐẠI HỌC

Quảng Nam, tháng 4 năm 2017

UBND TỈNH QUẢNG NAM
TRƯỜNG ĐẠI HỌC QUẢNG NAM
KHOA: CÔNG NGHỆ THÔNG TIN

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC

Tên đề tài:

NGHIÊN CỨU MỘT SỐ THUẬT TOÁN MÃ HÓA DỮ LIỆU VÀ ỨNG DỤNG MÃ HÓA EMAIL

Sinh viên thực hiện

NGUYỄN THỊ HOÀNG VY
MSSV: 2113021051

CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN

KHÓA 2013 – 2017

Cán bộ hướng dẫn

T.S VŨ ĐỨC QUẢNG
MSCB: ……

Quảng Nam, tháng 4 năm 2017

LỜI CẢM ƠN

Được sự phân công và tạo điều kiện của khoa công nghệ thông tin - trường đại học Quảng Nam và được sự đồng ý của thầy giáo hướng dẫn T.S Vũ Đức Quảng em đã thực hiện đề tài: "*Nghiên cứu một số thuật toán mã hóa dữ liệu và ứng dụng mã hóa Email* "cho bài khóa luận của mình.

Đầu tiên, em xin cảm ơn tất cả các thầy cô giáo trong trường đại học Quảng Nam nói chung và các thầy cô giáo khoa công nghệ thông tin nói riêng đã tận tình hướng dẫn, truyền đạt các kiến thức, kinh nghiệm vô cùng quí báu trong suốt thời gian học tập, nghiên cứu, rèn luyện tại trường đại học Quảng Nam.

Đặt biệt, em xin chân thành cảm ơn thầy giáo hướng dẫn TS. Vũ Đức Quảng, thầy đã tận tình giúp đỡ, chỉ bảo trực tiếp chu đáo trong suốt quá trình hoàn thành bài khóa luận.

Sau cùng em xin cảm ơn gia đình, bạn bè đã động viên, đóng góp ý kiến trong quá trình tìm hiểu, nghiên cứu và hoàn thành bài khóa luận.

Mặc dù đã có nhiều cố gắng, áp dụng tất cả nội dung đã học và tích cực tham khảo tài liệu em đã hoàn thành bài khóa luận. Song do hạn chế về khả năng, thời gian bài có thể xảy ra những thiếu sót mà bản thân không thể nhìn thấy. Vì vậy, em rất mong sự đóng góp ý kiến của thầy cô giáo, bạn bè để bài khóa luận được hoàn chỉnh hơn.

Em xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Từ viết đầy đủ
CSDL	Cơ sở dữ liệu
UCLN	Ước chung lớn nhất
CNTT	Công nghệ thông tin
WWW	World Wide Web
OSI	Open Systems Interconnection Reference Model
TCP	Transmission Control Protocol
SSL	Secure Sockets Layer
TLS	Transport Layer Security
PGP	Pretty Good Privacy

DANH MỤC HÌNH VẼ

PHẦN 1. MỞ ĐẦU

1.1. Lý do chọn đề tài

Từ xưa đến nay, thông tin luôn là yếu tố quan trọng trong các hoạt động của con người. Trước đây khi công nghệ máy tính chưa phát triển, khi nói đến vấn đề bảo mật thông tin, chúng ta thường nghĩ các biện pháp nhằm đảm bảo thông tin được trao đổi hay cất giữ một cách an toàn và bí mật. Chẳng hạn các biện pháp như: đóng dấu hay niêm phong một bức thư để biết được lá thư được chuyển nguyên vẹn đến người nhận hay không? Hiện nay, với sự bùng nổ của Internet, nó đã trở thành phương tiện trao đổi trên toàn cầu. Có thể dễ thấy rằng tất cả các hoạt động, công việc của chúng ta đều liên quan đến Internet: đọc báo, giải trí, học tập, mua sắm...và khi nó trở thành một phương tiện điều hành của các hệ thống thì việc bảo mật thông tin được đưa lên hàng đầu. Song song với sự phát triển đó là sự xâm nhập thông tin cá nhân cũng ngày càng được tăng lên và đa dạng hóa. Vấn đề đặt ra là bảo mật thông tin trong quá trình truyền tải thông tin nhất là thông tin chính trị, kinh tế, quân sự.

Trên thế giới hiện nay có khá nhiều giải pháp mã hóa thông tin công nghệ mới dựa trên các thuật toán có độ phức tạp cao, hàng loạt các giao thức các cơ chế đã được tạo ra để đáp ứng nhu cầu an toàn bảo mật thông tin. Thường thì mục tiêu của an toàn bảo mật thông tin không thể đạt đươc nếu đơn thuần chỉ là dựa vào các thuật toán toán học và các giao thức mà muốn đạt được điều này cần có các kỹ thuật mã hóa. Để giải quyết vấn đề đó hệ mật mã đã ra đời. Từ các hệ mật mã cổ điển như hệ dịch mã vòng... đến các mã hóa hiện đại như DES, các hệ mã khóa công khai như RSA. Kèm với sự phát triển các hệ mật mã là các phương pháp phá khóa của các hệ mật mã. Việc an toàn bảo mật thông tin và xâm nhập thông tin vẫn luôn luôn diễn ra và đấu tranh hằng ngày.

Ngày nay, Email (thư điện tử) không còn xa lạ với bất kì ai. Email là một công cụ mà hầu như ai cũng cần phải có, nó như là một địa chỉ thứ hai của bạn. Do đó, nếu ta đánh mất địa chỉ Email thì xem như đã tự đưa thông tin bản thân cho người khác. Và nếu đó là Email để làm công việc quan trọng như kinh doanh chẳng hạn thì xem như bí mật kinh doanh, thông tin khách hàng của bạn sẽ bị rò rĩ, thậm chí thông tin này sẽ rao bán nghĩa là mất Email thì công việc của bạn sẽ bị gián đoạn.

Từ những vấn đề đã được đề cập ở trên, để hiểu rõ hơn về các kỹ thuật mã hoá thông tin em đã chọn "**Nghiên cứu một số thuật toán mã hóa dữ liệu và ứng dụng mã hóa Email"** làm đề tài nghiên cứu khóa luận tốt nghiệp của mình.

1.2. Mục tiêu của đề tài

- Nghiên cứu về lý thuyết về mã hóa dữ liệu, các phương pháp mã hóa cổ điển, các phương pháp mã hóa hiện đại thường dùng hiện nay.
- Tìm hiểu các thuật toán mã hóa để xây dựng chương trình minh họa và các ví dụ minh họa.
- Giới thiệu một phần mềm để bảo mật Email cho người sử dụng Email.

1.3. Đối tượng và phạm vi nghiên cứu

1.3.1. Đối tượng nghiên cứu

- Các thuật toán mã hóa dữ liệu cổ điển, hiện đại.
- Hàm băm MD5, SHA-1.
- Email và phần mềm mã hóa Email PGP trên Thunderbird.

1.3.2. Phạm vi nghiên cứu

- Nghiên cứu các thuật toán mã hóa dữ liệu.
- Tìm hiểu và cài đặt một số chương trình minh họa bằng C# cho một số thuật toán mã hóa dữ liệu.
- Tìm hiểu và cài đặt một số phần mềm để mã hóa Email.

1.4. Phương pháp nghiên cứu

- Tìm hiểu, thu thập tài liệu thông tin trong giáo trình, Internet.
- Đưa ra và cài đặt thành công phần mềm mã hóa Email.

1.5. Lịch sử nghiên cứu

An toàn bảo mật thông tin là một trong những vấn đề mang tính quan trọng cao nên được nhiều nhà khoa học nghiên cứu và đã được ứng dụng trong nhiều lĩnh vực như chính trị, kinh tế, thương mại. Có thể kể ra những vấn đề được nghiên cứu gần đây như nghiên cứu các phương pháp thám mã, một số luật mã thuộc hệ mã cổ điển trên văn bản tiếng việt của Th.S Ngô Phương Nam, nghiên cứu các mô hình bảo mật thông tin và ứng dụng vào hệ thống thông tin của bộ giao thông vận tải... Kế thừa từ những đề tài đã được nghiên cứu em sẽ tìm hiểu kĩ, đi sâu, và làm rõ vấn đề hoàn thành bài khóa luận tốt nghiệp của mình.

1.6. Đóng góp của đề tài

- Làm rõ các vấn đề liên quan đến mã hóa dữ liệu.
- Phân tích ưu, nhược điểm của các thuật toán mã hóa.
- Giới thiệu và hướng dẫn sử dụng một phần mềm để bảo mật Email.

1.7. Cấu trúc đề tài

Đề tài gồm 4 chương:

- Chương 1: Tổng quan về mã hóa dữ liệu
- Chương 2: Các phương pháp mã hóa cổ điển
- Chương 3: Các phương pháp mã hóa hiện đại
- Chương 4: Ứng dụng mã hóa Email

PHẦN 2. NỘI DUNG NGHIÊN CỨU

CHƯƠNG 1: TỔNG QUAN VỀ MÃ HÓA DỮ LIỆU

1.1. Khái niệm về mã hóa dữ liệu

1.1.1. Giới thiệu chung về mã hóa dữ liệu

1.1.1.1. Khái niệm mã hóa dữ liệu

Mã hóa là phương pháp để biến đổi thông tin (phim, hình ảnh, văn bản...) từ định dạng bình thường sang dạng thông tin không thể hiểu được nếu không có phương tiện giải mã.

Mã hóa dữ liệu ngăn chặn được các việc sau:

- Nghe trộm và xem lén dữ liệu.
- Giả mạo thông tin.
- Chỉnh sửa và đánh cắp lén dữ liệu.
- Sự gián đoạn các dịch vụ mạng.

Mã hóa không thể ngăn chặn thông tin bị đánh cắp nhưng thông tin được lấy về không dùng được, không dùng được vì dữ liệu ban đầu đã biến sang dạng khác.

1.1.1.2. Một số thuật ngữ mã hóa dữ liệu

- Văn bản gốc: là loại văn bản bình thường còn nguyên chưa được mã hóa, ai cũng có thể đọc được và hiểu được.
- Văn bản mã hóa: Là loại văn bản gốc đã được mã hóa, chỉ có người nhận và người gửi mới đọc và hiểu được.
- Mã hóa: Là quá trình biến đổi thông tin ban đầu (có thể hiểu được) thành dạng không thể hiểu được nhằm mục đích giữ bí mật thông tin nào đó.
- Giải mã: Là quá trình ngược lại với mã hóa, nhằm khôi phục từ dữ liệu đã mã hóa về dữ liệu ban đầu.

- Thuật toán mã hóa: Là tập hợp các giao thức hay cách để chuyển đổi một văn bản gốc sang văn bản mã hóa. Để đảm bảo tính an toàn của thuật toán thì thuật toán giải mã phải được bảo mật chỉ cho người nhận biết.
- Khóa bí mật: Là khóa duy nhất được người dùng để lấy lại thông tin ban đầu đã được mã hóa và khóa này được giữ bí mật.
- Khóa công khai: Là loại khóa tất cả mọi người ai cũng biết, không cần phải bảo mật, với khóa này người dùng có thể trao đổi thông tin bí mật với nhau
- Sản phẩm mật mã: Bao gồm các hệ thống thiết bị, mạch tích hợp và các phần mềm mã hóa chuyên dụng có tích hợp các thuật toán mã hóa được thiết kế để bảo vệ thông tin giao dịch điện tử và lưu trữ ở dạng số.

1.1.2. Tầm quan trọng của mã hóa dữ liệu

Nó đóng vai trò rất lớn trong phát triển ngành công nghệ thông tin cũng như các ngành khác. Vai trò lớn nhất của mã hóa dữ liệu là giúp truyền thông tin đến người nhận không bị rò rĩ.

Mã hóa dữ liệu là một công cụ thiết yếu của bảo mật thông tin. Mã hóa đáp ứng được các nhu cầu về tính bảo mật, tính chứng thực, tính không từ chối của hệ truyền tin.

Mã hóa dữ liệu giúp người dùng bảo vệ thông tin cá nhân của mình cũng như thông tin gửi đến người khác.

1.2. Tiêu chuẩn để đánh giá hệ mã hóa

1.2.1. Độ an toàn của thuật toán

Độ an toàn của thuật toán được đặc trưng cho khả năng của hệ mật mã chống lại sự thám mã.

Nguyên tắc đầu tiên trong mã hóa dữ liệu là bất kì một thuật toán mã hóa nào cũng đều có thể bị phá vỡ. Do đó, không có thuật toán mã hóa được xem là an toàn mãi mãi. Mỗi thuật toán mã hóa đều có những độ phứt tạp khác nhau và cho những độ an toàn khác nhau. Độ an toàn của thuật toán dựa vào các nguyên tắc sau:

- Nếu chi phí để giải mã một khối lượng thông tin lớn hơn giá trị của khối lượng thông tin đó được tạm coi là an toàn.
- Nếu thời gian để phá vỡ thuật toán là quá lớn thì thuật toán đó được tạm coi làm an toàn.
- Nếu lượng dữ liệu cần thiết để phá vỡ thuật toán là quá lớn so với dữ liệu đã được mã hóa thì thuật toán đó được tạm coi là an toàn.

Độ an toàn của thuật toán ở trên chỉ đúng với môt thời điểm nhất định nào đó, luôn luôn có những khả năng cho phép những người phá mã tìm ra cách để phá vỡ thuật toán.

Trên thực tế, có thể xây dựng hệ mã hóa có độ an toàn tuyêt đối nhưng nó rất khó cho việc sử dụng và chi phí rất cao.

1.2.2. Tốc độ mã hóa và giải mã

Tốc độ mã hóa và giải mã thuật toán luôn là tiêu chí đầu tiên để người ta lựa chọn một thuật toán để mã hóa dữ liệu.

Một hệ mật tốt là hệ mật có tốc độ mã hóa và giải mã nhanh.

1.2.3. Phân phối khóa

Phân phối khóa là một trong những nhân tố quan trọng quyết định độ an toàn của thuật toán.

Nó thể hiện ở hai khía cạnh:

- Phân phối khóa công khai nhưng đảm bảo tính mật.
- Sử dụng khóa công khai để phân phối khóa bí mật, khóa bí mật để mã hóa dữ liệu.

Một hệ mật mã hiện đại nào hiện nay đều phụ thuộc vào khóa, nó được truyền bí mật hay công khai. Hệ mật có khóa công khai sẽ có chi phí rẻ hơn hệ mật mã khóa bí mật.

Phân phối khóa cũng là một trong các tiêu chí để người dùng lựa chọn thuật toán để mã hóa dữ liệu.

1.3. Phân loại các thuật toán mã hóa dữ liệu

1.3.1. Phân loại theo phương pháp

1.3.1.1. Mã hóa hai chiều:

Bao gồm mã hóa đối xứng và mã hóa bất đối xứng.

1.3.1.1.1. Mã hóa đối xứng

Mã hóa đối xứng (mã hóa khóa bí mật) là phương pháp mã hóa hai chiều. Mã hóa đối xứng sử dụng khóa giải mã và khóa mã hóa là như nhau.

Để sử dụng được mã hóa đối xứng thì hai bên mã hóa và giải mã phải thống nhất với nhau về cơ chế giải mã cũng như mã hóa, nếu không có công việc này thì hai bên không thể giao tiếp nói chuyện với nhau.

Đối với mã hóa đối xứng, việc truyền - nhận hai bên được tiến hành phải thực hiện theo hai bước sau:

- Đầu tiên bên gửi và bên nhận bằng cách nào đó phải thỏa thuận khóa bí mật được dùng để mã hóa và giải mã. Vì chỉ cần biết được khóa này thì kẻ thứ ba có thể giải mã được thông tin nên thông tin này phải được bí mật truyền đi.
- Sau đó bên gửi sẽ dùng một thuật toán mã hóa với khóa bí mật tương ứng để mã hóa dữ liệu sắp được truyền đi. Khi bên nhận nhận được sẽ dùng chính khóa bí mật đó để giải mã dữ liệu.

Vấn đề lớn nhất của mã hóa đối xứng làm sao để thỏa thuận khóa bí mật giữa bên nhận và bên gửi, vì nếu truyền khóa này từ bên gửi sang bên nhận mà không dùng một biện pháp bảo vệ nào thì kẻ thứ ba có thể dễ dàng

lấy được khóa bí mật này. Và nếu điều này xảy ra thông tin sẽ không được an toàn nữa. Vậy nên khóa bí mật này phải được truyền bí mật.

Mã hóa đối xứng có thể chia thành hai loại:

- Mật mã khối: Là mã hóa tác động trên bản rõ theo từng nhóm bít. Từng nhóm bít này thường được gọi là khối (block).Từng khối dữ liệu trong văn bản ban đầu được thay thế bằng một khối dữ liệu khác có cùng độ dài. Đối với các thuật toán ngày nay, kích thước chung của một block là 64 bits. Các thuật toán mã hóa này thường dùng cho những dữ liệu có độ dài biết trước.
- Mật mã dòng: Là mã hóa tác động trên bản rõ theo từng bit một. Dữ liệu của văn bản ban đầu được mã hóa theo từng bit một. Các thuật toán mã hóa theo từng bước một này có tốc độ nhanh hơn các thuật toán mã hóa khối và nó thường được áp dụng để mã hóa dữ liệu không biết độ dài trước.

Các thuật toán mã hóa đối xứng thường gặp là: DES, AES, RC4...

1.3.1.1.2. Mã hóa bất đối xứng

Mã hóa bất đối xứng (mã hóa khóa bí mật) là phương pháp mã hóa một chiều. Mã hóa bất đối xứng sử dụng khóa mã hóa và khóa giải mã là khác nhau. Tất cả mọi người đều có thể biết khóa công khai và có thể lấy khóa công khai này để mã hóa thông tin nhưng chỉ có người nhận nắm giữ khóa bí mật nên chỉ người nhận mới có thể giải mã được thông tin và lấy lại được thông tin ban đầu.

Để thực hiện mã hóa bất đối xứng thì phải qua các bước sau:

- Bên nhận phải tạo ra một cặp khóa. Bên nhận sẽ giữ lại khóa bí mật và truyền cho bên gửi khóa công khai. Vì khóa công khai này được công khai nên không cần bảo mật.

- Bên gửi trước khi gửi sẽ dùng thuât toán mã hóa bất đối xứng mã hóa với khóa là khóa công khai từ bên nhận.
- Bên nhận sẽ giải mã dữ liệu với khóa là khóa bí mật đã được gửi trước đó để lấy lại dữ liệu ban đầu.

Hạn chế lớn nhất của phương pháp mã hóa bất đối xứng là tốc độ mã hóa và giải mã chậm, nếu dùng để mã hóa dữ liệu truyền nhận sẽ mất rất nhiều chi phí và thời gian. Vì những hạn chế đó, người ta thường dùng mã hóa bất đối xứng để truyền khóa bí mật giữa bên gửi và bên nhận, và có thể dùng khóa bí mật bên nhận và trao đổi với nhau qua mã hóa đối xứng.

Thuật toán mã hóa bất đối xứng thường gặp là: RSA...

1.3.1.2. Mã hóa một chiều

Là phương pháp mã hóa dữ liệu chuyển một chuỗi thông tin thành một chuỗi đã được mã hóa có độ dài nhất định mà ta không có bất kỳ cách nào để khôi phục chuỗi đã được mã hóa về lại chuỗi ban đầu.

Trong xử lí của hàm băm, dù chuỗi đầu vào khác nhau và có độ dài khác nhau thì chuỗi băm vẫn ra độ dài nhất định và dù hai chuỗi đầu vào chỉ khác nhau vài kí tự thì chuỗi băm cho kết quả khác nhau hoàn toàn. Do đó, hàm băm dùng để kiểm tra tính toàn vẹn của dữ liệu.

Các thuật toán mã hóa một chiều thường gặp như: MD4, MD5, SHA...

1.3.1.3. Mã hóa cổ điển

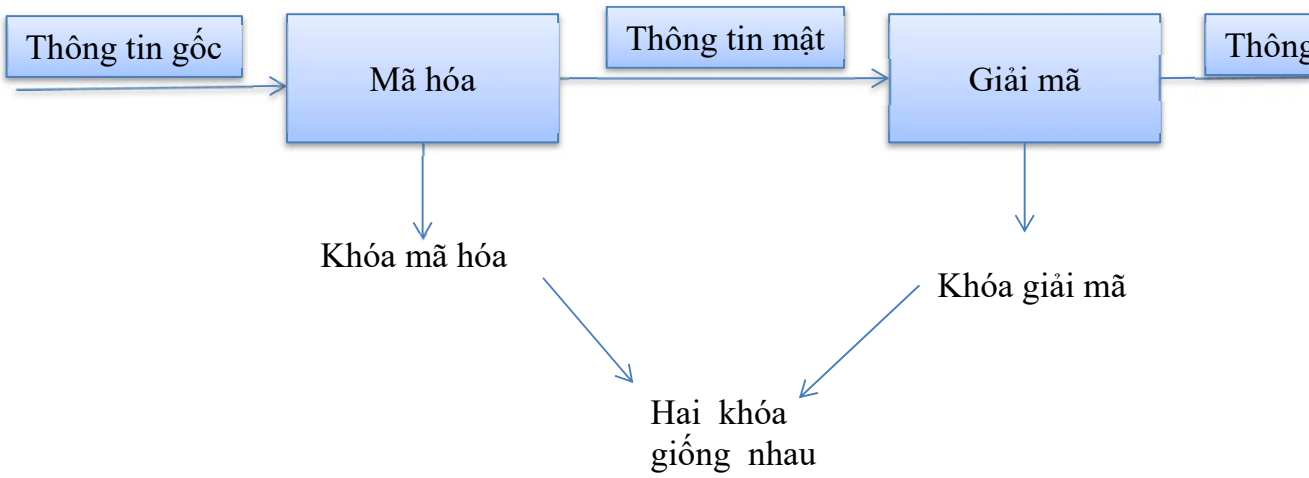
Mã hoá cổ điển là cách đơn giản nhất, tồn lại lâu nhất trên thế giới và không cần khóa bảo mật, chỉ cần người gửi và người nhận cùng biết về thuật toán chung thì hai bên có thể giao tiếp với nhau được.

Mã hóa này được xem là không an toàn, vì nếu một người thứ ba biết được thuật toán thì xem như thông tin không còn bảo mật nữa. Việc giữ bí mật thuật toán trở nên rất quan trọng và không phải ai cũng có thể giữ bí mật đó một cách trọn vẹn. Nếu thuật toán bị rò rĩ thì việc mã hóa thất bại.

1.3.2. Phân loại theo khóa

1.3.2.1. Mã hóa khóa bí mật

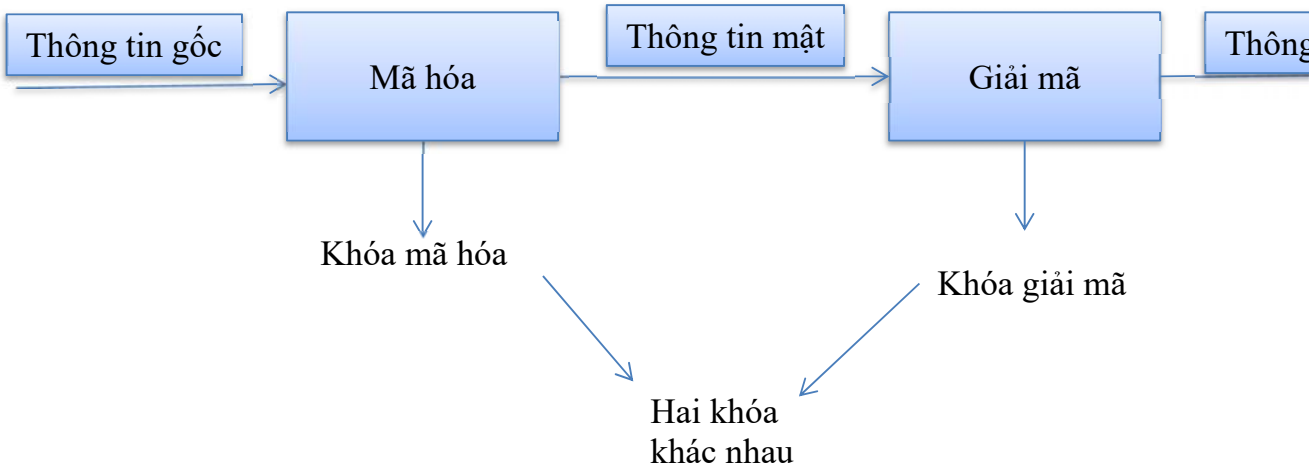
Là một dạng mật mã hóa cho phép người dùng trao đổi thông tin với nhau mà người gửi và người nhận không cần trao đổi khóa bí mật chung trước đó.



Hình 1. 1: Mã hóa khóa bí mật

1.3.2.2. Mã hóa khóa công khai

Là một dạng mật mã hóa cho phép người sử dụng trao đổi các thông tin mà người gửi và người nhận cần phải trao đổi các khóa chung bí mật trước đó.



Hình 1. 2: Mã hóa khóa công khai

1.4. Khóa

1.4.1. Khái niệm

Trong mật mã học, khóa là một đoạn thông tin điều khiển hoạt động của một thuật toán mã hóa dữ liệu. Nói một cách khác khóa là thông tin đặt biệt của quá trình mã hóa và giải mã.

Với các thuật toán tốt, mã hóa cùng một văn bản giống nhau mà khóa khác nhau sẽ cho ra các bản mã hoàn toàn khác nhau. Và ngược lại, nếu giải mã với khóa sai thì thuật toán sẽ không cho ra lại dữ liệu ban đầu.

1.4.2. Ví dụ

Ví dụ ta có một bức thư với nội dung như sau: " Chúng tôi sẽ đi học vào sáng thứ hai và chiều thứ tư hàng tuần" với khóa là " Thay thế mỗi kí tự thành kí tự thứ 3 đứng sau so với nó trong bức thư" và thay thế mỗi kí tự thành kí tự thứ 2 đứng sau so với nó trong bức thư thì bức thư đã mã hóa ra hoàn toàn khác nhau và nếu không biết được khóa thì không thể lấy lại nội dung ban đầu của bức thư gốc.

1.5. Các ứng dụng mã hóa dữ liệu

Với sự phát triển nhanh chóng của ngành công nghệ thông tin, ai ai cũng biết đến các thiết bị điện tử như: máy tính, tivi, điện thoại thông minh lần lượt được ra đời. Cùng với sự phát triển ấy là sự đột nhập các hacker, gây rất nhiều khó khăn cho người dùng hiện nay. Chính vì sự cần thiết ấy, việc bảo mật thông tin đã và đang được phát triển mạnh mẽ và trên nhiều lĩnh vực.

Tuy vậy, ứng dụng của mật mã hóa được chia thành các lĩnh vực nhỏ như:

- Bảo mật: Che dấu nội dung (file, CSDL...) của các thông điệp trong phiên giao tiếp trên hệ thống máy tính.
- Xác thực hóa: Đảm bảo tính chính xác nguồn gốc của một thông điệp hay người dùng nào đó.

- Toàn vẹn thông tin: Đảm bảo chỉ có nguồn gốc của thông điệp mới được phép thay đổi dữ liệu của thông điệp.
- Tính không từ chối: Khi một phiên trò chuyện được kết nối thành công, hai bên giao tiếp không thể chối từ mình đã trò chuyện**.**

Ngoài ra, còn các lĩnh vực quan trọng như: Chữ kí điện tử, xác thực...

1.6. Kết luận chương 1

Chương 1, đề tài chủ yếu tìm hiểu các nội dung cơ bản, tổng quát về mã hóa dữ liệu như:

- Khái niệm mã hóa dữ liệu
- Ứng dụng mã hoa dữ liệu
- Khóa
- Vài trò của mã hóa dữ liệu
- Độ an toàn của thuật toán
- Phân loại các thuật toán theo các phương pháp

Thông qua đó, hiểu rõ được tổng quát nhất về mã hóa dữ liệu.

CHƯƠNG 2: CÁC PHƯƠNG PHÁP MÃ HÓA CỔ ĐIỂN

2.1. Phương pháp đổi chỗ

Thuật toán phương pháp đổi chỗ mã hóa bằng cách thay đổi vị trí các kí tự trong văn bản gốc để tạo thành các văn bản mật mã.

Hiện nay, có rất nhiều biện pháp được áp dung. Vì vậy một bản rõ sẽ cho rất nhiều bản mã hoàn toàn khác nhau.

Một số kĩ thuật thường được sử dụng như:

2.1.1. Đảo ngược toàn bộ bản rõ

Phương pháp này mã hóa bằng cách đảo ngược toàn bộ văn bản gốc để tạo thành văn bản mật mã.

Ví dụ:

Bản rõ: HOANGVY.

Bản mã: YVGNAOH.

Kĩ thuật này hiện nay không được sử dụng nữa, bởi độ an toàn ít, dễ giải mã. Quá dễ dàng cho người thứ ba phá mã và lấy lại thông tin ban đầu.

2.1.2. Mã hóa theo hình học

Theo cách này các kí tự trong văn bản gốc sẽ được sắp xếp lại theo một mẫu hình học nào đó, thông thường người ta sẽ sắp xếp thành mảng hoặc ma trận hai chiều.

Ví dụ:

Bản rõ: TRUONGDAIHOCQUANGNAM

Dữ liệu TRUONGDAIHOCQUANGNAM sẽ được viết lại thành ma trận hai chiều 5x4 như hình bên dưới:

T	R	U	O
N	G	D	A
I	H	O	C
Q	U	A	N
G	N	A	M

Nếu ta lấy theo thứ tự cột là cột 1, cột 2, cột 3, cột 4 thì kết quả sẽ được bản mã là: TNIQGRGHUNUDOAAOACNM.

Kĩ thuật này tuy an toàn hơn kĩ thuật đổi chỗ toàn bộ bản rõ nhưng cũng không được nhiều người dùng đến. Vì độ an toàn chưa được đánh giá là cao.

2.1.3. Đổi chỗ các cột

Trước hết đổi chỗ kí tự trong văn bản gốc thành hình chữ nhật theo các cột, sau đó các cột được sắp xếp lại và các chữ lấy ra theo chiều ngang.

Ví dụ:

Bản rõ: NGUYEN THI HOANG VY LOP TIN K13.

Bản rõ trên sẽ được xếp thành ma trận 5x5 như sau:

N	N	O	V	I
G	T	A	L	N
U	H	N	O	K
Y	I	G	P	1
E	H	V	T	3

Bản mã sẽ được lấy ra từ ma trận trên nhưng theo chiều ngang. Ở đây chúng ta có 5! cột, vậy nên chúng ta sẽ có 5! bản rõ khác nhau và 5! bản mã mã này nội dung hoàn toàn khác nhau.

Nếu ta lấy lần lượt cột 5, cột 4, cột 3, cột 2, cột 1 ghép lại với nhau thì ta sẽ được bản mã là: IVONNNLATGKONHU1PGIY3TVHE.

Hiện nay, kĩ thuật này còn được sử dụng nhưng rất ít vì độ an toàn tương đối thấp, cách mã hóa dễ hiểu nên dễ bị các hacker phá mã.

2.2. Phương pháp thay thế

2.2.1. Mã Ceasar

Phương pháp này được phát hiện từ rất sớm bởi Julius Ceasar trong cuộc chiến Gallic Wars. Lần đầu tiên được sử dụng trong quân sự. Việc mã hoá được thực hiện đơn giản là thay thế từng kí tự trong văn bản gốc bằng

ba kí tự sau nó. Điều này tương ứng với phép dịch ba vị trí và được mô phỏng ở hình dưới đây:

A	B	C	D	E	F	G	H	I	J
D	E	F	G	H	I	J	K	L	M

K	L	M	N	O	P	Q	R	S	T
N	O	P	Q	R	S	T	U	V	W

U	V	W	X	Y	Z
X	Y	Z	A	B	C

Theo hình trên, chữ A sẽ được thay thế bởi chữ D, chữ B sẽ được thay thế bới chữ E, chữ Z sẽ được mã hóa thành chữ C...

Với Ceasar ông chỉ áp dụng phép dịch ba vị trí như trên hình minh họa. Nhưng sau này khi áp dụng nó người ta tìm đến các phép dịch lớn hơn. Ở một khía cạnh khác, các bạn đã biết bảng chữ cái tiếng anh chúng ta gồm 26 kí tự, do đó điểm yếu của phương pháp này nằm ở phạm vi phép dịch chỉ được phép nằm trong khoảng từ 0 -25 kí tự. Nếu ta sử dụng ở phép dịch 26 thì phép dịch sẽ bằng 0. Dựa trên phương pháp này người ta đã xây dựng nên công thức cho Ceasar như sau:

Đánh số thứ tự cho từng kí tự trong bảng chữ cái:

A	B	C	D	E	F	G	H	I	J
0	1	2	3	4	5	6	7	8	9

K	L	M	N	O	P	Q	R	S	T
10	11	12	13	14	15	16	17	18	19

U	V	W	X	Y	Z
20	21	22	23	24	25

Mã Ceasar được định nghĩa qua phép tịnh tiến các chữ như sau: Với mỗi chữ P sẽ được thay thế bằng chữ C trong đó: C = (p + k) mod 26

Và quá trình giải mã là: P = (c – k) mod 26

Trong đó:

p là số thứ tự của chữ trong bản rõ

c là số thứ tự của chữ tương ứng của bản mã

k là khoá của mã Ceasar

Khóa k cho hai quá trình phải là giống nhau. Nếu sai dù chỉ một kí tự thì sẽ không cho lại dữ liệu ban đầu.

Có 26 giá trị khác nhau của k, nên có 26 khoá khác nhau. Thực tế độ dài khoá ở đây chỉ là 1, vì mọi chữ đều tịnh tiến đi một khoảng như nhau.

Ví dụ:

Bản rõ: QUANG NAM

Khóa có độ dài là 3

Thì bản rõ sẽ là TXAQJQDP

Mã Ceasar là hệ mã cũ và trở nên không an toàn vì phép thử của nó quá ít chỉ với 26 lần. Do đó các hacker có thể dùng phương pháp vét cạn.

Để phá mã người ta tốn quá ít thời gian và chi phí nên độ an toàn thấp. Bởi vậy, nên thuật toán này cũng rất ít người còn dùng đến nó.

2.2.2. Hệ mã hóa Vinegere

Là phương pháp mã hóa văn bản bằng cách sử dụng xen kẻ một số phép mã hóa của mã hóa Ceasar khác nhau dựa trên các chữ cái trên cùng một từ khóa. Nó là một dạng đơn giản của mã hóa thay thế sử dụng nhiều bảng chữ cái.

Để mã hóa, ta dùng một hình vuông Vinegere. Nó gồm 26 hàng, mỗi hàng dịch về bên trái một bước so với hàng phía trên, tạo thành 26 bảng mã

Caesar. Trong quá trình mã hóa, tùy theo từ khóa mà mỗi thời điểm ta dùng một dòng khác nhau để mã hóa văn bản.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
4	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
5	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
6	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
7	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
8	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
9	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
10	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
11	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
12	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
13	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
14	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
15	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
16	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
17	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
18	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
19	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
20	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
21	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
22	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
23	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
24	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
25	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
26	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Hình 2. 1: Mã hóa Vinegere

Trong mã hóa Vinegere, để có thể mã hóa được văn bản thì bắt buộc chiều dài của khóa và chiều dài của văn bản phải bằng nhau. Nhưng trên thực tế không ai đặt khóa dài bằng văn bản. Mà chiều dài của khóa sẽ được lặp lại nhiều lần cho đến khi độ dài của nó bằng với chiều dài của văn bản.

Ví dụ:

Bản rõ: "SINHVIEN"

Khóa: "HOANGVY"

Vì bản rõ có chiều dài bằng 8 kí tự nhưng khóa chỉ có độ dài bằng 7 kí tự mà trong mã hóa Vinegere chiều dài của khóa và bản rõ phải bằng nhau nên khi mã hóa khóa sẽ trở thành "HOANGVYH"

Nhìn vào **Hình** 2.1 ta thấy ứng với mỗi kí tự trong bản rõ là một kí tự trên hàng của bảng biểu, ứng với mỗi kí tự từ khóa là một kí tự trên cột của bảng biểu. Giao giữa hàng kí tự khóa và cột kí tự bản rõ là kí tự được mã hóa.

Bản rõ	S	I	N	H	V	I	E	N
Khóa	H	O	A	N	G	V	Y	H
Bản mã	Z	W	N	U	B	C	C	U

Trong mã hóa này, hai kí tự giống nhau của bản rõ có thể mã hóa ra hai kí tự khác nhau hoàn toàn trong bản mã. Và ngược lại, hai kí tự của bản mã giống nhau cũng không hẳn được tạo ra bởi hai kí tự giống nhau của bản gốc.

Với mã hóa vinegere, việc mã hóa khá đơn giản, chỉ cần biết được khóa thì có thể dễ dàng giải mã. Vậy nên cần bảo mật tốt khóa để an toàn hơn.

2.3. Kết luận chương 2

Chương 2, đề tài tập trung vào mã hóa cổ điển dựa trên hai phương pháp: phương pháp thay thế và phương pháp đổi chỗ.

Trong phương pháp thay thế:

- Hệ mã Ceasar
- Hệ mã Vinegere

Trong phương pháp đổi chỗ có một số kĩ thuật như:

- Đảo ngược toàn bộ bản rõ
- Mã hóa theo hình học
- Đổi chỗ các cột

Thông qua đó, nắm bắt và hiểu rõ cách hoạt động của từng loại thuật toán và đưa ra các ví dụ minh họa. Bên cạnh đó, còn đưa ra được các ưu điểm, nhược điểm của từng loại mã hóa cổ điển trên.

CHƯƠNG 3: CÁC PHƯƠNG PHÁP MÃ HÓA HIỆN ĐẠI

3.1. Thuật toán mã hóa RSA

3.1.1. Lịch sử

Thuật toán RSA được Ron Rivest, Adi Shamir và Len Adleman mô tả lần đầu tiên vào năm 1977 tại Học viện Công nghệ Massachusetts. Tên của thuật toán lấy từ ba chữ cái đầu của tên ba tác giả(RSA).

Trước đó, vào năm 1973 nhà toán học người Anh Clifford Cocks đã mô tả một thuật toán tương tự. Với khả năng tính toán tại thời điểm đó thì thuật toán này không khả thi và chưa bao giờ được thực nghiệm. Tuy nhiên, năm 1997 phát minh này mới được công bố vì được xếp vào loại tuyệt mật.

3.1.2. Khái niệm

RSA là một thuật toán mật mã hóa khóa công khai, 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 khai.

RSA đang được sử dụng phổ biến trong thương mại điện tử và được cho là đảm bảo an toàn với điều kiện độ dài khóa đủ lớn.

3.1.3. Đặc điểm và khả năng bị tấn công

- Đặc điểm
    - Khóa bí mật khó bị tấn công vì chỉ có một người bảo mật nên khóa không bị truyền qua các môi trường không an toàn.
    - Khi biết các tham số ban đầu của hệ mã hóa việc tính ra cặp khóa công khai và bí mật dễ dàng (người gửi có bản rõ và khóa công khai thì dễ dàng tạo ra bản mã, và người nhận có bản mã và khóa bí mật dễ dàng lấy lại bản rõ ban đầu).
    - Độ an toàn cao.
    - Thao tác giải mã và mã hóa tốn ít thời gian.

- Khả năng bị tấn công
  - Thuật toán RSA có tính an toàn cao, ít bị tấn công.
  - Từ khi được công bố, RSA đã được phân tích tính an toàn bởi nhiều nhà khoa học. Và đã có một số cuộc tấn công lên hệ mật RSA song chúng chủ yếu là minh họa cho việc sử dụng RSA không đúng cách.
  - Để giải mã được thuật toán người ta cần khá nhiều thời gian và tiền bạc nên ít ai đi phá mã giải thuật này.
  - Phép thử lớn nên ít được các hacker quan tâm nên RSA đang tạm an toàn.

3.1.4. Thuật toán và sơ đồ

Giả sử V (người gửi) và T (người nhận) muốn trao đổi thông tin mật với nhau trong môi trường không an toàn internet. Với thuật toán RSA trước hết V phải tạo ra hai khóa là khóa công khai và khóa bí mật gồm 4 bước sau:

Bước 1: Bên T tạo ra hai số nguyên tố lớn ngẫu nhiên p và q

Bước 2: Tính n= p*q và giá trị hàm số ϕ(n) = (p-1)(q-1)

Bước 3: Chọn một số ngẫu nhiên e (sao cho 0 < e < $\phi(n)$)sao cho

UCLN(e, $\phi(n)) = 1$

Bước 4: Tính d= e^{-1} bằng cách dùng thuật toán Euclide. Tìm số tự nhiên x sao cho

$$d = \frac{x*\phi(n)+1}{e}$$

Bước 5: Ta có n và e là khóa công khai, d là khóa bí mật

Sau đó là quá trình mã hóa của bên gửi và giải mã của bên nhận. Quá trình này cũng thực hiện qua 5 bước:

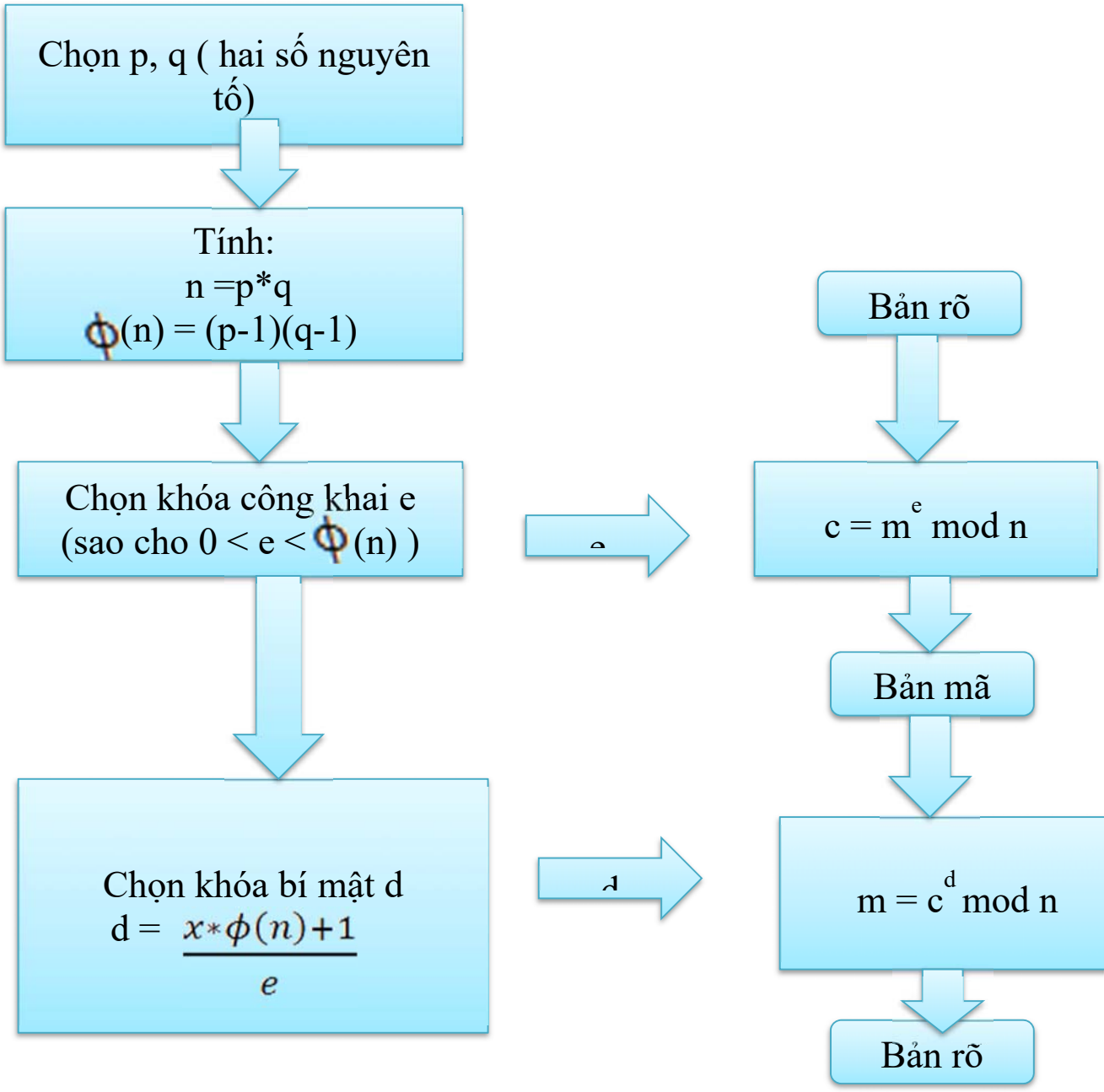
Bước 1: V nhận được khóa công khai của T

Bước 2: V biểu diễn thông tin cần gửi thành số m

Bước 3: Tính $c = m^e \bmod n$

Bước 4: Gửi c cho T

Bước 5: Giải mã: Tính $m = c^d \bmod n$ => m là thông tin từ V gửi sang cho T.



Hình 3. 1: Sơ đồ hoạt động của RSA

3.1.5. Ví dụ minh họa

Đầu tiên chúng ta sẽ tạo ra khóa công khai và khóa bí mật.

Bước 1: Chọn p= 11 và q = 23 (Đây là hai số nguyên tố)

Bước 2: Tính n và ϕ(n)

n = p*q = 11*23 = 253

ϕ(n) = (p-1)(q-1) = 10*22 = 220

Bước 3: Chọn e = 3 Vì UCLN(220,3) = 1

Bước 4: Tính d:

d = $\frac{x*\phi(n)+1}{e}$ <=> 3*d = x*220 + 1 => d = 807 và x = 11

Bước 5: Ta có: n = 253, e = 3, d = 807

Quá trình mã hóa và giải mã.

Bước 1: V nhận khóa công khai n = 253, e = 3

Bước 2: Biểu diễn thông tin cần gửi m =38

Bước 3: Tính c

c = m^e mod n = 38^3 mod 253 = 224

Bước 4: Gửi c cho T

Bước 5: Giải mã: Tính m = c^d mod n = 224^{807} mod 253 => m = 38

3.2. Thuật toán mã hóa MD5

3.2.1. Lịch sử

MD5 (Message Digest 5) là một loạt các giải thuật đồng hóa thông tin được thiết kế bởi Ronald Rivest. Khi công việc phân tích chỉ ra rằng giải thuật trước MD5 có vẻ không an toàn, ông đã thiết kế ra MD5 vào năm 1991 để thay thế MD4 an toàn hơn.

3.2.2. Khái niệm

MD5 là một hàm băm để mã hóa một chuỗi thông tin có độ dài bất kì thành một chuỗi có độ dài cố định là 128 bít, từng được xem là một chuẩn internet, MD5 được sử dụng rộng rãi trong các chương trình an ninh mạng và cũng thường được dùng để kiểm tra tính toàn vẹn của tệp tin.

3.2.3. Đặc điểm và khả năng bị tấn công

- Đặc điểm
  - Giải thuật dễ hiểu, dễ bị phá mã.
  - Với một bản rõ chỉ có một bản mã và không bao giờ có thể xuất hiện bản mã thứ hai.
  - Tất cả các bản rõ có độ dài khác nhau đều cho kết quả bản mã có độ dài cố định bằng 128 bít.
  - Với bất kì giá trị băm, không thể chuyển thành bản rõ cho dù biết bản mã.
  - Hàm băm có thể hoạt động trên khối dữ iệu có độ dài bất kì
- Khả năng bị tấn công
  - Khả năng bị tấn công cao.
  - Trên lí thuyết, đây là hàm băm một chiều nghĩa là người ta không thể phá mã và lấy lại bản gốc nhưng trên thực tế việc này đã được các hacker tìm đến và làm đề tài từ rất lâu vì nó tương đối dễ phá mã.

3.2.4. Thuật toán và sơ đồ

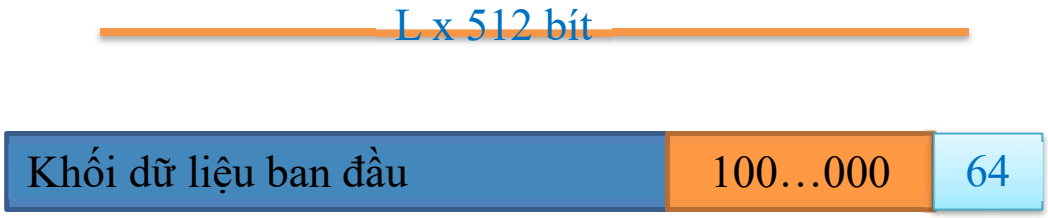
Input: Dữ liệu có độ dài bất kì.

Output: Giá trị băm 128 bít.

Giải thuật gồm 5 bước thực hiện trên khối dữ liệu có độ dài 512 bít

Bước 1: Nhồi dữ liệu

- Nhồi thêm các bit sao cho dữ liệu có độ dài l = n * 512 + 448 (n,l nguyên).
- Tất cả các khối dữ liệu ban đầu luôn được thực hiện nhồi dữ liệu ngay cả khi dữ liệu ban đầu có độ dài mong muốn.
- Các bít được nhồi theo nguyên tắc: Thêm 1 bit 1 và các bit 0 theo sau:

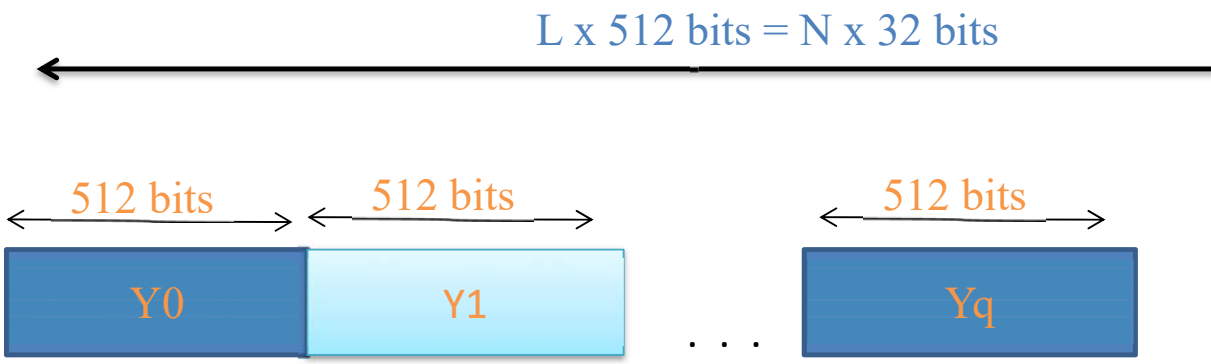


Hình 3. 2: Nhồi dữ liệu của MD5

Bước 2: Thêm vào độ dài.

- Độ dài của khối dữ liệu ban đầu được biểu diễn dưới dạng nhị phân 64 bit và được thêm vào cuối chuỗi nhị phân kết quả của bước 1.
- Kết quả có được từ hai bước đầu là một khối dữ liệu có độ dài là bội số của 512. Chiều dài của dãy bít cuối cùng của thông điệp chia thành L khối 512 bit:

  $Y_0, Y_1,\ldots, Y_L$.



Hình 3. 3: Thêm độ dài của MD5

Bước 3: Khởi tạo bộ đệm MD (MD buffer)

- Một bộ đệm 128 bit được dùng lưu trữ các giá trị băm trung gian và kết quả.
- Bộ đệm được biểu diễn 4 thanh ghi từ 32 bit với các giá trị khởi tạo ở dạng big-endian (byte có trọng số lớn nhất trong từ nằm ở địa chỉ thấp nhất) và có 2 bộ đệm. Bố thanh ghi của bộ đệm đầu tiên được

đánh đặt tên là A,B,C,D có giá trị như sau (theo dạng Hex) và các giá tri này là hằng số:

A=67 45 23 01

B=EF CD AB 89

C=98 BA DC FE

D=10 32 54 76

Các giá trị này tương đương với các từ 32-bit sau:

A = 01 23 45 67

B = 89 AB CD EF

C = FE DC BA 98

D = 76 54 32 10

Trước tiên khối L_1 kết hợp giá trị khởi tạo H_0 thông qua hàm F để tính giá trị hash H_1, sau đó khối L_2 kết hợp giá trị khởi tạo H_1 thông qua hàm F để tính giá trị hash H_2. Cứ như vậy cho đến khối L_N ta sẽ có giá trị băm cho toàn thông điệp là H_N.

Bước 4: Xử lý các khối dữ liệu 512 bit

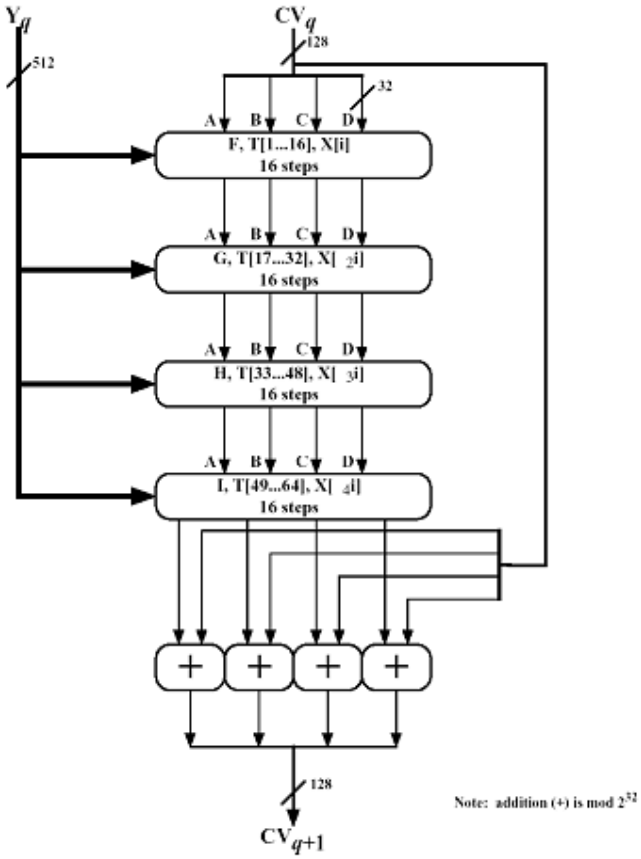
- Trọng tâm của giải thuật là hàm nén gồm 4 vòng xử lý. Các vòng này có cấu trúc giống nhau nhưng sử dụng các hàm luận lí khác nhau gồm F, G, H và I.

Vòng	Hàm	Giá trị
0 -15	F(X, Y, Z)	(X AND Y) OR ((NOT X) AND Z)
16-31	G(X, Y, Z)	(Z AND X) OR ((NOT Z) AND Y)
32-48	H(X, Y, Z)	X XOR Y XOR Z
49-63	I(X, Y, Z)	Y XOR (X AND (NOT Z))

- Tại mỗi bước, các giá trị ABCD của giá trị hash sẽ biến đổi qua 64 vòng. Tại vòng thứ j sẽ có hai tham số là K_j và W_j đều có kích thước

là 32 bít. Các hằng số K_j được tính: K_j là phần nguyên của số 2^{32} x abs(sin(i)), i được tính theo radian.

- Giá trị block L_i 512 bít được biến đổi qua một hàm mesage schedule cho ra 64 giá trị $W_0, W_1, ..., W_{63}$ và mỗi giá trị 32 bít.. Khối L_i được chia thành 16 khối 32 bít tương ứng với $W_0, W_1, .. ., W_{15}$. Tiếp theo, 16 giá trị này được lặp lại 3 lần tạo thành dãy có 64 giá trị.
- Kết quả của vòng cuối cùng được cộng theo modulo 2^{32} với đầu vào CV_q để tạo CV_{q+1}.

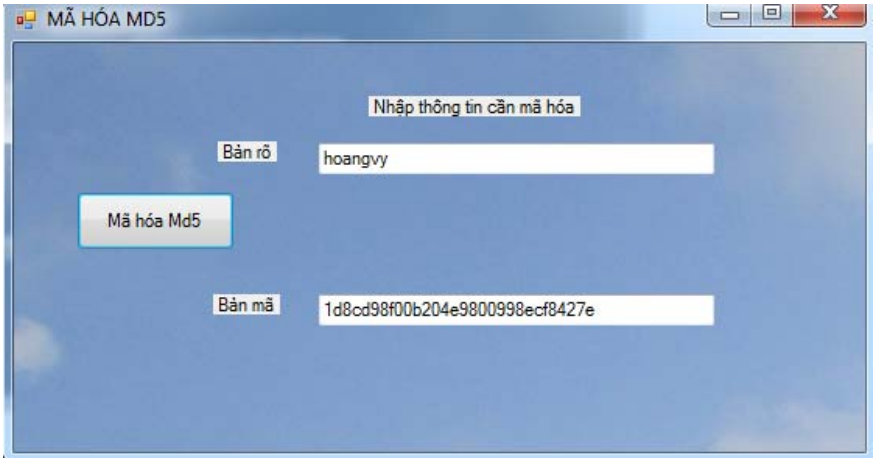


Hình 3. 4: Một vòng xử lí trên khối 512 bít của MD5

Bước 5: Xuất kết quả

Sau khi thao tác trên toàn bộ L blocks 512 bit. Kết quả của khối thứ L là bảng băm 128 bit.

3.2.5. Giao diện chương trình minh họa



Hình 3. 5: Giao diện chương trình minh họa của MD5

3.3. Thuật toán mã hóa SHA-1

3.3.1. Lịch sử

SHA-1 (Secure Hash Algorithm) là một trong năm thuật toán an toàn được chấp nhận bởi FIFS dùng để chuyển một đoạn dữ liệu nhất định thành một đoạn dữ liệu có chiều dài không đổi với xác suất khác biệt cao. Theo chuẩn FIPS 180-2 phát hành ngày 1 tháng 8 năm 2002 những thuật giải này được gọi là "an toàn".

3.3.2. Khái niệm

SHA-1 là một hàm băm mật mã được thiết kế bởi cơ quan an ninh quốc gia và được công bố bởi NIST hay còn gọi là Cục Xử Lý Thông Tin Tiêu Chuẩn Liên Bang của Mỹ.

SHA là viết tắt của Secure Hash Algorithm. Ba thuật toán SHA có cấu trúc khác nhau và được phân biệt là: SHA-0, SHA-1, và SHA-2. SHA-1 gần tương tự như SHA-0, nhưng sửa chữa một lỗi trong các đặc tả kỹ thuật gốc của hàm băm SHA dẫn đến những điểm yếu quan trọng. Các thuật toán SHA-0 đã không được sử dụng trong nhiều ứng dụng. SHA-2 mặt khác có những điểm khác biệt quan trong so với hàm băm SHA-1.

3.3.3. Đặc điểm và khả năng bị tấn công

- Đặc điểm

- Dữ liệu đầu vào có nhiều giá trị khác nhau nhưng dữ liệu đầu ra đều có độ dài cố định là 160 bít.
- Không có khả năng lấy lại dữ liệu ban đầu sau khi giải mã.
- Dễ hiểu, độ phứt tập thấp.

- Khả năng bị tấn công
  - Độ an toàn thấp
  - Trên nguyên tắc thì đây là hàm băm một chiều nhưng nó đã được các hacker phá mã từ rất lâu.

3.3.4. Thuật toán và sơ đồ

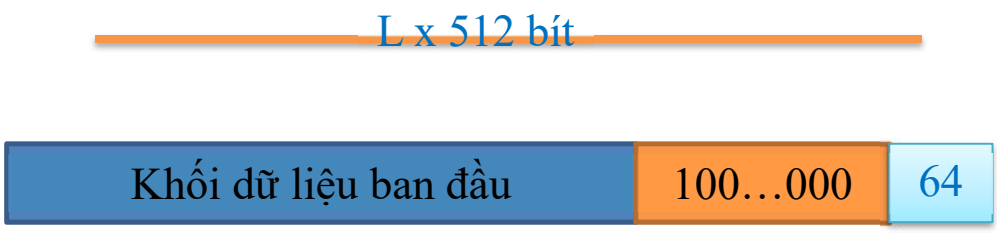
Input: Chuỗi có độ dài bất kì và tối đa là 2^{64} bít.

Output: Chuỗi có độ dài cố định 160 bít.

Thuật toán gồm 5 bước:

Bước 1: Nhồi dữ liệu

- Nhồi thêm các bít sao cho dữ liệu có độ dài l = n * 512 + 448 (n,l nguyên).
- Tất cả các khối dữ liệu ban đầu luôn được thực hiện nhồi dữ liệu ngay cả khi dữ liệu ban đầu có độ dài mong muốn.
- Các bít được nhồi theo nguyên tắc: Thêm 1 bit 1 và các bit 0 theo sau:

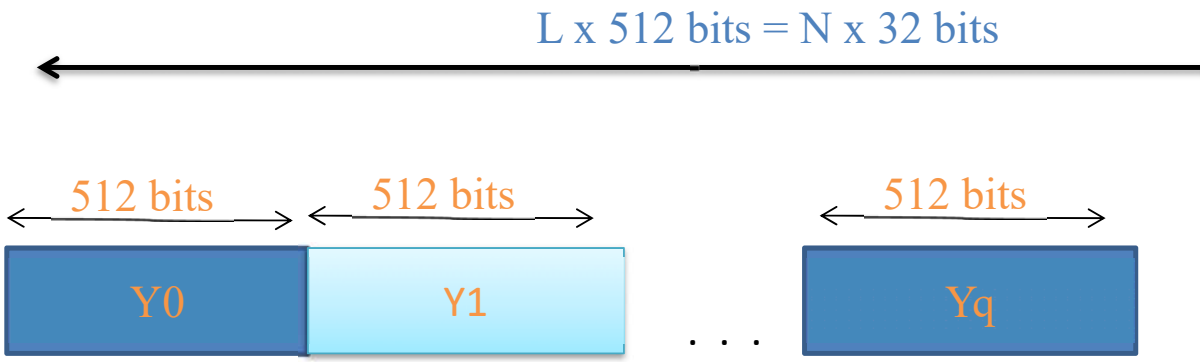


Hình 3. 6: Nhồi dữ liệu của SHA-1

Bước 2: Thêm vào độ dài.

- Độ dài của khối dữ liệu ban đầu được biểu diễn dưới dạng nhị phân 64 bit và được thêm vào cuối chuỗi nhị phân kết quả của bước 1.

- Kết quả có được từ hai bước đầu là một khối dữ liệu có độ dài là bội số của 512. Chiều dài của dãy bít cuối cùng của thông điệp chia thành L khối 512 bit Y_0, Y_1…, Y_L.



Hình 3. 7: Thêm độ dài của SHA-1

Bước 3: Khởi tạo bộ đệm MD (MD buffer)

- Một bộ đệm 160 bit được dùng lưu trữ các giá trị băm trung gian và kết quả.
- Bộ đệm được biểu diễn bằng 5 thanh ghi 32 bit với các giá trị khởi tạo ở dạng big-endian (byte có trọng số lớn nhất trong từ nằm ở địa chỉ thấp nhất)và có 2 bộ đệm. Năm thanh ghi của bộ đệm đầu tiên được đánh đặt tên là A, B, C,D, E có giá trị như sau (theo dạng Hex) và các các giá trị này là hằng số:

  A=67 45 23 01

  B=EF CD AB 89

  C=98 BA DC FE

  D=10 32 54 76

  E=C3 D2 E1 F0.

Các giá trị này tương đương với các từ 32 bit sau:

A = 01 23 45 67

B = 89 AB CD EF

C = FE DC BA 98

D = 76 54 32 10

E = C3 D2 E1 F0

Bước 4: Xử lý các khối dữ liệu 512 bít

- Trọng tâm của giải thuật bao gồm 4 vòng lặp thực hiện tất cả 80 bước. Bốn vòng lặp có cấu trúc như nhau, chỉ khác nhau ở các hàm logic F_1, F_2, F_3, F_4. Mỗi vòng có đầu vào gồm khối 512 bit hiện thời và một bộ đệm 160 bit ABCDE. Các thao tác sẽ cập nhật giá trị bộ đệm.

Bước	Hàm	Giá trị
$(0 \leq t \leq 19)$	$F_1 = F(X, Y, Z)$	(X AND Y) OR ((NOT Y) AND Z)
$(20 \leq t \leq 39)$	$F_2 = F(X, Y, Z)$	Y XOR Y XOR Z
$(40 \leq t \leq 59)$	$F_3 = F(X, Y, Z)$	(X AND Y) OR (X AND Z) OR (Y AND Z)
$(60 \leq t \leq 79)$	$F_4 = F(X, Y, Z)$	X XOR Y XOR Z

- Mỗi bước sử dụng một hằng số Kt $(0 \leq t \leq 79)$

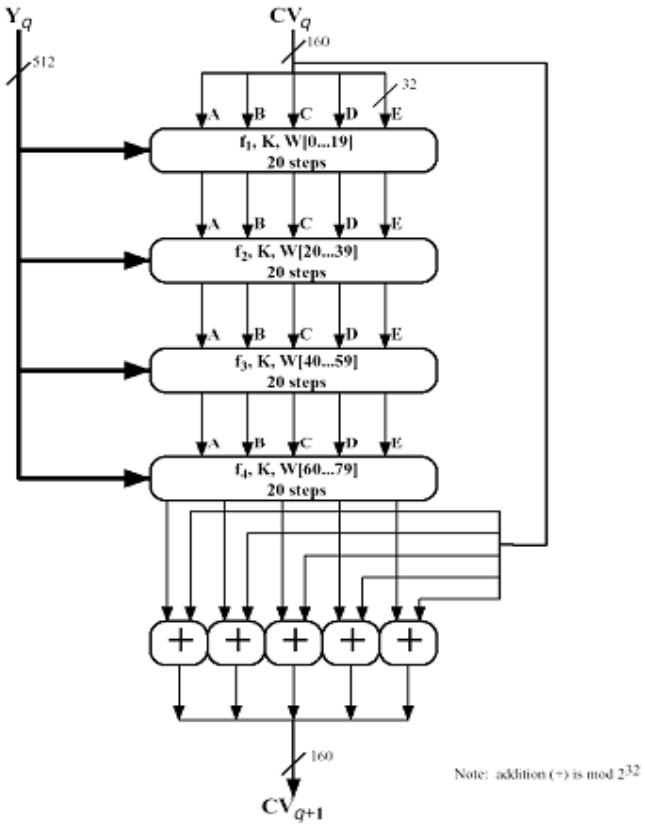
  Kt = 5A827999 $(0 \leq t \leq 19)$

  Kt = 6ED9EBA1 $(20 \leq t \leq 39)$

  Kt = 8F1BBCDC $(40 \leq t \leq 59)$

  Kt = CA62C1D6 $(60 \leq t \leq 79)$

- Đầu ra của 4 vòng (bước 80) được cộng với đầu ra của bước CV_q để tạo ra CV_{q+1}

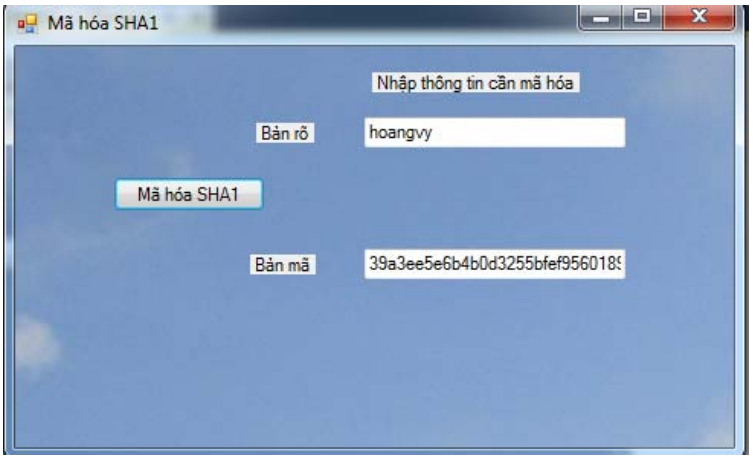


Hình 3. 8: Một vòng xử lí trên khối 512 bít của SHA-1

Bước 5: Xuất kết quả

Sau khi xử lý hết L khối 512 bit, đầu ra của lần xử lý thứ L là giá trị băm 160 bits.

3.3.5. Giao diện chương trình minh họa



Hình 3. 9: Giao diện chương trình minh họa SHA-1

3.4. Kết luận chương 3

Ở chương 3, đề tài tìm hiểu và nghiên cứu một số thuật toán mã hóa hiện đại như:

- Hệ mã RSA
- Hệ mã MD5
- Hệ mã SHA-1

Bên cạnh đó, nắm được các ưu điểm, nhược điểm, độ an toàn của từng thuật toán mã hóa. Đồng thời đưa ra được các ví dụ minh họa của hệ mã RSA, và chương trình mô phỏng cho MD5 và SHA-1 bằng ngôn ngữ C#.

CHƯƠNG 4: ỨNG DỤNG MÃ HÓA EMAIL PGP TRÊN THUNDERBIRD

4.1. Cách thức mã hóa email

4.1.1. Giới thiệu tổng quát về dịch vụ Email.

Thư điện tử (Email) là một dịch vụ chuyển thư qua mạng máy tính. Email là dịch vụ chuyển thư rất nhanh và được thông dụng nhất hiện nay Bởi ở nó có rất nhiều tiện lợi. Dù bạn ở bất cứ nơi đâu, chỉ cần hai bên có sử dụng mạng internet là có thể giao tiếp với nhau. Một nguồn tin nhắn gửi đi không những có thể gửi tới một mà nhiều người ở dưới dạng dữ liệu bình thường hay đã được mã hóa.

Hiện nay có rất nhiều biện pháp để bảo đảm an toàn cho Email.

Email dần dần đươc phát triển hơn, ngày nay email không chỉ gửi được văn bản chữ mà còn nhiều dữ liệu khác nhau như: hình ảnh, âm thanh...

So với dịch vụ gửi thư thông thường thì Email đứng đầu với nhiều tiện ích vượt trội như:

- Về độ an toàn thư: Vì khi gửi dịch vụ email có dùng mật khẩu và không phải qua một ai trung gian nên so với gửi thư thông thường là an toàn hơn.
- Về khối lượng thông tin: Email có thể chuyển khối lượng dữ liệu lớn so với cách gửi thư bằng bưu điện gấp nhiều lần.
- Về thời gian: Chỉ vài giây hoặc vài phút là có thể gửi thư đến người nhận, điều này không đáng kể so với chuyển thư bình thường bằng bưu điện. Điều này làm cho người sử dụng tiết kiệm được nhiều thời gian cũng như tiền bạc.

Email còn nhiều tính năng nổi bật khác như một người có thể có nhiều địa chỉ và không bao giờ bị trùng, không bị hư các phần vật lí.

Một email thông thường thuờng có dạng:Tên_định_dạng@tên_miền.

Ví dụ: Nguyenthihoangvy123@gmail.com

4.1.2. Các phương thức gửi -nhận mail.

4.1.2.1. POP3 (Post ofice Protocol 3)

Là giao thức nhận email, cho phép người dùng tải thư về máy có thể đọc và quản lí thư trên máy cục bộ. Vì dung lượng trên máy chủ hạn chế nên thư tải về bị xóa khỏi máy chủ hoặc lưu dưới dạng bản sao.

POP3 sử dụng cổng 110 để thực hiện thủ tục nhận thư, tuy nhiên có thể sử dụng cổng 995 để mã hóa kết nối trên kênh truyền SSL.

Thủ tục của POP3 là thủ tục rất có ích và đơn giản.

4.1.2.2. SMTP (Simple Mail Transfer Protocol)

Là giao thức gửi email sử dụng kết nối TCP/IP nhưng không có tính bảo mật.

Là giao thức gửi thông tin và không cho phép ai lấy thông tin ở máy chủ từ xa theo yêu cầu của mình một cách tùy ý.

Là giao thức được phát triển ở tầng ứng dụng của mô hình 7 tầng OSI và sử dụng cổng 25 để lắng nghe yêu cầu từ các client . Để tăng tính bảo mật, nhà cung cấp email sẽ yêu cầu client gửi đến cổng 465 hoặc cổng 587 trên kênh truyền SSL/TLS.

4.1.2.3. IMAP (Internet Message Access Protocol)

Là giao thức nhận email, ra đời sau POP3 và được cải tiến bởi nó, vì vậy IMAP có nhiều chức năng và phứt tạp hơn POP3.

IMAP được khắc phục hạn chế về dung lượng của POP3 nên thư được lưu trên server và có thể tải về trên nhiều máy nhưng dữ liệu vẫn được đồng bộ hóa.

IMAP4 là giao thức được sử dụng rộng rãi hiện nay. IMAP sử dụng cổng 143 của TCP.

4.1.3. Các hình thức mã hóa Email

Hiện nay, có nhiều hình thức để mã hóa Email nhưng thông thường có ba cách để mã hóa: mã hóa kết nối, mã hóa thư gửi đi và mã hóa lưu trữ.

4.1.3.1. Mã hóa kết nối

Đây là loại mã hóa đối với người cung cấp dịch vụ Email. Điều này rất quan trọng trong môi trường như hiện nay, máy tính của bạn đang sử dụng mạng wifi tràn lan. Khi bạn gửi một Email nào đó đến người khác, trước khi đến người nhận tin nhắn của bạn sẽ đến máy chủ và trong thời gian này các hacker có thể đọc lén dữ liệu của bạn một cách dễ dàng nếu không được bảo mật.

Để đảm bảo việc kết nối giữa máy tính chủ và máy tính của bạn cần phải cài đặt hai mã hóa là SSL và TLS. Hai giao thức này giúp bảo vệ sự kết nối của bạn ít bị hacker xâm phạm.

4.1.3.2. Mã hóa thư gửi đi

Đây là loại mã hóa dữ liệu trước khi bạn truyền lên mạng. Hiện nay, có rất nhiều phần mềm hỗ trợ việc mã hóa email như PGP...Chỉ cần bên gửi và bên nhận trao đổi và thực hiện vài công việc thì dữ liệu giao tiếp đã được bảo mật hơn.

4.1.3.3. Mã hóa lưu trữ

Đây là loại mã hóa tất cả các thông tin đã được lưu trữ. Loai mã hóa này rất cần thiết, cho rằng dữ liệu của bạn được truyền đi được bảo mật hoàn toàn nhưng khi bạn lưu trữ nó bạn không dùng một biện pháp nào. Lỡ như ai đột nhập vào máy tính cá nhân của bạn hay bạn mất thiết bị cá nhân thì việc người khác lấy được mọi thông tin của bạn gần như là dễ dàng.

4.1.4. Các nguy cơ khi sử dụng, biện pháp và bảo vệ Email

4.1.4.1. Các nguy cơ khi sử dụng Email

- Spam: Là dạng các tin nhắn quảng cáo hay tin nhắn rác quấy rầy trong hộp thư của chúng ta. Các tin nhắn này sẽ gửi đến với số lượng nhiều gây phiền toái cho chúng ta, đặc biệt nó rất

dễ cho chúng ta sự nhầm lẫn giữa tin nhắn rác và tin nhắn quan trọng.

- Bom thư: Là dạng thư có vi rút xâm nhập. Đây là nguy cơ các thiết bị cá nhân của chúng ta bị vi rút tấn công.
- Mất tài khoản: Là dạng tài khoản cá nhân bị người khác đột nhập và sử dụng. Tài khoản có thể không sử dụng được nữa hoặc chúng ta không thể vào bằng mật khẩu cũ.
- Thông tin bị đánh cắp: Là dạng thông tin trong thư của bạn bị người thứ ba biết và sử dụng. Điều này để lại hậu quả hết sức nguy hiểm.

Trên thực tế còn rất nhiều nguy cơ khác như bị lừa qua Email...

4.1.4.2. Các biện pháp và bảo vệ Email

- Sử dụng mật khẩu dài và khó: Hãy giúp email của mình an toàn hơn bằng cách đặt mật khẩu dài và khó. Tránh tình trạng lấy ngày sinh, số điện thoại... rất dễ để các hacker xâm nhập.
- Sử dụng nhiều Email: Đây là phương pháp nhằm đánh lạc hướng cho các hacker. Họ sẽ không biết email nào là email bạn thực sự dùng.
- Không chia sẻ thông tin email một cách tùy ý: Hiện nay, các ứng dụng thật và ảo đang tràn lan. Vì vậy khi chia sẻ thông tin bạn nên cẩn thận đọc kĩ để không bị đánh cắp các thông tin email.
- Không mở file khi không biết người gửi là ai: Dù đây là phương pháp bất lợi nhưng để an toàn bạn cần phải phòng tránh.
- Thường xuyên duyệt vi rút cho thiết bị cá nhân: Các phần mềm duyệt vi rút sẽ giúp bạn thông báo nếu có vi rút ở các đường link bạn kết nối.

- Hạn chế kết nối và sử dụng Wifi cộng đồng : Dễ bị đánh cắp thông tin nếu bạn thường xuyên giao tiếp, giao dịch bằng wiffi cộng đồng

Hiện nay, trên thực thế có rất nhiều biện pháp và phần mềm để bảo vệ Email.

4.2. Ứng dụng mã hóa email PGP trên Thunderbird

4.2.1. Giới thiệu về mã hóa Email PGP trên Thunderbird

Mã hóa Email PGP trên thunderbird là một công cụ rất tiện ích và được sử dụng nhiều nhất cho ai muốn bảo mật Email.

Nó sử dụng thuật toán mã hóa đối xứng để mã hóa và giải mã.

Nó cho phép chúng ta thu hồi các khóa khi không muốn dùng nữa.

4.2.2. Các yêu cầu để gửi -nhận Email đã mã hóa trên Thunderbird

Giả sử bên gửi (V) và bên nhận (T) muốn giao tiếp với nhau.

Để V và T giao tiếp thành công, việc đầu tiên, cả V và T đều phải cài đặt đầy đủ các phần mềm: Thunderbird, GnuPG, Enigmail.

Để V có thể gửi một thông tin đã được mã hóa cho T cần:

- T tạo ra hai khóa là khóa công khai và khóa bí mật và phải gửi khóa công khai này cho V để V mã hóa thông tin
- V lưu và import khóa công khai của T để tiến hành mã hóa thông tin.
- Soạn tin nhắn muốn gửi và đánh đấu mã hóa.

Để T có thể giải mã thông tin vừa được V gửi cần:

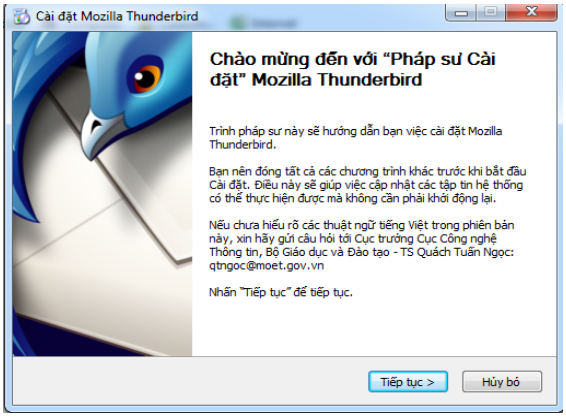
- T sẽ lấy khóa bí mật của mình vừa tạo để giải mã bằng cách nhập mật khẩu khi đã tạo thành công hai khóa. Đây là mật khẩu để bảo vệ khóa của T. Sau khi T nhập mật khẩu đúng , thông tin sẽ được giải mã lại như ban đầu.

4.2.3. Cài đặt một số phần mềm

4.2.3.1. Cài đặt phần mềm Mail Client Thunderbird

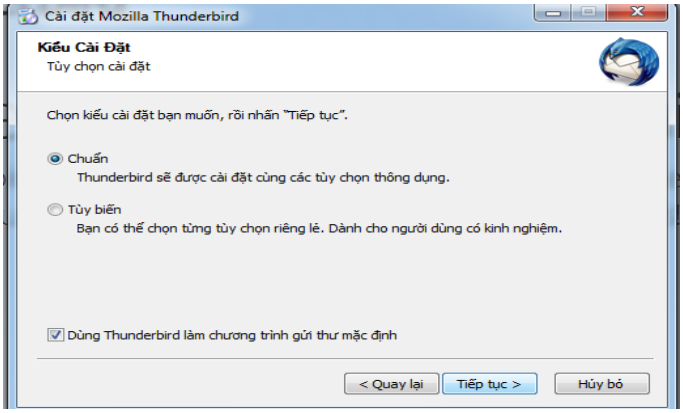
Thunderbird là phần mềm cho phép bạn lưu và trao đổi trên nhiều tài khoản với nhiều nhà cung cấp dịch vụ khác nhau.

Là phần mềm dễ sử dụng và được tích hợp nhiều chức năng giúp bảo mật Email an toàn hơn.



Hình 4. 1: Giao diện cài đặt Thunderbird

Chọn [Tiếp tục] để tiếp tục cài đặt.



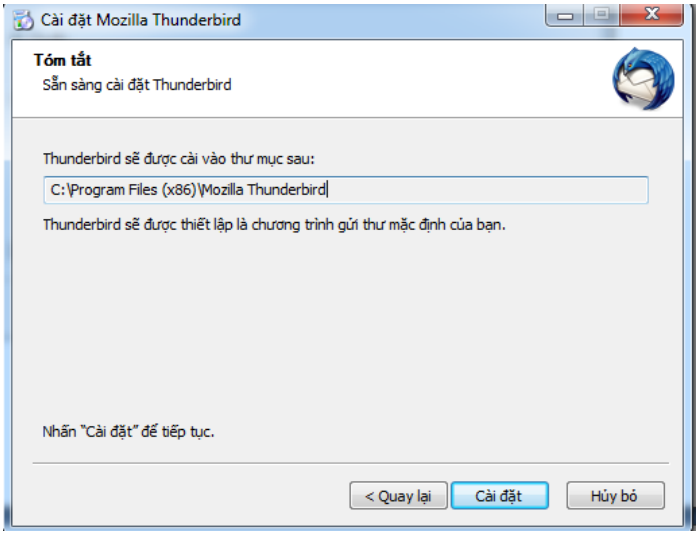
Hình 4. 2: Giao diện cài đặt Thunderbird

Màn hình này có hai sự lựa chọn:

- Chuẩn: Là mặc định của thunderbird.
- Tùy biến: Có thể cài đặt với những tham số tùy ý: Có tạo icon trên màn hình desktop hay không ...

Bạn có thể chọn một trong hai lựa chọn . Thông thường người ta sẽ chọn [Chuẩn]

Sau đó, chọn [Tiếp tục] để tiếp tục cài đặt.



Hình 4. 3: Giao diện cài đặt Thunderbird

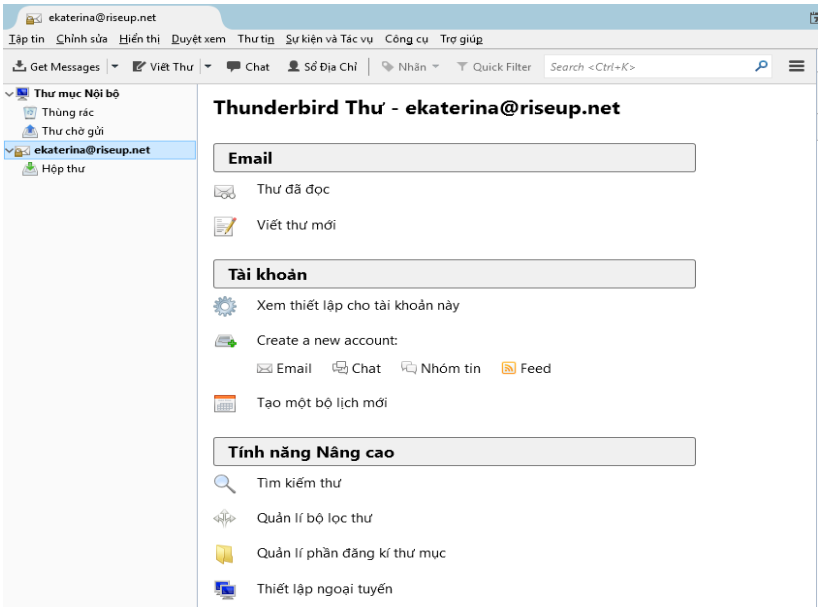
Ở trên là đường link của Thunderbird sau khi cài đặt xong.

Đường link trên màn hình là mặc định. Chúng ta có thể thay đổi nếu muốn.

Sau đó, chọn [Cài đặt] để tiếp tục cài đặt.

Hình 4. 4: Giao diện cài đặt Thunderbird

Chọn [Hoàn thành] để kết thúc quá trình cài đặt.



Hình 4. 5: Giao diện cài đặt Thunderbird thành công

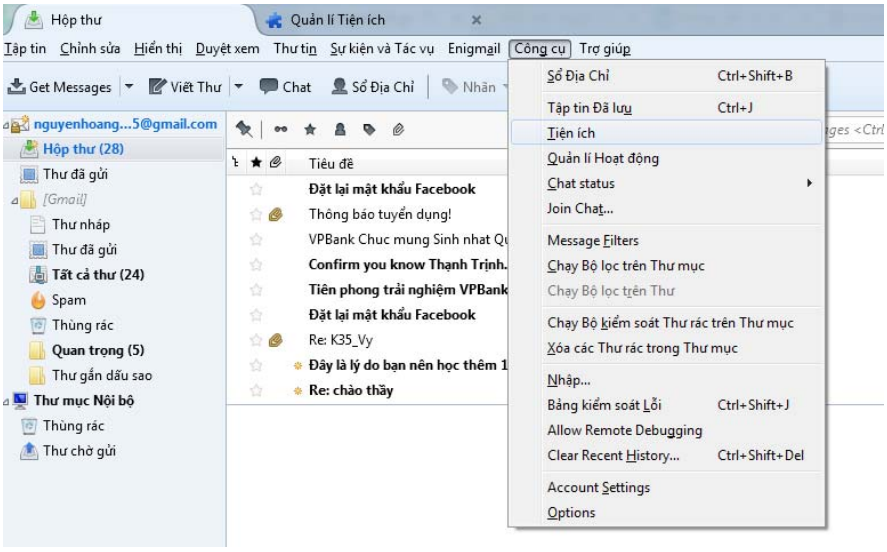
Sau khi cài đặt xong, Thunderbird sẽ có giao diện như hình bên trên.

4.2.3.2. Cài đăt Enigmail

Enigmail là giao diện để chúng ta có thể sử dụng chương trình mã hóa email GnuPG, nó giúp truy cập đến các chức năng mã hóa do GnuPG cung cấp.

Để tiến hành cài đặt thành công cần thực hiện qua 3 bước sau:

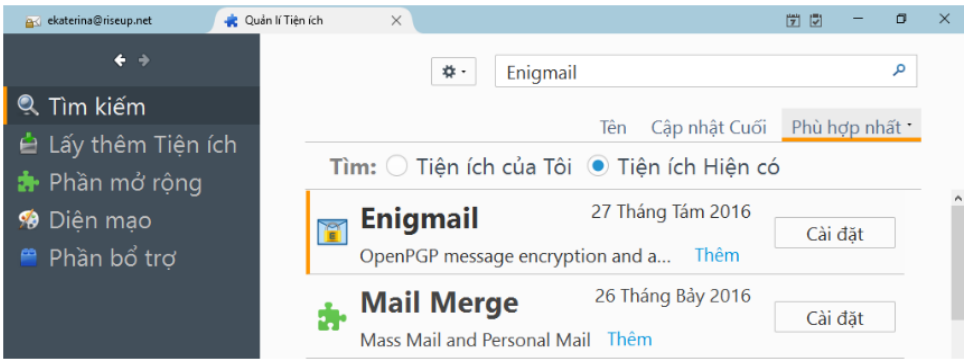
Bước 1: Mở tab quản lí tiện ích



Hình 4. 6: Giao diện cài đặt Enigmail

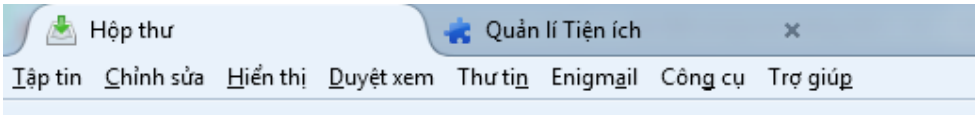
Chọn [Công cụ] → chọn [Tiện ích].

Bước 2: Tìm và cài đặt Enigmail



Hình 4. 7: Giao diện cài đặt Enigmail

- Gõ 'Enigmail' vào ô tìm kiếm và enter.
- Chọn [Cài đặt] cho phần mềm hiển thị đầu tiên.



Hình 4. 8: Giao diện cài đặt Enigmail thành công

Sau khi cài đặt xong, hệ thống sẽ bắt buộc bạn restart máy tính.

Sau khi mở thunderbird , thanh công cụ có Enigmail như trên nghĩa là chúng ta đã cài đặt thành công.

4.2.3.3. Cài đặt GnuPG

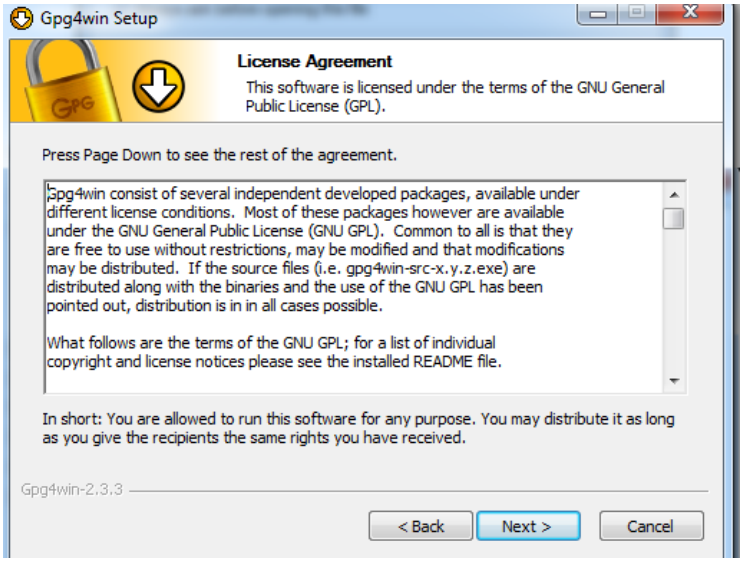
GnuPG là phần mềm mã nguồn mở và miễn phí cho phép mã hóa Email.

Nó còn giúp chúng ta tạo ra khóa công khai và khóa bí mật phục vụ cho việc mã hóa Email, bên cạnh đó còn giúp chúng ta lấy lại khóa bằng cách tạo chứng chỉ thu hồi.

Chọn [Next] để tiếp tục

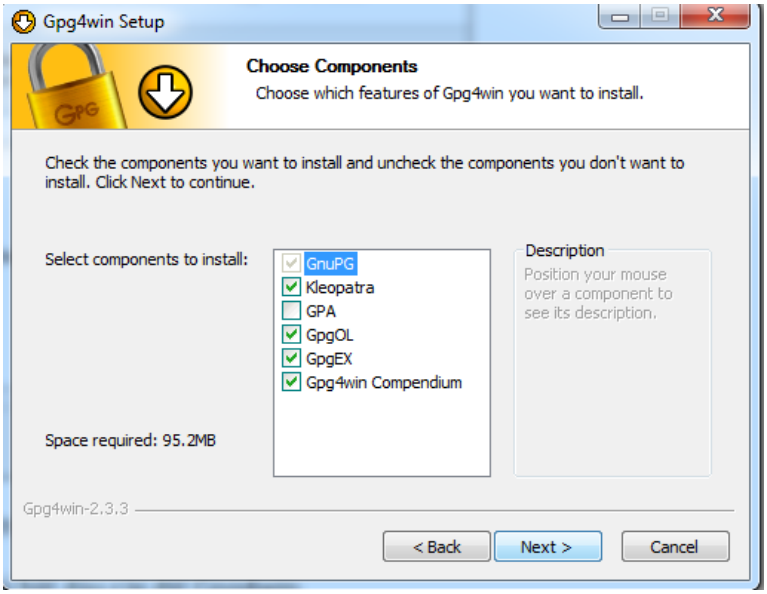
Hình 4. 9: Giao diện cài đặt GnuPG

Chọn [Next] để tiếp tục.



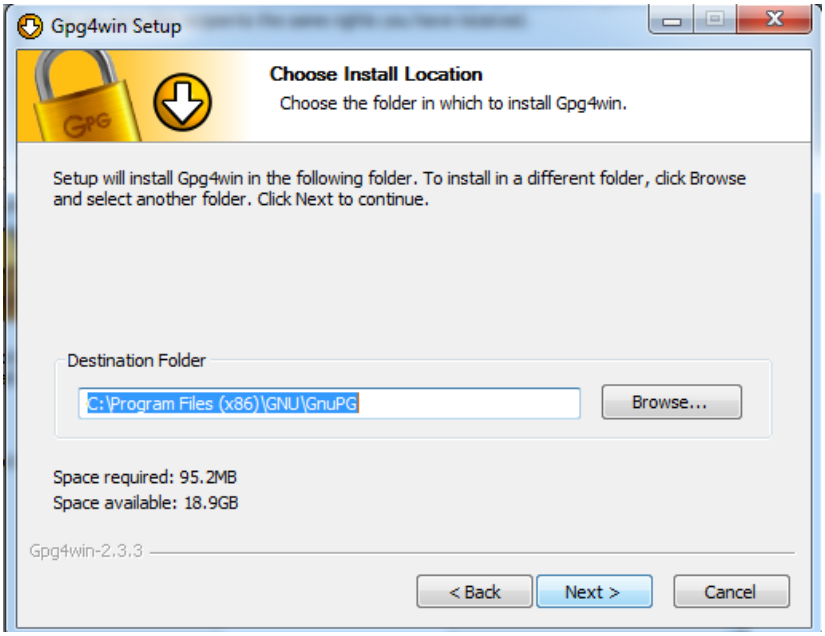
Hình 4. 10: Giao diện cài đặt Enigmail

Chọn [Next] để tiếp tục.

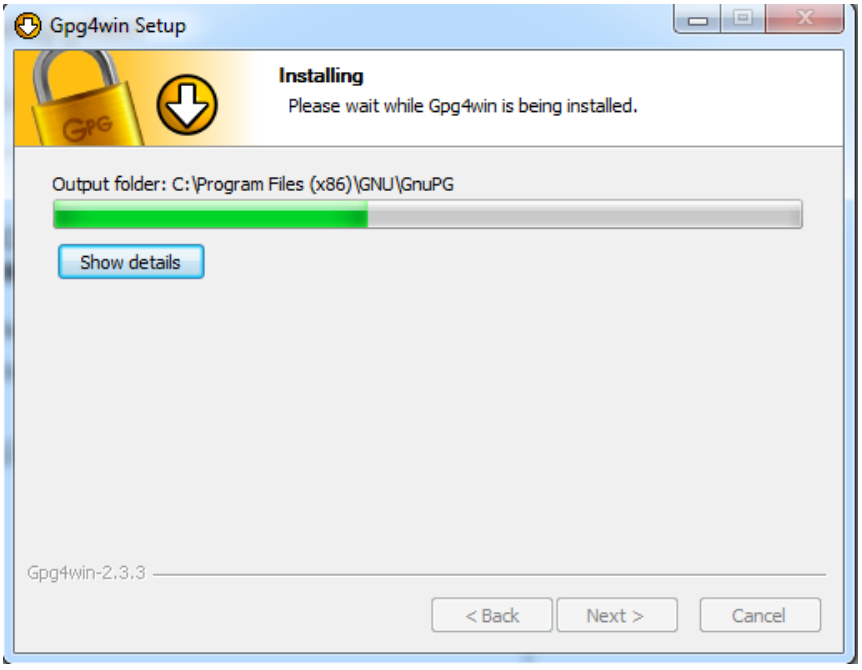


Hình 4. 11: Giao diện cài đặt GnuPG

Chọn [Next] để tiếp tục.



Hình 4. 12: Giao diện cài đặt GnuPG



Hình 4. 13: Giao diện cài đặt GnuPG

Hình trên là giao diện đang cài đặt GnuPG.

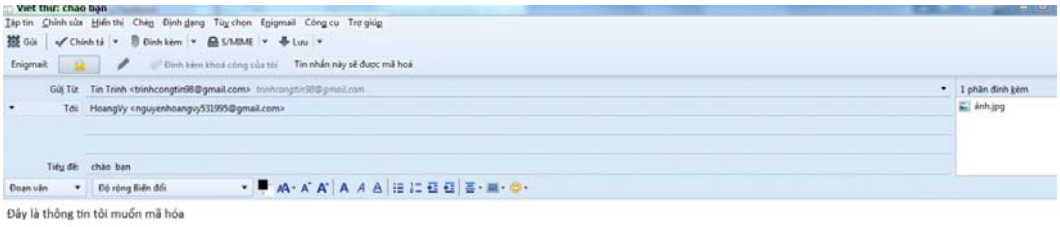
Chọn [Finish] để kết thúc cài đặt GnuPG.

Hình 4. 14: Giao diện cài đặt xong GnuPG

Hoàn thành xong quá trình cài đặt GnuPG

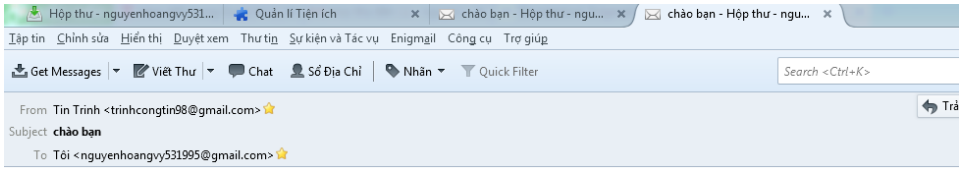
4.3. Demo mã hóa Email PGP trên Thunderbird

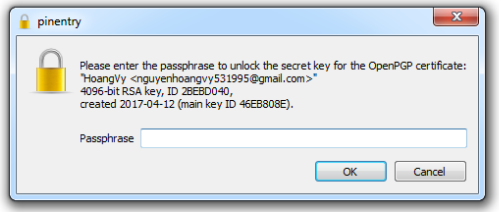
4.3.1. Giao diện gửi thư



Hình 4. 15: Giao diện gửi thư

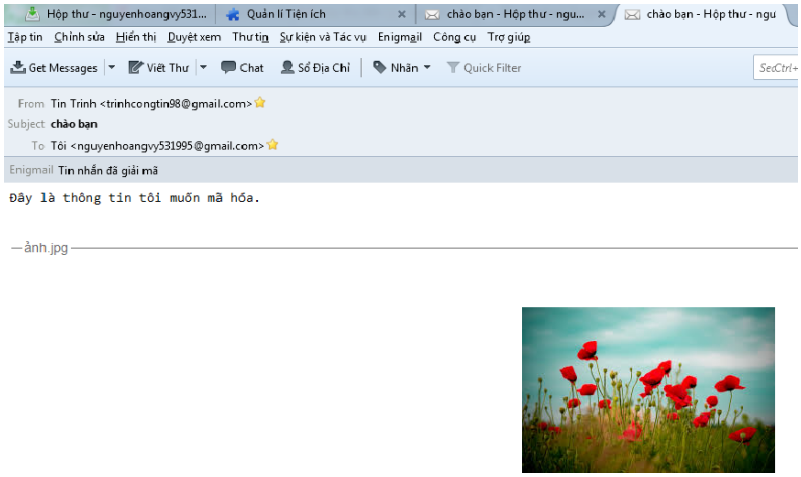
4.3.2. Giao diện nhận và chưa giải mã thư





Hình 4. 16: Giao diện nhận và chưa giải mã thư

4.3.3. Giao diện nhận và giải mã thư-



Hình 4. 17: Giao diện nhận và giải mã thư

4.4. Kết luận chương 4

Tìm hiểu và cài đặt thành công phần mềm Thunderbird, GnuPG vàcác chương trình hỗ trợ mã hóa email.

Hiểu được cách hoạt động của phần mềm mã hoá Email PGP trên Thunderbird

Đưa ra được chương trình mô phỏng mã hóa email.

- Người gửi gửi tin nhắn đã mã hóa thành công
- Người nhận nhận thư nhưng chưa giải mã
- Người nhận nhận và giải mã Email thành công

PHẦN 3. KẾT LUẬN VÀ KIẾN NGHỊ

1. Kết luận

1.1. Phần đạt được

Sau thời gian nghiên cứu, mặc dù đã có nhiều cố gắng, tìm hiểu kĩ các tài liệu đã học và kết hợp tìm các tài liệu chuyên ngành nhưng do còn hạn chế về thời gian, khả năng nên không thể tránh khỏi các thiếu sót nên đề tài hoàn thành ở mức độ sau:

- Về lý thuyết:
  - Trình bày một cách tổng quan về mã hóa dữ liệu, tìm hiểu tiêu chuẩn đánh giá hệ mật mã và các ứng dụng của mã hóa dữ liệu. Bên cạnh đó, đề tài còn nêu được một số thuật toán mã hóa dữ liệu cổ điển và công khai cũng như tìm các ưu nhược điểm của nó.
  - Trình bày tổng quát về Email, các giao thức được sử dụng trong email và các ưu điểm của Email so với gửi thư thông thường.
- Về phần demo
  - Cài đặt thành công phần mềm thunderbird, các chương trình hỗ trợ mã hóa và giải mã email như Enigmail, GnuPG.
  - Hiểu rõ được cách thức mã hóa Email PGP trên thunderbird.

1.2. Phần hạn chế

Do một số hạn chế nên đề tài còn nhiều thiếu sót về lĩnh vực đáng quan tâm như:

- Tìm hiểu còn chưa sâu về mã hóa dữ liệu, các thuật toán mã hóa dữ liệu và Email.
- Chưa tìm hiểu được tất cả các chức năng của PGP trên thunderbird, chỉ tập trung chức năng mã hóa Email.

2. Kiến nghị

Với nỗ lực của bản thân, em đã cố gắng hoàn thành đề tài này tốt nhất. Do thời gian và hạn chế về năng lực đề tài chỉ tập trung tìm hiểu và nghiên cứu. Thông qua đề tài này, em mong trường sẽ tao điều kiện hơn về mặt tài liệu và cơ sở thực tế về mã hóa dữ liệu để chúng em có cơ hội tìm hiểu sâu hơn.

PHẦN 4. TÀI LIỆU THAM KHẢO

[1]. Ths Hồ Văn Canh, NhẬp môn phân tích thông tin có bẢo mẬt(2011), NXB Thông Tin và Truyên Thông.

[2]. ThS Đặng Văn Sơn, Bảo mật thông tin(2013), NXB tổng hợp.

[3]. TS Nguyễn Khanh Văn, An toàn và bảo mật thông tin, NXB tổng hợp.

[4]. Ths Nguyễn Bình, Mật mã học(2008), NXB Bưu điện

[5]. Ths Trần Văn Sinh, An toàn và bảo mật máy tính (2013), NXB tổng hợp.

Và một số tài liệu liên quan trên Internet.

PHẦN 5. PHỤ LỤC

NHẬN XÉT CỦA GIẢNG VIÊN

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Giảng viên hướng dẫn	Sinh viên thực hiện
T.S Vũ Đức Quảng	Nguyễn Thị Hoàng Vy
Giảng viên chấm 1	Giảng viên chấm 2
ThS. Nguyễn Văn Khương	ThS. Dương Phương Hùng