TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CHUYÊN ĐỀ HỌC PHẦN**

**LẬP TRÌNH HỆ THỐNG**

**ĐỀ TÀI:**

**MÃ HÓA VÀ TRUYỀN NHẬN NHẬN DỮ LIỆU QUA SOCKET BẰNG THUẬT TOÁN MÃ HÓA CÔNG KHAI**

**RSA**

|  |  |
| --- | --- |
|  |  |
| |  |  | | --- | --- | | **Sinh viên thực hiện** | **: TRẦN ĐỨC SOẠN**  **NGÔ THẾ HẢI**  **ĐỖ TRỌNG DŨNG**  **KHỔNG NGỌC ÁNH**  **NGUYỄN TRƯỜNG THỊNH** | | **Giảng viên hướng dẫn** | **: NGÔ HOÀNG HUY** | | | **Ngành** | **: CÔNG NGHỆ THÔNG TIN** | | | **Chuyên ngành** | **: CÔNG NGHỆ PHẦN MỀM** | | | **Lớp** | **: D12CNPM2** | | | **Khóa** | **: 2017-2022** | | |  |

***Hà Nội, tháng 01 năm 2020***

**PHÂN CÔNG CÔNG VIỆC VÀ ĐÁNH GIÁ KẾT QUẢ**

|  |  |  |
| --- | --- | --- |
| **Họ Tên** | **Nhiệm Vụ** | **Kết Quả** |
| Trần Đức Soạn  1781310065 | Code đề tài, code bài tập bắt buộc | Hoàn thành |
| Khổng Ngọc Ánh  1781310009 | Code bài tập bắt buộc | Hoàn thành |
| Ngô Thế Hải  1781310029 | Code đề tài | Hoàn thành |
| Đỗ Trọng Dũng  1781310025 | Code bài tập bắt buộc | Hoàn thành |
| Nguyễn Trường Thịnh  1781310071 | Code bài tập bắt buộc | Hoàn thành |

**Giảng viên chấm điểm**:

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Chữ ký** | **Ghi chú** |
| Giảng viên chấm 1 : |  |  |
| Giảng viên chấm 2 : |  |  |

**MỤC LỤC**

[**LỜI CẢM ƠN** 1](#_Toc29488588)

[**LỜI NÓI ĐẦU** 2](#_Toc29488589)

[**CHƯƠNG 1: GIỚI THIỆU** 3](#_Toc29488590)

[**1.1** **Mã hóa công khai** 3](#_Toc29488591)

[**1.2** **Mã hóa đối xứng** 4](#_Toc29488592)

[**1.3** **Kết luận chương 1** 6](#_Toc29488593)

[**CHƯƠNG 2 : THUẬT TOÁN MÃ HÓA CÔNG KHAI RSA** 7](#_Toc29488594)

[**2.1 Giới thiệu về RSA** 7](#_Toc29488595)

[**2.2 Mô tả hoạt động của RSA** 8](#_Toc29488596)

[**2.2.1. Quá trình tạo khóa cho hệ mật RSA.** 9](#_Toc29488597)

[**2.2.2. Quá trình mã hóa:** 9](#_Toc29488598)

[**2.2.3. Quá trình giải mã***:* 10](#_Toc29488599)

[**2.2.4. Ví dụ:** 11](#_Toc29488600)

[**2.2.5. Chuyển đổi văn bản rõ** 11](#_Toc29488601)

[**2.2.6. Tạo chữ ký vào văn bản*.*** 13](#_Toc29488602)

[**2.4. Ứng dụng thực tế của RSA.** 15](#_Toc29488603)

[**2.4.2. SSL (Secure Socket Layer)** 16](#_Toc29488604)

[**2.3 Kết quả thực nghiệm** 17](#_Toc29488605)

[**CHƯƠNG III: KẾT LUẬN** 19](#_Toc29488606)

[**TÀI LIỆU THAM KHẢO** 20](#_Toc29488607)

**DANH MỤC HÌNH ẢNH**

***Hình 1.1:*** Mã hóa công khai 7

***Hình 1.2:*** Mã hóa đối xứng 8

***Hình 1.3:*** Mô hình mã hóa đối xứng 8

***Hình 2.1:*** Các tác giả Ronal Rivest, Adi Shamir và Leonard Adleman

tại Học Viện Công nghệ Masachusetts (MIT) vào năm 1977 12

**Hình 2.2:** Sơ đồ biểu diễn thuật toán mã RSA 14

***Hình 2.3:*** Mô hình chung của chữ ký điện tử 21

***Hình 2.4:*** Kết quả giải mã 23

***Hình 2.5:*** Kết quả mã hóa 23

# **LỜI NÓI ĐẦU**

Hiện nay, mạng điện thoại di động đã trở nên phổ biến trong đồi sống sinh hoạt của con người. Điều này đã đáp ứng được nhu cầu liên lạc và trao đổi thông tin thông qua mạng di động của con người. Mã hóa công khai áp dụng vào việc truyền dữ liệu trong di động. Chính vì vậy, nhóm chúng em đã chọn đề tài: Các thuật toán mã hóa công khai và áp dụng trong truyền nhận dữ liệu. để nêu rõ công dụng của mã hóa.

Báo cáo được chia thành các chương sau:

* Chương 1: Giới thiệu mã hóa công khai
* Chương 2: Tuật toán mã hóa công khai RSA
* Chương 3: Kết luận

Do điều kiện thời gian và khả năng có hạn nên báo cáo này không thể tránh khỏi nhầm lẫn và thiếu sót, chúng em rất mong sự đóng góp ý kiến của thầy để báo cáo ngày càng hoàn thiện hơn.

# **LỜI CẢM ƠN**

Qua thời gian học tập ở lớp, chúng em đã đạt được những kiến thức cơ bản về bộ môn lập trình hệ thống và hoàn thành đề tài: “Mã hóa và truyền nhận nhận dữ liệu qua socket bằng thuật toán mã hóa công khai RSA ”. Để đạt được kết quả này, chúng em đã nỗ lực thực hiện và đồng thời cũng nhận được rất nhiều sự giúp đỡ, quan tâm, ủng hộ của các thầy cô, bạn bè và gia đình.

Chúng em xin chân thành cảm ơn thầy giáo Ngô Hoàng Huy đã tận tình hướng dẫn, giúp đỡ chúng em trong quá trình học tập.

Vì thời gian có hạn nên không thể tránh khỏi những thiếu sót, chúng em rất mong được sự đóng góp ý kiến từ thầy cô và các bạn.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

Trần Đức Soạn

Khổng Ngọc Ánh

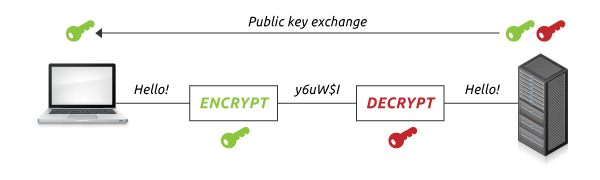
Ngô Thế Hải

Đỗ Trọng Dũng

Nguyễn Trường Thịnh

**CHƯƠNG 1: GIỚI THIỆU**

* 1. **Mã hóa công khai**
* Mã hóa là quá trình biến đổi một nội dũng đọc được sang nội dung không thể đọc được, đảm bảo khi lưu trữ hoặc truyền đi qua hệ thống truyền tin không bị lỗi bị lộ thông tin.
* Các cách thức mã hóa đều phải có một quy ước hay còn gọi là khóa để giải mã một tín hiệu, khi khóa bị lộ các thông tin mã hóa sẽ đọc được.
* Mã hóa công khai (Public-key cryptography) :
* Sử dụng hai khóa khác nhau để mã hóa và giải mã thông tin.
* Khóa dùng để giải mã hóa là khóa bí mật (Public key).
* Khóa còn lại để giải mã là khóa bí mật ( Private key ) .
* Nếu ai đó lấy được khóa công khai cũng không thể giải mã được nội dung nếu không có khóa bí mật .
* Tên gọi khác là mã hóa bất đối xứng (Asymmetric Encryption ) .

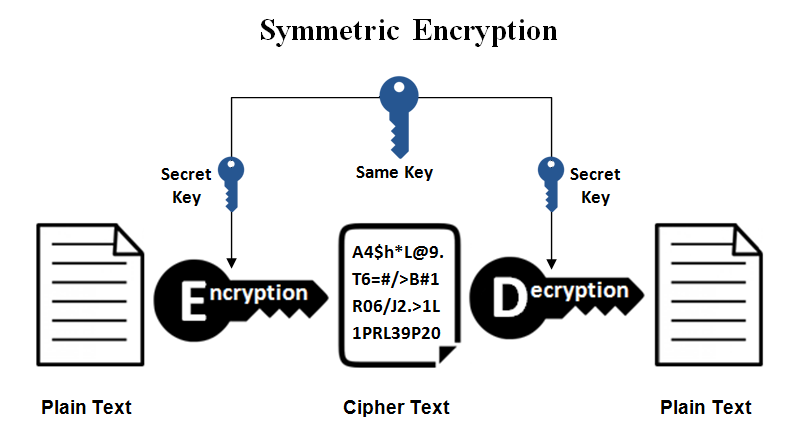


**Hình 1.1. Mã hóa công khai**

* Khóa công khai được công khai cho cả 2 đầu cuối của đường truyền . Để giải mã thì sử dụng khóa bí mật .
* Các loại mã hóa công khai:
* Trao đổi khóa Difie-hellman.
* DSS ( Tiêu chuẩn chứ ký số ).
* ElGamal.
* Các kỹ thuật Mã hóa đường cong elliptic.
* Hệ thống mã Paillier.
* Thuật toán mã hóa RSA.

Vào năm 1976 Whitefield Diffe và Martin Hellman đã tìm ra một phương pháp mã hoá khác mà có thể giải quyết được hai vấn đề trên, đó là mã hoá công khai(public key cryptograph) hay còn gọi là mã hoá bất đối xứng (asymertic cryptography). Đây có thể xem là một bước đột phá quan trọng nhất trong lĩnh vực mã hoá.

* 1. **Mã hóa đối xứng**

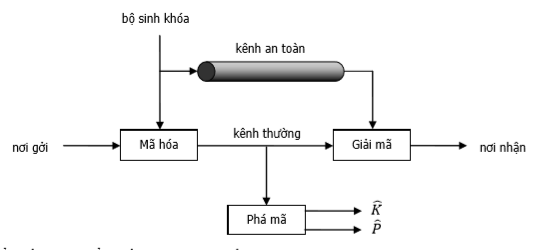
****

***Hình 1.2. Mã hóa đối xứng***

Bản mã C được gửi đi trên kênh truyền. Do bản mã C đã được biến đổi so với bản rõ P, cho nên những người thứ ba can thiệp vào kênh truyền để lấy được bản mã C, thì không hiểu dược ý nghĩa của bản mã.

Khóa phải được giữ bí mật giữa người dửi và người nhận, hay nói cách khác khóa phải được chuyển một cách an toàn từ người gửi đến người nhận.

* Xét mô hình mã hóa đối xứng:



***Hình 1.3. Mô hình mã hóa đối xứng***

Để khắc phục điểm yếu của mã hoá đối xứng người fta tập trung vào nghiên cứu theo hướng: có phương pháp nào để việc mã hoá và giải mã dùng 2 khoá khác nhau? Nghĩa là C = E(P, K1) và P = D(C,K2). Nếu thực hiện được như vậy thì chúng ta sẽ có 2 phương án áp dụng:

**Phương án 1:** Người nhận (Bob) giữ bí mật khoá K2 , còn khóa K1 thì công khai cho tất cả mọi người biết. Alice muốn gởi dữ liệu cho Bob thì dùng khóa K1 để mã hóa. Bob dùng K2 để giải mã. Ở đây Trudy cũng biết khóa K1, tuy nhiên không thể dùng chính K1 để giải mã mà phải dùng K2. Do đó chỉ có duy nhất Bob mới có thể giải mã được. Điều này bảo đảm tính bảo mật của quá trình truyền dữ liệu. Ưu điểm của phương án này là không cần phải truyền khóa K1 trên kênh an toàn.



**Phương án 2:** người gửi (Alice) giữ bí mật khóa K1, còn khóa K2 thì công khai cho tất cả mọi người biết. Alice muốn gởi dữ liệu cho Bob thì dùng khóa K1 để mã hóa. Bob dùng K2 để giải mã. Ở đây Trudy cũng biết khóa K2 nên Trudy cũng có thể giải mã được. Do đó phương án này không đảm bảo tính bảo mật. Tuy nhiên lại có tính chất quan trọng là đảm bảo tính chứng thực và tính không từ chối. Vì chỉ có duy nhất Alice biết được khóa K1, nên nếu Bob dùng K2 để giải mã ra bản tin, thì điều đó có nghĩa là Alice là người gửi bản mã. Nếu Trudy cũng có khóa K1 để gửi bản mã thì Alice sẽ bị quy trách nhiệm làm lộ khóa K1. Trong phương án này cũng không cần phải truyền K2 trên kênh an toàn.

Vì vậy nếu kết hợp phương án 1 và phương án 2, thì mô hình đề xuất của chúng ta khắc phục được các nhược điểm của mã hóa đối xứng.

Trong cả hai phương án, một khóa được giữ bí mật chỉ một người biết, còn khóa kia được công khai. Do đó mô hình mã hóa trên được gọi là mã hóa khóa công khai (hay mã hóa bất đối xứng). Để thuận tiện ta quy ước lại các ký hiệu như sau:

* Để tránh nhầm lẫn với khóa bí mật của mã đối xứng, khóa bí mật trong mô hình trên được gọi là khóa riêng (private key) và ký hiệu là KR.
* Khóa công khai (public key) được ký hiệu là KU.
* Bản rõ được ký hiệu là M, còn bản mã giữ nguyên ký hiệu là C

**Phương án 1** viết lại thành:

C = E(M, KU)

M = D(C, KR)

**Phương án 2** viết lại thành:

C = E(M, KR)

M = D(C, KU)

Vấn đề còn lại ở đây là liệu có tồn tại một mô hình mã hóa và giải mã dùng hai khóa khác nhau như vậy không? Dĩ nhiên là hai khóa KU và KR không thể nào hoàn toàn độc lập với nhau. Phải có một mối quan hệ nào đó giữa KU và KR thì mới có thể tiến hành giải mã hóa và giải mã được. Có nghĩa là KR = f(KU). Tuy nhiên một yêu cầu rất quan trọng là việc tính KR = f(KU) phải là bất khả thi về mặt thời gian. Nếu nguyên tắc này bị vi phạm thì việc giữ bí mật khóa KR không còn ý nghĩa vì từ khóa công khai KU có thể tính được KR.

Để có được cặp khóa KR và KU như trên, người ta thường dùng các hàm một chiều (oneway function). Các hàm một chiều có tính chất là hàm nghịch đảo của chúng rất khó thực hiện. Sau đây là ví dụ về hàm một chiều: việc sinh ra hai số nguyên tố lớn p, q và tính tích N = pq thì thực hiện dễ dàng. Tuy nhiên nếu chỉ cho trước N và thực hiện phân tích N để tìm lại hai số nguyên tố p, q là việc hoàn toàn bất khả thi về mặt thời gian. Chúng ta sẽ xem cách thức áp dụng hàm một chiều này để tạo khóa KR và KU trong phần mã hóa RSA.

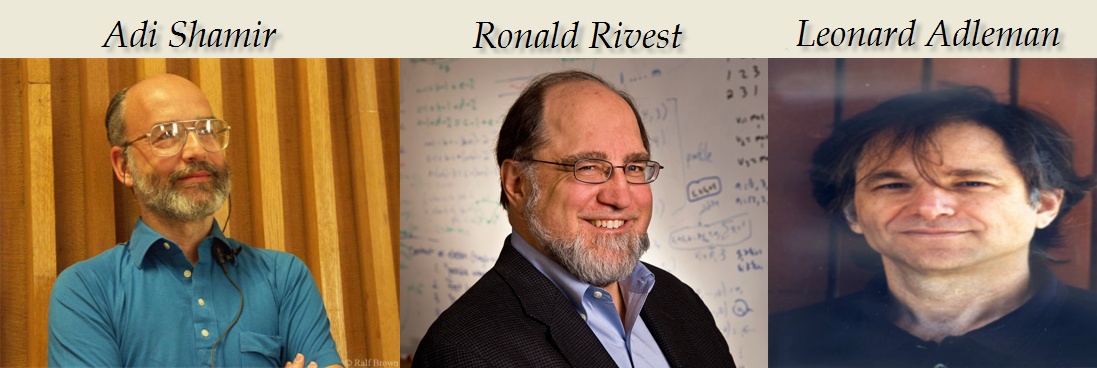
Có nhiều phương pháp mã hóa thuộc loại mã hóa khóa công khai. Đó là các phương pháp Knapsack, RSA, Elgaman, và phương pháp đường cong elliptic ECC…. Mỗi phương pháp có cách thức ứng dụng hàm một chiều khác nhau. Trong tài liệu này, chúng ta chỉ tập trung vào tìm hiểu phương pháp RSA. Bên cạnh đó, chúng ta cũng đề cập đến phương pháp trao đổi khóa Diffie-Hellman, một cách áp dụng hàm một chiều nhưng không phải để mã hóa. Tuy nhiên trước tiên chúng ta sẽ tìm hiểu sơ lược về lý thuyết số, đây là nền tảng toán học của phương pháp mã hóa khóa công khai.

* 1. **Kết luận chương 1**

Qua chương 1, chúng em dã tổng hợp được kiến thức về các loại mã hóa, đực biệt là mã hóa công và mã mã hóa đối xứng, kết hợp đó là các phương pháp thực hiện và chức năng của nó.

**CHƯƠNG 2 : THUẬT TOÁN MÃ HÓA CÔNG KHAI RSA**

1. **Giới thiệu về RSA**

Tiêu chuẩn Rivest-Shamir-Adleman (RSA) - Giải thuật mã hóa khóa công khai RSA là một tiêu chuẩn được các tác giả Ronal Rivest, Adi Shamir và Leonard Adleman phát triển tại Học Viện Công nghệ Masachusetts (MIT) vào năm 1977, tên tiêu chuẩn được lấy từ 3 chữ cái đầu của tên 3 tác giả, hiện tiêu chuẩn được các tổ chức Viện Tiêu chuẩn Quốc gia Hoa Kỳ (American National Standards Institute – ANSI), Viện Kỹ nghệ Điện và Điện tử (Institute of Electrical and Electronics Engineers – IEEE) và Phòng thí nghiệm RSA công nhận (RSA Laboratories là một bộ phận của Tập đoàn EMC). Đây là thuật toán đầu tiên phù hợp với việc tạo ra [chữ ký điện tử](http://vi.wikipedia.org/wiki/Ch%E1%BB%AF_k%C3%BD_%C4%91i%E1%BB%87n_t%E1%BB%AD) đồng thời với việc [mã hóa](http://vi.wikipedia.org/wiki/M%C3%A3_h%C3%B3a). Nó đánh dấu một sự tiến bộ vượt bậc của lĩnh vực [mật mã học](http://vi.wikipedia.org/wiki/M%E1%BA%ADt_m%C3%A3_h%E1%BB%8Dc) trong việc sử dụng khóa công cộng. RSA đang được sử dụng phổ biến trong [thương mại điện tử](http://vi.wikipedia.org/wiki/Th%C6%B0%C6%A1ng_m%E1%BA%A1i_%C4%91i%E1%BB%87n_t%E1%BB%AD) và được cho là đảm bảo an toàn với điều kiện độ dài [khóa](http://vi.wikipedia.org/wiki/Kh%C3%B3a_%28m%E1%BA%ADt_m%C3%A3%29) đủ lớn.

***Hình 2.1.* Các tác giả Ronal Rivest, Adi Shamir và Leonard Adleman tại Học Viện Công nghệ Masachusetts (MIT) vào năm 1977**

Trước đó vài năm, Clifford Cox, một chuyên gia mã hóa người Anh đã phát triển riêng một biến thể của RSA. Tuy nhiên, Chính phủ Anh xem đây là vấn đề mật và đã không công bố. Khi Rivest, Shamir và Adleman công bố RSA trong ấn phẩm Scientific American tháng 9/1977, Cơ quan An ninh quốc gia Hoa Kỳ (NSA) không đồng ý về việc phổ biến rộng rãi RSA và ra lệnh cấm, tuy nhiên lệnh cấm này không có cơ sở pháp lý.

# **Mô tả hoạt động của RSA**

* Định lý cơ sở:

*Định lý nhỏ của Fermat: Với p là một số nguyên tố khác 2 thì chia một số a lũy thừa p cho p sẽ có số dư chính bằng a: *

*Mở rộng ta có: *

*Với  là số nguyên tố cùng nhau với m và nhỏ hơn m*

- Thuật toán RSA có 2 khóa:

* Khóa công khai (Public Key)
* Khóa bí mật (Private Key)

- 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.

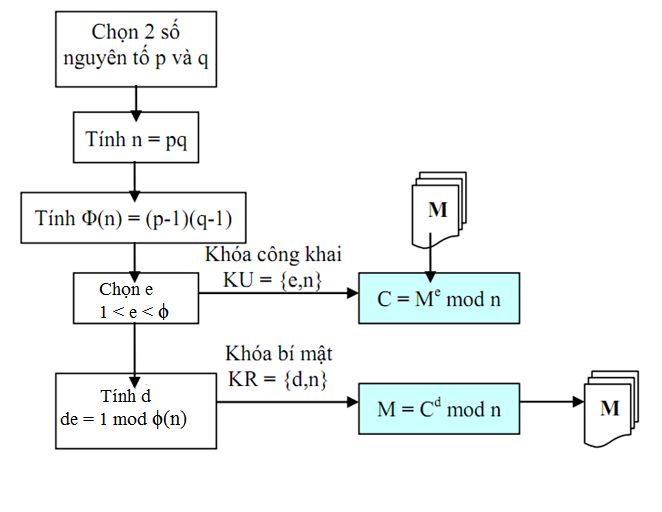
- Khóa bí mật được cá nhân giữ kín và dùng để giải mã

**Ví dụ:**

Bod muốn gửi cho Alice một thông tin mật M mà Bod muốn duy nhất Alice có thể đọc được.

**Bod**  **Alice**

( *Mã hóa M bằng Public Key của A đc C) (Giải mã C bằng Private Key của A đc M )*

**Hình 2.2: Sơ đồ biểu diễn thuật toán mã RSA**



# **2.2.1 Quá trình tạo khóa cho hệ mật RSA.**

Giả sử Alice và Bob cần trao đổi thông tin bí mật thông qua một kênh không an toàn (ví dụ như [Internet](http://vi.wikipedia.org/wiki/Internet)). Với thuật toán RSA, Alice đầu tiên cần tạo ra cho mình cặp khóa gồm khóa công khai và khóa bí mật theo 5 bước sau:

1. Chọn 2 số nguyên tố lớn khác nhau p, q thỏa mãn điều kiện 
2. Tính tích của nó 
3. Tính giá trị hàm Phi Euler của n: .
4. Chọn số nguyên d, sao cho và gcd(d,)=1.
5. Tính giá trị e thỏa mãn điều kiện:.

**Khóa công khai** bao gồm: n và e. **Khóa mật:** d còn p,q và thường là xóa sau khi tính toán khóa.



# **2.2.2 Quá trình mã hóa**

Giả sử Bob muốn gửi đoạn thông tin M cho Alice. Đầu tiên Bob chuyển M thành một số m < n theo một hàm có thể đảo ngược (từ m có thể xác định lại M) được thỏa thuận trước. Quá trình này được mô tả ở phần Chuyển đổi văn bản rõ.

Lúc này Bob có m và biết n cũng như e do Alice gửi. Bob sẽ tính c là bản mã hóa của m theo công thức:



Hàm trên có thể tính dễ dàng sử dụng phương pháp tính hàm mũ (môđun) bằng phương pháp bình phương. Cuối cùng Bob gửi C cho Alice.

**2.2.3 Quá trình giải mã**

Alice nhận *c* từ Bob và khóa bí mật *d*. Alice có thể tìm được m từ *c* theo công thức sau:



Biết m, Alice tìm lại M theo phương pháp đã thỏa thuận trước. Quá trình giải mã hoạt động vì ta có:



Do *ed* ≡ 1 (mod *p*-1) và *ed* ≡ 1 (mod *q*-1), theo [Định lý Fermat nhỏ](http://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BB%8Bnh_l%C3%BD_Fermat_nh%E1%BB%8F&action=edit) nên:



và



Do p và q là hai số nguyên tố cùng nhau, áp dụng định lý phần dư trung hoa, chúng ta có:



Hay



**2.2.4 Ví dụ**

Sau đây là một ví dụ với những số cụ thể. Ở đây chúng ta sử dụng những số nhỏ để tiện tính toán còn trong thực tế phải dùng các số có giá trị đủ lớn.

Lấy:

|  |  |
| --- | --- |
| *p* = 61 | — số nguyên tố thứ nhất (giữ bí mật hoặc hủy sau khi tạo khóa) |
| *q* = 53 | — số nguyên tố thứ hai (giữ bí mật hoặc hủy sau khi tạo khóa) |
| *n* = *pq* = 3233 | — môđun (công bố công khai) |
| *e* = 17 | — số mũ công khai |
| *d* = 2753 | — số mũ bí mật |

Khóa công khai là cặp (*e*, *n*). Khóa bí mật là *d*. Hàm mã hóa là:

encrypt(*m*) = *me* mod *n* = *m*17 mod 3233

với *m* là [văn bản rõ](http://vi.wikipedia.org/w/index.php?title=V%C4%83n_b%E1%BA%A3n_r%C3%B5&action=edit&redlink=1). Hàm giải mã là:

decrypt(*c*) = *cd* mod *n* = *c*2753 mod 3233

với *c* là [văn bản mã](http://vi.wikipedia.org/w/index.php?title=V%C4%83n_b%E1%BA%A3n_m%C3%A3&action=edit&redlink=1).

Để mã hóa văn bản có giá trị 123, ta thực hiện phép tính:

encrypt(123) = 12317 mod 3233 = 855

Để giải mã văn bản có giá trị 855, ta thực hiện phép tính:

decrypt(855) = 8552753 mod 3233 = 123

Cả hai phép tính trên đều có thê được thực hiện hiệu quả nhờ giải thuật bình phương và nhân

**2.2.5 Chuyển đổi văn bản rõ**

Trước khi thực hiện mã hóa, ta phải thực hiện việc chuyển đổi văn bản rõ ( chuyển đổi từ M sang m) sao cho không có giá trị nào của M tạo ra văn bản mã không an toàn.

Nếu không có quá trình này, RSA sẽ gặp phải một số vấn đề sau :

* Nếu m = 0 hoặc m = 1 sẽ tạo ra các bản mã có giá trị là 0 và 1 tương ứng
* Khi mã hóa với số mũ nhỏ (chẳng hạn e =3) và m cũng có giá trị nhỏ, giá trị me cũng nhận giá trị nhỏ (so với n). Như vậy phép môđun không có tác dụng và có thể dễ dàng tìm được m bằng cách khai căn bậc e của c ( bỏ qua môđun).
* RSA là phương pháp mã hóa xác định ( không có thành phần ngẫu nhiên) nên kẻ tấn công có thể thực hiện tấn công lựa chọn bản rõ bằng cách tạo ra một bảng tra giữa bản rõ và bản mã. Khi gặp một bản mã, kẻ tấn công sử dụng bảng tra để tìm ra bản rõ tương ứng.

Trên thực tế, ta thường gặp 2 vấn đề đầu khi gửi các bản tin ASCII ngắn với m là nhóm vài ký tự ASCII. Một đoạn tin chỉ có 1 ký tự NUL sẽ được gán giá trị m = 0 và cho ra bản mã là 0 bất kể giá trị của e và N. Tương tự, một ký tự ASCII khác, SOH, có giá trị 1 sẽ luôn cho ra bản mã là 1. Với các hệ thống dùng giá trị e nhỏ thì tất cả ký tự ASCII đều cho kết quả mã hóa không an toàn vì giá trị lớn nhất của m chỉ là 255 và 2553 nhỏ hơn giá trị n chấp nhận được. Những bản mã này sẽ dễ dàng bị phá mã.

Để tránh gặp phải những vấn đề trên, RSA trên thực tế thường bao gồm một hình thức chuyển đổi ngẫu nhiên hóa m trước khi mã hóa. Quá trình chuyển đổi này phải đảm bảo rằng m không rơi vào các giá trị không an toàn. Sau khi chuyển đổi, mỗi bản rõ khi mã hóa sẽ cho ra một trong số khả năng trong tập hợp bản mã. Điều này làm giảm tính khả thi của phương pháp tấn công lựa chọn bản rõ (một bản rõ sẽ có thê tương ứng với nhiều bản mã tùy thuộc vào cách chuyển đổi).

Một số tiêu chuẩn, chẳng hạn như PKCS, đã được thiết kế để chuyển đổi bản rõ trước khi mã hóa bằng RSA. Các phương pháp chuyển đổi này bổ xung thêm bit vào M. Các phương pháp chuyển đổi cần được thiết kế cẩn thận để tránh những dạng tấn công phức tạp tận dụng khả năng biết trước được cấu trúc của bản rõ. Phiên bản ban đầu của PKCS dùng một phương pháp đặc ứng (ad-hoc) mà về sau được biết là không an toàn trước tấn công lựa chọn bản rõ thích ứng (adaptive choén ciphertext attack). Các phương pháp chuyển đổi hiện đại sử dụng các kỹ thuật như chuyển đổi mã hóa bất dối xứng tối ưu như (Optimal Asymmetric Encryption Padding – OAEP) để chống lại tấn công dạng này. Tiêu chuẩn PKCS còn được bổ xung các tính năng khác để đảm bảo an toàn cho chữ ký RSA (Probabilistic Signature Scheme for RSA-RSA-PSS).

**2.2.6 Tạo chữ ký vào văn bản**

Thuật toán RSA còn được dùng để tạo chữ ký số cho văn bản.

Giả sử Alice muốn gửi cho Bob một văn bản có chữ ký của mình. Để làm việc này, Alice tạo ra một ***giá trị băm* (**hask value) của văn bản cần ký và tính giá trị mũ **d mod n** của nó (giống như khi Alice thực hiện giải mã). Giá trị cuối cùng chính là chữ ký điện tử của văn bản đang xét.

Khi Bob nhận được văn bản cùng với chữ ký điện tử, anh ta tính giá trị mũ **e mod n** của chữ ký đồng thời với việc tính giá trị băm của văn bản. Nếu 2 giá trị này như nhau thì Bob biết rằng người tạo ra chữ ký biết khóa bí mật của Alice và văn bản đã không bị thay đổi sau khi ký.

Cần chú ý rằng các phương pháp chuyển đổi bản rõ ( như RSA-PSS) giữ vai trò quan trọng đối với quá trình mã hóa cũng như chữ ký điện tử và không được dùng chung cho đồng thời cho cả hai mục đích trên.

**2.3 Các vấn đề của RSA**

Sau đây ta sẽ xem xét một số các tấn công phương pháp RSA.

* Vét cạn khóa: cách tấn công này thử tất cả các khóa d có thể có để tìm ra bản giải mã có ý nghĩa, tương tự như cách thử khóa K của mã hóa đối xứng. Với N lớn, việc tấn công là bất khả thi.
* Phân tích N thành thừa số nguyên tố N = pq: Chúng ta đã nói rằng việc phân tích phải là bất khả thi thì mới là hàm một chiều, là nguyên tắc hoạt động của RSA. Tuy nhiên, nhiều thuật toán phân tích mới đã được đề xuất, cùng với tốc độ xử lý của máy tính ngày càng nhanh, đã làm cho việc phân tích N không còn quá khó khăn như trước đây. Năm 1977, các tác giả của RSA đã treo giải thưởng cho ai phá được RSA có kích thước của N vào khoảng 428 bít, tức 129 chữ số. Các tác giả này ước đoán phải mất 40 nghìn triệu triệu năm mới có thể giải được. Tuy nhiên vào năm 1994, câu đố này đã được giải chỉ trong vòng 8 tháng. Bảng sau liệt kê kích thước N của các RSA đã phá mã được cho đến hiện nay.

|  |  |  |  |
| --- | --- | --- | --- |
| **Số chữ số của N** | **Số bit** | **Năm phá mã** | **Thuật toán** |
| 100 | 322 | 1991 | Quadratic sieve |
| 110 | 365 | 1992 | Quadratic sieve |
| 120 | 398 | 1993 | Quadratic sieve |
| 129 | 428 | 1994 | Quadratic sieve |
| 130 | 431 | 1996 | GNFS |
| 140 | 465 | 1999 | GNFS |
| 155 | 512 | 1999 | GNFS |
| 160 | 530 | 2003 | Lattice sieve |
| 174 | 576 | 2003 | Lattice sieve |
| 200 | 633 | 2005 | Lattice sieve |

Dĩ nhiên là việc phá mã trên chỉ được thực hiện trong phòng thí nghiệm. Tuy nhiên người ta cho rằng kích thước của N phải khoảng 1024 bít (309 chữ số) thì mới bảo đảm an toàn thật sự.

* Đo thời gian: Đây là một phương pháp phá mã không dựa vào mặt toán học của thuật toán RSA, mà dựa vào một “hiệu ứng lề” sinh ra bởi quá trình giải mã RSA. Hiệu ứng lề đó là thời gian thực hiện giải mã. Giả sử người phá mã có thể đo được thời giải mã dùng thuật toán bình phương liên tiếp. Trong thuật toán bình phương liên tiếp, nếu một bít của d là 1 thì xảy ra hai phép modulo, nếu bít đó là 0 thì chỉ có một phép modulo, do đó thời gian thực hiện giải mã là khác nhau. Bằng một số phép thử chosen-plaintext, người phá mã có thể biết được các bít của d là 0 hay 1 và từ đó biết được d.

**2.4 Ứng dụng thực tế của RSA.**

Tiêu chuẩn RSA được ứng dụng rộng rãi trong nhiều lĩnh vực như chữ ký số, thương mại điện tử, bảo mật, xác thực… Trong [Thông tư số 01/2011/TT-BTTTT](http://mic.gov.vn/vbqppl/Lists/Vn%20bn%20QPPL/DispForm.aspx?ID=7734) ngày 04/01/2011 của Bộ trưởng Bộ Thông tin và Truyền thông Công bố Danh mục tiêu chuẩn kỹ thuật về ứng dụng công nghệ thông tin trong cơ quan nhà nước quy định **Khuyến nghị áp dụng** tiêu chuẩn RSA, là một trong những giải thuật mã hóa và được xếp vào nhóm **Tiêu chuẩn về an toàn thông tin**.

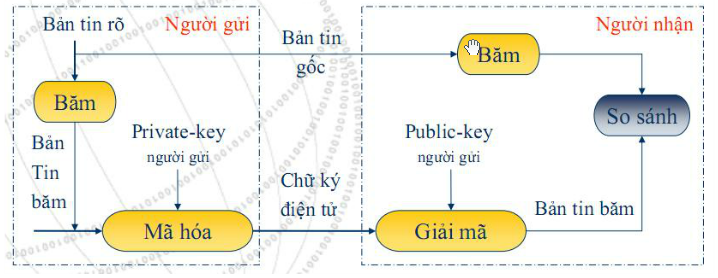
# **2.4.1 Chữ ký điện tử (Digital Signature)**

Chữ ký điện tử là chữ ký được tạo lập dưới dạng từ, chữ số, ký hiệu, âm thanh hoặc các hình thức khác bằng phương tiện điện tử, gắn liền hoặc kết hợp một cách logic với thông điệp dữ liệu. Chữ ký điện tử có giá trị xác nhận người ký thông điệp dữ liệu và xác nhận sự chấp thuận của người đó đối với nội dung thông điệp dữ liệu được ký.

Chữ ký điện tử là thông tin đi kèm theo dữ liệu (văn bản, hình ảnh , video ….) nhằm mục đích xác định người chủ của dữ liệu đó.

Chữ ký điện tử được tạo ra bằng cách áp dụng thuật toán băm một chiều trên văn bản gốc để tạo ra bản phân tích văn bản (**mesage digest**) hay còn gọi là **fingerprint**, sau đó mã hóa bằng private key tạo ra chữ ký số đính kèm với văn bản gốc được tính lại **fingerprint** để so sánh với **fingerprint** cũ cũng được phục hồi từ việc giải mã chữ ký số.

Mô hình chung của chữ kí điện tử :



***Hình 2.3:*** **Mô hình chung của chữ ký điện tử**

Đặc điểm của chữ ký điện tử rất đa dạng, có thể là một tên hoặc hình ảnh cá nhân kèm theo dữ liệu điện tử, một mã khóa bí mật, hay một dữ liệu sinh trắc học ( chẳng hạn như hình ảnh mặt, dấu vân tay, hình ảnh mống mắt … ) có khả năng xác thực người gửi. Độ an toàn của từng dạng là khác nhau.

* **Quy trình thực hiện chữ ký điện tử :**
* **Các bước mã hóa :**

Bước 1 : Dùng giải thuật băm để thay đổi thông điệp cần truyền đi, kết quả là ta được một message digest, dùng giải thuật md5 (message digest 5) ta được digest có chiều dài 128 bit, dùng giải thuật sha ( secure hash algorithm ) ta có chiều dài 160 bit.

Bước 2 :Sử dụng khóa private key của người gửi để mã hóa mesage 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 mesage ban đầu.

Bước 3 : Gộp digital signature vào message ban đầu, công việc này gọi là ‘’ ký nhận ’’ vào message. Sau khi đã ký nhận vào message, mọi sự thay đổi trên message sẽ bị phát hiện trong giai đoạn kiểm tra. Ngoài ra, việc ký nhận này đảm bảo người nhận tin tưởng message này xuất phát từ người gửi chứ không phải là ai khác.

* **Các bước kiểm tra :**

Bước 1 : Dùng public key của người gửi (khóa này được thông báo đến mọi người) để giải mã chữ ký số của message.

Bước 2 : Dùng giải thuật (md5 hoặc sha) hàm băm message đính kèm.

Bước 3 : So sánh kết quả thu được ở bước 1 và 2. Nếu trùng nhau, ta kết luận message này không bị thay đổi trong quá trình truyền và message này là của người gửi.

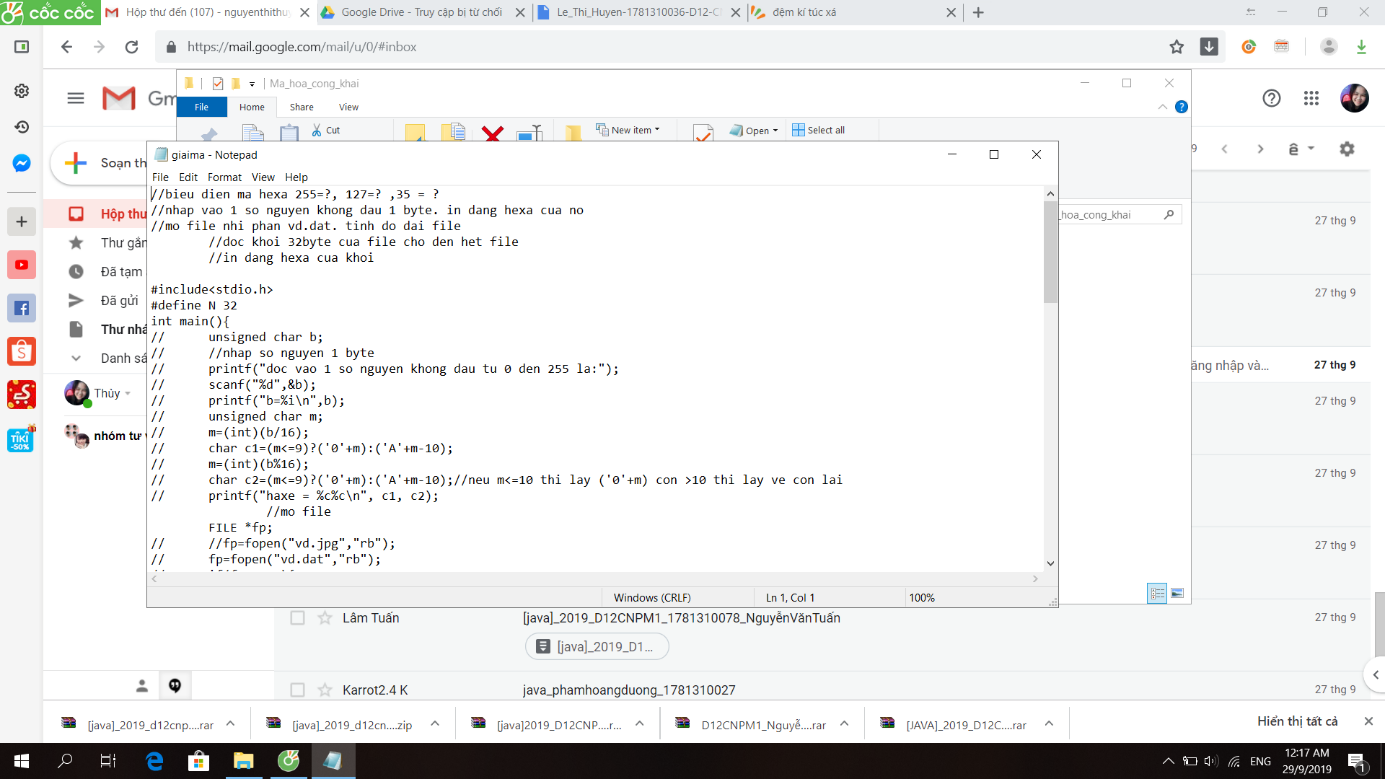
Mỗi cá nhân khi tham gia vào hệ thống chữ ký điện tử cần phải được cung cấp một bộ khóa (Public key, Private key) dùng để định danh cá nhân đó bởi một tổ chức cơ quan có thẩm quyền và được công nhận trong phạm vi sự dụng.

**2.4.2 SSL (Secure Socket Layer)**

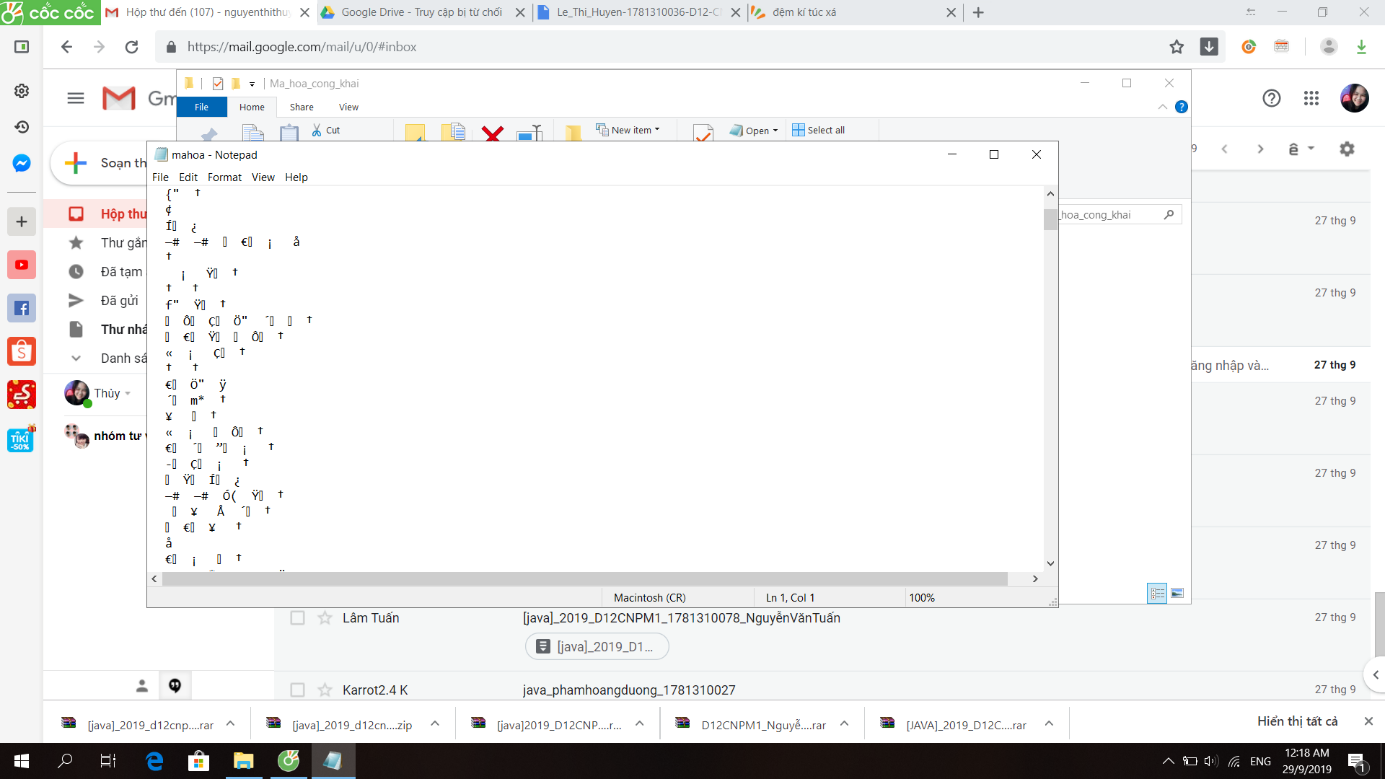
SSL là giao thức đa mục đích được thiết kế nhằm mã hóa toàn bộ thông tin đến/ đi giữa hai chương trình ứng dụng một cổng định trước (socket 443). Giao thức SSL được hình thành và phát triển đầu tiên năm 1994 bởi nhóm nghiên cứu Netscape và ngay nay trở thành chuẩn bảo mật thực hành trên mạng Internet.

Phiên bản hiện nay là SSL 3.0 và đang tiếp tục được bổ xung hoàn thiện.

**2.3 Kết quả thực nghiệm**

****

***Hình 2.4 Kết quả giải mã***

******

***Hình 2.5 Kết quả mã hóa***

**CHƯƠNG III:** **KẾT LUẬN**

Sau thời gian thực hiện đề tài môn học, chúng em đã đạt được nhiều tiến bộ cả về mặt tìm hiểu, nghiên cứu lý thuyết lẫn kỹ năng lập trình. Có thể nói, thông qua đồ án môn học, chúng em đã đạt được:

-Hiểu biết nhiều hơn về các kỹ thuật và kỹ xảo trong môn Lập Trình Hệ Thống

-Nâng cao tinh thần tự học, tự nghiên cứu.

-Hiểu hơn về mã hoá công khai, về cách hoạt động cũng như cách làm việc chính của mã hoá công khai.

-Ứng dụng: Có thể phân loại các ứng dụng cảu khoá công khai thành 3 loại khác nhau:

* Mã/giải mã – cung cấp bảo mật. Đây là ứng dụng bảo mật truyền thông như ta vẫn thường dùng với khoá đối xứng.
* Chữ kí điện tử - cung cấp xác thực. Một trong các ứng dụng mới của khoá công khai mà khoá đối xứng không thực hiện được, đó lá khoá công khai có đủ cơ sở để xác nhận người gửi và có thể là lựa chọn để tạo chữ ký điện tử của người gửi.
* Trao đổi khoá: Hai bên hợp tác để trao đổi session key. Có một số phương pháp tiếp cận khác nhau liên quan đên các khoá bí mật của một hoặc cả hai bên.

**TÀI LIỆU THAM KHẢO**

**Tài liệu tiếng Việt**

[1]. Bài giảng An ninh mạng viễn thông. TS. Nguyễn Phạm Anh Dũng – Học viện công nghệ BCVT.

[2]. Cơ sở lý thuyết mật mã. PGS. TS. Nguyễn Bình, TS. Trần Đình Sự. Học viện kỹ thuật mật mã. 2006

[3]. Thuật toán mã hóa và ứng dụng. TS. Dương Anh Đức, ThS. Trần Minh Triết. Đại học Khoa học tự nhiên – Đại học Quốc gia TP. HCM. 2005

[4]. Bài giảng An ninh mạng (Network security) . Học viện kỹ thuật mật mã. Phan Đình Diệu, Giáo trình lý thuyết mật mã và an toàn thông tin, Nhà xuất bản Đại học Quốc gia Hà Nội, 1999.

**Tài liệu Tiếng Anh**

[5]. RSA Cryptography Standard, October 27, 2012.

**Địa chỉ trên mạng INTERNET**

[7]. <http://aita.gov.vn>

[8]. <http://tiasang.com.vn/>

[9]. <http://antoanthongtin.vn/>