INDUSTRIAL UNIVERSITY OF HOCHIMINH CITY

# MẬT MÃ HỌC
# (Cryptography)

## Mã hóa bất đối xứng
## ASYMMETRIC CIPHERS

# NỘI DUNG

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

( Cryptography and Network Security: Principles and Practices (3rd Ed.) – Chapter 9, 10)

# Đặt vấn đề

Khuyết điểm của mã hóa 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.

# 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

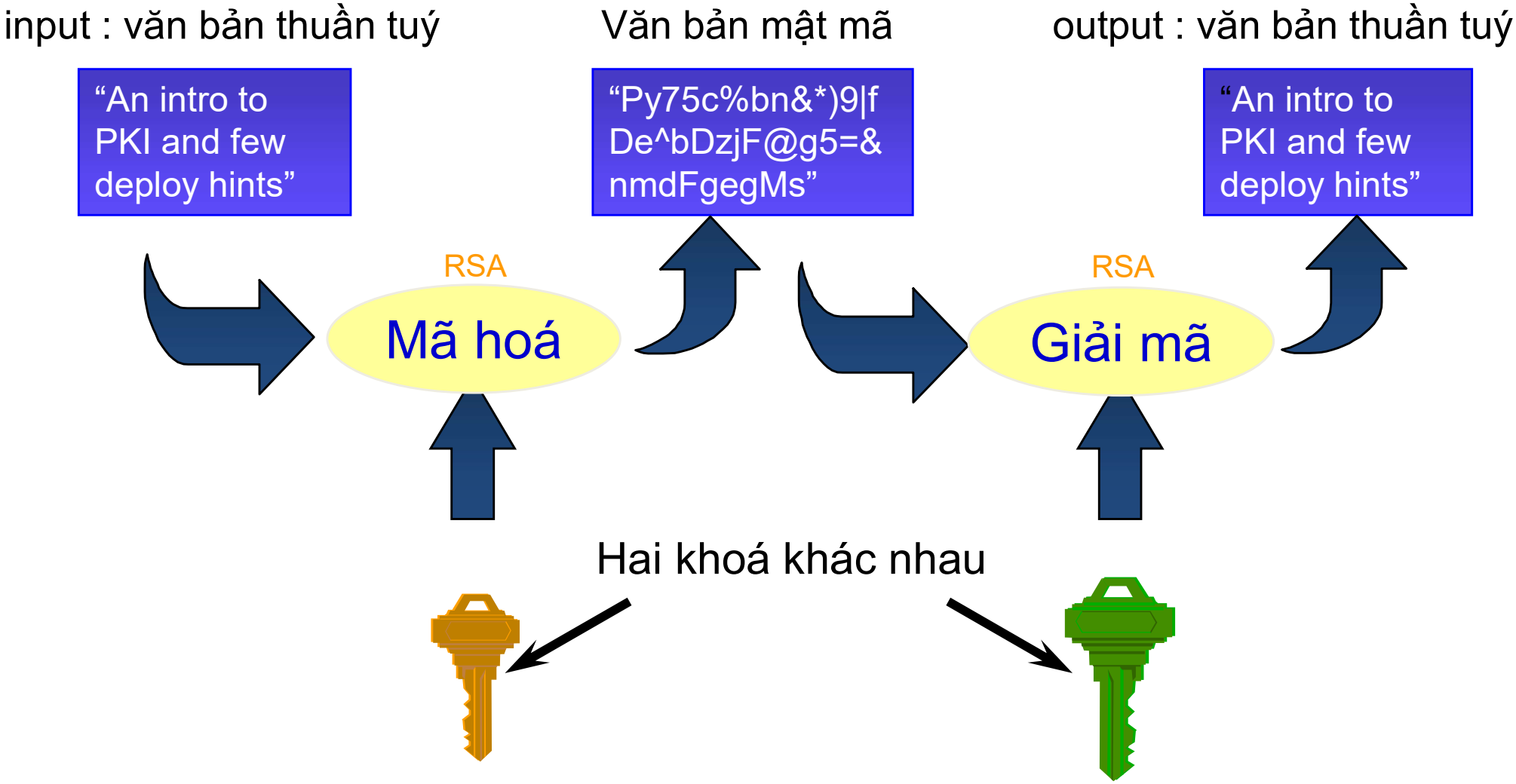
**MÃ HÓA KHÓA CÔNG KHAI**

(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



# 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)

- Hai vấn đề của Khóa bí mật

- Hai cơ chế của mã hóa khóa công khai

# Mã hóa công khai (Public-Key Cryptosystems)

- ***Vấn đề phân phối khóa***:
  - Khó đảm bảo chia sẻ mà không làm lộ khóa bí mật
  - Trung tâm phân phối khóa có thể bị tấn công.

- ***Không thích hợp cho chữ ký số***:
  - Bên nhận có thể làm giả thông điệp và nói rằng nhận từ bên gửi.

# Mã hóa khóa công khai Public-Key Cryptosystems

- **Mã hóa khóa công khai (*Public-Key Cryptosystems*)**
- Phát minh bởi Whitfield Diffie & Martin Hellman - Stanford Unit, vào năm 1976
- Mục tiêu là khắc phục điểm yếu của mã hóa đối xứng
- Một số ứng dụng cụ thể: SSL, VPN, SSH.
- Phương pháp: ***dùng hai khóa khác nhau cho quá trình mã hóa và giải mã***

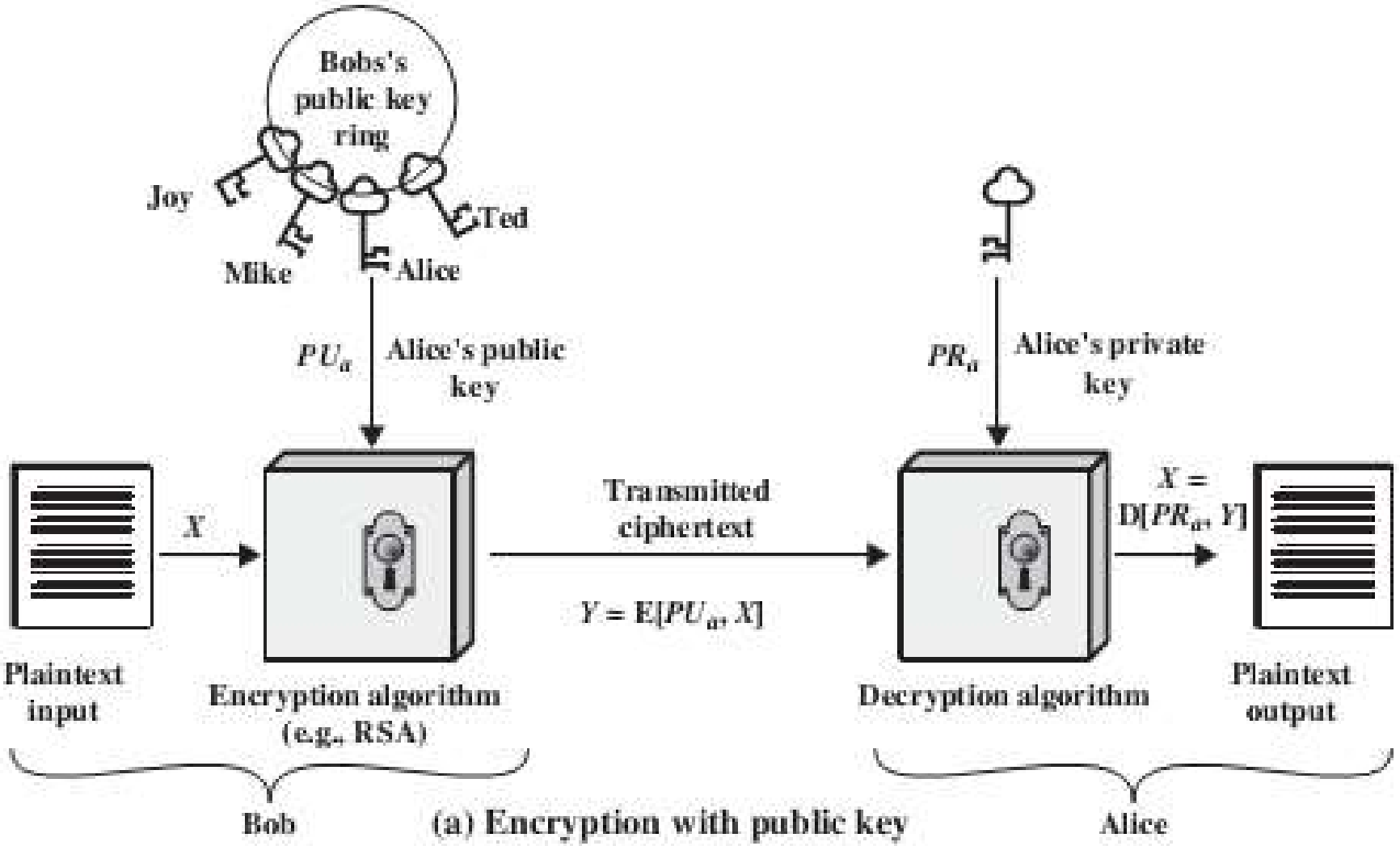
$$C = E(P, K_1) \text{ và } P = D(C, K_2)$$

# Giải thuật Mã hóa công khai (Public-Key Cryptosystems)

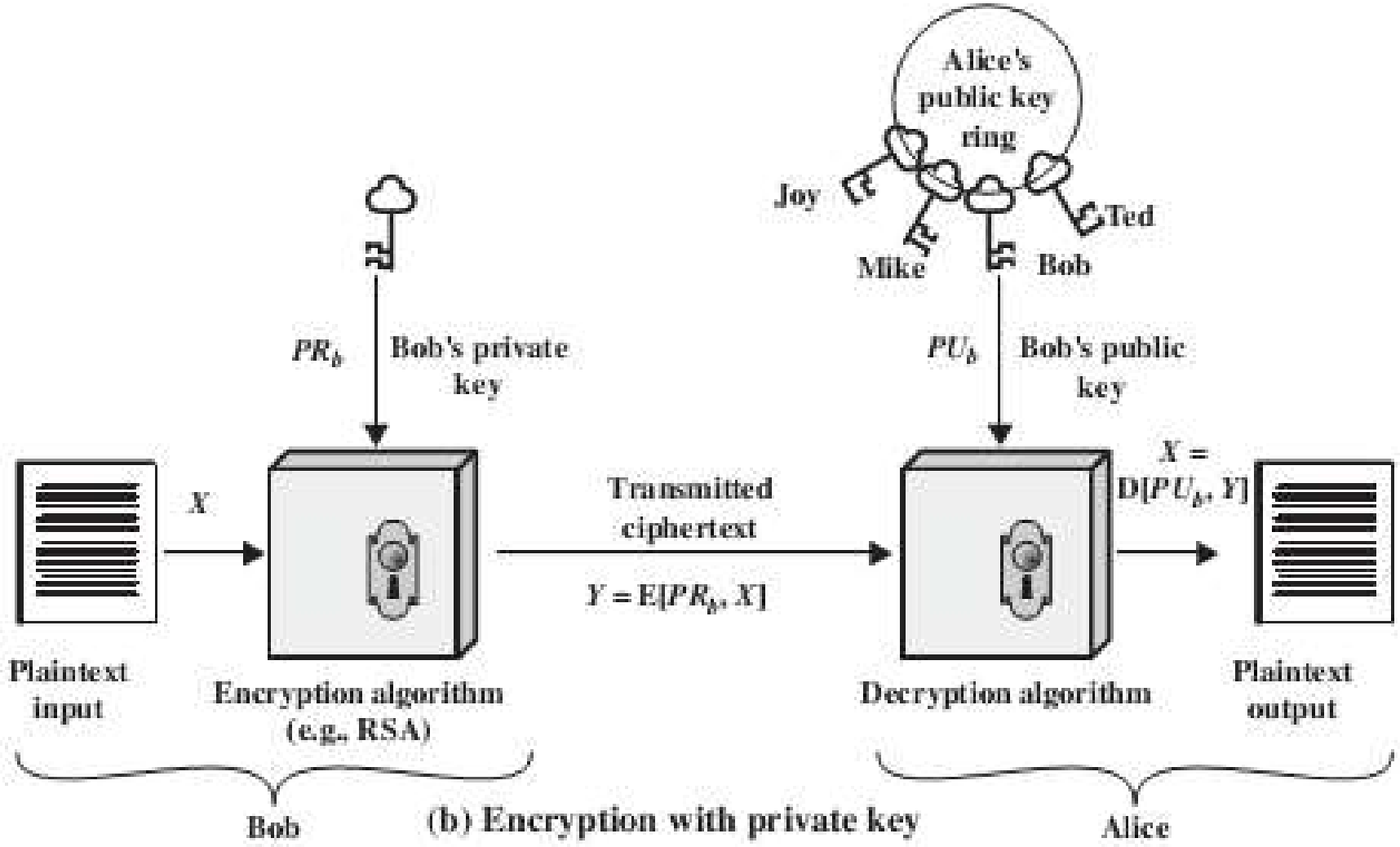
**Giải thuật khóa công khai gồm 6 thành phần:**

- **Đầu vào của giải thuật:** Bản rõ (thông điệp có thể đọc)
- **Giải thuật mật 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)

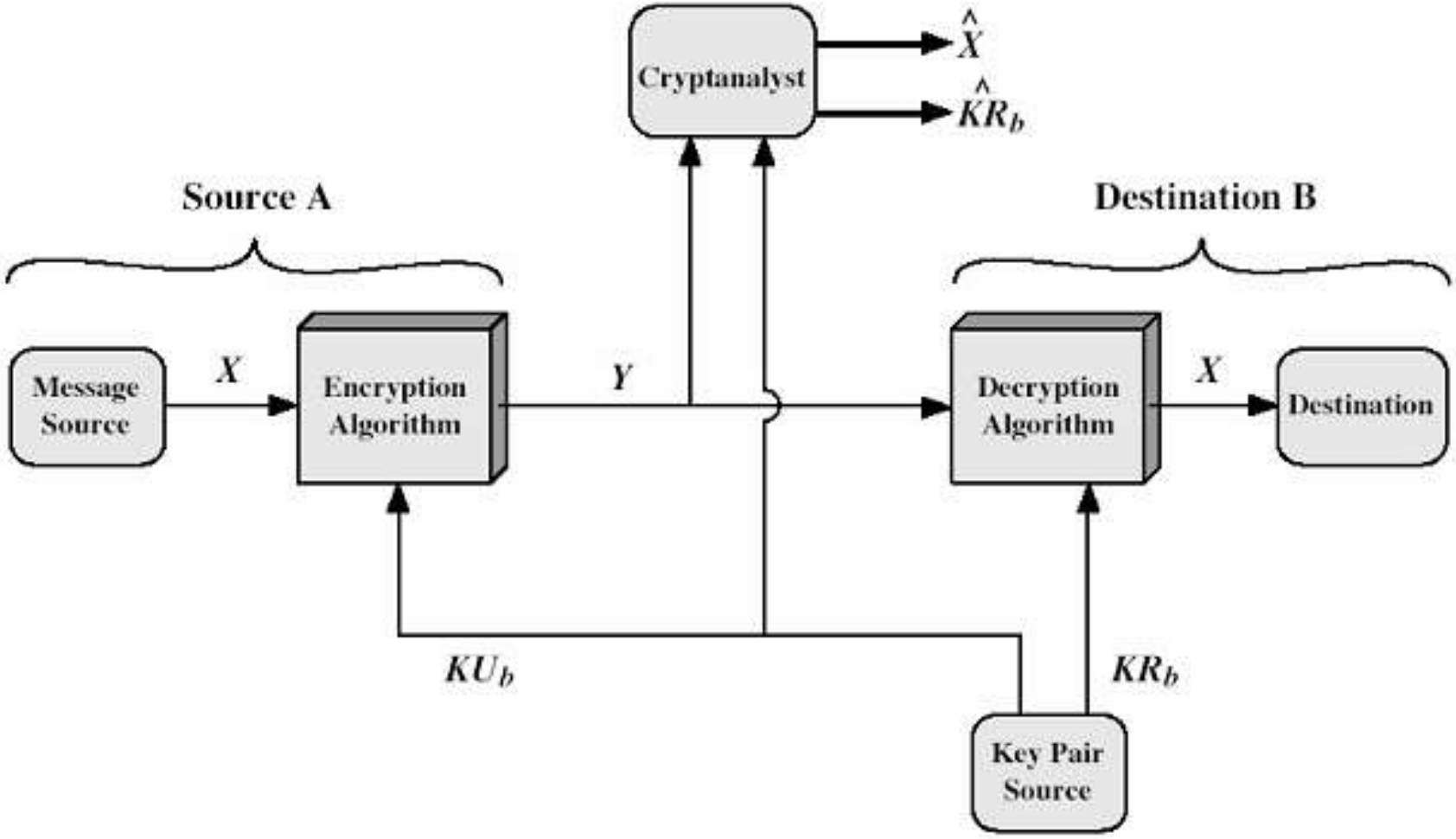
- Một trong hai khóa có thể dùng cho việc mã hóa (encryption), Khóa còn lại dùng cho giải mã (đối với thuật toán RSA)

# So sánh

- **Conventional Encryption (Mã hóa thông thường)**
  - 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.

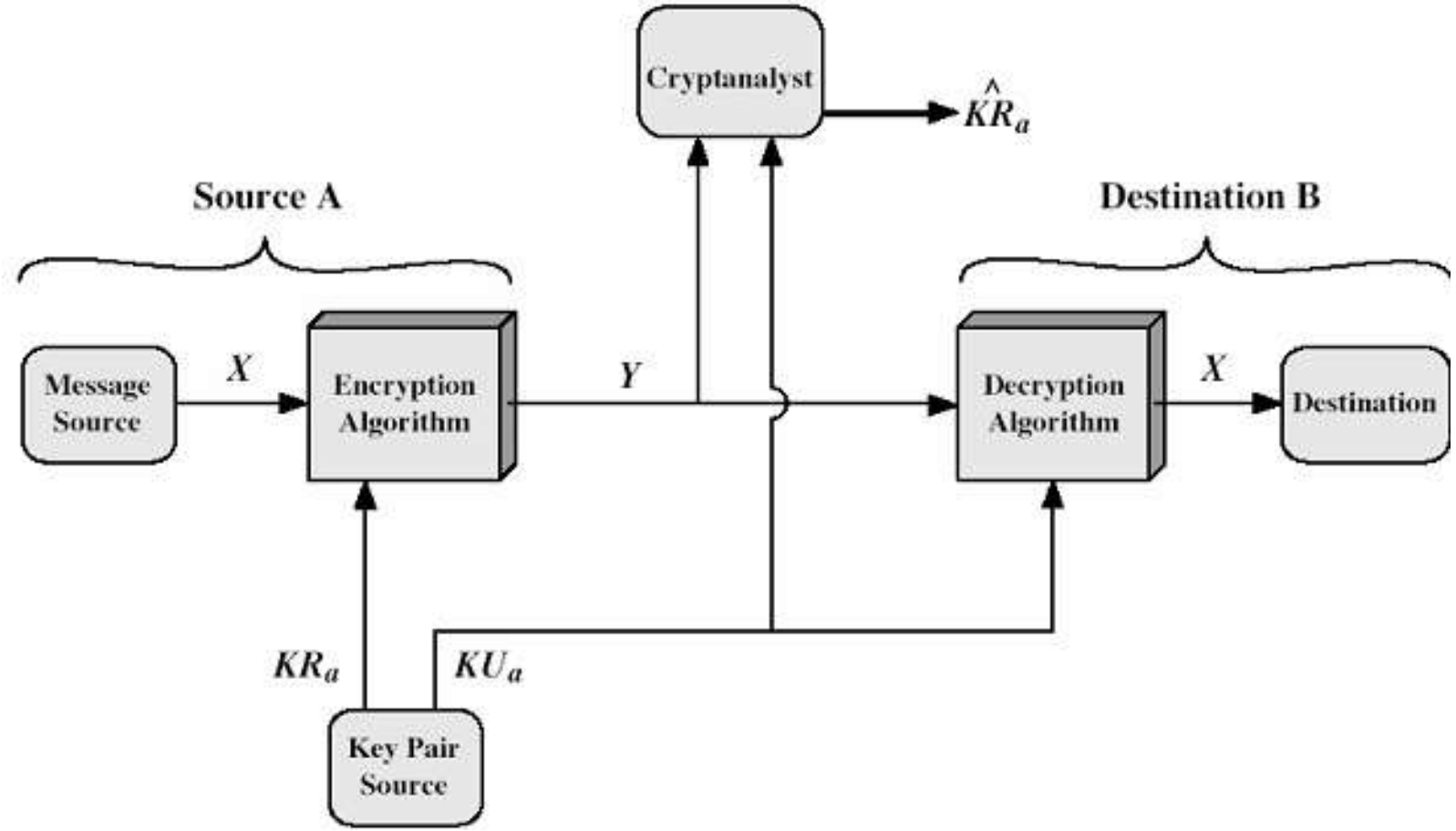
- **Public-key Encryption**
  - Một thuật toán được dùng để mã hóa và giải mã với một cặp khóa, một khóa dành cho mã hóa và một dành do giải mã
  - Sender và receiver phải có một trong cặp khóa (không giống nhau)
  - Một trong hai khóa phải được giữ bí mật
  - Không thể hoặc ít nhất không thực thể để 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

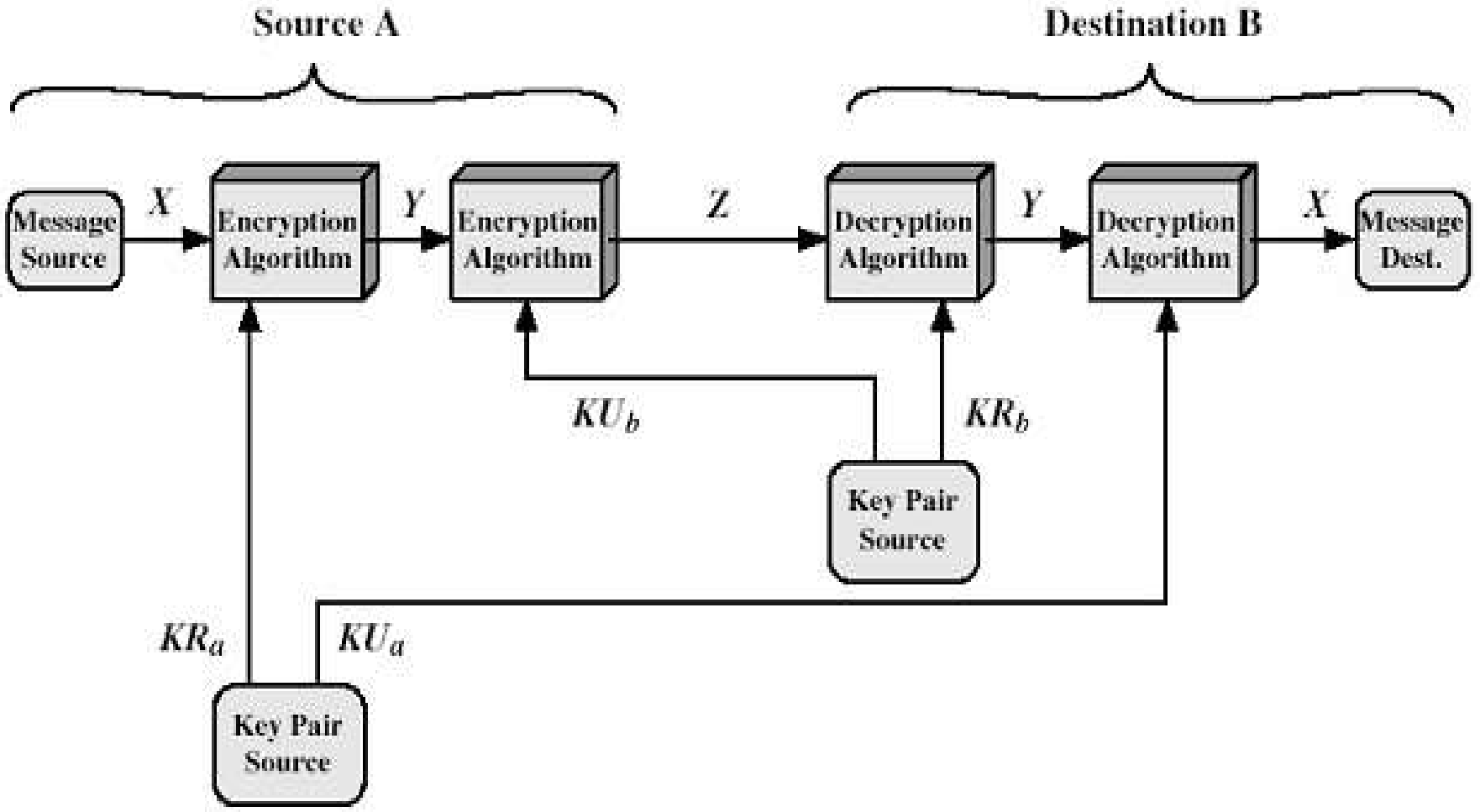


# Public-Key Cryptosystems: Authentication

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



# Public-Key Cryptosystems: Secrecy and Authentication



# Public-Key Application

- Có thể phân thành 3 loại:
  - **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 dộ 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, KU) 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

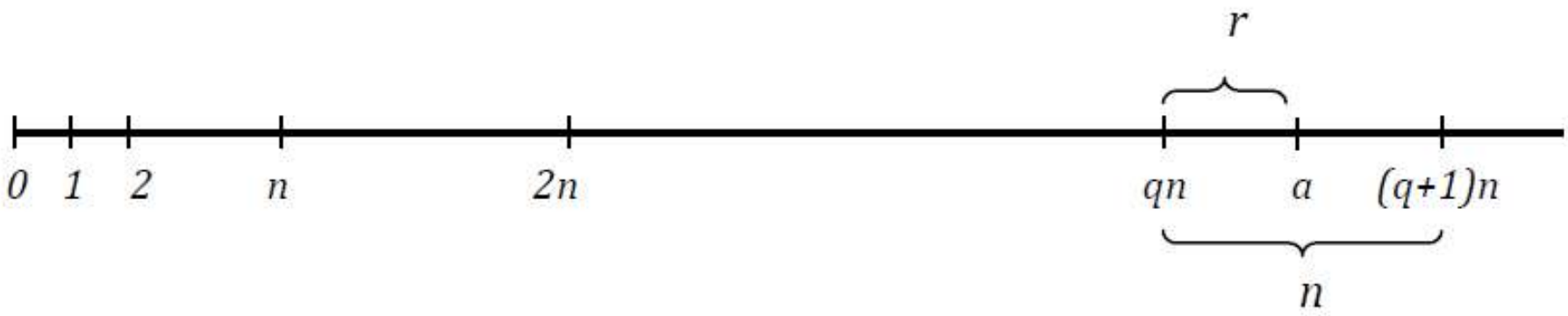
- Tốc độ xử lý
  - Các giải thuật khóa công khai chủ yếu dùng các phép nhân chậm hơn nhiều so với các giải thuật đối xứng
  - Không thích hợp cho mã hóa thông thường
  - Thường dùng trao đổi khóa bí mật đầu phiên truyền tin
- Tính xác thực của khóa công khai
  - Bất cứ ai cũng có thể tạo ra một khóa 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

# Nhắc lại lý thuyết số

*1. Phép chia modulo:*

- Phép chia modulo là phép chia lấy phần dư.
- Ví dụ: 27 *mod* 8 = 3, 35 *mod* 9 = 8.
- Một cách tổng quát:

$$a \bmod n = r \quad \text{với} \quad a \geq 0;\ n > 0;\ 0 \leq r \leq n-1$$



# Nhắc lại lý thuyết số

*1. Phép chia modulo:*

- Nếu hai số *a*, *b* có cùng số dư trong phép chia cho *n* thì ta nói rằng *a* và *b* là đồng dư trong phép chia modulo cho *n*, phép so sánh đồng dư được ký hiệu bằng dấu ≡:

$$a \equiv b \pmod{n} \text{ hay viết tắt là } a \equiv b \bmod n$$

- Ví dụ với *n* = 4 ta có 4 lớp tương đương sau:

$$\{0, 4, 8, 12, 16 \ldots\}$$
$$\{1, 5, 9, 13, 17 \ldots\}$$
$$\{2, 6, 10, 14, 18 \ldots\}$$
$$\{3, 7, 11, 15, 19 \ldots\}$$

# Nhắc lại lý thuyết số

*2. Một số tính chất của phép modulo:*

- Cho *a*, *b* và *n* là các số nguyên, phép modulo có các tính chất:

a) $(a + b) \bmod n = [(a \bmod n) + (b \bmod n)] \bmod n$

b) $(a - b) \bmod n = [(a \bmod n) - (b \bmod n)] \bmod n$

c) $(a \times b) \bmod n = [(a \bmod n) \times (b \bmod n)] \bmod n$

# Nhắc lại lý thuyết số

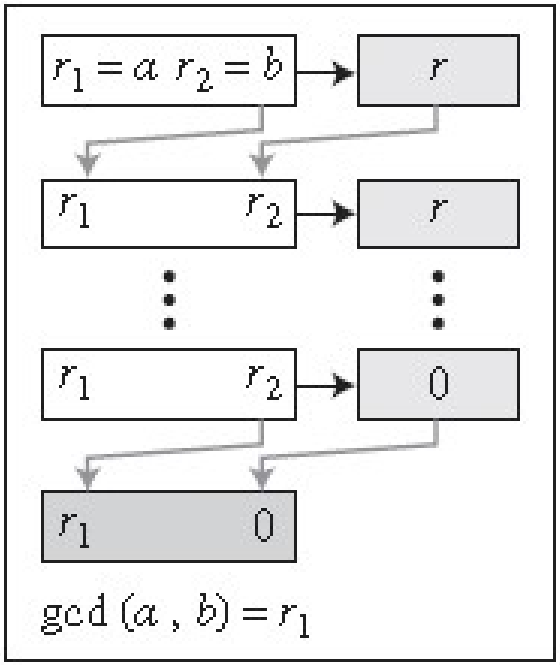
*3. Ước số:*

- Nếu a mod n =0 (viết cách khác a ≡0 mod n) thì có nghĩa là *a* chia hết cho *n*, hay *n* là ước số của *a*.
- Ước số chung lớn nhất của hai số: ký hiệu *gcd*(*a, b*) . Để tìm USCLN của hai số a, b, chúng ta có thể dùng thuật toán Euclid.

# Ước số chung lớn nhất (Greatest Common Divisor –gcd)

- Cho hai số a, b $\in Z \setminus \{0\}$, c $\in Z$ là ước số chung (common divisor) của a và b nếu c|a và c|b
- C được gọi là ước số chung lớn nhất (greatest common divisor), ký hiệu gcd(a, b), ***nếu nó là số nguyên lớn nhất được chia hết bởi cả a và b***.
- Ví dụ: gcd(12,18) =6, gcd(-18,27) = 9

# Thuật toán Euclid tìm gcd(a,b)




a. Process

```
r1 = a; r2 = b;   (Initialization)
while (r2 > 0)
  {
   q = r1 / r2;
   r = r1 - q * r2;
   r1 = r2; r2 = r;
  }

  gcd (a, b) = r1
```

b. Algorithm

# Ví dụ: Tìm gcd(2740,1760)

| $q$ | $r_1$ | $r_2$ | $r$ |
|---|---|---|---|
| 1 | 2740 | 1760 | 980 |
| 1 | 1760 | 980 | 780 |
| 1 | 980 | 780 | 200 |
| 3 | 780 | 200 | 180 |
| 1 | 200 | 180 | 20 |
| 9 | 180 | 20 | 0 |
| | **20** | 0 | |

➔ gcd (2740, 1760) = 20

# Ví dụ: Tìm gcd(25,60)

| $q$ | $r_1$ | $r_2$ | $r$ |
|---|---|---|---|
| 0 | 25 | 60 | 25 |
| 2 | 60 | 25 | 10 |
| 2 | 25 | 10 | 5 |
| 2 | 10 | 5 | 0 |
| | **5** | 0 | |

→ gcd(25,60)=5

# Nguyên tố cùng nhau
# (co-prime hay relatively prime )

- Hai số nguyên a, b ∈ Z \{0} được gọi là **nguyên tố cùng nhau nếu gcd(a, b)=1**.
- Ví dụ:  (5,8) , (9,14) là các cặp nguyên tố cùng nhau

## *Phần tử nghịch đảo của phép nhân modulo:*

- Nếu hai số nguyên *a* và *n* nguyên tố cùng nhau, thì tồn tại số nguyên *w* sao cho:

$$a.w \equiv 1 \bmod n$$

- Ta gọi *w* là phần tử nghịch đảo của *a* trong phép modulo cho *n* và ký hiệu là $a^{-1}$

# Nghịch đảo nhân multiplicative inverse

- Nếu tồn tại 1 số $b \in Z_n$ sao cho

$$ab \equiv 1 (\text{mod } n)$$

thì b được gọi là nghịch đảo nhân của a modulo n

- Ký hiệu

$$b = a^{-1} \pmod{n}$$

# Nghịch đảo nhân multiplicative inverse

- ***Điều kiện để số a có nghịch đảo nhân khi và chỉ khi gcd(a, n)=1***
- Ví dụ: a=22, n=25

  Gcd(22,25)= 1 → a có nghịch đảo nhân

  $8 = 22^{-1} \pmod{25}$ vì $8.22 = 176 = 1 \pmod{25}$

  → 8 là nghịch đảo nhân của 22 modulo 25

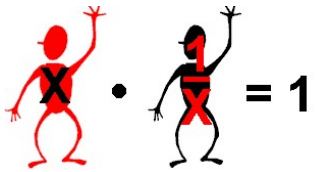
# Cách tìm nghịch đảo nhân

- Cách 1: dùng giải thuật Euclid mở rộng

  $au + pv = 1$ với u,v số nguyên

  $u = a^{-1} \bmod p$

# Thuật toán Euclid mở rộng (extended Euclidean algorithm)




- Cho 2 số nguyên a và b, tìm 2 số nguyên khác s và t sao cho:

$$s \times a + t \times b = \gcd(a, b)$$

- Thực hiện phép mod cả 2 vế

(s x n + t x b) mod n = 1 mod n

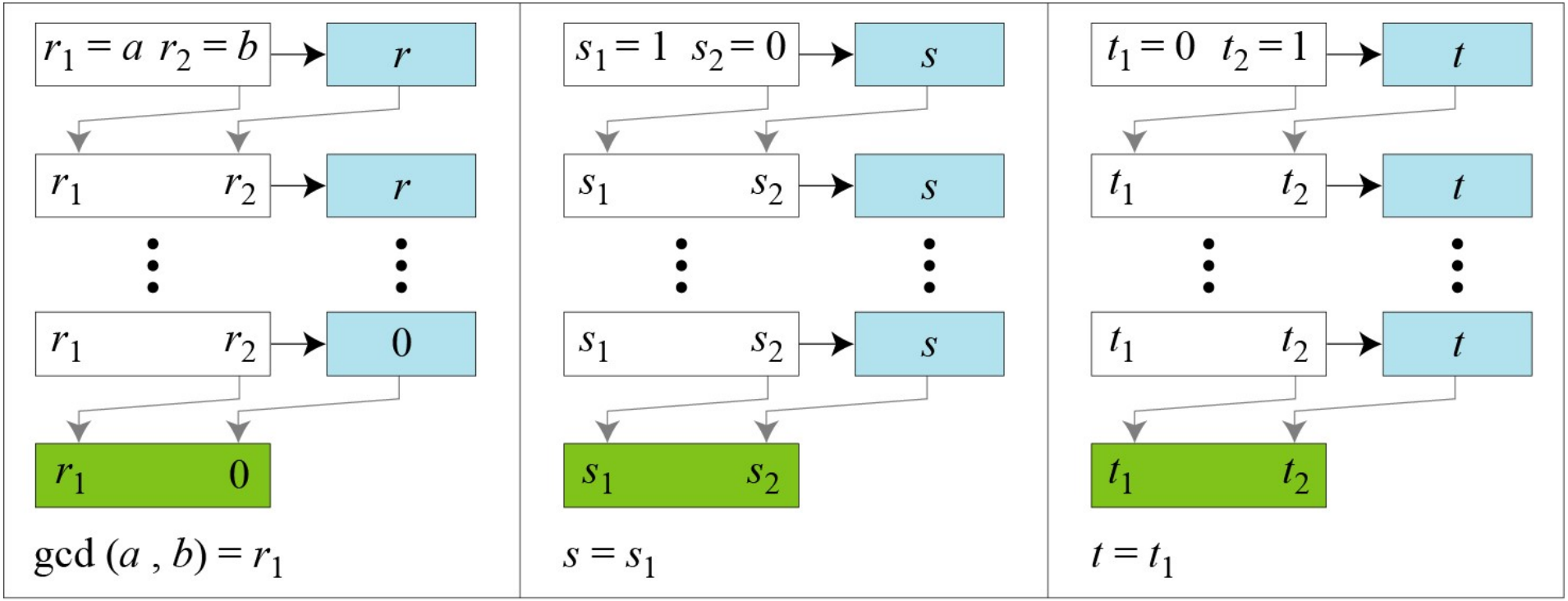
[(s x n) mod n] + [(txb) mod n] = 1 mod n

0 + [(txb) mod n] = 1

➔ ( t x b ) mod n = 1 → t chính là nghịch đảo nhân của b

- Thuật toán này vừa có thể tính được gcd(a,b) vừa tính được các giá trị s và t

# Thuật toán Euclid mở rộng (extended Euclidean algorithm)




a. Process

# Thuật toán Euclid mở rộng (extended Euclidean algorithm)

$r_1 \leftarrow a;$ $r_2 \leftarrow b;$
$s_1 \leftarrow 1;$ $s_2 \leftarrow 0;$
$t_1 \leftarrow 0;$ $t_2 \leftarrow 1;$ (Initialization)

while $(r_2 > 0)$

{

$q \leftarrow r_1 / r_2;$

$r \leftarrow r_1 - q \times r_2;$
$r_1 \leftarrow r_2;$ $r_2 \leftarrow r;$ (Updating $r$'s)

$s \leftarrow s_1 - q \times s_2;$
$s_1 \leftarrow s_2;$ $s_2 \leftarrow s;$ (Updating $s$'s)

$t \leftarrow t_1 - q \times t_2;$
$t_1 \leftarrow t_2;$ $t_2 \leftarrow$ t; (Updating $t$'s)

}

gcd $(a, b) \leftarrow r_1;$ $s \leftarrow s_1;$ $t \leftarrow t_1$

b. Algorithm

# Ví dụ:

## a = 161 và b = 28, tìm gcd (a, b) và giá trị s và t.

Giải: $r = r_1 - q \times r_2$; $s = s_1 - q \times s_2$; $t = t_1 - q \times t_2$

| $q$ | $r_1$ | $r_2$ | $r$ | $s_1$ | $s_2$ | $s$ | $t_1$ | $t_2$ | $t$ |
|---|---|---|---|---|---|---|---|---|---|
| 5 | 161 | 28 | 21 | 1 | 0 | 1 | 0 | 1 | −5 |
| 1 | 28 | 21 | 7 | 0 | 1 | −1 | 1 | −5 | 6 |
| 3 | 21 | 7 | 0 | 1 | −1 | 4 | −5 | 6 | −23 |
| | **7** | 0 | | **−1** | 4 | | **6** | −23 | |

→ gcd (161, 28) = 7, s = −1 và t = 6.

NẾU T1>0 → d=T1,

NẾU T1<0→d = Φ(N)+T1

# Ví dụ: Tìm nghịch đảo nhân của 23 trong $Z_{100}$.

$$t = t_1 - q \times t_2$$

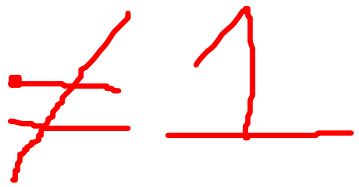
| $q$ | $r_1$ | $r_2$ | $r$ | $t_1$ | $t_2$ | $t$ |
|---|---|---|---|---|---|---|
| 4 | 100 | 23 | 8 | 0 | 1 | −4 |
| 2 | 23 | 8 | 7 | 1 | −4 | 19 |
| 1 | 8 | 7 | 1 | −4 | 9 | −13 |
| 7 | 7 | 1 | 0 | 9 | −13 | 100 |
| | 1 | 0 | | −13 | 100 | |

- GCD(Φ(N),E) =gcd(100, 23) là 1; nghịch đảo nhân của 23 là –13 hoặc 87.

# Ví dụ: Tìm nghịch đảo nhân của 12 trong $Z_{26}$.

| $q$ | $r_1$ | $r_2$ | $r$ | $t_1$ | $t_2$ | $t$ |
|---|---|---|---|---|---|---|
| 2 | 26 | 12 | 2 | 0 | 1 | –2 |
| 6 | 12 | 2 | 0 | 1 | –2 | 13 |
| | 2 | 0 | | –2 | 13 | |

- gcd(26, 2) là 2; nghịch đảo nhân không tồn tại



# Cách tìm nghịch đảo nhân

- Cách 2: dùng giải thuật tính số mũ nhanh (với p là số nguyên tố)

$$a^1 \equiv a^{p-2} \pmod{p}$$

# *Phần tử nghịch đảo của phép nhân modulo:*

- Ví dụ:

- $n = 10, a = 7$ là hai số nguyên tố cùng nhau, do đó tìm được $a^{-1} = 3$ $(21 \equiv 1 \ mod \ 10)$

| $a^{-1}$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| $a^{-1}$x7 mod 10 | 0 | 7 | 4 | 1 | 8 | 5 | 2 | 9 | 6 | 3 |

- $n = 10, a = 2$ không phải là hai số nguyên tố cùng nhau, ta có bảng phép nhân sau:

| $a^{-1}$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| $a^{-1}$x2 mod 10 | 0 | 2 | 4 | 6 | 8 | 0 | 2 | 4 | 6 | 8 |

- Trong bảng trên không tồn tại số $a$-1 nào sao cho $a.a^{-1} \equiv 1 \ mod \ 10$. Vậy không tồn tại phần tử nghịch đảo.
- Để tính chúng ta dùng thuật toán Euclid mở rộng $s \times a + t \times b = \gcd(a, b)$

# Bài tập tìm nghịch đảo nhân

- Cho n = 5 và  a = 2. Tìm nghịch đảo nhân
- Vì gcd(2, 5) = 1, do đó 2 sẽ có nghịch đảo nhân modulo 5

  → $3 = 2^{-1}$ (mod) 5 vì $2 \cdot 3 \equiv 1 \pmod 5$.
- gcd(4, 15) = 1 vì vậy 4 có nghịch đảo nhân modulo 15.

  →Vì $4 \cdot 4 \equiv 1 \pmod{15}$ nên $4 = 4^{-1} \pmod{15}$

# 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

# Thuật toán RSA Phát sinh khóa

**Key Generation Alice**

| | |
|---|---|
| Select $p, q$ | $p$ and $q$ both prime, $p \neq q$ |
| *Calculate* $n = p \times q$ | p, q là số nguyên tố |
| 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 | $PR = \{d, n\}$ |

(The **G**reatest **C**ommon **D**ivisor (GCD)- ước số chung lớn nhất)

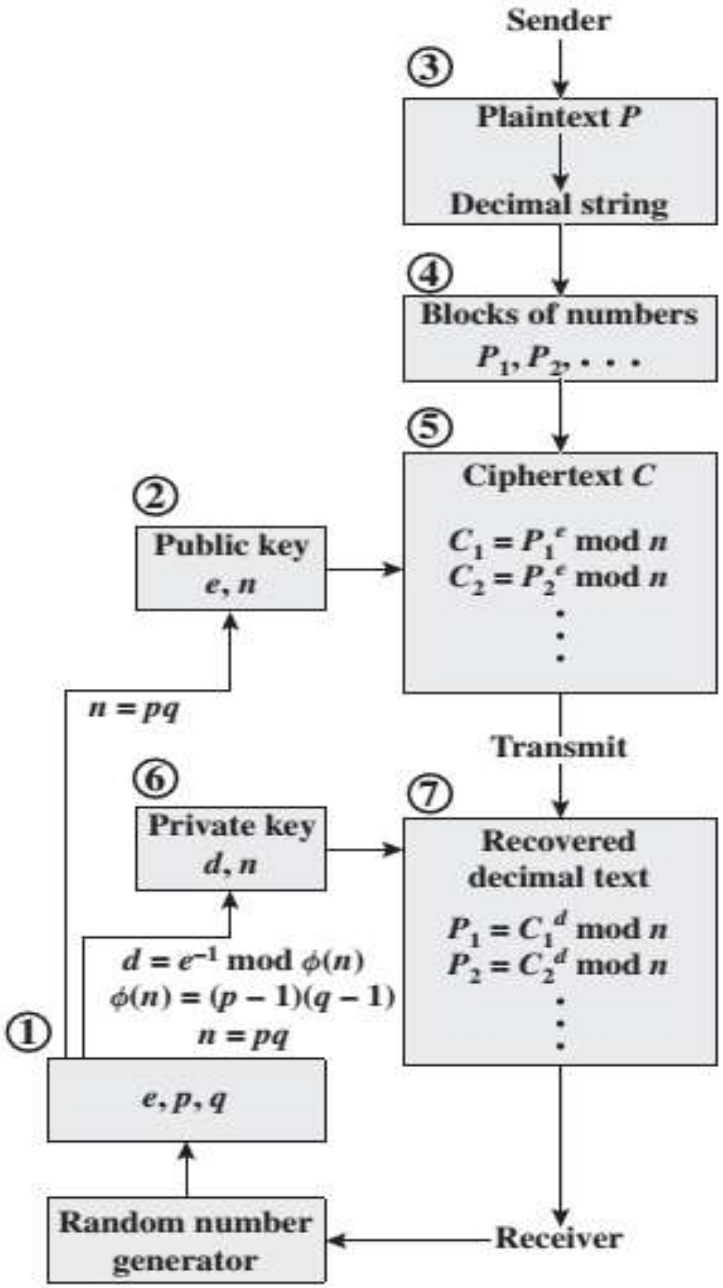
# Thuật toán RSA Thực hiện RSA

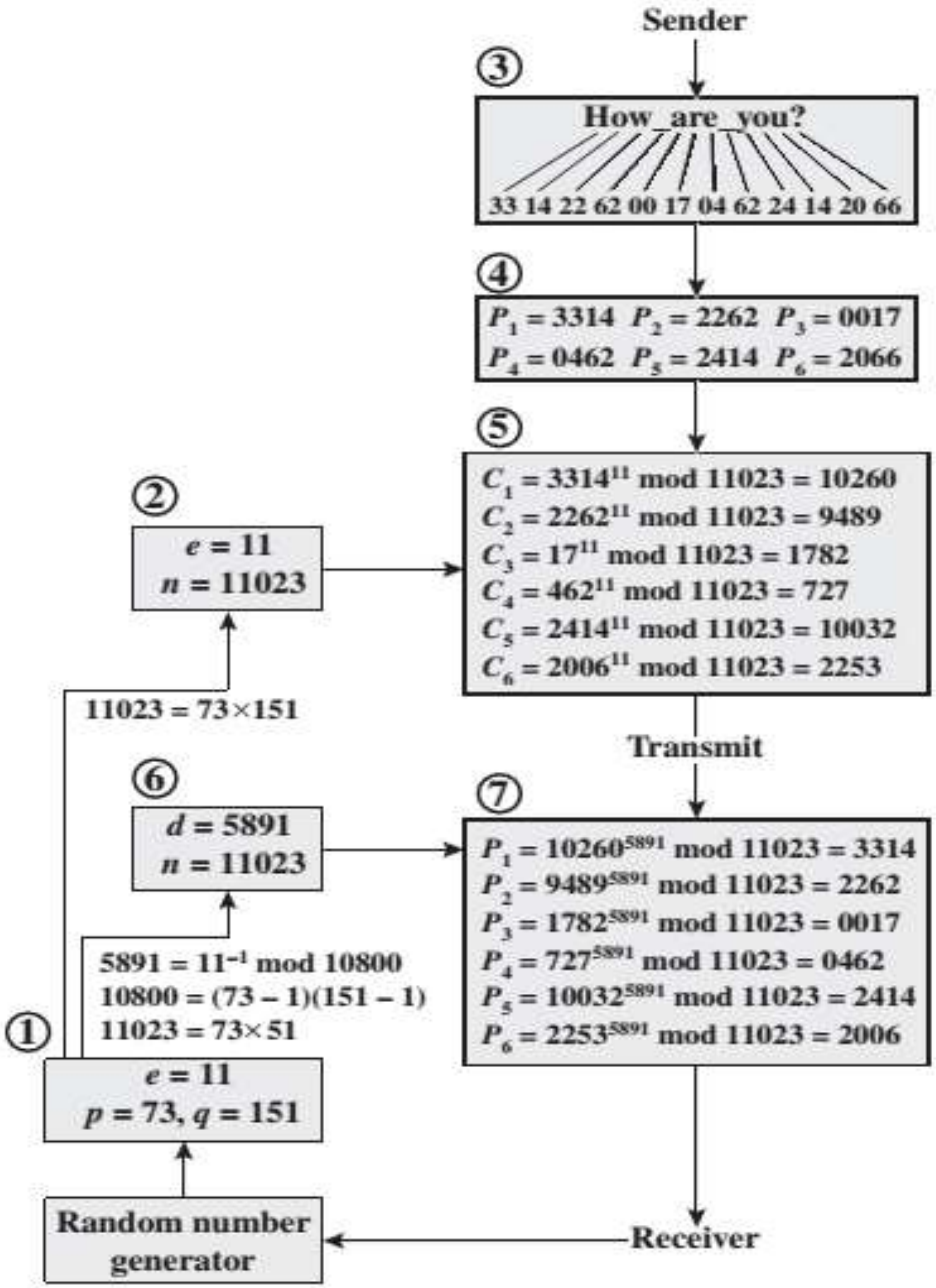
**Encryption by Bob with Alice's Public Key**

| | |
|---|---|
| Plaintext: | $M < n$ |
| Ciphertext: | $C = M^e \bmod n$ |

**Decryption by Alice with Alice's Public Key**

| | |
|---|---|
| Ciphertext: | $C$ |
| Plaintext: | $M = C^d \bmod n$ |


Sender
③
Plaintext $P$
Decimal string
④
Blocks of numbers
$P_1, P_2, \ldots$
⑤
Ciphertext $C$
$C_1 = P_1^e \bmod n$
$C_2 = P_2^e \bmod n$
②
Public key
$e, n$
$n = pq$
Transmit
⑥
Private key
$d, n$
⑦
Recovered decimal text
$P_1 = C_1^d \bmod n$
$P_2 = C_2^d \bmod n$
$d = e^{-1} \bmod \phi(n)$
$\phi(n) = (p-1)(q-1)$
$n = pq$
①
$e, p, q$
Random number generator
Receiver


Sender
③
How_are_you?
33 14 22 62 00 17 04 62 24 14 20 66
④
$P_1 = 3314$ $P_2 = 2262$ $P_3 = 0017$
$P_4 = 0462$ $P_5 = 2414$ $P_6 = 2066$
⑤
$C_1 = 3314^{11} \bmod 11023 = 10260$
$C_2 = 2262^{11} \bmod 11023 = 9489$
$C_3 = 17^{11} \bmod 11023 = 1782$
$C_4 = 462^{11} \bmod 11023 = 727$
$C_5 = 2414^{11} \bmod 11023 = 10032$
$C_6 = 2006^{11} \bmod 11023 = 2253$
②
$e = 11$
$n = 11023$
$11023 = 73 \times 151$
Transmit
⑥
$d = 5891$
$n = 11023$
⑦
$P_1 = 10260^{5891} \bmod 11023 = 3314$
$P_2 = 9489^{5891} \bmod 11023 = 2262$
$P_3 = 1782^{5891} \bmod 11023 = 0017$
$P_4 = 727^{5891} \bmod 11023 = 0462$
$P_5 = 10032^{5891} \bmod 11023 = 2414$
$P_6 = 2253^{5891} \bmod 11023 = 2006$
$5891 = 11^{-1} \bmod 10800$
$10800 = (73-1)(151-1)$
$11023 = 73 \times 51$
①
$e = 11$
$p = 73, q = 151$
Random number generator
Receiver

# 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 $n$
4. Tính nghịch đảo của $e$ trong phép modulo $\phi(n)$ được $d = 7$ ($3x7 = 21$) →gcd($\phi(n)$ , $e$) →gcd(20,3)=1
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

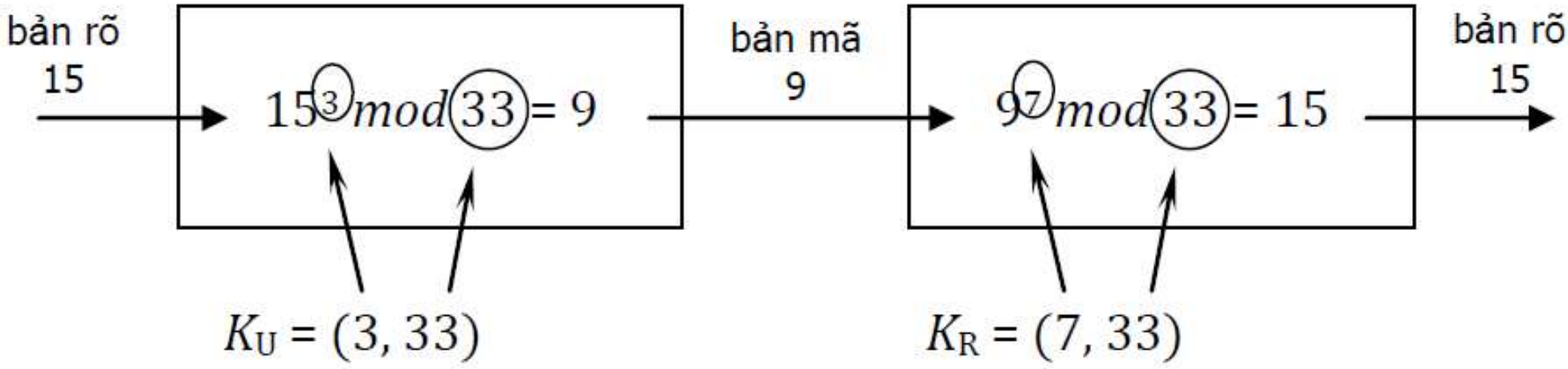
*Theo phương án 1 (mã hóa bảo mật):*

6) Mã hóa bản rõ $M = 15$:

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

7) Giải mã bản mã C = 9:

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



# Mã hóa và Giải mã RSA

*Theo phương án 2 (mã hóa chứng thực):*

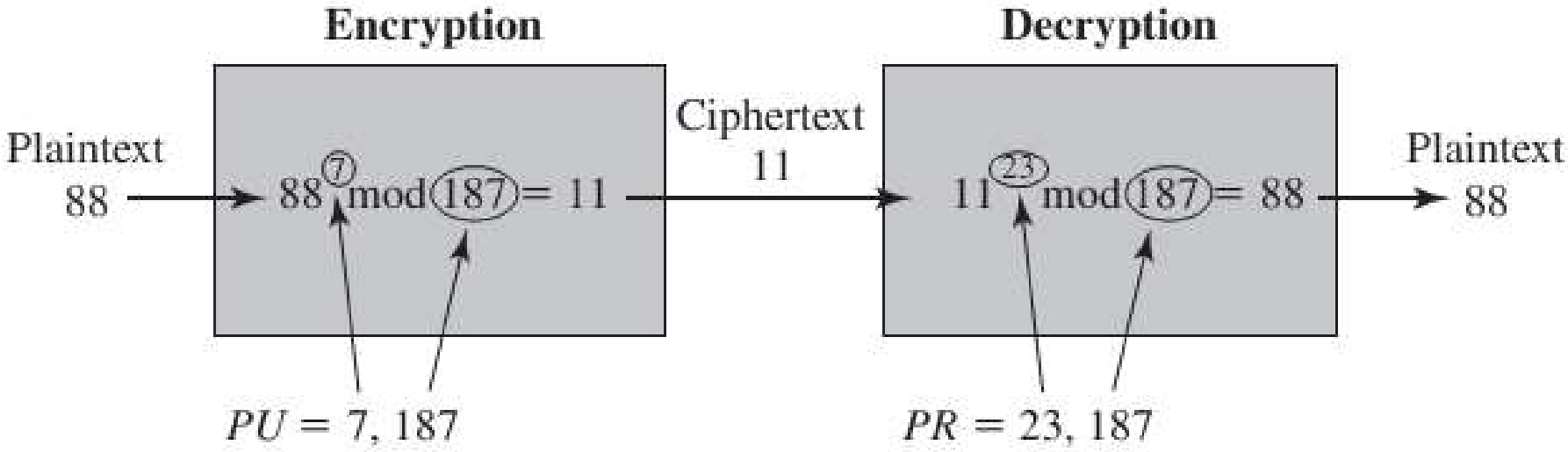
6) Mã hóa bản rõ $M = 15$:

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

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

# Bài tập

1)Tìm cặp khóa bí mật và công khai với p=7 và q=19. Thực hiện mã hóa và giải mã với M=DHKHMT (11,08,24,22,41,05) đảm bảo tính bí mật, tính toàn vẹn/chứng thực và cả hai.

2) Xây dựng mã giả cho thuật toán RSA

3) Demo thuật toán RSA

# Giải thuật tính $a^c \bmod n$

```
1. c = 0;
2. d = 1;
3. for i = k downto 1 do
6.       if bi = 1 then
4.                c = c × 2 + 1;
5.                d = (d × d × a) mod n;
6.       else
7.                c = c × 2;
8.                d = (d × d) mod n;
9. return d;
```

# 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)

# An ninh của hệ mã hóa RSA

**Hiện tượng lộ bản rõ:**

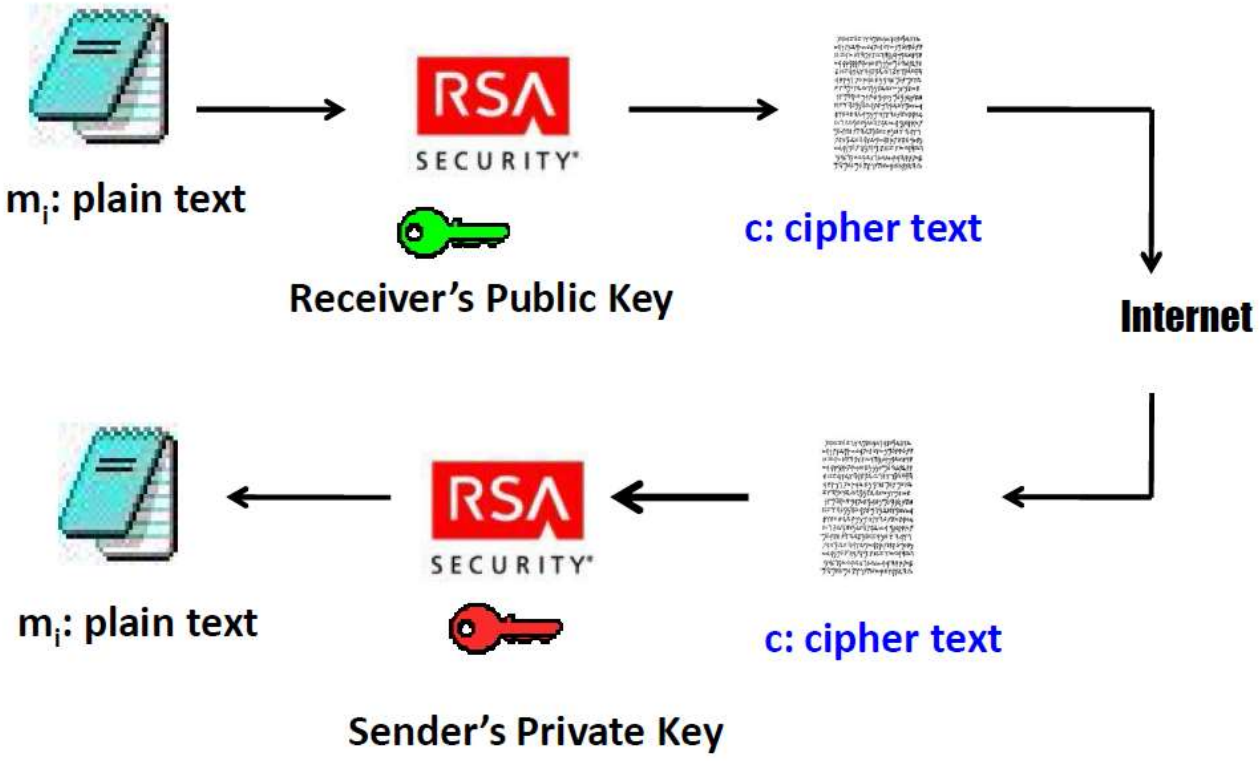
- Hệ mã RSA có N = p*q = 5*7, e = 17, với m = 6 ta có

$$C = 6^{17} \bmod N = 6.$$

- Hệ mã RSA có N = p*q = 109*97, e = 865, với mọi m ta đều có $m^e \bmod N = M$.
- Với hệ mã RSA có N = p*q và e bất kỳ, số lượng bản rõ bị lộ mã hóa sẽ là (1 + (e-1, p-1))*(1 + (e-1, q-1)).
- Trong thực tế RSA thường được sử dụng với các thông điệp có kích thước nhỏ (secsion key), và thường sử dụng lai ghép với các hệ mật đối xứng (DES,AES…)

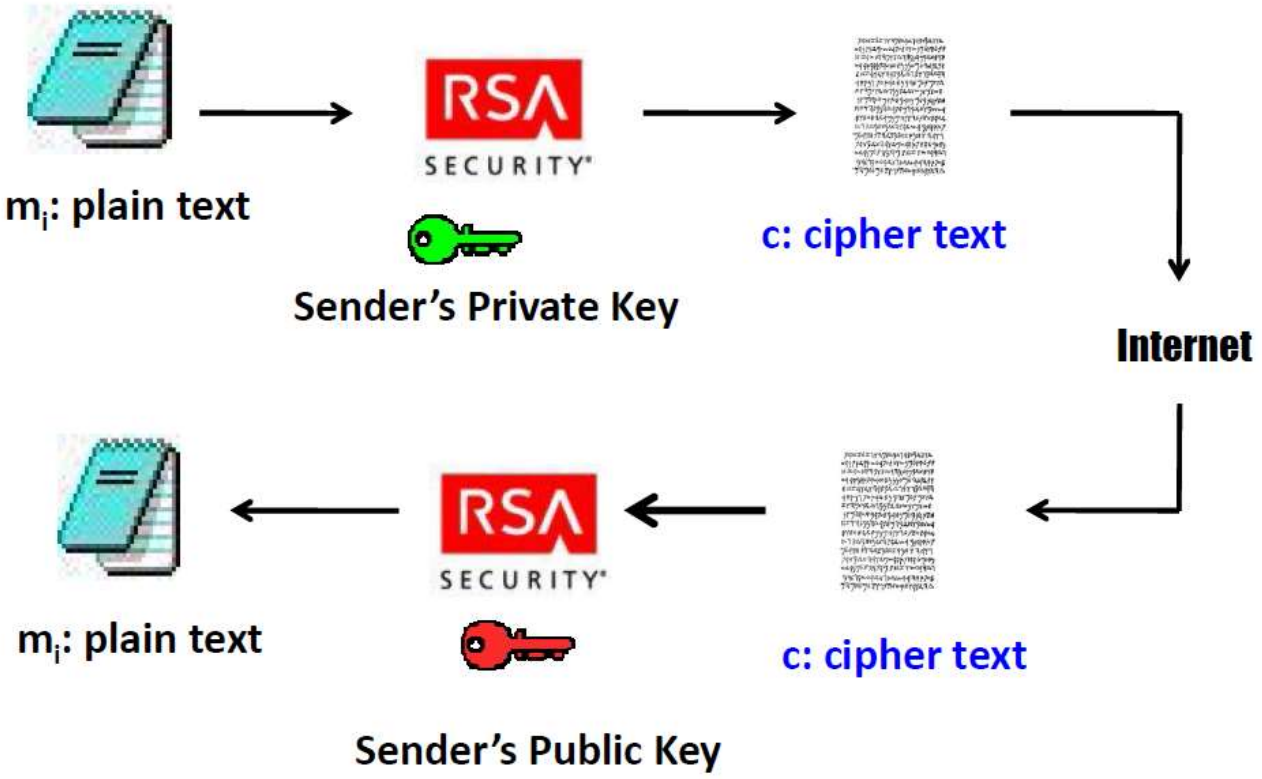
# Ứng dụng của hệ mã hóa RSA

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



# Phạm vi ứng dụng của hệ mã hóa RSA

- Mạng hành chính công, E-Business, E-Government
- Kinh doanh thương mại điện tử : Thanh toán điện tử, bảo mật các dữ liệu điện tử, chứng thực chữ ký điện tử. . .
- Đào tạo, thi cử từ xa,bảo mật dữ liệu tuyển sinh.
- Ngân hàng thương mại: Giao dịch, thanh toán qua mạng.
- Xuất nhập cảnh
- . . . . .

# 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 gởi cho Bob bản tin $M = 9$ vừa áp dụng chứng thực và bảo mật như ở sơ đồ 4-3. Hãy thực hiện quá trình mã hóa và giải mã.

# Bài tập

1. Cho N = 1517. Hãy tính 131435 mod N.
2. Trong hệ mã RSA có N = p * q = 103 * (219 – 1) thì có thể sử dụng tối đa là bao nhiêu gía trị của e để làm khóa mã hóa, giải thích.
3. Trong hệ mã RSA có N = p*q = 103 * 113 sẽ có bao nhiêu trường hợp lộ bản rõ.
4. Cho hệ RSA có n = 1363, biết phi(n) = 1288 hãy mã hóa bản rõ M = 2007.
5. Tương tự Câu 1 với n = 215629 và phi(n) = 214684 hãy giải mã bản mã M = 2007.

# Bài tập

6. 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 đợợc viết bằng tiếng Anh ngợời 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 đợợc 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 đợợc là một tập các số ∈ZN. Hãy thực hiện mã hóa xâu P = ”SERIUS”.
- c) Giả sử bản mã thu đợợc là C = <201, 793, 442, 18> hãy thực hiện giải mã để tìm ra thông điệp bản rõ ban đầu.

# Bài tập

7. Cho hệ mã RSA có p = 29, q = 43, e = 11.

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

b) Để mã hóa các thông điệp đợợc viết bằng tiếng Anh ngợời 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 đợợc 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 đợợc là một tập các số ∈ ZN. Hãy thực hiện mã hóa xâu P = ”TAURUS”.
- c) Giả sử bản mã thu đợợc là C = <1, 169, 1206, 433> hãy thực hiện giải mã để tìm ra thông điệp bản rõ ban đầu.

# Bài tập

8. Cho hệ mã RSA có n = 1363, e = 57.

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

b) Giả sử bản rõ P = 102 hãy mã hóa và đưa ra bản mã C.

c) Giả sử hệ mã trên được dùng làm hệ chữ ký điện tử, hãy tính chữ ký với thông điệp M = 201

# 3. Mã 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 $\alpha$ 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 $(0 \leq a_u, a_V \leq p-2)$, và tính giá trị $b_U, b_V$ công khai tương ứng :

$$b_U = \alpha^{a}{}_{u} \bmod p \text{ và}$$

$$b_V = \alpha^{a}{}_{v} \bmod p$$

- 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 $\alpha$ 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 $\alpha = 2$ biết công khai ($p$ là SNT và $\alpha$ 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: $\alpha$
- Bên tạo khóa (A) chọn giá trị bí mật Xa (Xa<q-1) và tính giá trị $Ya=\alpha^{X}_{A} \bmod q$. Khi đó, bộ khóa K={PU,PR} của A, với khóa công khai $PU=\{q, \alpha, Y_A\}$ 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 $K = (Y_A)^K \bmod q$, $C_1 = \alpha^k \bmod q$, $C_2 = KM \bmod q$. Khi đó (C1,C2) là bản mã được truyền đi
- Quá trình bên nhận (A) giải mã:
  - Tính khóa $K = (C_1)^{XA} \bmod q$
  - Tìm bản gốc theo công thức: $M = (C_2 K^{-1}) \bmod q$

# 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

# So sánh chiều dài khóa ứng với an toàn tương đương

| **Symmetric scheme (key size in bits)** | **ECC-based scheme (size of *n* in bits)** | **RSA/DSA (modulus size in bits)** |
|---|---|---|
| 56 | 112 | 512 |
| 80 | 160 | 1024 |
| 112 | 224 | 2048 |
| 128 | 256 | 3072 |
| 192 | 384 | 7680 |
| 256 | 512 | 15360 |

# Câu hỏi và bài tập

1. Khái niệm mã hóa khóa công khai, cơ chế, các thành phần của hệ mã hóa công khai
2. Các đặc điểm và yêu cầu của hệ mã hóa công khai
3. Nêu nguyên tắc của mã hóa khóa công khai? Tại sao trong mã hóa khóa công khai không cần dùng đến kênh an toàn để truyền khóa?
4. Trong mã hóa khóa công khai, khóa riêng và khóa công khai có phải là 2 khóa tùy ý, không liên quan? Nếu có liên quan, tại sao không thể tính khóa riêng từ khóa công khai?

# Câu hỏi và bài tập

5. Ngoài vấn đề truyền khóa, mã hóa khóa công khai còn ưu điểm hơn mã hóa đối xứng ở điểm nào?
6. Nêu nhược điểm của mã hóa khóa công khai.
7. Hãy nêu các vấn đề về RSA
8. Cho các cặp số nguyên tố sau: (13,23); (11,19); (23,29). Hãy thực hiện các bước phát sinh khóa để đưa ra khóa công khai, khóa bí mật
9. Dùng các cặp khóa trên để mã hóa thông điệp có chiều dài 88
10. Thế nào là độ an toàn của một thuật toán mã hóa?

# THANKS YOU

*INNOVATION - UNITY - HUMANITY*

www.iuh.edu.vn