Bài 5:
Mã khoá công khai và RSA

Thời lượng: 6 tiết

Lương Thái Lê

Nội dung

- Khái niệm mã công khai
- Mã công khai RSA
- Trao đổi khóa Diffie-Hellman

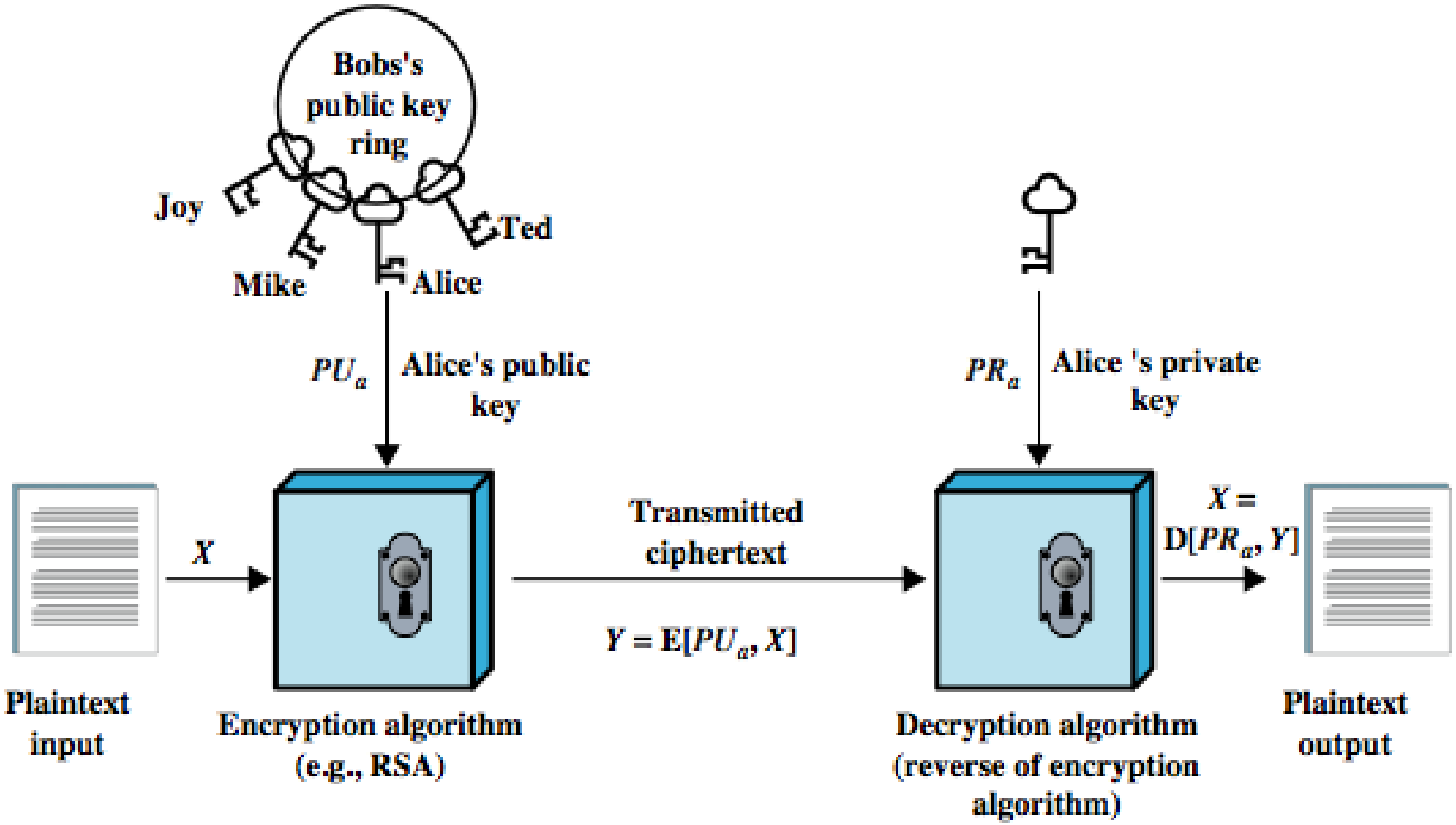
Giới thiệu về Mã khoá công khai

- Mã khối đối xứng hiện đại có một số yếu điểm:
  - Phải bảo vệ bí mật cho khóa
  - Không đảm bảo tính xác thực

=>Whitfield Diffie & Martin Hellman ở Đại học Stanford phát minh ra Mã khóa công khai (1976)

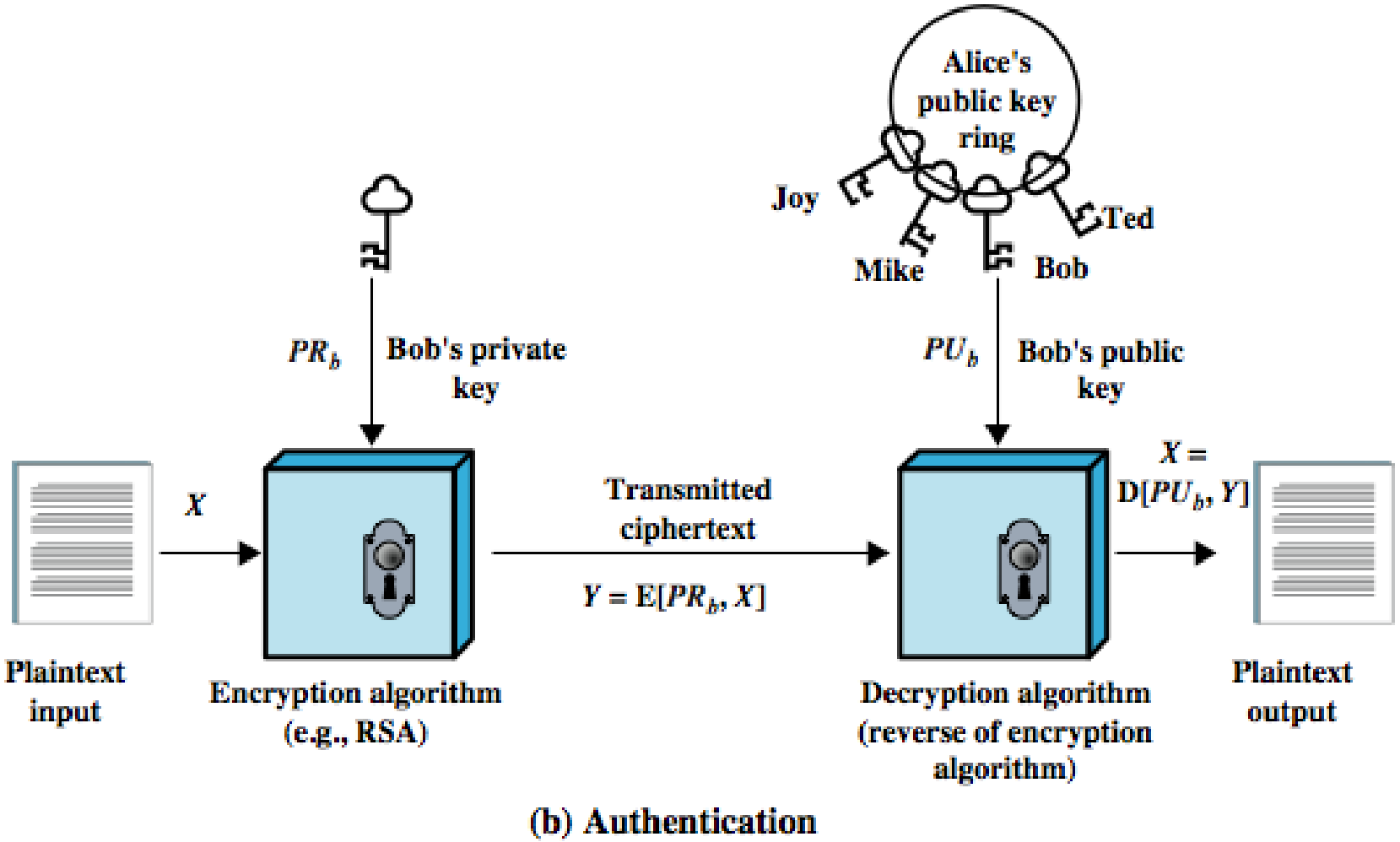
- Mỗi người tham gia hệ mã công khai tạo cặp 2 khoá:
  - **Khoá công khai** (public key)
  - **Khoá riêng** (private key)
- Không đối xứng vì hai phía không như nhau
- Được dùng cho vấn đề bảo mật và xác thực
- Hệ mã hóa sử dụng mã công khai phổ biến nhất là RSA
- Hỗ trợ thêm chứ không phải thay thế khoá riêng.

Sử dụng Mã khóa công khai để bảo mật



(a) Confidentiality

Sử dụng Mã khóa công khai để xác thực



(b) Authentication

2. Các yêu cầu cho hệ mật mã khóa công khai

- Các điều kiện được Diffie và Hellman đưa ra [DIFF76b].

1. Dễ dàng tạo ra một cặp khóa (PU, PR).
2. Dễ dàng mã hóa M để tạo ra bản mã tương ứng: C = E (PU, M)
3. Dễ dàng giải mã C để phục hồi M = D (PR, C) = D [PR, E (PU, M)]
4. Không thể thực hiện được đối với kẻ tấn công, dù biết PU_b cũng không thể xác định PR_b.
5. Tính toán không thể thực hiện được đối với kẻ tấn công, dù biết PU, và một bản mã C, cũng không khôi phục lại được M.
6. Hai khóa có thể được áp dụng theo thứ tự tùy ý: M = D[PUb, E(PRb, M)] = D[PRb, E(PUb, M)].

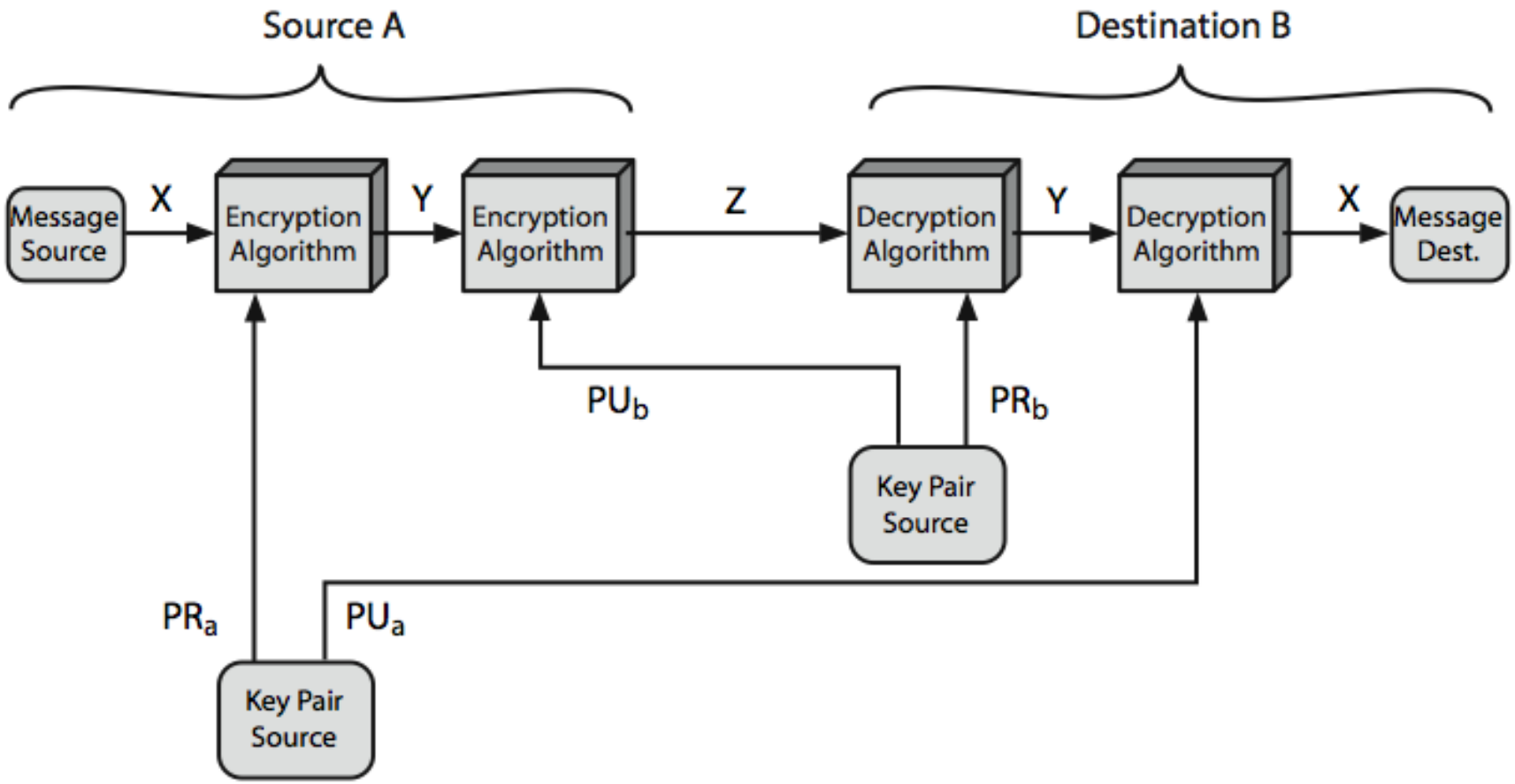
Ứng dụng khoá công khai

- Có thể phân loại ứng dụng thành 3 loại:
  - Mã/giải mã: dùng trong bảo mật
  - Chữ ký điện tử: dùng trong xác thực
  - Trao đổi khoá
- Một số thuật toán phù hợp với mọi ứng dụng, còn một số chuyên dùng cho ứng dụng cụ thể

Table 9.3 Applications for Public-Key Cryptosystems

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

Mô hình hệ thống Bảo mật và Xác thực sử dụng mã khóa công khai



Mã công khai RSA

- Được sáng tạo bởi Rivest, Shamir & Adleman ở MIT vào năm 1977
- Là mã công khai được biết đến nhiều nhất và sử dụng rộng rãi nhất
- Dựa trên lũy thừa trên trường hữu hạn các số nguyên modulo nguyên tố
- Là mã khối, mỗi khối là một số nguyên thuộc đoạn [0;n-1] (n thường <= 2^{1024}) => kích thước khối ≤ $\log_2(n)+1$
- Là mã công khai an toàn nhất cho đên nay

Nguyên lý thuật toán mã hóa RSA

- Có M là plaintext block; C là ciphertext block. Khi đó:

$$C = M^e \bmod n$$
$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

$\Rightarrow$Khóa công khai PU = {e,n}

$\Rightarrow$Khóa riêng PR = {d,n}

Khởi tạo khóa RSA

1. Sinh khóa (Alice)	Ví dụ
Chọn p, q là hai số nguyên tố khác nhau	p = 17 & q = 11
Tính n = pq	n = pq =17 x 11=187
Tính $\phi(n) = (p - 1)(q - 1)$	ø(n) = 16x10=160
Chọn số nguyên e, gcd ($\phi(n)$, e) = 1; 1 <e < $\phi(n)$	Chọn e = 7 thỏa mãn gcd(e,160)= 1
Tính $d \equiv e^{-1} \pmod{\phi(n)}$	d = 23 vì: 23x7(mod 160)=161 mod 160=1
Khóa công khai: PU = {e, n}	PU = {7,187}
Khóa riêng: PR = {d, n}	PR = {23,187}

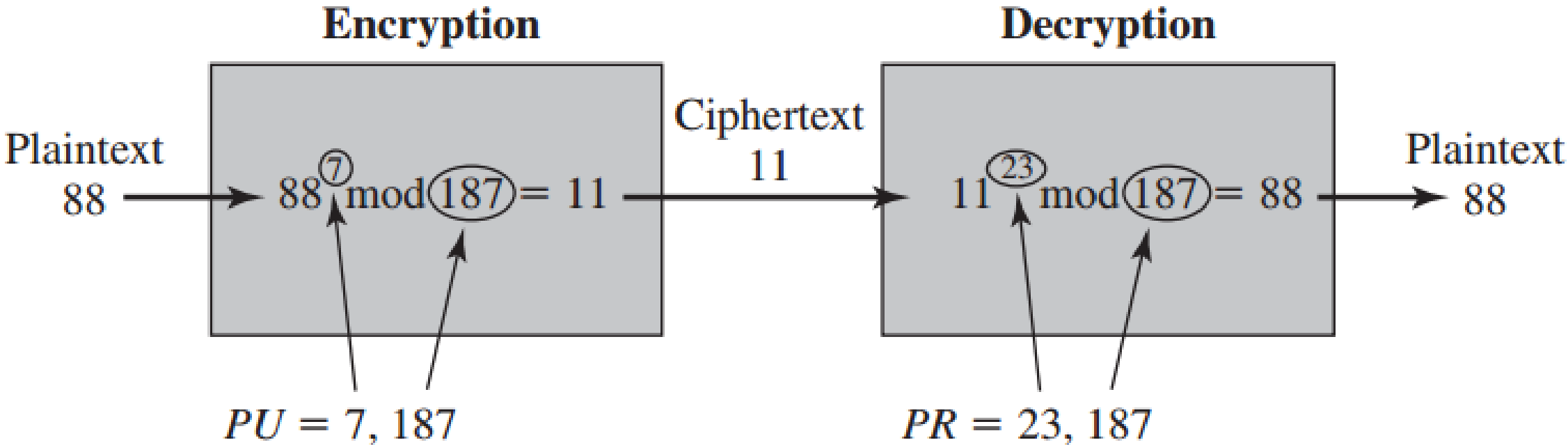
Mã hóa và Giải mã với RSA

2. Bob mã hóa với khóa công khai của Alice	
Bản rõ	$M < n$
Bản mã	$C = M^e \bmod n$

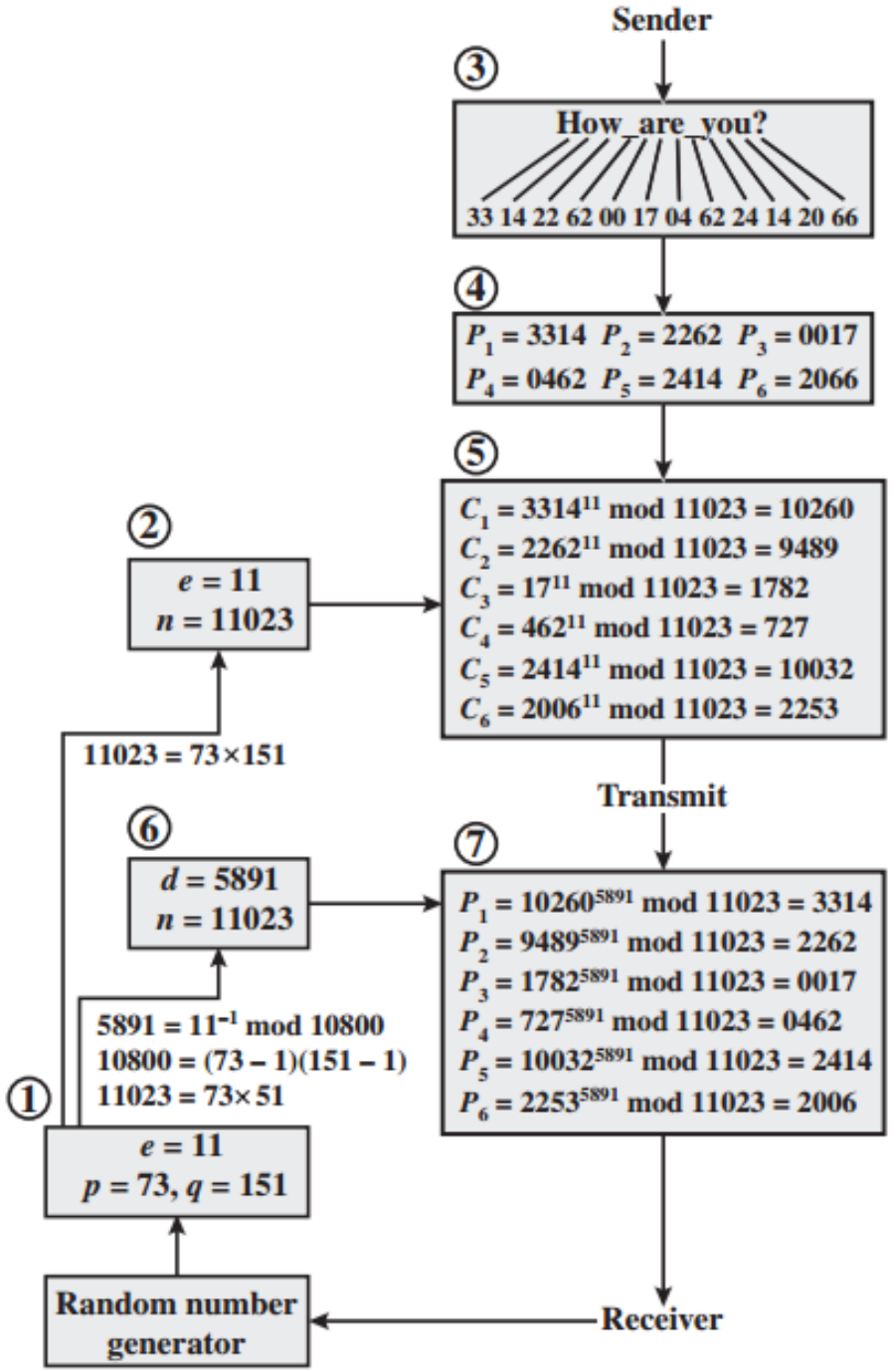
Ví dụ
$M = 88$
$C = 88^7 \bmod 187 = 11$

3. Alice giải mã bằng khóa riêng của Alice	
Bản mã	C
Bản rõ	$M = C^d \bmod n$

Ví dụ
$C = 11$
$M = 11^{23} \bmod 187 = 88$


Encryption
Decryption
Plaintext
88
$88^{7} \bmod 187 = 11$
Ciphertext
11
$11^{23} \bmod 187 = 88$
Plaintext
88
$PU = 7, 187$
$PR = 23, 187$

RSA Example


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

An toàn của mã RSA

- Độ an toàn của RSA phụ thuộc vào độ khó của bài toán phân tích số nguyên tố lớn.

=> n phải là 1 số nguyên đủ lớn để khó bị phân tích

- Mã sử dụng lũy thừa của e => Nếu e nhỏ thì sẽ nhanh, nhưng không an toàn

=>Phổ biến nhất là chọn e = 65537= $(2^{16} + 1)$

  - Nếu e cố định thì cần chọn n sao cho: $\gcd(e,\phi(n))=1$

- Mã hóa lâu hơn DES, AES

$\Rightarrow$thường được dùng để mã hóa khóa bí mật (độ dài ngắn)

- RSA là phương pháp mã hóa xác định (ko ngẫu nhiên) nên dễ bị tấn công lựa chọn bản rõ

Thuật toán Trao đổi khoá Diffie-Hellman

- Là thuật toán mã hóa công khai đầu tiên, được Diffie và Hellman đề xuất năm 1976
- Là mã khoá công khai giúp giải bài toán phân phối khoá an toàn
- Dựa trên nguyên lý : bài toán tìm dlog là khó
- Sử dụng mã khoá công khai để phân phối khoá mật

Thuật toán Diffie – Hellman và ví dụ

1. Các giá trị công khai chung
q là số nguyên tố
a là một căn nguyên thủy của q, a < q

Ví dụ
q =353
a = 3

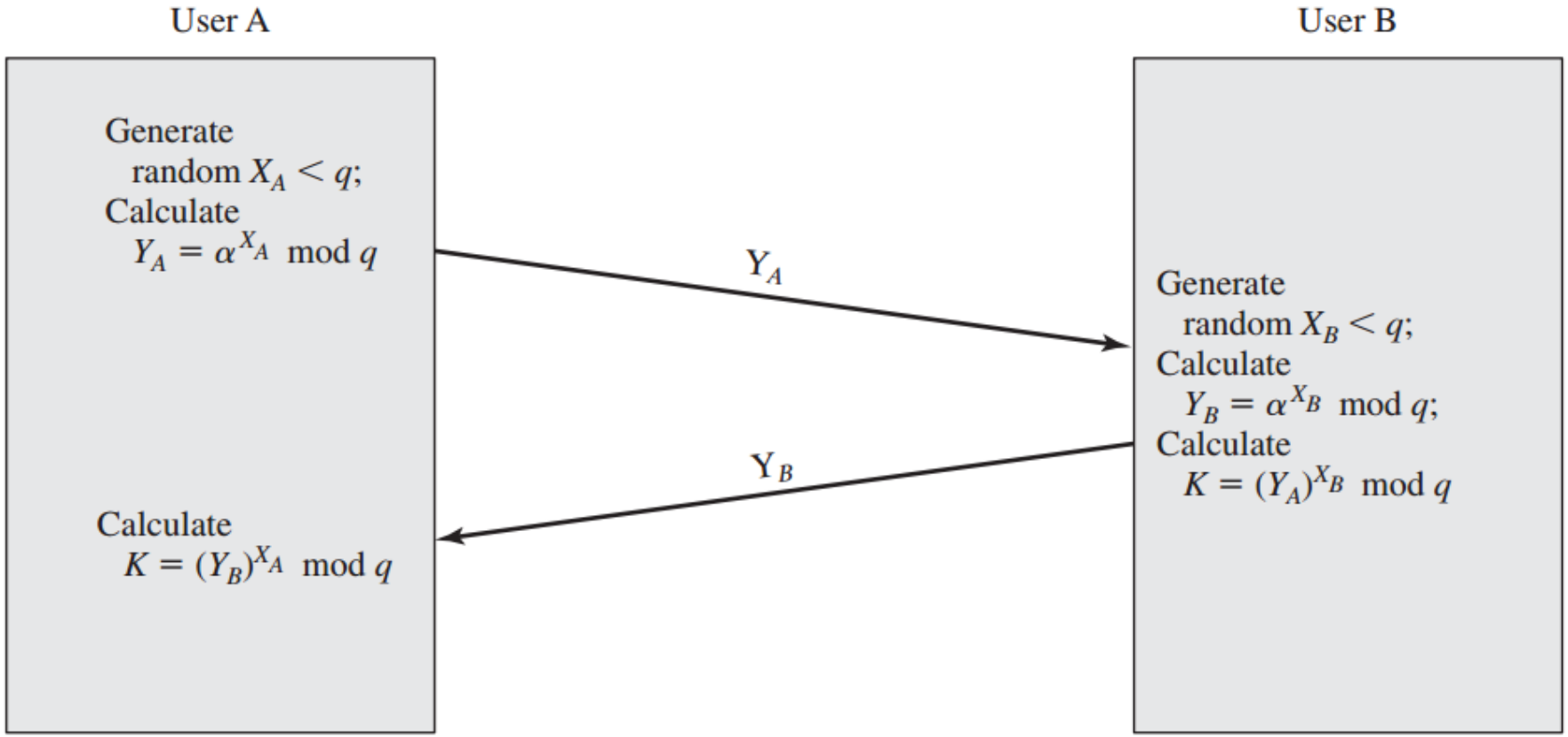
2. Alice tạo khóa
Chọn khóa riêng $X_A < q$
Tính khóa công khai $Y_A = a^{XA} \bmod q$

Ví dụ
$x_A = 97$
$y_A = 3^{97} \bmod 353 = 40$

2. Bob tạo khóa
Chọn khóa riêng $X_B < q$
Tính khóa công khai $Y_B = a^{XB} \bmod q$

Ví dụ
$x_B = 233$
$y_B = 3^{233} \bmod 353 = 248$

Trao đổi khóa Diffie-Hellman



- $K = \alpha^{x_A . x_B} \bmod q = ?$ **160**
- K được sử dụng như khóa phiên (khóa bí mật chung)

Tấn công vào hệ trao đổi khóa Diffie-Hellman

- Mô hình trao đổi khóa này không có tính xác thực nên dễ bị giả mạo:
  - David tạo ra 2 khóa riêng X_{D1} và X_{D2} rồi tính 2 khóa công khai tương ứng Y_{D1} và Y_{D2}
  - Khi Alice gửi PU Y_A cho Bob thì David chặn lấy, thay vào đó David gửi Y_{D1} cho Bob. Đồng thời David tính $K_2 = (Y_A)^{X_{D2}} \ mod \ q$
  - Bob nhận được Y_{D1} và tính $K_1 = (Y_{D1})^{X_B} \ mod \ q$
  - Bob gửi Y_B cho Alice
  - David chặn lấy Y_B thay vào đó gửi Y_{D2} cho Alice. Đồng thời David tính $K_1 = (Y_B)^{X_{D1}} \ mod \ q$
  - Alice nhận dược Y_{D2} và tính khóa PR $K_2 = (Y_{D2})^{X_A} \ mod \ q$

  => David và Alice chung khóa K_2 còn David và Bob chung khóa K_1

Mật mã Elgaman

- Được đề xuất bởi T.Elgaman năm 1984
- Mã Elgaman được dùng trong chuẩn chữ ký số (Digital Signature Standard – DSS) và email standard S/MIME

1. Các giá trị công khai chung
q Là số nguyên tố
a là một căn nguyên thủy của q (a < q)

Ví dụ
q = 19
a =10

2. Alice tạo khóa
Chọn $X_A < q - 1$
Tính $Y_A = a^{XA} \bmod q$
Khóa công khai: PU = {q, a, Y_A}
Khóa riêng: X_A

Ví dụ
$X_A = 5$
$Y_A = a^5 \bmod 19 = 3$
{19, 10, 3}
5

Mật mã Elgaman

3. Bob muốn gửi tin nhắn cho Alice
Bản gốc: M < q
Chọn ngẫu nhiên k < q
Tính K= $(Y_A)^k$ mod q
Tính C_1= a^k mod q;
Tính C_2 = KM mod q
Bản mã: (C_1, C_2)

Ví dụ
M = 17
k = 6
K = 3^6 mod 19 = 7.
C_1 = 10^6 mod 19 = 11
C_2 = 7 x 17 mod 19 = 5
(11, 5)

4. Alice giải mã tin nhắn từ Bob
Bản mã: (C_1, C_2)
Tính K= $(C_1)^{XA}$ mod q
Bản rõ: M = (C_2K^{-1}) mod q

Ví dụ
(11, 5)
K = 11^5 mod 9 = 7.
7^{-1} mod 19 = 11. M = 5 x 11 mod 19 = 17.