Public-Key Encryption
公钥加密(非对称加密)

公钥密码学

Public-Key Cryptography

Asymmetric Encryption非对称加密

对称钥算法优缺点

uses **one** key
shared by both sender and receiver
symmetric, parties are equal

- **对称加密的优点**
  - 速度快,处理量大，适用: 数据的直接加密
  - 加密密钥长度相对较短,如64比特---256比特
  - 除了加密，还可构造各种加密体制，如产生伪随机数，HASH函数等
- **对称加密的缺点**
  - 密钥双方一致、保密，传递较难
  - 大型网络中密钥量大，难以管理，一般需要KDC
  - 密钥需要经常更换

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 key
- **digital signatures** – how to verify a message comes intact from the claimed sender

公钥密码体制

Stanford 大学的Diffie 和Hellman 于1976年首次提出

- 两把不同的密钥，将加密功能和解密功能分开

  一把: 私钥，秘密保存

  另一把: 公钥，不需要保密
- 两把相关的密钥任何一把都可以用于加密,而另外一把用作解密
- 公开钥密码算法基于数学函数，而不是基于替代和置换

- 给定公钥，要确定出私钥，在计算上是不可行的
- 可以简化密钥的管理，并且可以通过公开系统来分配密钥
- 用途:
  - 加解密
  - 数字签名
  - 密钥交换

非对称密码算法

每个实体产生一对密钥

- 具有两把不同密钥，一把称为公钥（Public-Key）
  一把称为私钥（Private-key）

  } 密钥对

- 任何一把都可用于加密,而另外一把则用作解密
- 特点
  - ✓ 用公钥加密的数据只能用私钥解密，而用私钥加密的数据只能用公钥解密
  - ✓ 公钥公开存放/发布，以供访问获取，所有的人或实体都可得到它
  - ✓ 私钥是私有的，不应被其他人或实体得到，且保持机密性

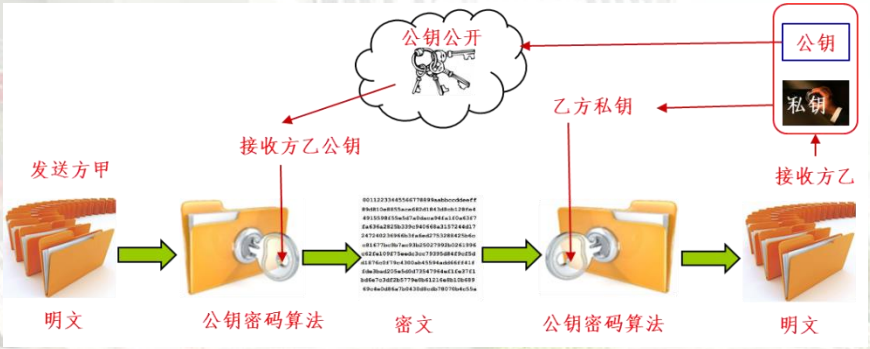
非对称密码算法-保密模型

➢ 工作原理

若发送方甲（A）想发数据给接收方乙（B），则甲方用乙方的公钥对数据进行加密，由于只有乙方拥有私钥，故只有乙方才可对收到的密文进行解密得到原数据，其他任何接收方均不能解密数据。过程可描述如下：

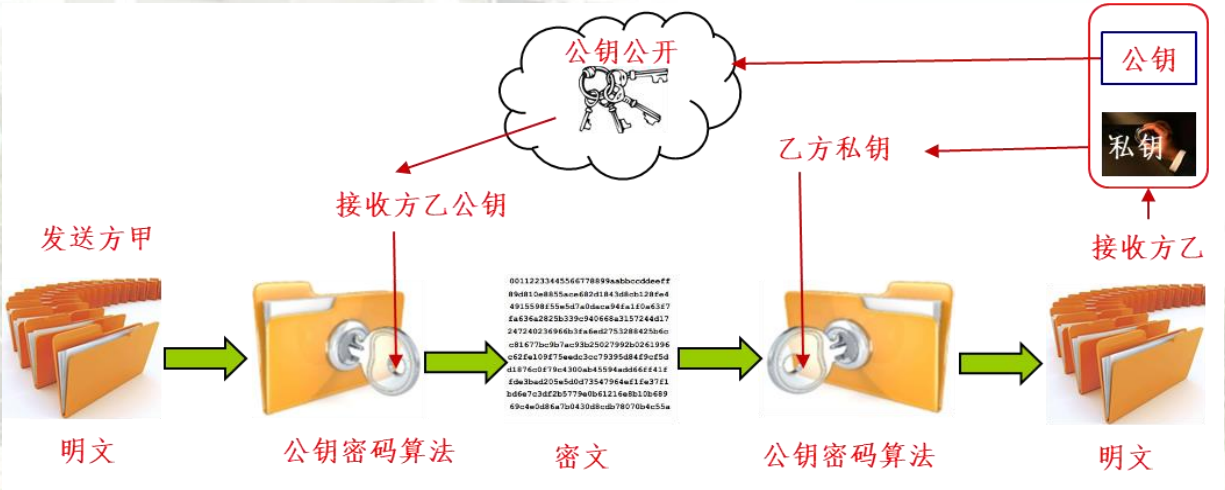
$$C = E(PU_b, M), \qquad M = D(PR_b, C)$$

其中：PU_b, PR_b,分别为B的公钥及私钥，C是密文，M为原数据



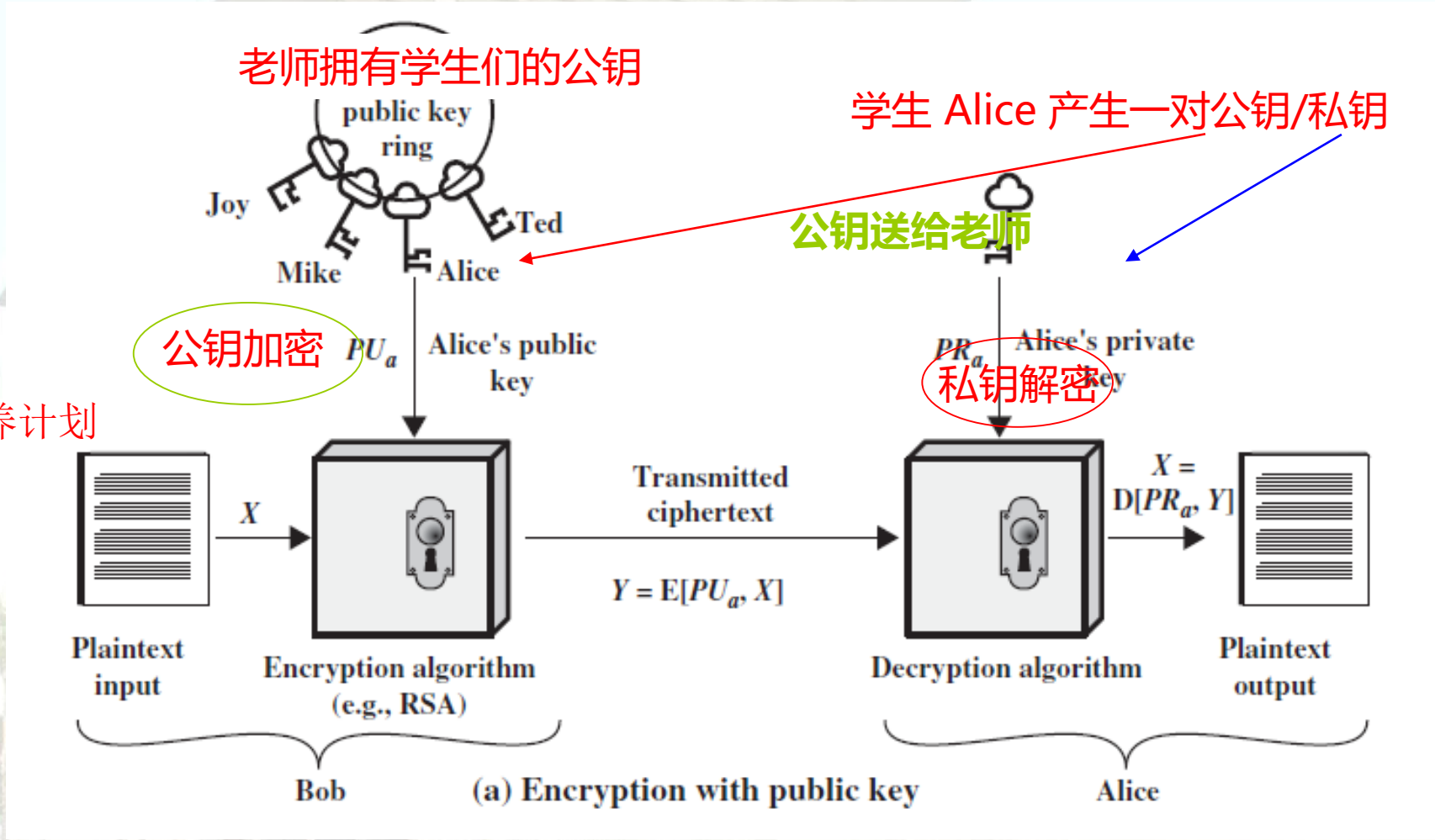
非对称密码算法模型

- 非对称密码模型包括：明文、加密演算法、公钥、私钥、密文、解密算法
- 常见非对称加密算法有RSA，ECC，以及SM2
- 特点
  - ✓ 仅根据公开密码算法和加密密钥来确定解密密钥在计算上是不可行的
  - ✓ 两把密钥中的任何一把可用来加密，另一把则用来解密，但作用则不同



保密模型

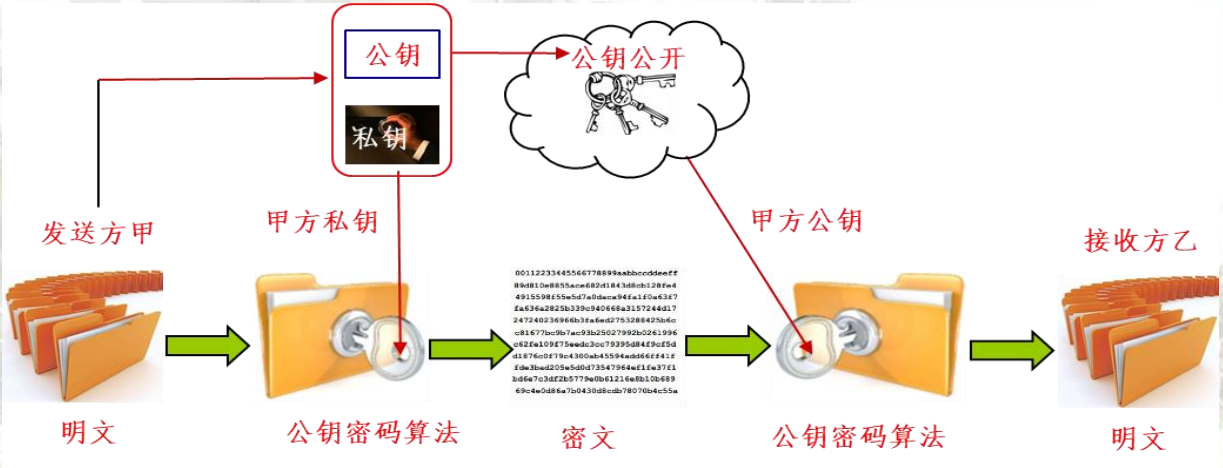
Public-Key Cryptography - encryption



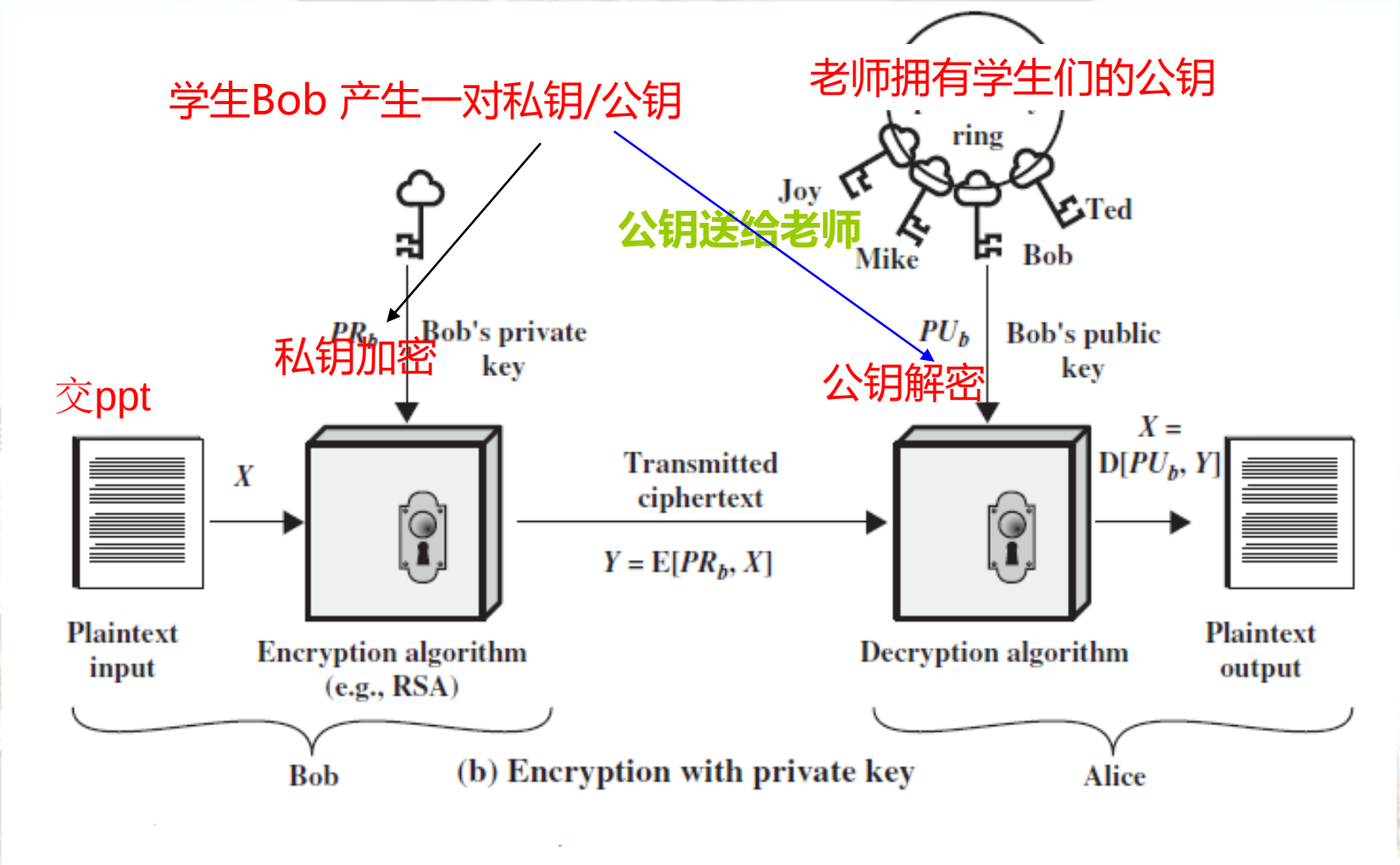
非对称密码算法-鉴别模型

➢ 工作原理

- ✓ 发送方甲想发数据给接收方乙，则用甲方A的私钥对数据进行加密，乙方B只有用A的公钥才能对数据进行解密得到原数据
- ✓ 请注意，所有实体只要能从公开网路获得A的公钥，都能用A的公钥对数据进行解密得到原数据，因此该模型只用于鉴别数据来源于甲方A，即验证了该数据由甲方A签发，但并没有对该数据进行保密。



Public-Key Cryptography - Authentication



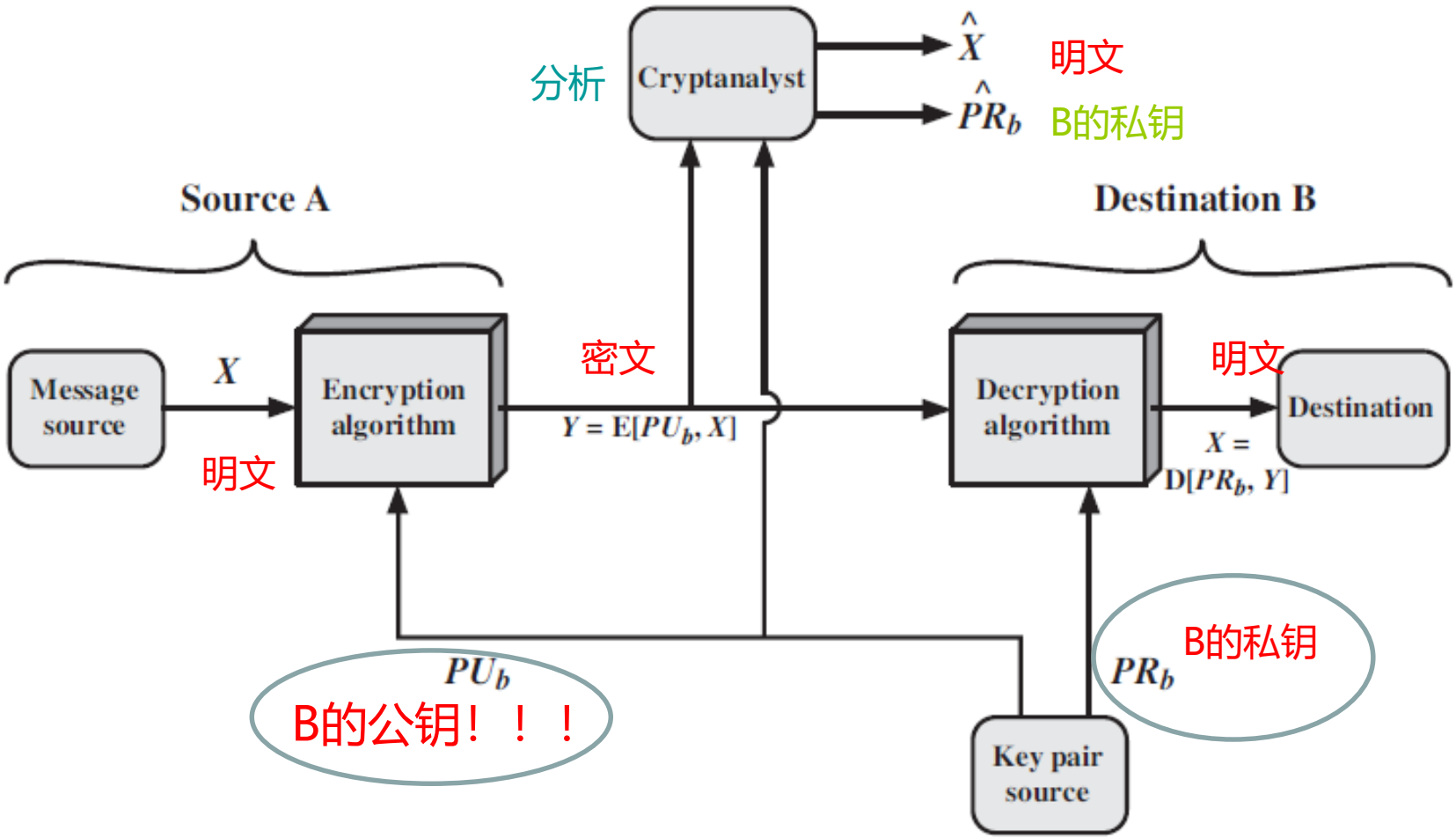
Public-Key Characteristics

- it is computationally infeasible（不可行） to find decryption key knowing only algorithm & encryption key
- it is computationally easy to en/decrypt messages when the relevant (en/decrypt) key is known
- either of the two related keys can be used for encryption, with the other used for decryption (for some algorithms)

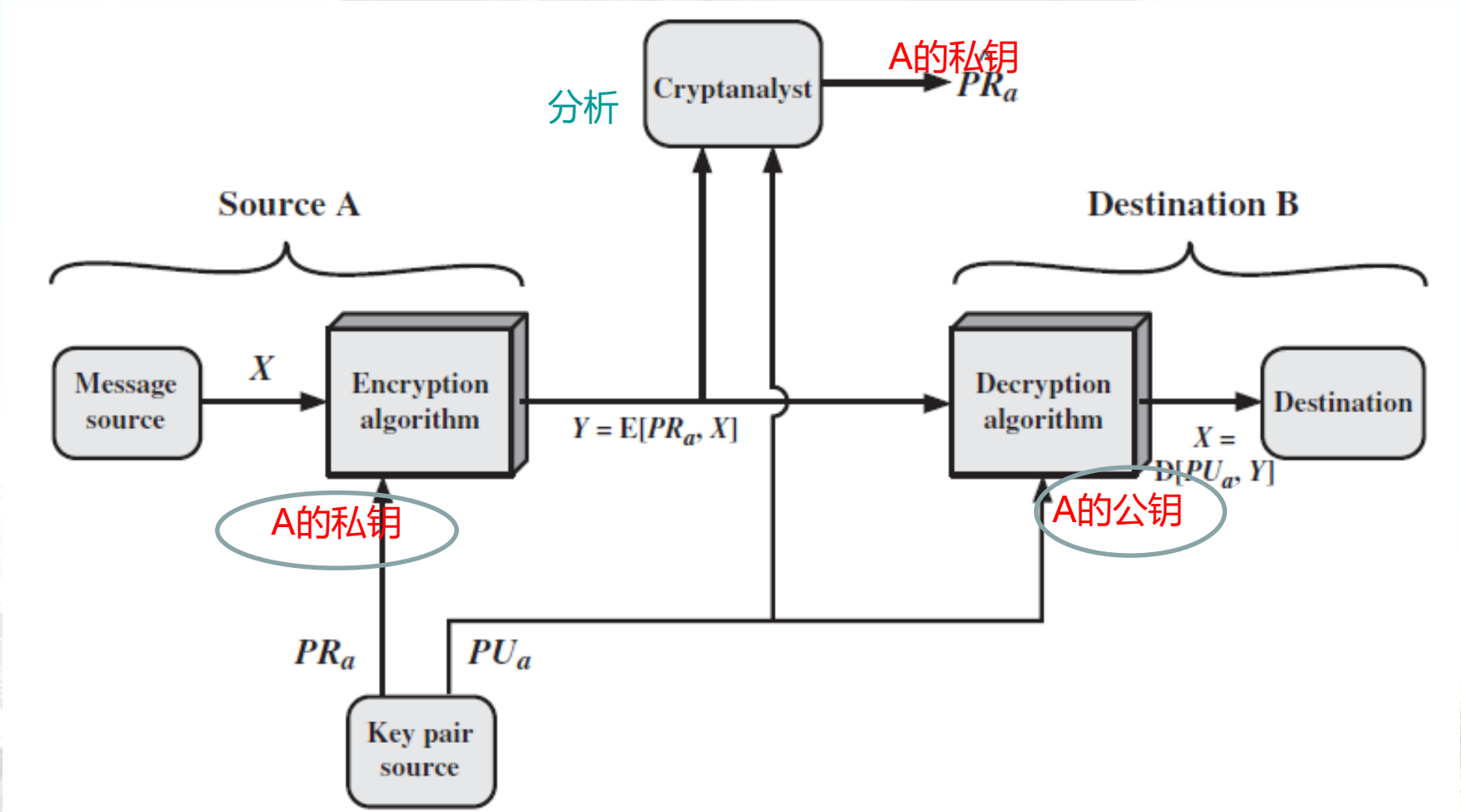
Table 9.1 CONVENTIONAL AND PUBLIC-KEY ENCRYPTION

Conventional Encryption	Public-Key Encryption
Needed to Work: 起码要求	*Needed to Work:*
1. The same algorithm with the same key is used for encryption and decryption.	1. One algorithm is used for encryption and decryption with a pair of keys, one for encryption and one for decryption.
2. The sender and receiver must share the algorithm and the key.	2. The sender and receiver must each have one of the matched pair of keys (not the same one).
Needed for Security:	*Needed for Security:*
1. The key must be kept secret.	1. One of the two keys must be kept secret.
2. It must be impossible or at least impractical to decipher a message if no other information is available.	2. It must be impossible or at least impractical to decipher a message if no other information is available.
3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.	3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.

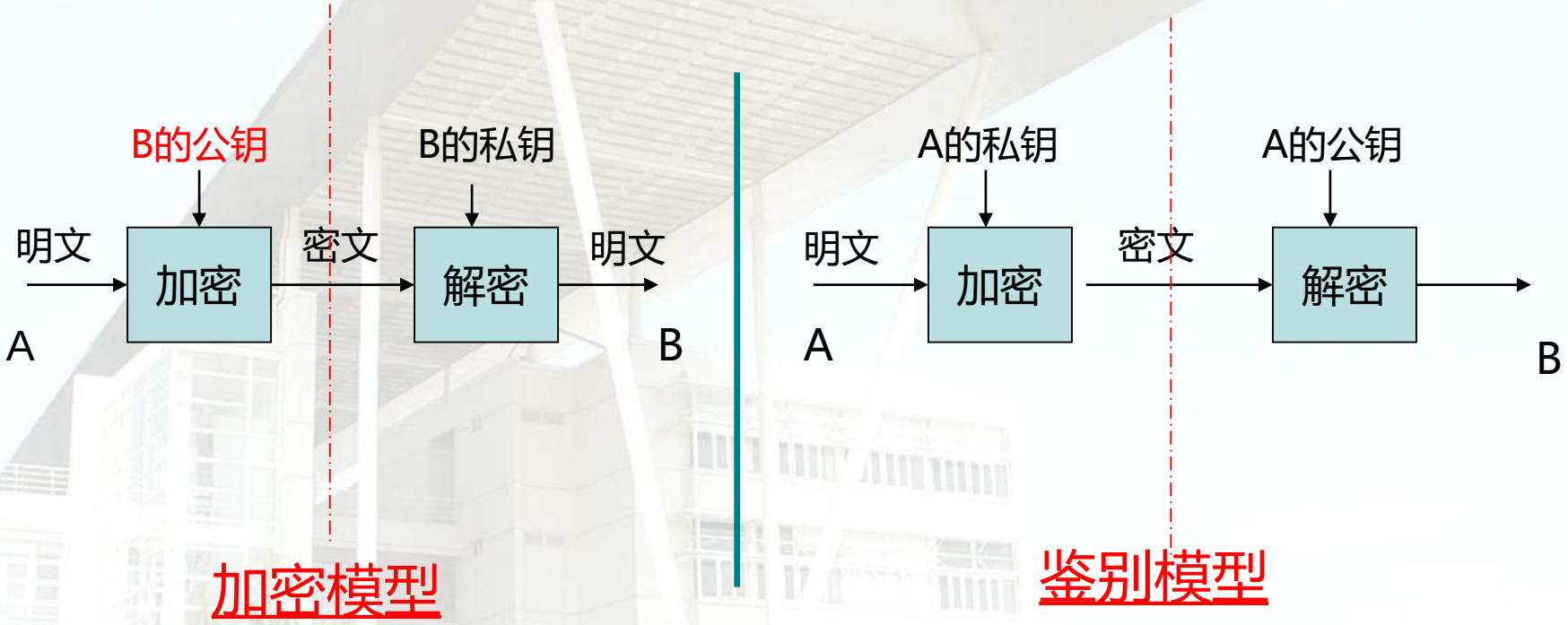
Public-Key Cryptosystems - 保密性



Public-Key Cryptosystems - 鉴别



公钥加密及鉴别



加密模型

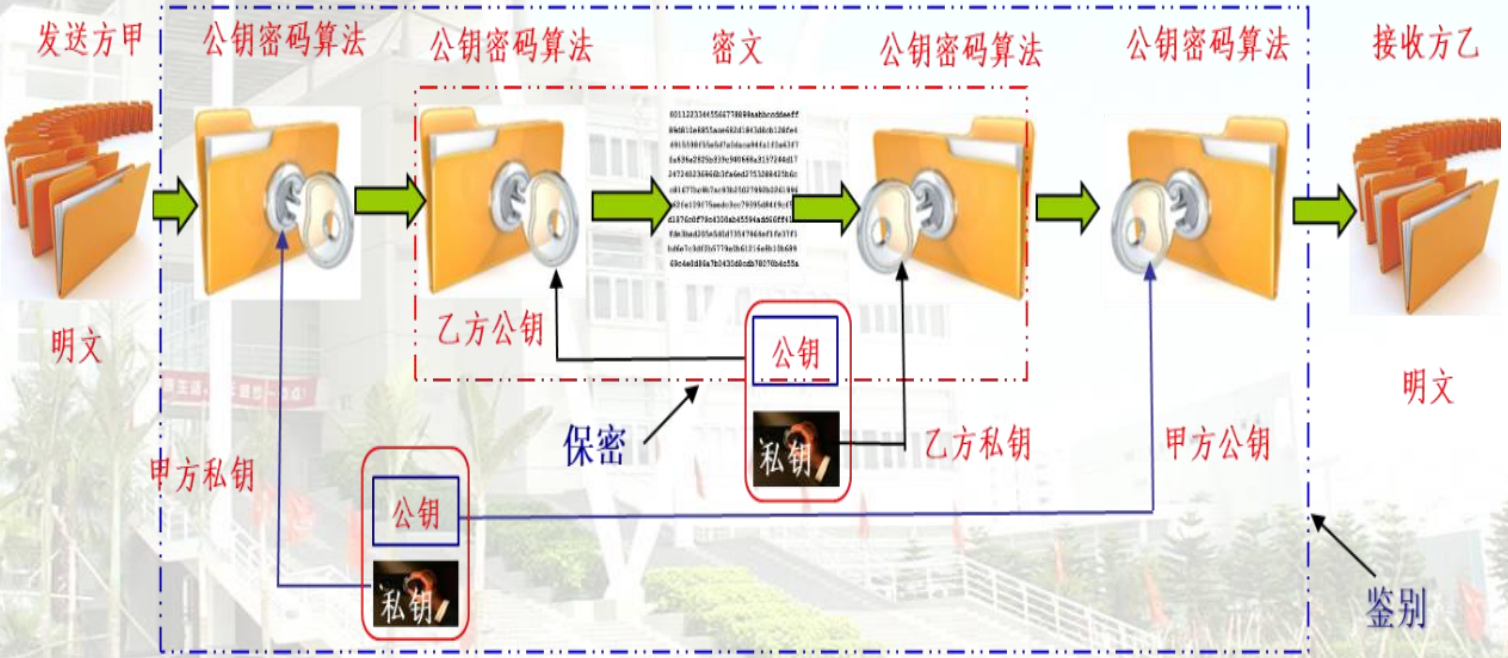
鉴别模型

不必进行共享密钥分发！！！

任何时候可改变私钥，公开发布公钥！！！

具有保密和鉴别功能的公开密码模型

- 为了同时提供数据的机密性以及鉴别性，则需要先用发送方甲A的私钥进行加密完成签名，再用接收方乙B的公钥对整个消息进行加密。
- 代价是每次要执行四次比对称密码算法费时得多的公开密钥密码演算法



Public-Key Cryptography

- **public-key/two-key/asymmetric** cryptography
  - 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

公钥算法优缺点

- **公钥加密的优点**
  - 私钥保密，公钥公开
  - 密钥生命周期相对较长
  - 可以产生数字签名机制
  - 在大型网络上，所需的密钥相对较少
- **公钥加密的缺点**
  - 速度慢，处理量少，适用于密钥交换
  - 密钥长度相对较长

Public-Key Applications 应用

- classify into 3 categories:

  - ✓ **encryption/decryption** (provide secrecy)
  - ✓ **digital signatures** (provide authentication)
  - ✓ **key exchange** (of session keys)

    wakeup and re-key

- some algorithms are suitable for all, others are specific to one only

Requirement for public-key cryptography

- Diffie and Hellman proposed the system requirement:
  - It is computationally easy to generate a pair of keys
  - It is computationally easy for a sender to encrypt: $C = E_{KUb}(M)$
  - It is computationally easy for a receiver to decrypt $M = D_{KRb}(C)$
  - It is computationally infeasible to determine the private key, knowing the public key
  - It is computationally infeasible to recover the plaintext, knowing the public key and ciphtertext

Security of Public Key Schemes

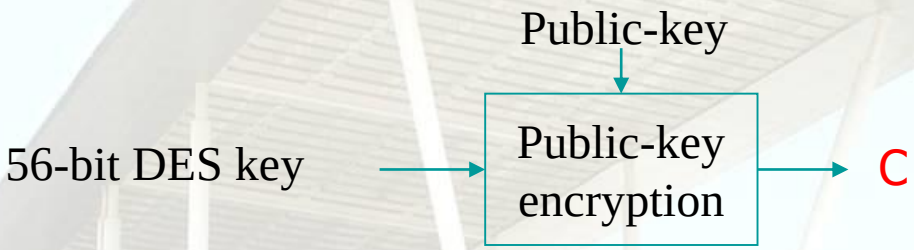
- brute force （强力） **exhaustive search** attack is always theoretically possible
- use large Keys (>1024bits)－tradeoff
- Compute the private key, given the public keys？
- **Slow** compared to symmetric key schemes,
  So for Digital signature and Key management

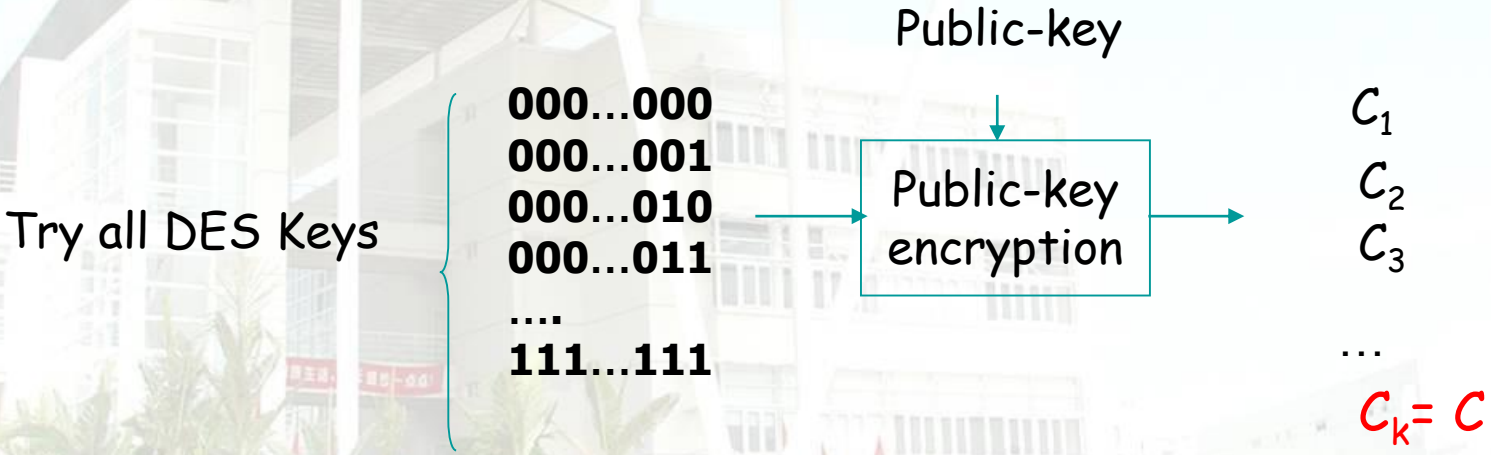
Probable-message attack

- Public key is known
- Encrypt all possible messages
- Try to find a match between the ciphertext and one of the above encrypted messages

Probable-message attack

encrypt a 56-bit DES key （as input message):





Attack: 无论公钥有多长，总可以找到 $C_k = C$ 对应的P_k

Solution: append Nonce in the message

Turing Award（图灵奖）

- 1966年由美国计算机协会ACM（Association for Computing Machinery）设立
  - ✓ 奖励在计算机领域作出重要贡献的个人
  - ✓ 计算机领域的国际最高奖项
  - ✓ 誉为“计算机界的诺贝尔奖”
- 与密码学相关领域的图灵奖得主

（）

1995年	曼纽尔·布卢姆	Manuel Blum	计算复杂度理论，及密码学和程序校验上的应用
2000年	姚期智（华人）	Andrew Chi-Chih Yao	计算理论，伪随机数生成，密码学与通信复杂度，安全多方计算（Secure Multi-Party Computation）
	罗纳德·李维斯特	Ronald L. Rivest	公钥密码学（RSA加密算法）
2002年	阿迪·萨莫尔	Adi Shamir	
	伦纳德·阿德曼	Leonard M. Adleman	
2012年	莎菲·戈德瓦塞尔	Shafi Goldwasser	在密码学和复杂理论领域做出创举性工作
	希尔维奥·米卡利	Silvio Micali	
2015年	惠特菲尔德·迪菲	Whitfield Diffie	非对称加密创始人
	马丁·赫尔曼	Martin Hellman	

Popular Public Key Algorithms

- RSA (Rivest-Shamir-Adleman)
  - Number theory
- Elliptic curve (椭圆曲线) cryptography

Discrete Logarithms (离散对数)

素数：大于1的自然数，除了1和其自身外，不能被其它自然数整除的数

原根：如果a是素数p的**原根**，那么

$a \mod p,\quad a^2 \mod p,\quad \cdots,\quad a^{p-1} \mod p,$

产生出各不相同的整数，并且以某种排列方式组成从1到p－1的所有整数

对一个整数b和素数p的一个原根a，可以找到一个唯一的指数i，使得：

$$b = a^i \mod p \quad (\ 0 \leqslant i \leqslant (p-1)\ )$$

指数 i 称为 b 的以 a 为基数的模 p 的**离散对数**或**指数**。

如果已知b和a，p，而p 是一个大素数，计算i是不可行的。

RSA Algorithm

- by Rivest, Shamir & Adleman of MIT， 1978
- best known & widely used public-key scheme
- based on exponentiation (求幂) in finite field （有限域）over integers modulo (prime)
- uses large integers (eg. 1024 bits)

RSA Key Setup（密钥生成）

each user generates a public/private key pair by:

- selecting two large primes (两个大素（质）数) at random : p, q
- computing their system modulus
  n=p.q and ø(n)=(p-1)(q-1)
- selecting an encryption key e， randomly （随机选取密钥）
- where 1<e<ø(n), gcd(e,ø(n))=1 （最大公因子）

互素,若N个整数的最大公因数是1，则称这N个整数互素。

e和ø(n)互素

- solve following equation to find decryption key d
  e.d=1 mod ø(n) and 0≤d≤n
  $d= e^{-1} \bmod \phi(n)$
- publish public key（公钥）: {e,n}
- keep private key （私钥）: {d,n}

RSA Use

- ✓ **encrypt a message M :**
  - ➢ **obtains public key of recipient PU={e,n}**
  - ➢ **computes: $C = M^e \bmod n$**

- ✓ **decrypt the ciphertext C:**
  - ➢ **uses its private key PR={d,n}**
  - ➢ **computes: $M = C^d \bmod n$**

RSA Use

- note that the message M must be smaller than the modulus n

- if M is large，block the message

Why/How RSA Works

- **Euler's Theorem (欧拉定理):**
  $a^{ø(n)} \bmod n = 1$ where $\gcd(a,n)=1$

- **in RSA:**
  $n=p.q$
  $ø(n)=(p-1)(q-1)$
  carefully chose e & d to be inverses mod ø(n):
  $e.d=1+k.ø(n)$ for some k
- **hence :**

$$M = C^d \bmod n$$
$$= (M^e)^d \bmod n$$
$$= M^{ed} \bmod n$$
$$= M^{k(p-1)(q-1)+1} \bmod n$$
$$= M.\, M^{k(p-1)(q-1)} \bmod n \quad (\text{根据欧拉定理})$$
$$= M$$

Key Generation

Select p, q	p and q both prime, $p \quad q$
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	$d \equiv e^{-1} \bmod \phi(n)$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

Encryption

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

Decryption

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

Figure 9.5 The RSA Algorithm

RSA Example - Key Setup

1. **Select primes: p=17 & q=11**
2. **Compute: $n = pq$ =17 x 11=187**
3. **Compute $\phi(n)=(p-1)(q-1)$=16 x 10=160**
4. **Select e: gcd(e,160)=1; choose e=7**

   e < 160
5. **Determine d: de=1 mod 160 and d < 160**

   d=23 since 23x7=161= 10x160+1
6. **Publish public key PU={7,187}**
7. **Keep secret private key PR={23,187}**

RSA Example - En/Decryption

- given message M = 88 (88<187)
- encryption:

  $C = 88^7 \bmod 187 = 11$
- decryption:

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

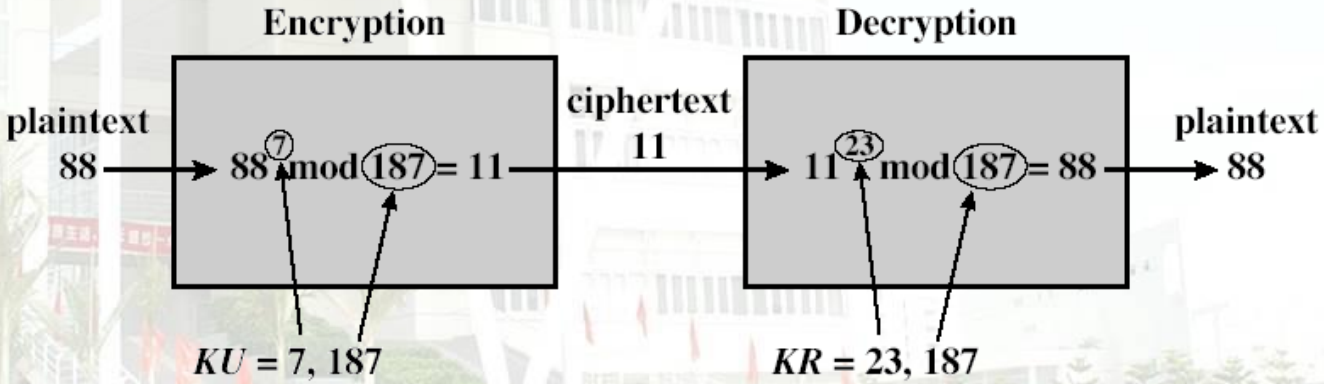


Figure 9.6 Example of RSA Algorithm

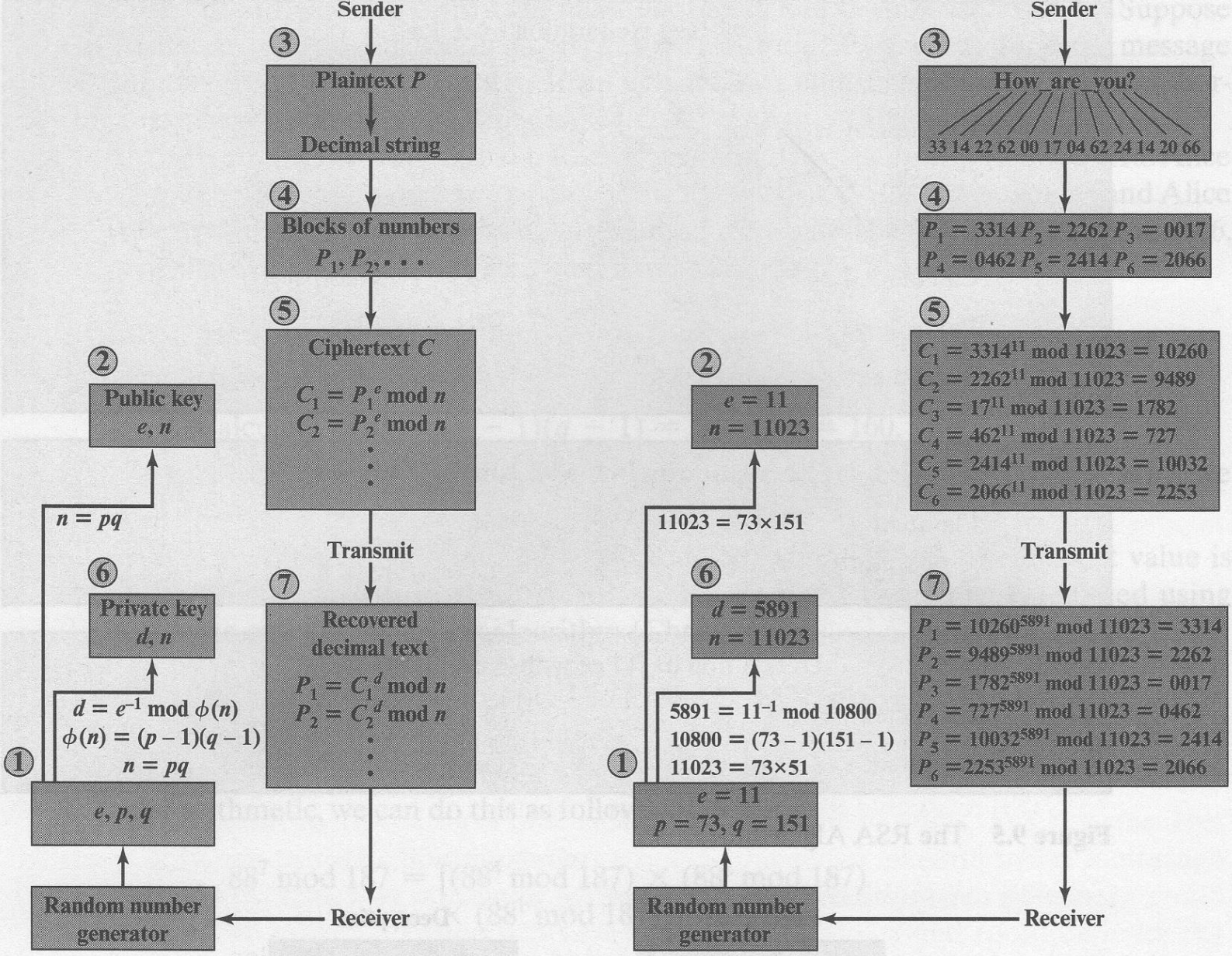


Figure 9.7 RSA Processing of Multiple Blocks

RSA Security

- possible approaches to attacking RSA are:
  - brute force（强力法） key search—大密钥
  - mathematical attacks (based on difficulty of computing ø(n), by factoring modulus n)
  - timing attacks (on running of decryption)

RSA的限制：

✓产生密钥很麻烦，受到素数产生技术的限制
✓分组长度太大，使运算代价很高，尤其是速度较慢，较对称密码算法慢几个数量级