Chương 5

HỆ MÃ HÓA BẤT ĐỐI XỨNG

Giáo viên: Lê Quốc Anh

Nội dung

1. Mã hóa khóa công khai (Public-Key Cryptosystems)
2. Thuật toán RSA
3. Một số thuật toán mã hóa khóa công khai khác

Một số khuyết điểm của hệ mã đối xứng

- Vấn đề trao đổi khóa giữa người gửi và người nhận: Cần phải có một kênh an toàn để trao đổi khóa sao cho khóa phải được giữ bí mật chỉ có người gửi và người nhận biết. Điều này tỏ ra không hợp lý khi mà ngày nay, khối lượng thông tin luân chuyển trên khắp thế giới là rất lớn. Việc thiết lập một kênh an toàn như vậy sẽ tốn kém về mặt chi phí và chậm trễ về mặt thời gian.
- Tính bí mật của khóa: không có cơ sở quy trách nhiệm nếu khóa bị tiết lộ.
- Cần quá nhiều khóa cho nên việc quản lý khóa phức tạp (Trên môi trường mạng có N người dùng, thì cần N(N-1)/2)
- Không thể thiết lập được chữ ký điện tử

Ý tưởng

- Vào năm 1976 Whitfield Diffie và Martin Hellman đã tìm ra một phương pháp mã hóa khác mà có thể giải quyết được hai vấn đề trên, đó là mã hóa khóa công khai (public key cryptography) hay còn gọi là mã hóa bất đối xứng (asymetric cryptography).
- Whitfield Diffie và Martin Hellman đưa ra 2 phương án sau:

Ý tưởng

- **Phương án 1:** người nhận (Bob) giữ bí mật khóa *K*2, còn khóa *K*1 thì công khai cho tất cả mọi người biết.
- Alice muốn gởi dữ liệu cho Bob thì dùng khóa *K*1 để mã hóa. Bob dùng *K*2 để giải mã.
- Ở đây Trudy cũng biết khóa *K*1, tuy nhiên không thể dùng chính *K*1 để giải mã mà phải dùng *K*2. 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 *K*1 trên kênh an toàn.

Ý tưởng

- **Phương án 2:** người gửi (Alice) giữ bí mật khóa *K*1, còn khóa *K*2 thì công khai cho tất cả mọi người biết. Alice muốn gởi dữ liệu cho Bob thì dùng khóa *K*1 để mã hóa. Bob dùng *K*2 để giải mã.
- Ở đây Trudy cũng biết khóa *K*2 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 *K*1, nên nếu Bob dùng *K*2 để giải mã ra bản tin, thì điều đó có nghĩa là Alice là người gửi bản mã. Nếu Trudy cũng có khóa *K*1 để gửi bản mã thì Alice sẽ bị quy trách nhiệm làm lộ khóa *K*1.
- Trong phương án này cũng không cần phải truyền *K*2 trên kênh an toàn→Mã bất đối xứng kết hợp 2 phương án trên

Mã hóa công khai (Public-Key Cryptosystems)

- Mã bất đối xứng là một dạng của hệ thống mật mã mà trong đó mã hóa (encryption) và giải mã (decryption) được thực hiện bằng cách dùng hai khóa (Key) khác nhau
- Một là khóa công khai (Public key) và một là khóa bí mật (Private key).
- Nó cũng được gọi tên là

MÃ HÓA KHÓA CÔNG KHAI

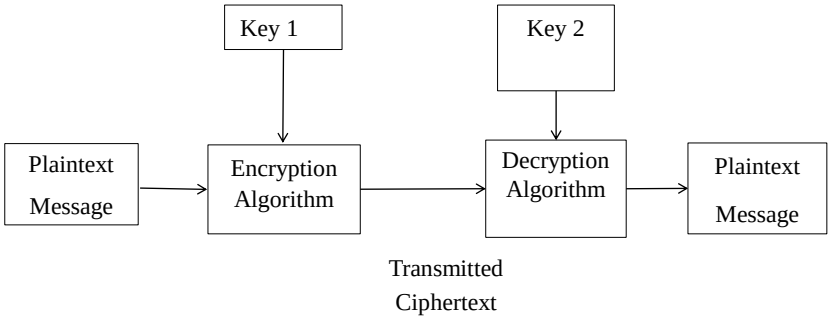
(Public-key Encryption)

Có hai mode làm việc :

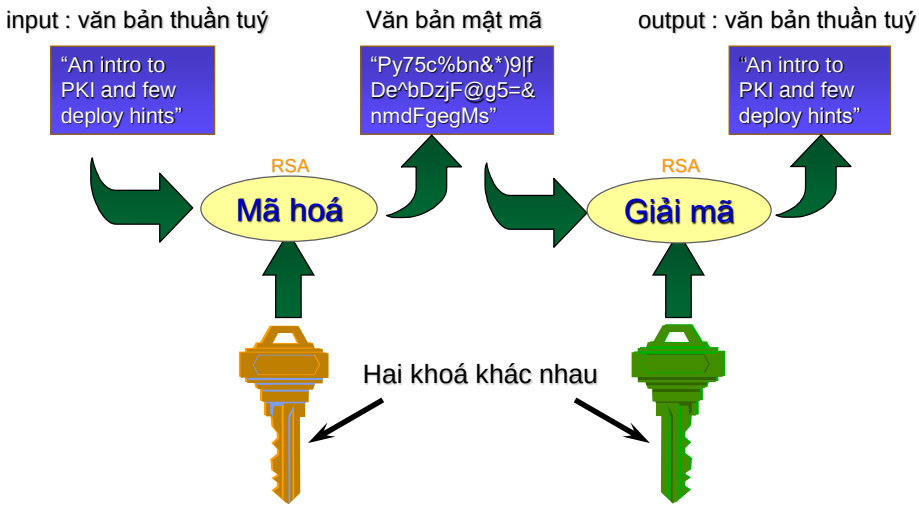
- Bảo mật : Mã bằng public key → giải mã bằng private key
- Xác thực : Mã bằng private key →giải mã bằng public key

Mã hoá bất đối xứng (asymmetric cipher model)

Sơ đồ mã hóa bất đối xứng



Mã hoá bất đối xứng



Mã hóa công khai (Public-Key Cryptosystems)

- Mã đối xứng có thể dùng để bảo mật (Confidentiality), chứng thực (Authentication), hoặc cả hai.
- Hiện nay, mã hóa khóa công khai được ứng dụng rộng rãi trong nhiều lĩnh vực, trong đó bao gồm: trao đổi, phân phối khóa, chữ ký số, bảo mật dữ liệu.
- Một số thuật toán mã hóa đối xứng: Diffie-Hellman, El-Gamal, RSA, ECC …

Mã hóa công khai (Public-Key Cryptosystems)

- Mã hóa khóa công khai được dùng rộng rãi nhất là mã RSA.
- Độ khó của việc tấn công được dựa vào độ khó của việc tìm thừa số nguyên tố (Prime factors) của một số composite number (hợp số).

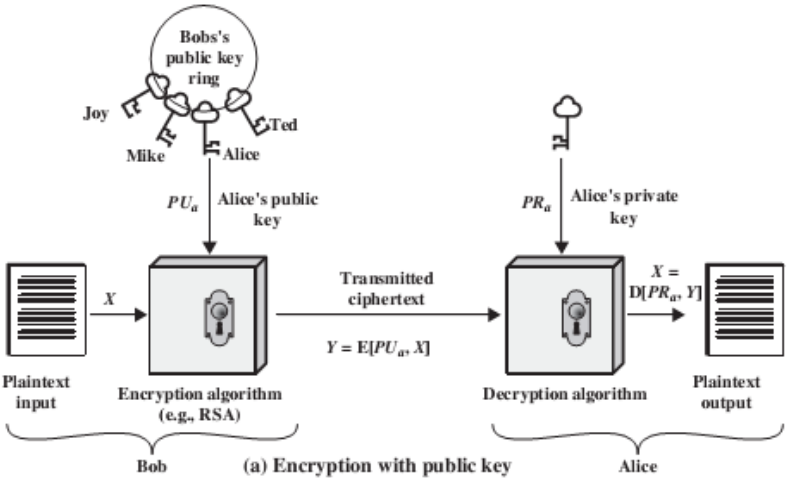
Mã hóa công khai (Public-Key Cryptosystems)

- Tên gọi:
  - **Mã hóa hóa công khai** (***Public-key Cryptosystems***)
  - **Mã hóa hai khóa** (***two-key Cryptosystems***)
  - **Mã hóa bất đối xứng** (***asymmetric Cryptosystems***)
- Hai khóa:
  - Một khóa **public-key**, có thể biết bất cứ ai, và có thể được dùng để mã hóa thông điệp.
  - Khóa **private-key**, chỉ được biết bởi người nhận, dùng để giải mã thông điệp
- Bất đối xứng là bởi vì:
  - Người mã hóa thông điệp không thể giải mã thông điệp do chính mình mã hóa
  - Người thẩm tra chữ ký không thể tạo ra chữ ký

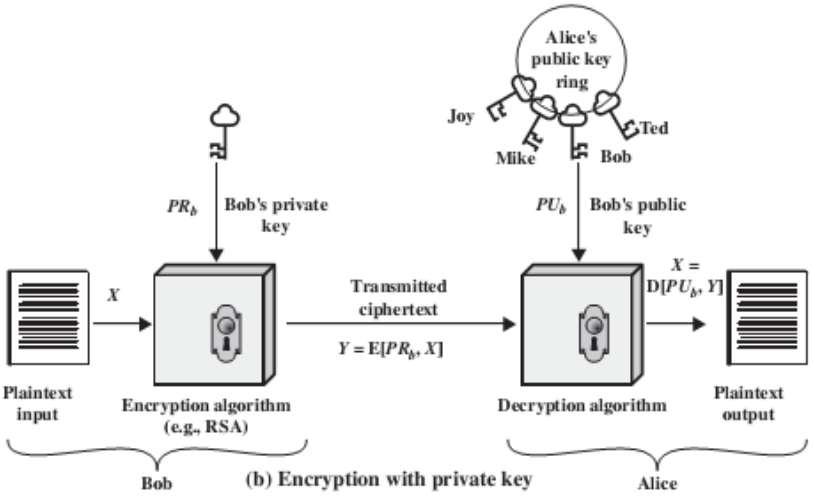
Mã hóa công khai (Public-Key Cryptosystems)

- **Giải thuật khóa công khai gồm các bước:**
- **Đầu vào của giải thuật:** Bản rõ (thông điệp có thể đọc)
- **Giải thuật sinh khóa**
- **Giải thuật mã hóa**
  - **Khóa công khai và bí mật:** một cặp khóa được chọn sao cho 1 khóa dùng để mã hóa và 1 khóa dùng để giải mã.
  - **Bản mã:** thông điệp đầu ra ở dạng không đọc được, phụ thuộc vào bản rõ và khóa. Nghĩa là với cùng một thông điệp, 2 khóa khác nhau sinh ra 2 bảng mã khác nhau
- **Giải thuật giải mã**

Public-key encryption scheme: Encryption



Public-key encryption scheme: Authentication



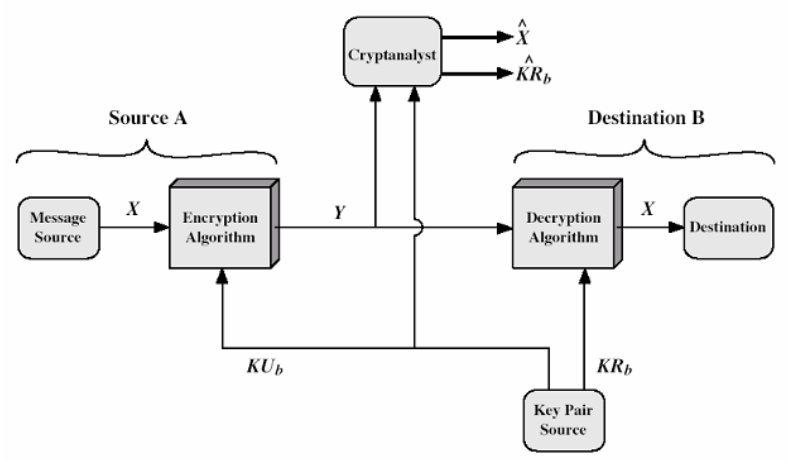
Đặc điểm Public-Key Cryptosystems

- Không thể tính toán để tìm khóa giải mã (decryption key) khi chỉ biết thuật toán và khóa mã hóa (encryption key)
- Một trong hai khóa có thể dùng cho việc mã hóa (encryption), Khóa còn lại dùng cho giải mã (đối với thuật toán RSA)

So sánh hệ mã đối xứng và bất đối xứng

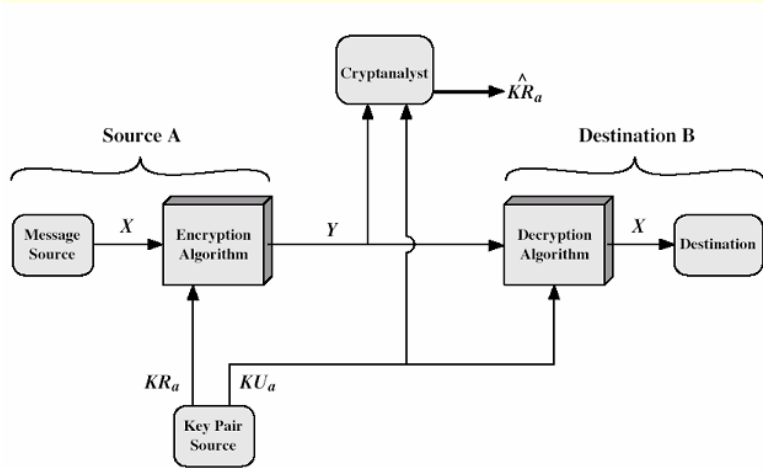
- **Symmetric-key Encryption**
  - Cùng thuật toán với cùng khóa được dùng cho việc mã hóa và giải mã
  - Sender và Receiver phải cùng chia sẻ thuật toán và khóa
  - Khóa phải giữ bí mật
  - Không thể hoặc ít nhất không thực thế để giải mã một thông điệp nếu những thông tin khác có sẵn.
  - Sự hiểu biết về thuật toán cộng với các mẫu ciphertext phải đủ thì mới xác định ra được khóa
- **Public-key Encryption**
  - Một thuật toán được dùng để mã hóa và giải mã với một cặp khóa, một khóa dành cho mã hóa và một dành do giải mã
  - Sender và receiver phải có một trong cặp khóa (không giống nhau)
  - Một trong hai khóa phải được giữ bí mật
  - Không thể hoặc ít nhất không thực thế để giải mã một thông điệp nếu những thông tin khác có sẵn.
  - Sự hiểu biết về thuật toán + một trong hai khóa + các mẫu ciphertext phải đủ thì mới có thể xác định được khóa còn lại.

Public-Key Cryptosystems: Secrecy

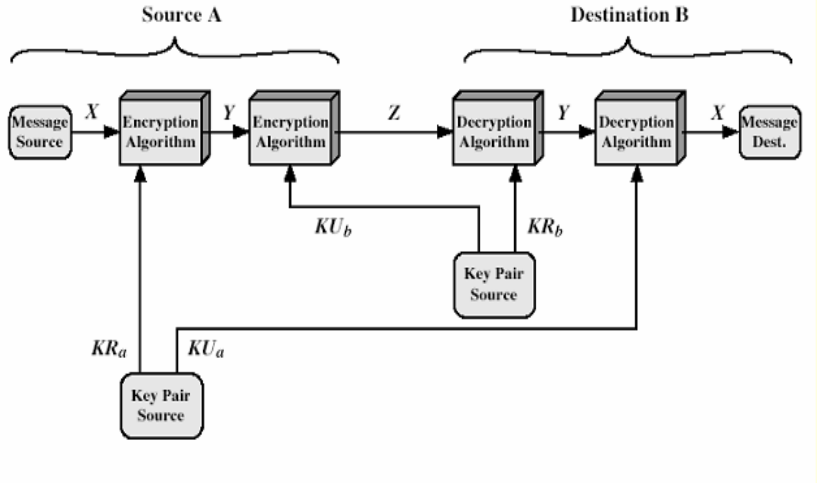


Public-Key Cryptosystems: Authentication

Thông điệp mã hóa được coi là một **digital signature**



Public-Key Cryptosystems: Secrecy and Authentication



Public-Key Application

- **Mã hóa/giải mã (*Encryption/decryption*):** Sender mã hóa thông điệp bằng khóa public key của người nhận.
- **Chữ ký số (*Digital signatures*)** – cung cấp chứng thực (authentication): Sender mã hóa thông điệp bằng khóa public key của người nhận. Chữ ký được lưu bằng một thuật toán áp đặt vào message hoặc gắn vào một khối nhỏ dữ liệu mà là một hàm của message
- **Trao đổi khóa (*Key exchange*)**: Hai bên hợp tác để trao đổi **khóa phiên (session key)**

Public-Key Application

- Một vài thuật toán thì phù hợp cho tất cả các ứng dụng, loại khác thì chỉ dành riêng cho một loại ứng dụng

Algorithm	Encryption/Decryption	Digital Signature	Key Exchange
RSA	Yes	Yes	Yes
Elliptic Curve	Yes	Yes	Yes
Diffie-Hellman	No	No	Yes
DSS	No	Yes	No

Phá mã Public-key

- Tấn công vét cạn (Brute Force attack): Luôn luôn là có thể về mặt lý thuyết
  - → Sử dụng khóa đủ lớn (>512 bits)
  - → khóa lớn ảnh hưởng đến tốc độ của việc mã hóa và giải mã
- Tìm Private key khi biết Public key:
  - →Chưa được chứng minh tính khả thi của phương pháp này

An ninh Public-Key Cryptosystems

- An toán của hệ mã hóa khóa công khai dựa trên độ khó của việc giải bài toán ngược.
- Tính bền của sự an toàn này còn phụ thuộc vào phương pháp tấn công của các thám mã

Ưu điểm mã hóa khóa công khai

- Đơn giản trong việc lưu chuyển khóa: Chỉ cần đăng ký một khóa công khai → mọi người sẽ lấy khóa này để trao đổi thông tin với người đăng ký → không cần thêm kênh bí mật truyền khóa.
- Mỗi người chỉ cần một cặp khóa (PR, PU) là có thể trao đổi thông tin với tất cả mọi người.
- Là tiền đề cho sự ra đời của chữ ký số và các phương pháp chứng thực điện tử.

Hạn chế của mã Public keys

- Tốc độ xử lý
  - Các giải thuật khóa công khai chủ yếu dùng các phép nhân chậm hơn nhiều so với các giải thuật đối xứng
  - Không thích hợp cho mã hóa thông thường
  - Thường dùng trao đổi khóa bí mật đầu phiên truyền tin
- Tính xác thực của khóa công khai
  - Bất cứ ai cũng có thể tạo ra một khóa công khai
  - Chừng nào việc giả mạo chưa bị phát hiện có thể đọc được nội dung các thông báo gửi cho người kia
  - Cần đảm bảo những người đăng ký khóa là đáng tin

2. Hệ mã hóa RSA

- Đề xuất bởi Rivest, Shamir & Adleman – MIT, 1977
- Là hệ mã hóa khóa công khai phổ biến nhất
- Là cơ chế mã hóa khối, plaintext và ciphertext là các số nguyên từ 0 đến n-1. Kích cỡ n thường là 1024 bits, hoặc 309 chữ số thập phân (nghĩa là n $<2^{1024}$)
- Dựa trên hàm mũ (exponentiation) trong trường hữu hạn (finite field)
- Bảo mật cao vì chi phí phân tích thừa số của một số nguyên lớn là rất lớn

Mã hóa và Giải mã RSA

Thuật toán mã hóa và giải mã RSA, RSA dùng phép lũy thừa modulo của lý thuyết số.

1. Chọn hai số nguyên tố lớn *p* và *q* và tính *N* = *pq*. Cần chọn *p* và *q* sao cho:

M < 2^{i-1} < *N* < 2^{i} . Với *i* = 1024 thì *N* là một số nguyên dài khoảng 309 chữ số.

2. Tính $\phi(n)$ = (*p* - 1)(*q* - 1)

3. Tìm một số *e* sao cho *e* nguyên tố cùng nhau với $\phi(n)$

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

4. Tìm một số *d* sao cho e.d≡1 mod $\phi(n)$ (*d* là nghịch đảo của *e* trong phép modulo *n*)

5. Hủy bỏ *n*, *p* và *q*. Chọn khóa công khai K_U là cặp (*e*, *N*), khóa riêng K_R là cặp (*d*, *N*)

Mã hóa và Giải mã RSA

6) Việc mã hóa thực hiện theo công thức:
   - Theo phương án 1, mã hóa bảo mật: $C = E(M, K_U) = M^e \ mod \ N$
   - Theo phương án 2, mã hóa chứng thực: $C = E(M, K_R) = M^d \ mod \ N$
7) Việc giải mã thực hiện theo công thức:
   - Theo phương án 1, mã hóa bảo mật: $\bar{M} = D(C, K_R) = C^d \ mod \ N$
   - Theo phương án 2, mã hóa chứng thực: $\bar{M} = D(C, K_U) = C^e \ mod \ N$

Bản rõ M có kích thước i-1 bít, bản mã C có kích thước i bít.

Thuật toán RSA sinh khóa

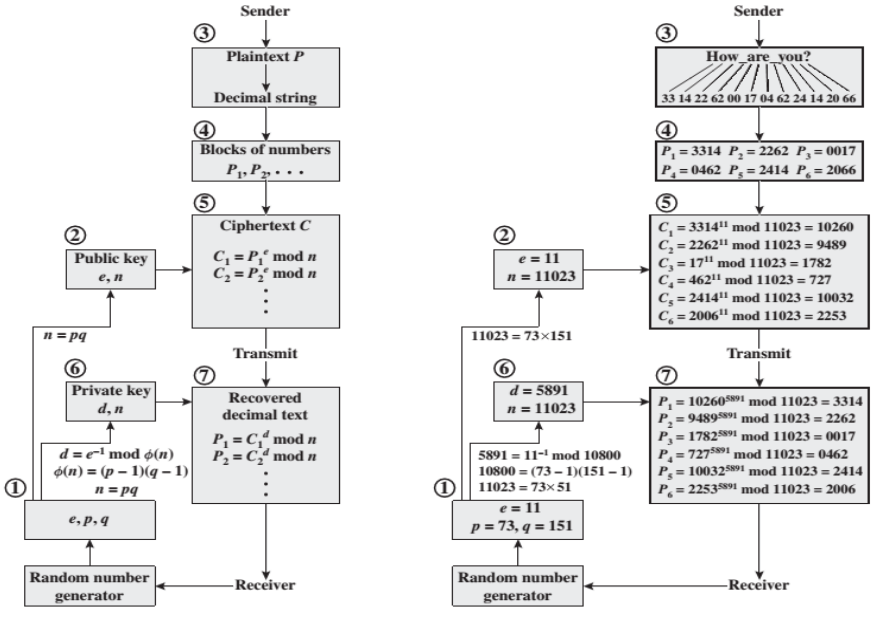
Key Generation Alice

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calcuate $\phi(n) = (p-1)(q-1)$	
Select integer e	gcd $(\phi(n), e) = 1; 1 < e < \phi(n)$
Calculate d	$d \equiv e^{-1} \pmod{\phi(n)}$
Public key	$PU = \{e, n\}$
Private key	$PR = \{d, n\}$

Thuật toán RSA Thực hiện RSA

Encryption by Bob with Alice's Public Key	
Plaintext:	$M < n$
Ciphertext:	$C = M^e \bmod n$

Decryption by Alice with Alice's Public Key	
Ciphertext:	C
Plaintext:	$M = C^d \bmod n$



Quá trình xử lý của RSA và ví dụ minh họa

Mã hóa và Giải mã RSA

- **Ví dụ RSA:** Để minh họa ta sẽ thực hiện một ví dụ về mã hóa RSA với kích thước khóa là 6 bít.

1. Chọn $p = 11$ và $q = 3$, do đó $N = pq = 33$ ($2^5 = 32 < 33 < 64 = 2^6$)
2. $\phi(n) = (p-1)(q-1) = 20$
3. Chọn $e = 3$ nguyên tố cùng nhau với $\phi(n)$
4. Tính nghịch đảo của e trong phép modulo n được $d = 7$ (3x7 = 21)
5. Khóa công khai $K_U = (e, N) = (3, 33)$. Khóa bí mật $K_R = (d, N) = (7, 33)$

Mã hóa và Giải mã RSA

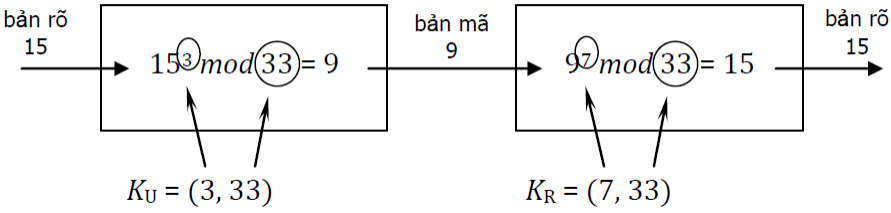
Theo phương án 1 (mã hóa bảo mật):

6) Mã hóa bản rõ $M = 15$:

$$C = M^e \bmod N = 15^3 \bmod 33 = 9 \quad (\text{vì } 15^3 = 3375 = 102 \times 33 + 9)$$

7) Giải mã bản mã C = 9:

$$\bar{M} = C^d \bmod N = 9^7 \bmod 33 = 15 = M \quad (\text{vì } 9^7 = 4.782.696 = 144.938 \times 33 + 15)$$



Mã hóa và Giải mã RSA

Theo phương án 2 (mã hóa chứng thực):

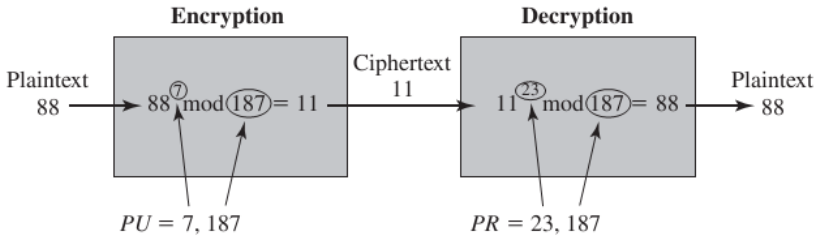
6) Mã hóa bản rõ *M* = 15:

$$C = M^d \bmod N = 15^7 \bmod 33 = 27 \quad (\text{vì } 15^7 = 170.859.375 = 5177.556 \times 33 + 27)$$

7) Giải mã bản mã C = 9:

$$\bar{M} = C^e \bmod N = 27^3 \bmod 33 = 15 = M \quad (\text{vì } 27^3 = 19.683 = 596 \times 33 + 15)$$

Ví dụ thực hiện RSA



Ghi chú : RSA sử dụng các sô nguyên tố lớn p,q để việc phân tích N với (N= pq) là vô cùng khó khăn.

Example: Confidentiality

- Take $p = 7$, $q = 11$, so $n = 77$ and $\phi(n) = 60$
- Alice chooses $e = 17$, making $d = 53$
- Bob wants to send Alice secret message HELLO (07 04 11 11 14)
  - $07^{17} \bmod 77 = 28$
  - $04^{17} \bmod 77 = 16$
  - $11^{17} \bmod 77 = 44$
  - $11^{17} \bmod 77 = 44$
  - $14^{17} \bmod 77 = 42$
- Bob sends 28 16 44 44 42

Example

- Alice receives 28 16 44 44 42
- Alice uses private key, $d = 53$, to decrypt message:
  - $28^{53} \bmod 77 = 07$
  - $16^{53} \bmod 77 = 04$
  - $44^{53} \bmod 77 = 11$
  - $44^{53} \bmod 77 = 11$
  - $42^{53} \bmod 77 = 14$
- Alice translates message to letters to read HELLO
  - No one else could read it, as only Alice knows her private key and that is needed for decryption

Example: Integrity/Authentication

- Take $p = 7$, $q = 11$, so $n = 77$ and $\phi(n) = 60$
- Alice chooses $e = 17$, making $d = 53$
- Alice wants to send Bob message HELLO (07 04 11 11 14) so Bob knows it is what Alice sent (no changes in transit, and authenticated)
  - $07^{53} \bmod 77 = 35$
  - $04^{53} \bmod 77 = 09$
  - $11^{53} \bmod 77 = 44$
  - $11^{53} \bmod 77 = 44$
  - $14^{53} \bmod 77 = 49$
- Alice sends 35 09 44 44 49

Example

- Bob receives 35 09 44 44 49
- Bob uses Alice's public key, $e = 17$, $n = 77$, to decrypt message:
  - $35^{17} \bmod 77 = 07$
  - $09^{17} \bmod 77 = 04$
  - $44^{17} \bmod 77 = 11$
  - $44^{17} \bmod 77 = 11$
  - $49^{17} \bmod 77 = 14$
- Bob translates message to letters to read HELLO
  - Alice sent it as only she knows her private key, so no one else could have enciphered it
  - If (enciphered) message's blocks (letters) altered in transit, would not decrypt properly

Example: Both

- Alice wants to send Bob message HELLO both enciphered and authenticated (integrity-checked)
  - Alice's keys: public (17, 77); private: 53
  - Bob's keys: public: (37, 77); private: 13
- Alice enciphers HELLO (07 04 11 11 14):
  - $(07^{53} \bmod 77)^{37} \bmod 77 = 07$
  - $(04^{53} \bmod 77)^{37} \bmod 77 = 37$
  - $(11^{53} \bmod 77)^{37} \bmod 77 = 44$
  - $(11^{53} \bmod 77)^{37} \bmod 77 = 44$
  - $(14^{53} \bmod 77)^{37} \bmod 77 = 14$
- Alice sends 07 37 44 44 14

→Sinh viên suy ra giải mã

Phá mã hệ mã hóa RSA

Bốn hướng có thể để tấn công RSA:

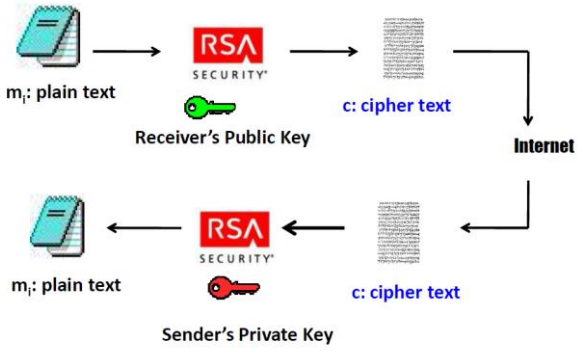
- **Vét cạn (Brute force attacks)**: Thử tất cả các khóa private key có thể. Điều này phụ thuộc vào độ dài khóa. → dùng khóa đủ lớn
- **Phân tích toán học (Mathematical attacks)**: Có vài hướng, nhưng tất cả đều tập trung vào việc phân tích thừa số tích của hai số nguyên tố.
- **Phân tích thời gian (Timing attacks)**: Cách này tùy thuộc vào thời chạy của thuật toán giải mã.
- **Phân tích bản mã được chọn (Chosen ciphertext attacks)**: khám phá các thuộc tính của thuật toán RSA. → ngăn ngừa bằng cách làm nhiễu

An ninh của hệ mã hóa RSA

- An ninh của RSA dựa trên độ khó của việc phân tích ra thừa số nguyên tố các số nguyên tố lớn.
- Thời gian cần thiết để phân tích thừa số một số lớn tăng theo hàm mũ với số bit của số đó
  - Mất nhiều năm khi số chữ số thập phân của n vượt quá 100 (giả sử làm 1 phép tính nhị phân mất 1 ηs)
- Kích thước khóa lớn đảm bảo an ninh cho RSA
  - Từ 1024 bit trở lên
  - Gần đây nhất năm 1999 đã phá mã được 512 bit (155 chữ số thập phân)

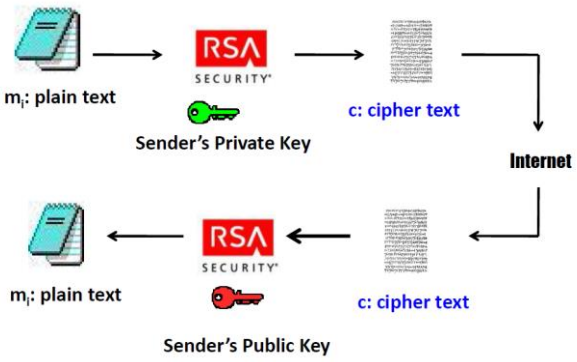
Ứng dụng của hệ mã hóa RSA

1. Bảo mật thông điệp : Sử dụng khoá công khai của bên nhận để mã, khoá riêng của bên nhận để giải mã



Ứng dụng của hệ mã hóa RSA

2. Xác thực thông điệp : Dùng khoá cá nhân của bên gửi để mã , khoá công khai của bên gửi để giải mã



Bài tập

1. Cho p = 5, q= 11, e = 7. Tính khóa riêng (d, N) trong phương pháp RSA.
2. Thực hiện mã hóa và giải mã bằng phương pháp RSA với p = 3, q = 11, e = 7, M = 5 theo hai trường hợp mã hóa bảo mật và mã hóa chứng thực.
3. Alice chọn p = 7, q = 11, e = 17, Bob chọn p = 11, q = 13, e = 11:

   a. Tính khóa riêng KRA của Alice và KRB của Bob

   b. Alice muốn gửi cho Bob bản tin M = 9 vừa áp dụng chứng thực và bảo mật. Hãy thực hiện quá trình mã hóa và giải mã.

Bài tập

4. Cho hệ mã RSA có p = 31, q = 41, e = 271.

a) Hãy tìm khóa công khai KP, và khóa bí mật KS của hệ mã trên.

b) Để mã hóa các thông điệp đươc viết bằng tiếng Anh người ta dùng một hàm chuyển đổi các ký tự thành các số thập phân có hai chữ số như sau:

Ký tự	A	B	C	D	E	F	G	H	I	J	K	L	M
Mã hóa	00	01	02	03	04	05	06	07	08	09	10	11	12
Ký tự	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Mã hóa	13	14	15	16	17	18	19	20	21	22	23	24	25

Khi đó ví dụ xâu ABC sẽ đươc chuyển thành 00 01 02 và sau đó cắt thành các số có 3 chữ số 000 (bằng 0) và 102 để mã hóa. Bản mã thu đươc là một tập các số $\in Z_N$. Hãy thực hiện mã hóa xâu P = ”SERIUS”.

c) Giả sử bản mã thu được là C = <201, 793, 442, 18> hãy thực hiện giải mã để tìm ra thông điệp bản rõ ban đầu.

3. Một số thuật toán mã hóa khóa công khai khác

3.1 Trao đổi khóa Diffie-Hellman

(Diffie-Hellman Key Exchange)

3.2 Mật mã Elgamal

(Elgamal Cryptographic System)

3.3 Mật mã ECC

(Elliptic Curve Cryptography)

Trao đổi khóa Diffie-Hellman

- Giải thuật mật mã khóa công khai đầu tiên
- Đề xuất bởi Whitfield Diffie và Martin Hellman vào năm 1976
- Chỉ dùng để trao đổi khóa bí mật một cách an ninh trên các kênh thông tin không an ninh
- Khóa bí mật được tính toán bởi cả hai bên
- An ninh phụ thuộc vào độ phức tạp của việc tính log rời rạc

Trao đổi khóa Diffie-Hellman

a. Tạo khóa

- Ta có p là số nguyên tố ($p \epsilon Z_p$) .
- Giả sử $\alpha \in Z_p$ là một số nguyên thuỷ (primitive element)
- Các giá trị p và α được công bố công khai trên mạng.
- UID thông tin định danh hợp lệ cho từng user U trên mạng ("tên"," e-mail address"," telephone number"...)
- Từng "user U,V" có một số mũ a_u, a_V với $(0 \le a_u, a_V \le p-2)$, và tính giá trị b_U, b_V công khai tương ứng :

$$b_U = \alpha^{a}{}_{u} \text{ modp và}$$
$$b_V = \alpha^{a}{}_{v} \text{ modp}$$

- Khoá chung $K_{u,v}$ được tính $K_{u,v} = \alpha^{a}{}_{u}, {}^{a}{}_{v} \; modp$

Trao đổi khóa Diffie-Hellman

b. Thuật giải

- Input : *p* SNT và α primitive element $\in Z^*_p$ → *truyền công khai trên mạng*

  Từng "*user U,V*" có một số mũ a_u, a_v với :

  $(0 \le a_u , a_v \le p-2)$,

- Output :

  Hai bên cùng tính $b_u = \alpha^{a}{}_u \bmod p$ và $b_v = \alpha^{a}{}_v \bmod p$

  Hai bên gửi cho nhau : b_u và b_v.

  1. Bên V tính : $K_{U,V} = \alpha^{a}{}_u \,,^{a}{}_v \bmod p = b_u{}^{a}{}_v \bmod p$

     Dùng b_U từ U cùng với giá trị mật a_u

  2. Bên U tính : $K_{U,V} = \alpha^{a}{}_u \,,^{a}{}_v \bmod p = b_v{}^{a}{}_u \bmod p$

     Dùng b_V gửi từ V cùng với giá trị mật a_v

Trao đổi khóa Diffie-Hellman

c. Ví dụ Diffie- Hellman

- Giả sử *p* = 25307 và α = 2 biết công khai (*p* là SNT và α là số nguyên thuỷ gốc modulo *p*).
- User U Chọn a_U = 3578. Tính

$$b_U = \alpha^{a_U} \bmod p$$
$$= 2^{3578} \bmod 25307$$
$$= 6113,$$

Dùng để chứng nhận U

- User V chọn a_V = 19956. Tính

$$b_V = \alpha^{a_V} \bmod p$$
$$= 2^{19956} \bmod 25307$$
$$= 7984,$$

Dùng để chứng nhận V

Trao đổi khóa Diffie-Hellman

Ví dụ Diffie- Hellman (tiếp)

- User U tính khoá của mình

$$\begin{aligned} K_{U,V} &= b_V{}^{a_U} \bmod p \\ &= 7984^{3578} \bmod 25307 \\ &= 3694, \end{aligned}$$

- User V tính khoá của mình

$$\begin{aligned} K_{U,V} &= b_U{}^{a_V} \bmod p \\ &= 6113^{19956} \bmod 25307 \\ &= 3694. \end{aligned}$$

Mật mã ElGamal

- Được đề xuất năm 1985, dựa vào độ phức tạp của bài toán logarit rời rạc.
- Mã ElGamal được dùng trong số tiêu chuẩn như: Digital Signature Standard (DSS) và S/MIME e-mail standard
- An ninh của ElGamal dựa trên độ khó của việc tính logarit rời rạc

Mật mã ElGamal

- Quá trình tạo khóa của A sử dụng hệ ElGamal gồm các bước chính sau:
- A, B thống nhất số nguyên tố q và phần tử sinh q: α
- Bên tạo khóa (A) chọn giá trị bí mật Xa (Xa<q-1) và tính giá trị Ya=α^{X}_{A} mod q. Khi đó, bộ khóa K={PU,PR} của A, với khóa công khai PU={q, α,Y_A} và khóa cá nhân PR={X_A}

Mật mã ElGamal

- Quá trình B sử dụng bộ khóa của A trong việc truyền dữ liệu M (M<q):
- B chọn giá trị k (k<q) và tính toán khóa K =$(Y_A)^K$ mod q, C_1= α^k mod q, C_2 =KM mod q. Khi đó (C1,C2) là bản mã được truyền đi
- Quá trình bên nhận (A) giải mã:
  - Tính khóa K = $(C_1)^{XA}$ mod q
  - Tìm bản gốc theo công thức: M =(C_2K^{-1})mod q

Mật mã đường cong Elliptic

- ECC- Elliptic Elliptic Curve Cryptography
- Ưu điểm:
  - ECC sử dụng khoá có độ dài nhỏ hơn so với RSA. → làm tăng tốc độ xử lý một cách đáng kể; với cùng một độ dài khoá thì ECC có nhiều ưu điểm hơn so với các giải thuật khác
  - Có thể dụng cả 3 ứng dụng: bảo mật, trao đổi khóa, chữ ký số.
- An ninh ECC dựa trên vấn đề logarit đường cong elliptic
- Tính tin cậy vẫn chưa cao bằng RSA

Xin chân thành cảm ơn!