

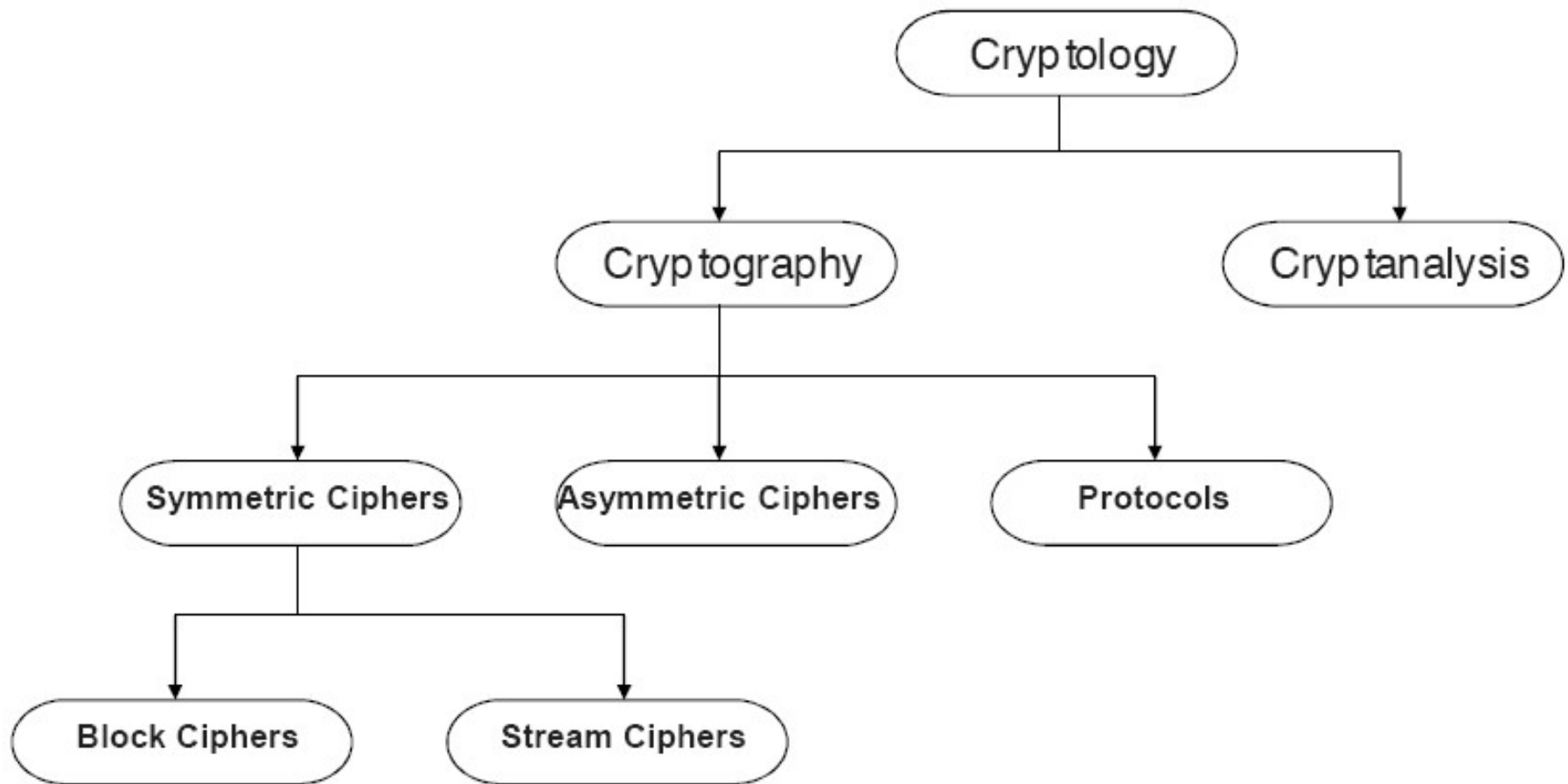
Information Security

CS3002

Lecture 7
9th September 2024

Dr. Rana Asif Rehman
Email: r.asif@lhr.nu.edu.pk

Classification of the Field of Cryptology



Adopted with thanks from: Chapter 1 of Understanding Cryptography by Christof Paar and Jan Pelzl

Private-Key Cryptography

- Traditional **private/secret/single key** cryptography uses **one** key.
- Shared by both sender and receiver.
- If this key is disclosed communications are compromised.
- Also is **symmetric**, parties are equal.
 - Hence does not protect sender from receiver forging a message & claiming is sent by sender.

Public-Key Cryptography

- **Public-key/two-key/asymmetric** cryptography involves the use of **two** keys:
 - A **public-key**, which may be known by anybody, and can be used to **encrypt messages**, and **verify signatures**.
 - A **private-key**, known only to the recipient, used to **decrypt messages**, and **sign** (create) **signatures**.
- Is **asymmetric** because
 - Those who encrypt messages or verify signatures **cannot** decrypt messages or create signatures.

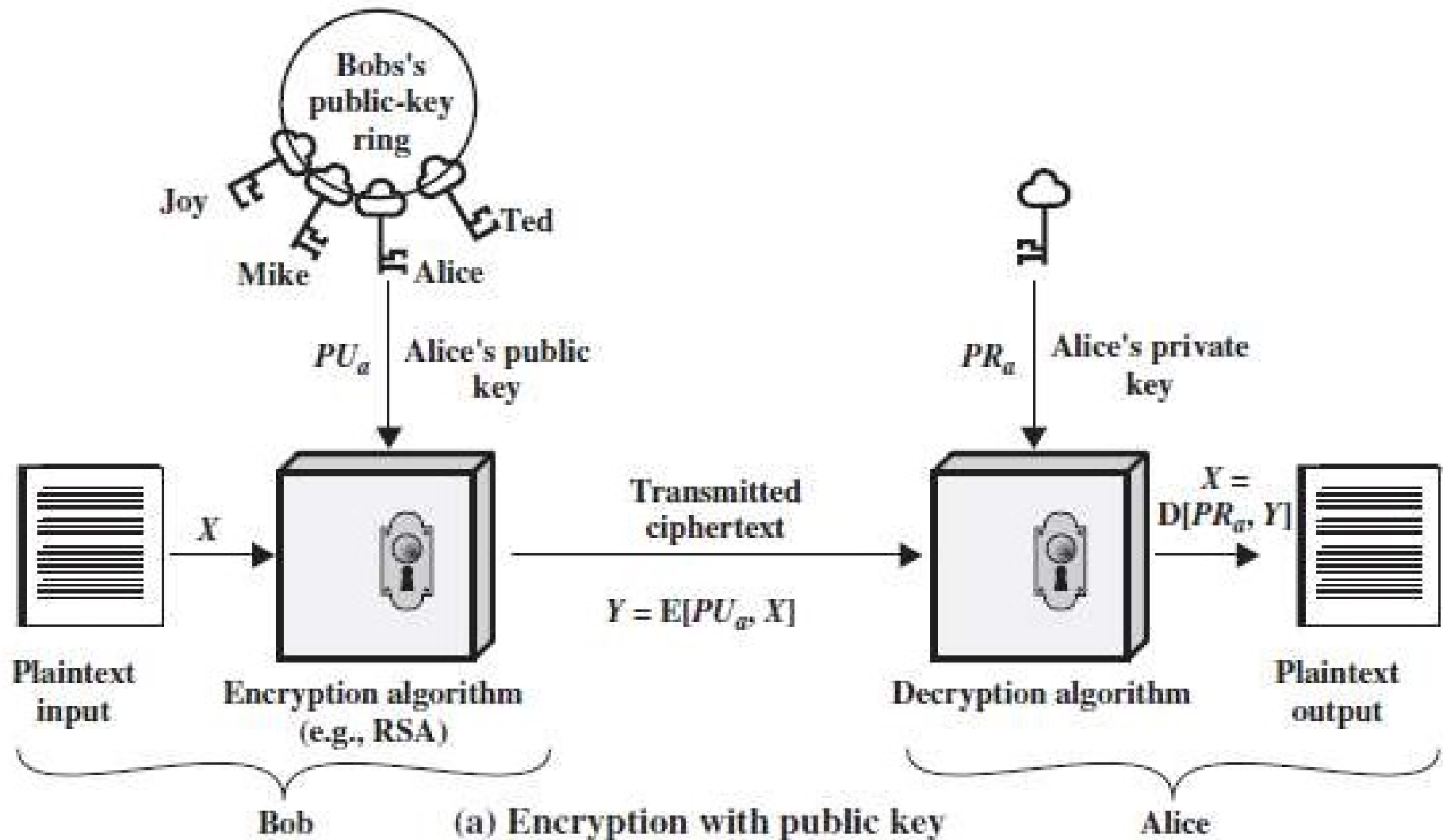
Need for Both

There is a very important fact that is sometimes misunderstood: The advent of asymmetric-key cryptography does not eliminate the need for symmetric-key cryptography.

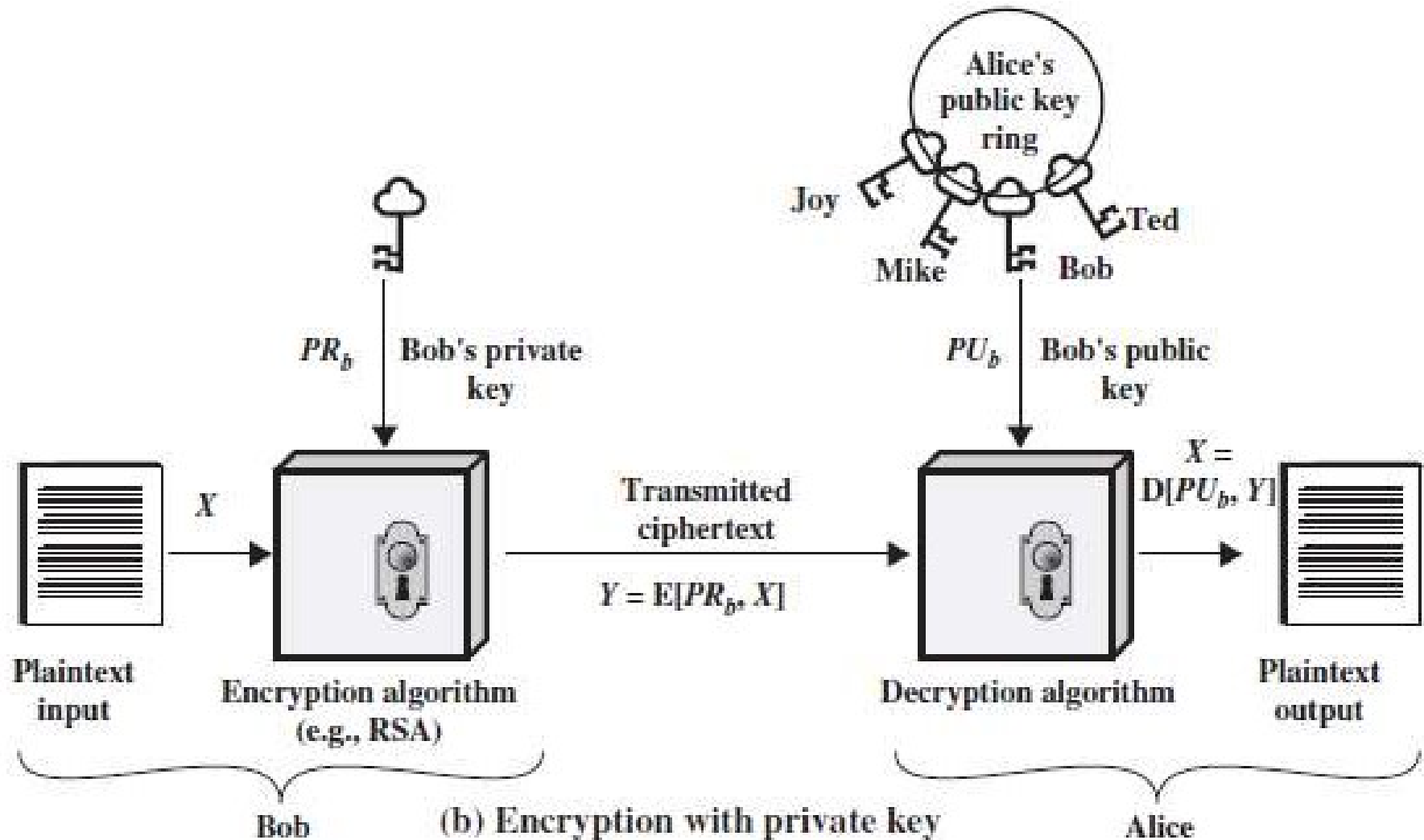
Why Public-Key Cryptography?

- Developed to address two key issues:
 - **Key distribution** – how to have secure communications in general without having to trust a KDC with your key.
 - **Digital signatures** – how to verify a message comes intact from the claimed sender.
- Public invention due to Whitfield Diffie & Martin Hellman at Stanford University in 1976
 - Known earlier in classified community.

Encryption



Authentication



Advantages

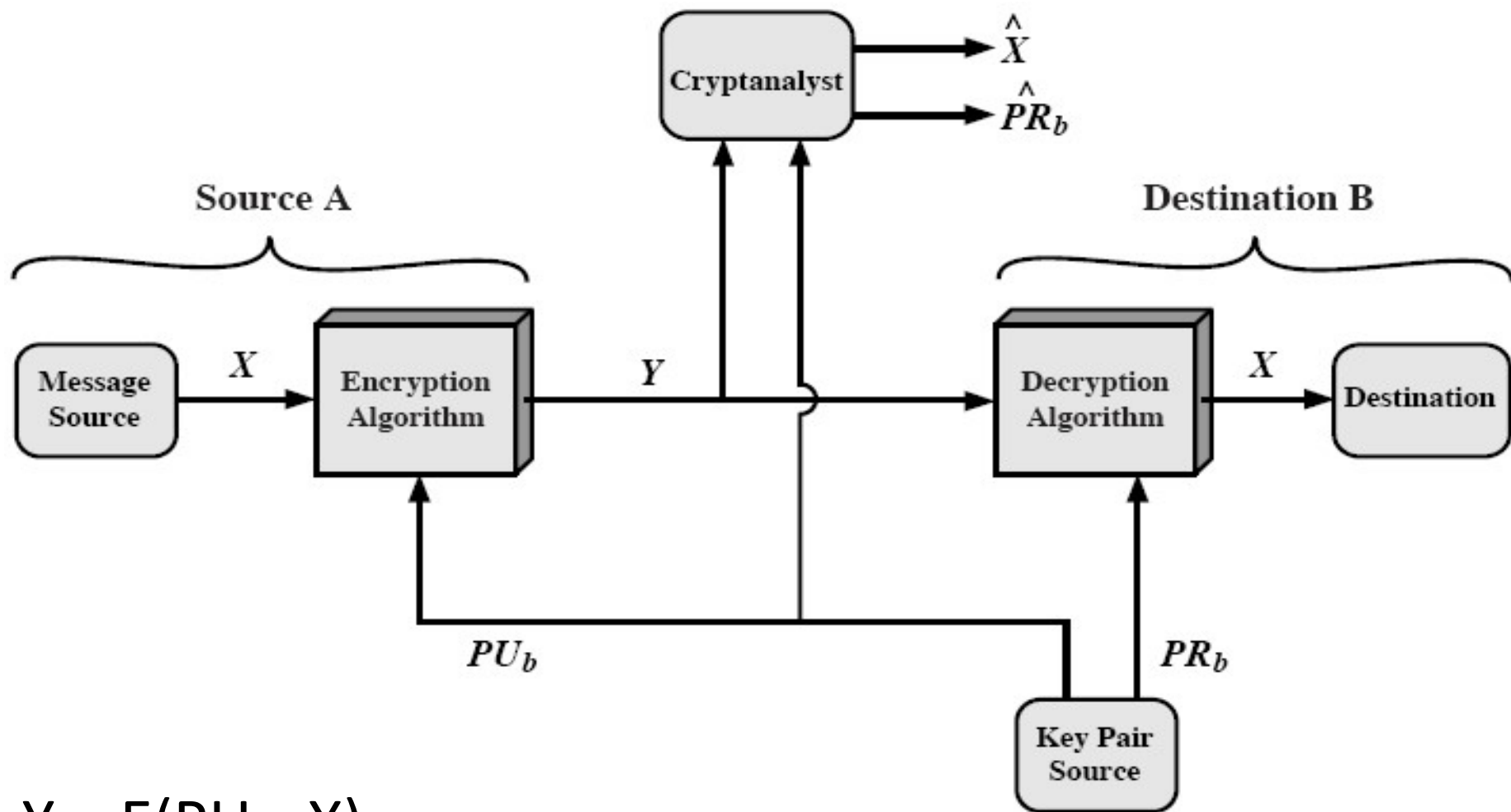
- Each user generates pair of keys.
- Place one of keys in public register or accessible file (public key).
- Private keys generated locally.
- Private key need not to be distributed.
- Keys can be changed at any time.
 - At any time, a system can change its private key and publish the companion public key to replace its old public key.

Public-Key Applications

- Can classify uses into 3 categories:
 - **Encryption/decryption** (provide secrecy)
 - **Digital signatures** (provide authentication)
 - **Key exchange** (of session keys)
- Some algorithms are suitable for all uses, others are specific to one.

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

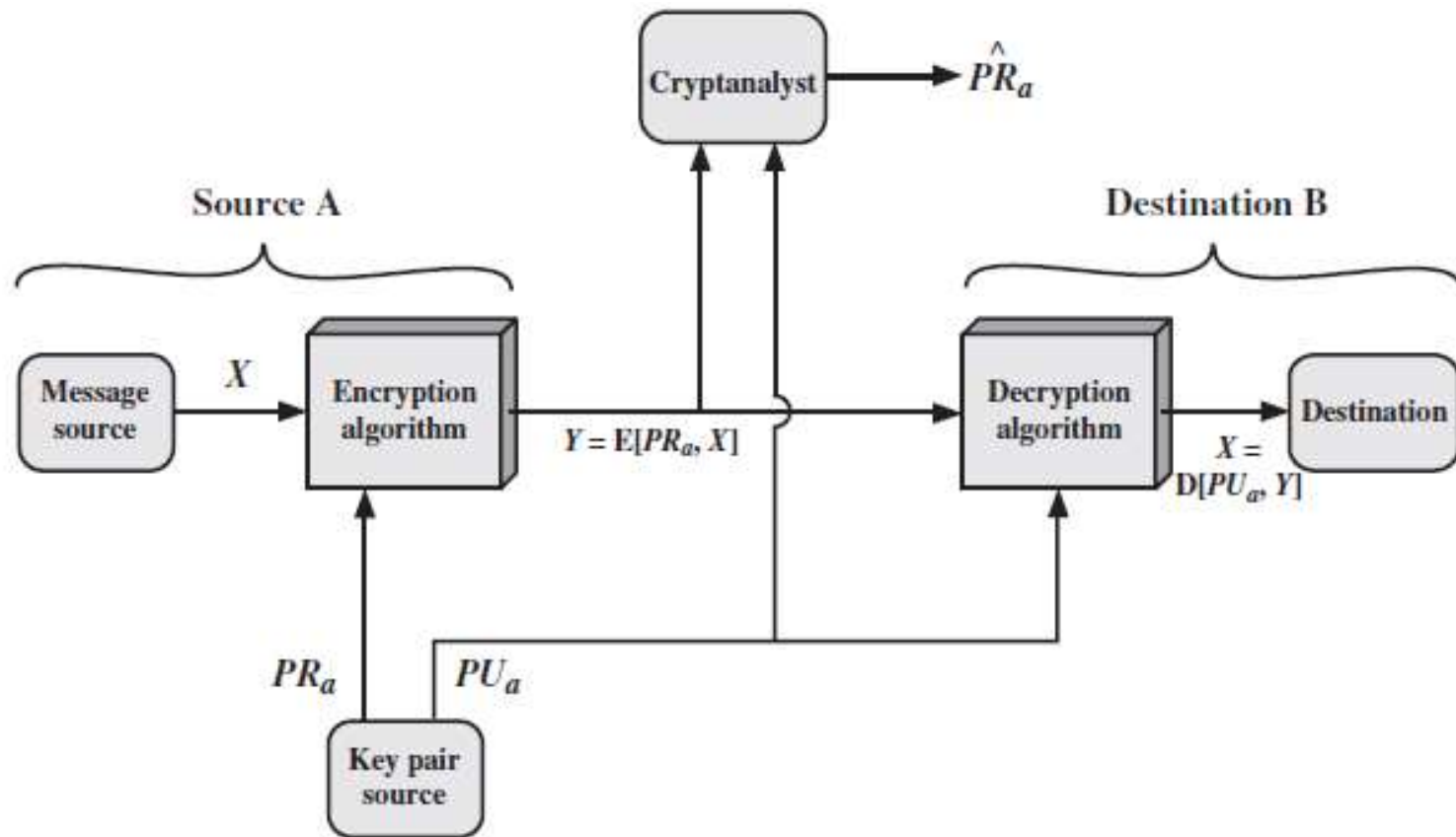
Applications: Confidentiality



$$Y = E(PU_b, X)$$

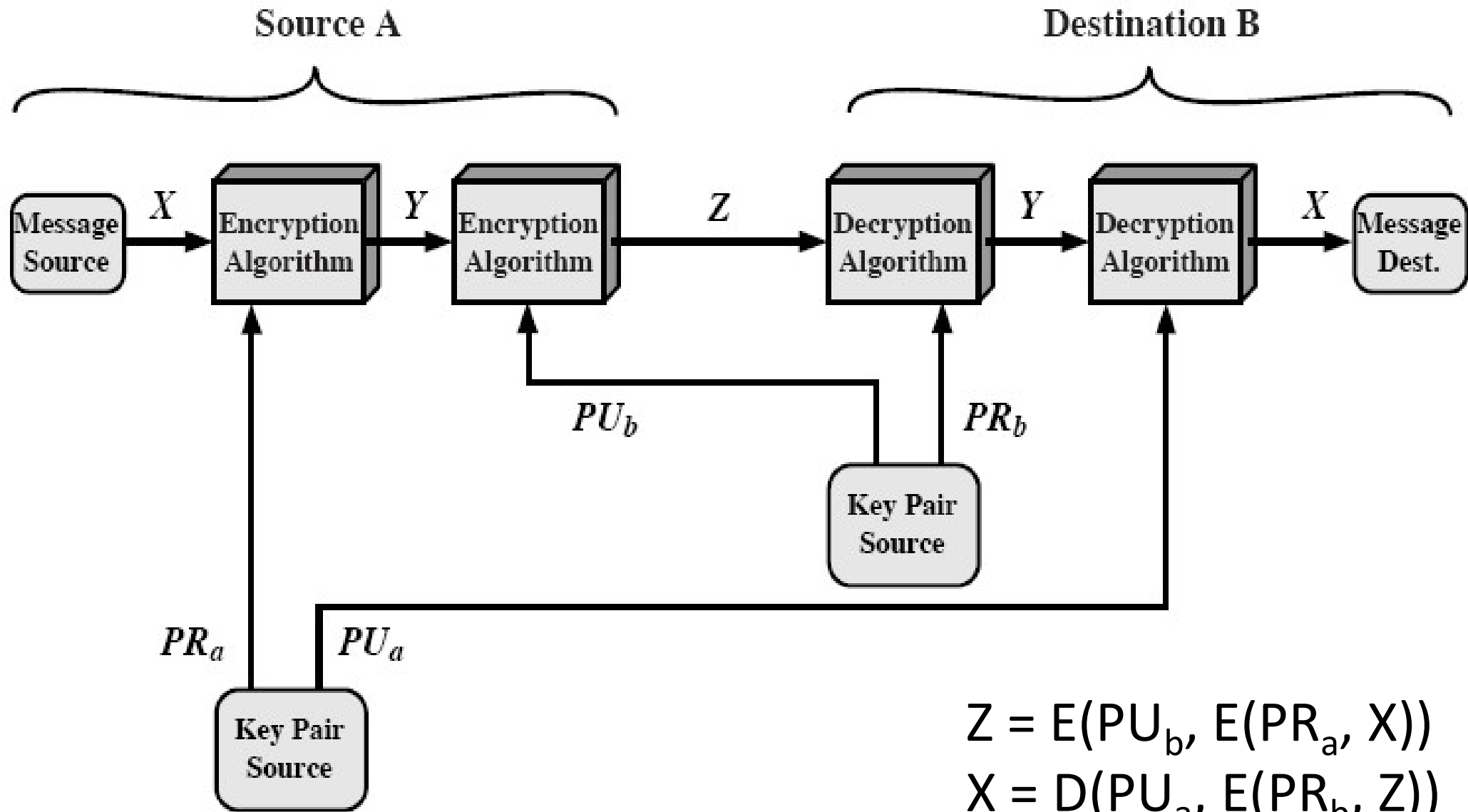
$$X = D(PR_b, Y)$$

Applications: Authentication



$$Y = E(PR_a, X)$$
$$X = D(PU_a, Y)$$

Applications: Confidentiality + Authentication



Requirements for Public-Key Cryptography (1/2)

1. Computationally easy for a party B to generate a pair (public key PU_b , private key PR_b).

2. Easy for sender to generate ciphertext:

$$C = E(PU_b, M)$$

3. Easy for the receiver to decrypt ciphertext using private key:

$$M = D(PR_b, C) = D[PR_b, E(PU_b, M)]$$

Requirements for Public-Key Cryptography (2/2)

4. Computationally infeasible to determine private key (PR_b) knowing public key (PU_b).
5. Computationally infeasible to recover message M , knowing PU_b and ciphertext C .
6. Either of the two keys can be used for encryption, with the other used for decryption:

$$M = D[PU_b, E(PR_b, M)] = D[PR_b, E(PU_b, M)]$$

RSA

RSA

- By Rivest, Shamir & Adleman of MIT in 1977.
- Best known & widely used public-key scheme.
- Based on exponentiation in a finite (Galois) field over integers modulo a prime.
 - nb. exponentiation takes $O((\log n)^3)$ operations (easy)
- Uses large integers (eg. 1024 bits).
- Security due to cost of factoring large numbers.
 - nb. factorization takes $O(e^{\log n \log \log n})$ operations (hard)

RSA Key Setup

- Each user generates a public/private key pair by:
- Selecting two large primes at random - p, q
- Computing their system modulus $n=p \cdot q$
 - note $\phi(n) = (p-1)(q-1)$
 - >512 bits $\rightarrow 1.340e^{154}$
- Selecting at random the encryption key e
 - where $1 < e < \phi(n)$, $\gcd(e, \phi(n)) = 1$
- Solve following equation to find decryption key d
 - $e \cdot d = 1 \pmod{\phi(n)}$ and $0 \leq d \leq n$
- Publish their public encryption key: $PU=\{e,n\}$
- Keep secret private decryption key: $PR=\{d,n\}$

RSA Use

- To encrypt a message M the sender:
 - Obtains **public key** of recipient $PU = \{e, n\}$
 - Computes: $C = M^e \bmod n$, where $0 \leq M < n$
- To decrypt the ciphertext C the owner:
 - Uses their private key $PR = \{d, n\}$
 - Computes: $M = C^d \bmod n$
- Note that the message M must be smaller than the modulus n (block if needed) ??

RSA Example - Key Setup (cont.)

Key Generation by Alice

Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p - 1)(q - 1)$	
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	$PR = \{d, n\}$

Encryption by Bob with Alice's Public Key

Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod n$

Decryption by Alice with Alice's Public Key

Ciphertext:	C
Plaintext:	$M = C^d \pmod n$

Why RSA Works

- Because of Euler's Theorem
- $a^{\phi(n)} \bmod n = 1$ where $\gcd(a, n) = 1$
- In RSA have:
 - $n = p \cdot q$
 - $\phi(n) = (p-1)(q-1)$
 - Carefully chose e & d to be inverses mod $\phi(n)$
 - Hence $e \cdot d = 1 + k \cdot \phi(n)$ for some k

$$\begin{aligned} C^d &= M^{e \cdot d} = M^{1+k \cdot \phi(n)} = M^1 \cdot (M^{\phi(n)})^k \\ &= M^1 \cdot (1)^k = M^1 = M \bmod n \end{aligned}$$

RSA Example - Key Setup

1. Select primes: $p=17$ & $q=11$
2. Compute $n = pq = 17 \times 11 = 187$
3. Compute $\phi(n) = (p-1)(q-1) = 16 \times 10 = 160$
4. Select e : $\gcd(e, 160) = 1$; **choose** $e=7$
5. **Determine** d : $de=1 \pmod{160}$ **and** $d < 160$ **Value** is $d=23$ **since** $23 \times 7 = 161 = 10 \times 160 + 1$
Extended Euclidean Method
6. **Publish public key** $PU = \{7, 187\}$
7. **Keep secret private key** $PR = \{23, 187\}$

RSA Example - En/Decryption (1/2)

- Sample RSA encryption/decryption is:
- Given message $M = 88$ ($88 < 187$)

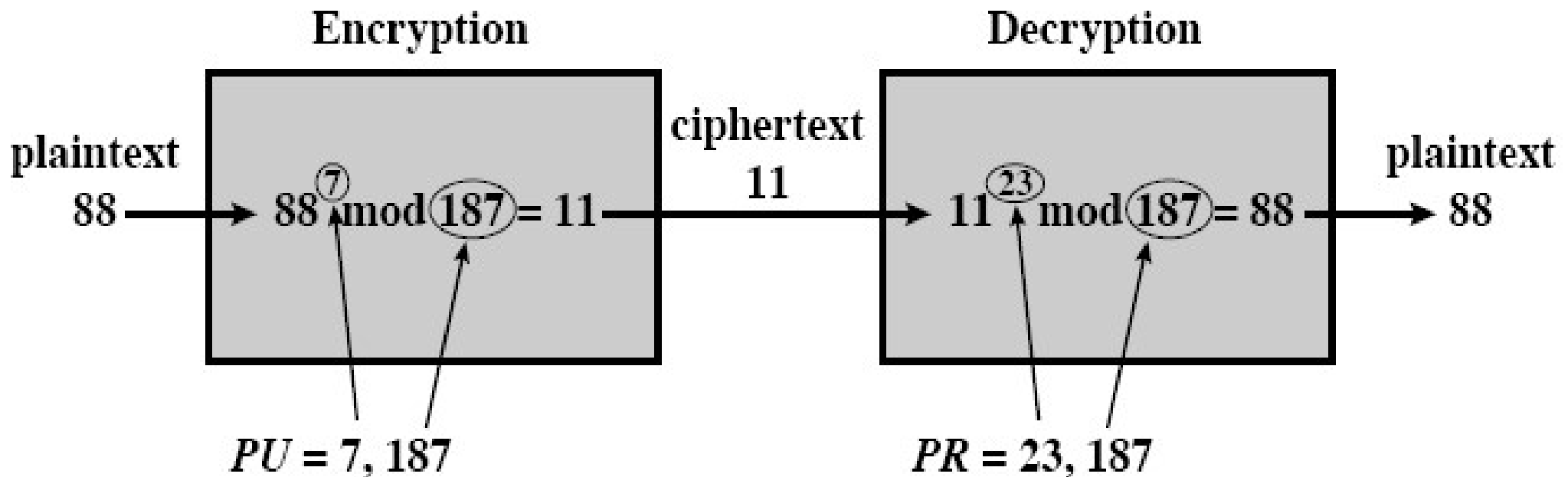
- Encryption:

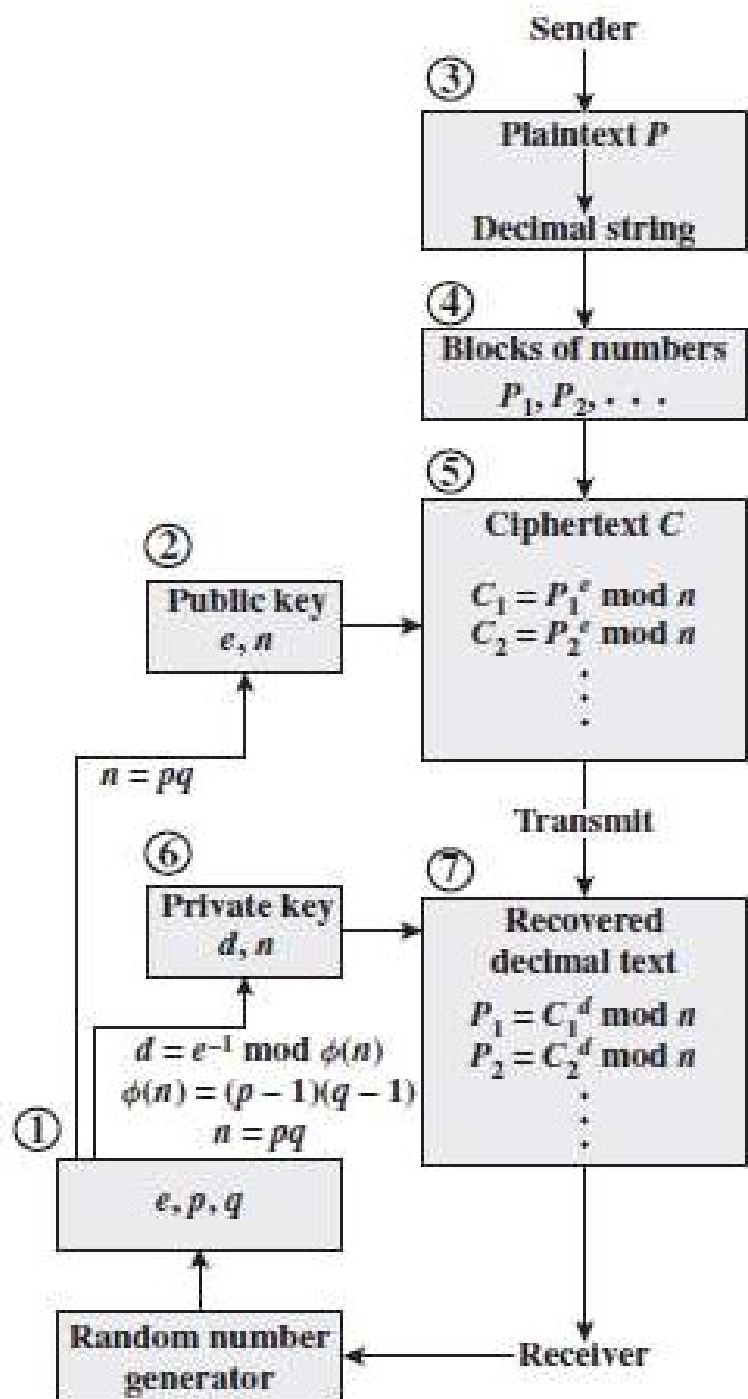
$$C = 88^7 \bmod 187 = 11$$

- Decryption:

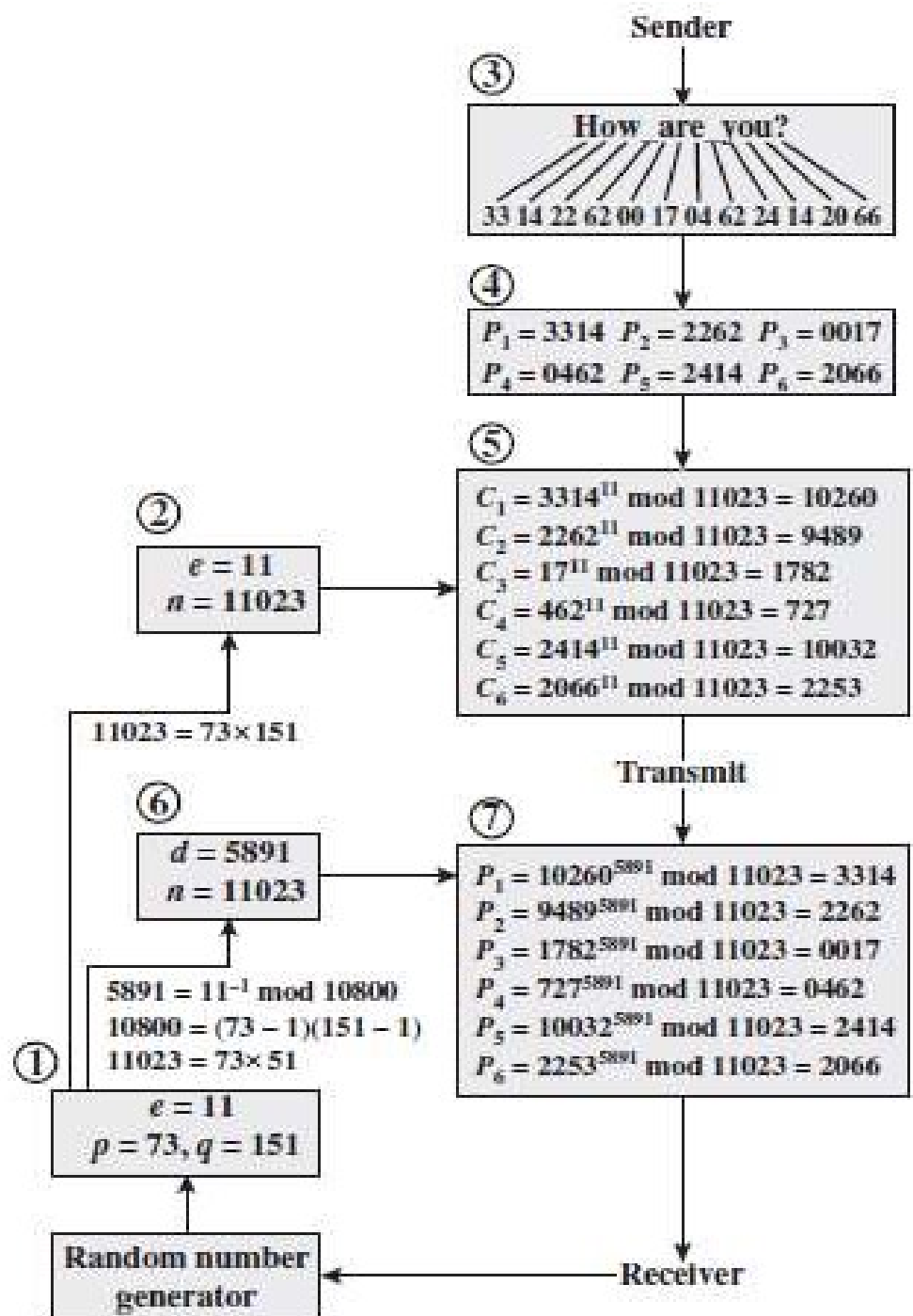
$$M = 11^{23} \bmod 187 = 88$$

RSA Example - En/Decryption (2/2)





(a) General approach



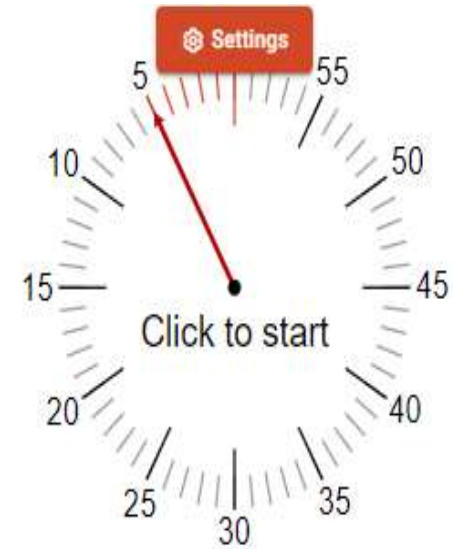
(b) Example

Activity Time

Generate the public and private key pair by applying the RSA algorithm:

$$p = 47$$

$$q = 59$$



Answer:

Step 1: Let $p = 47$ and $q = 59$. Thus $n = 47 \times 59 = 2773$

Step 2: Select $e = 17$

Step 3: Publish $(n,e) = (2773, 17)$

Step 4: $(p-1) \times (q-1) = 46 \times 58 = 2668$

Use the Euclidean Algorithm to compute the modular inverse of 17 modulo 2668 . The result is $d = 157$

<< Check: $17 \times 157 = 2669 = 1(\text{mod } 2668)$ >>

Public key is $(2773, 17)$

Private key is 157