



# *Cryptography and Network Security*

---

Sixth Edition  
by William Stallings



# Chapter 9

---

Public Key Cryptography and RSA

# Misconceptions Concerning Public-Key Encryption

- Public-key encryption is more secure from cryptanalysis than symmetric encryption
- Public-key encryption is a general-purpose technique that has made symmetric encryption obsolete
- There is a feeling that key distribution is trivial when using public-key encryption, compared to the cumbersome handshaking involved with key distribution centers for symmetric encryption



# Table 9.1

## Terminology Related to Asymmetric Encryption

### **Asymmetric Keys**

Two related keys, a public key and a private key that are used to perform complementary operations, such as encryption and decryption or signature generation and signature verification.

### **Public Key Certificate**

A digital document issued and digitally signed by the private key of a Certification Authority that binds the name of a subscriber to a public key. The certificate indicates that the subscriber identified in the certificate has sole control and access to the corresponding private key.

### **Public Key (Asymmetric) Cryptographic Algorithm**

A cryptographic algorithm that uses two related keys, a public key and a private key. The two keys have the property that deriving the private key from the public key is computationally infeasible.

### **Public Key Infrastructure (PKI)**

A set of policies, processes, server platforms, software and workstations used for the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain, and revoke public key certificates.

# Principles of Public-Key Cryptosystems

- The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems associated with symmetric encryption:

## Key distribution

- **How to have secure communications in general without having to trust a KDC with your key**

## Digital signatures

- **How to verify that a message comes intact from the claimed sender**

- Whitfield Diffie and Martin Hellman from Stanford University achieved a breakthrough in 1976 by coming up with a method that addressed both problems and was radically different from all previous approaches to cryptography

# Public-Key Cryptosystems

- A public-key encryption scheme has six ingredients:

Plaintext

The readable message or data that is fed into the algorithm as input

Encryption algorithm

Performs various transformations on the plaintext

Public key

Used for encryption or decryption

Private key

Used for encryption or decryption

Ciphertext

The scrambled message produced as output

Decryption algorithm

Accepts the ciphertext and the matching key and produces the original plaintext

# Public-Key Cryptography

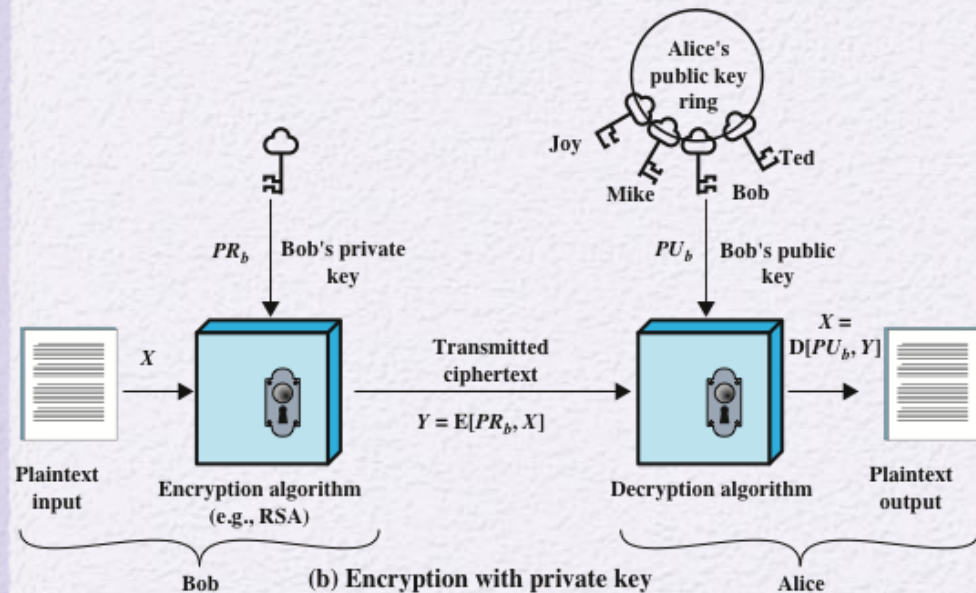
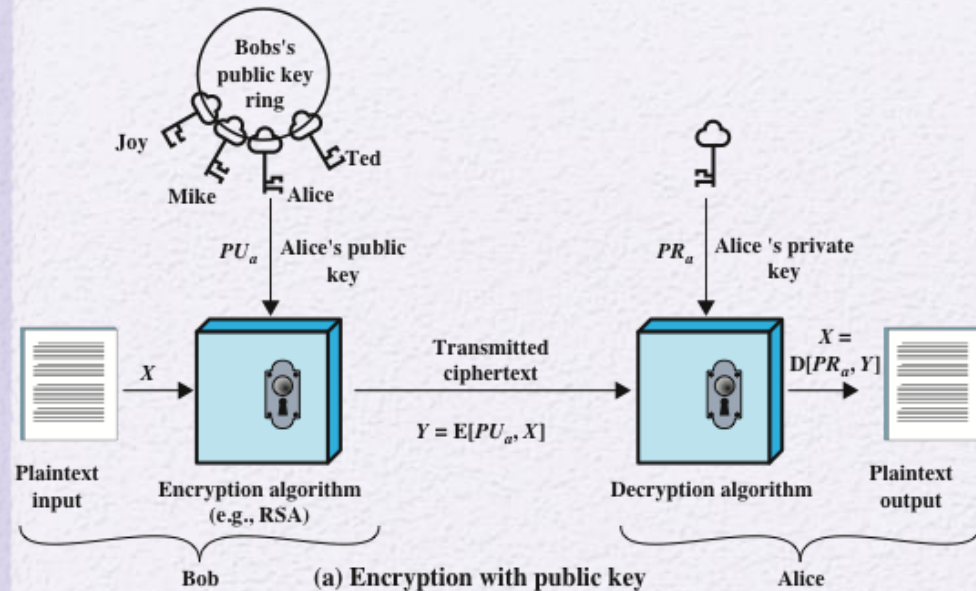
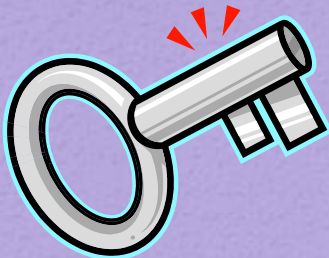


Figure 9.1 Public-Key Cryptography

# Table 9.2

## Conventional and Public-Key Encryption

<b>Conventional Encryption</b>	<b>Public-Key Encryption</b>
<p data-bbox="193 325 463 357"><i>Needed to Work:</i></p> <ol data-bbox="231 414 946 628" style="list-style-type: none"><li data-bbox="231 414 946 492">1. The same algorithm with the same key is used for encryption and decryption.</li><li data-bbox="231 549 946 628">2. The sender and receiver must share the algorithm and the key.</li></ol> <p data-bbox="193 685 521 721"><i>Needed for Security:</i></p> <ol data-bbox="231 778 946 1170" style="list-style-type: none"><li data-bbox="231 778 946 813">1. The key must be kept secret.</li><li data-bbox="231 863 946 992">2. It must be impossible or at least impractical to decipher a message if the key is kept secret.</li><li data-bbox="231 1049 946 1170">3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.</li></ol>	<p data-bbox="985 325 1255 357"><i>Needed to Work:</i></p> <ol data-bbox="1023 414 1738 763" style="list-style-type: none"><li data-bbox="1023 414 1738 585">1. One algorithm is used for encryption and a related algorithm for decryption with a pair of keys, one for encryption and one for decryption.</li><li data-bbox="1023 642 1738 763">2. The sender and receiver must each have one of the matched pair of keys (not the same one).</li></ol> <p data-bbox="985 821 1313 856"><i>Needed for Security:</i></p> <ol data-bbox="1023 913 1738 1349" style="list-style-type: none"><li data-bbox="1023 913 1738 949">1. One of the two keys must be kept secret.</li><li data-bbox="1023 999 1738 1128">2. It must be impossible or at least impractical to decipher a message if one of the keys is kept secret.</li><li data-bbox="1023 1185 1738 1349">3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.</li></ol>



# Public-Key Cryptosystem: Secrecy

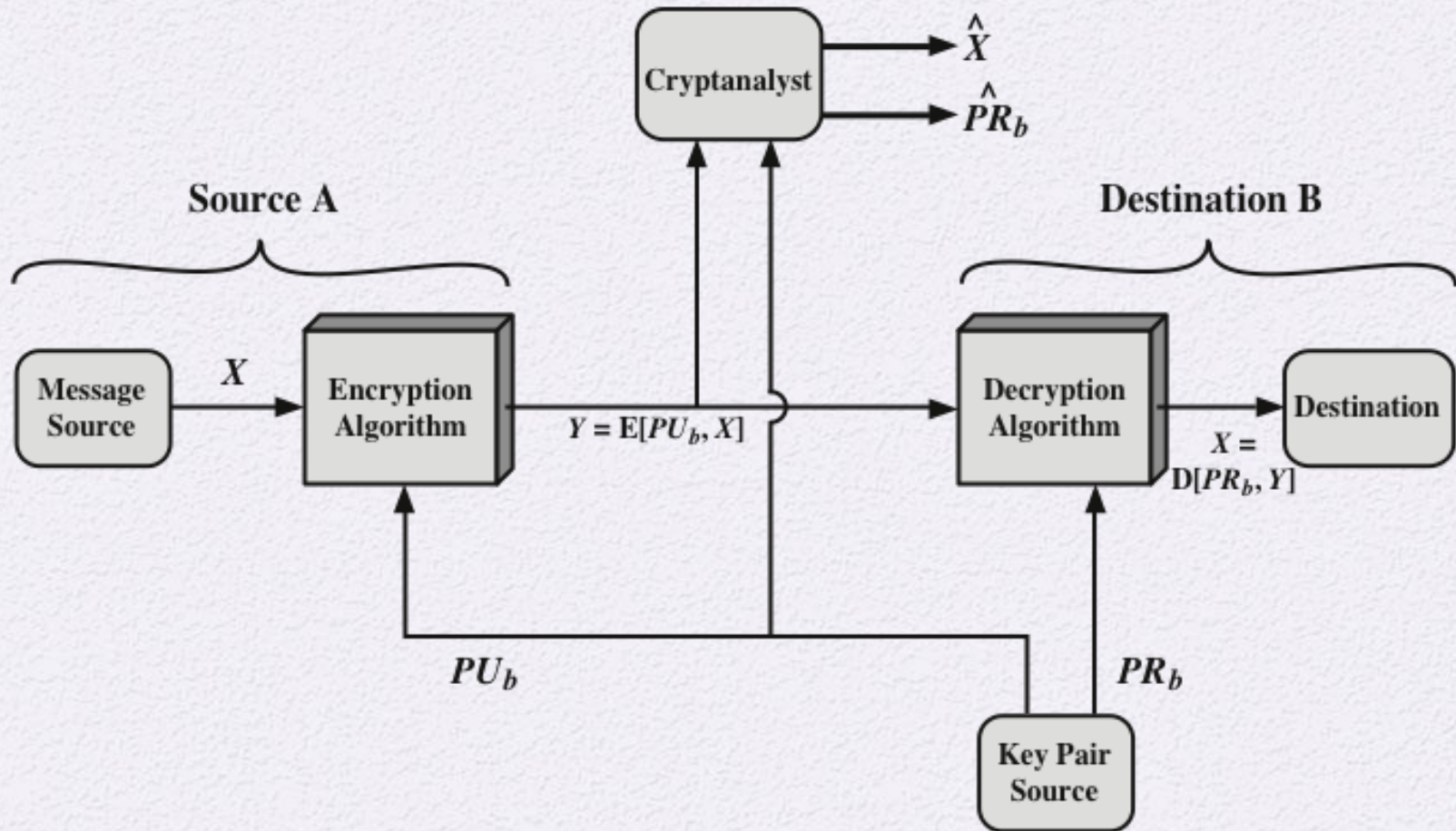


Figure 9.2 Public-Key Cryptosystem: Secrecy

# Public-Key Cryptosystem: Authentication

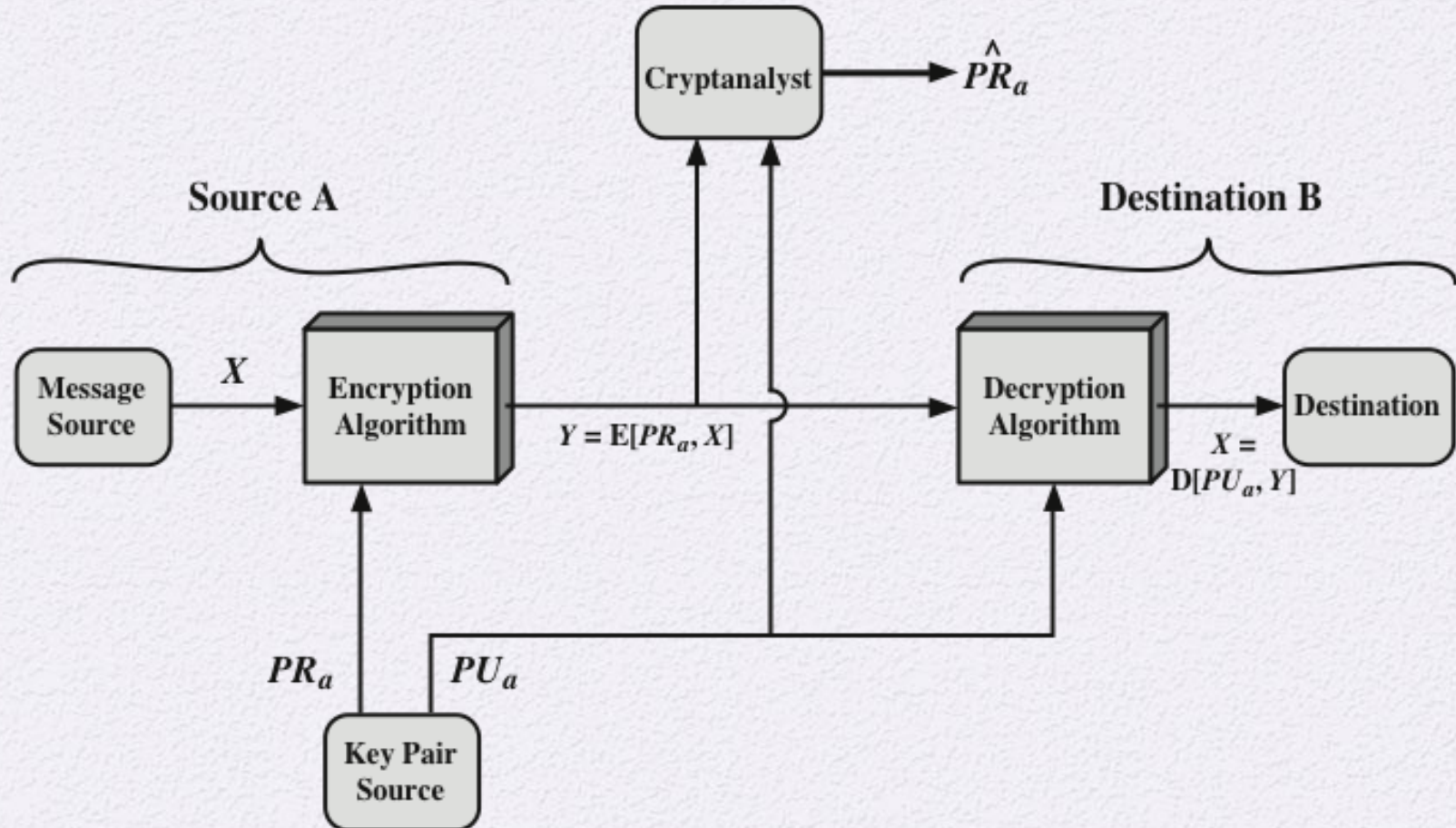


Figure 9.3 Public-Key Cryptosystem: Authentication

# Public-Key Cryptosystem: Authentication and Secrecy

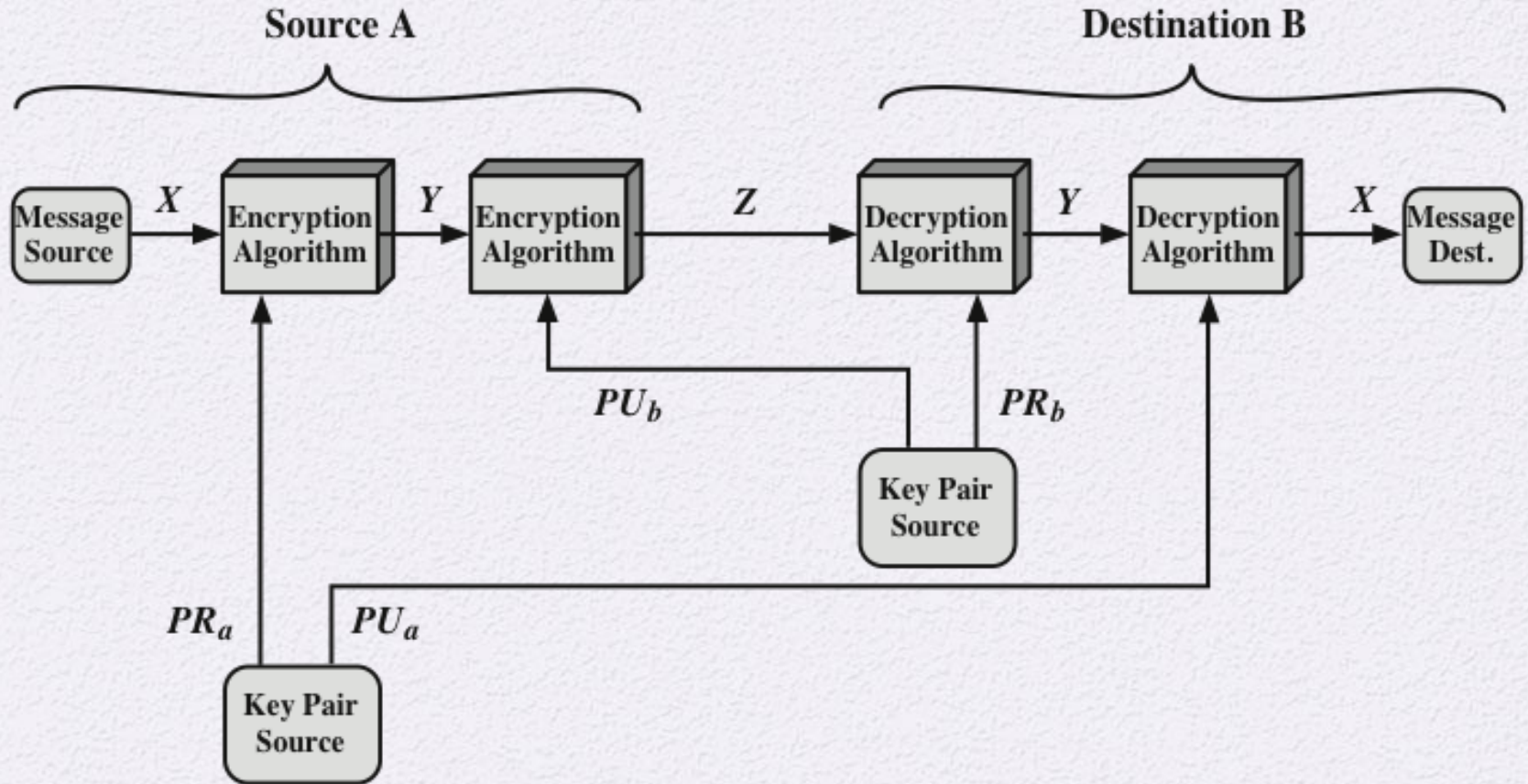
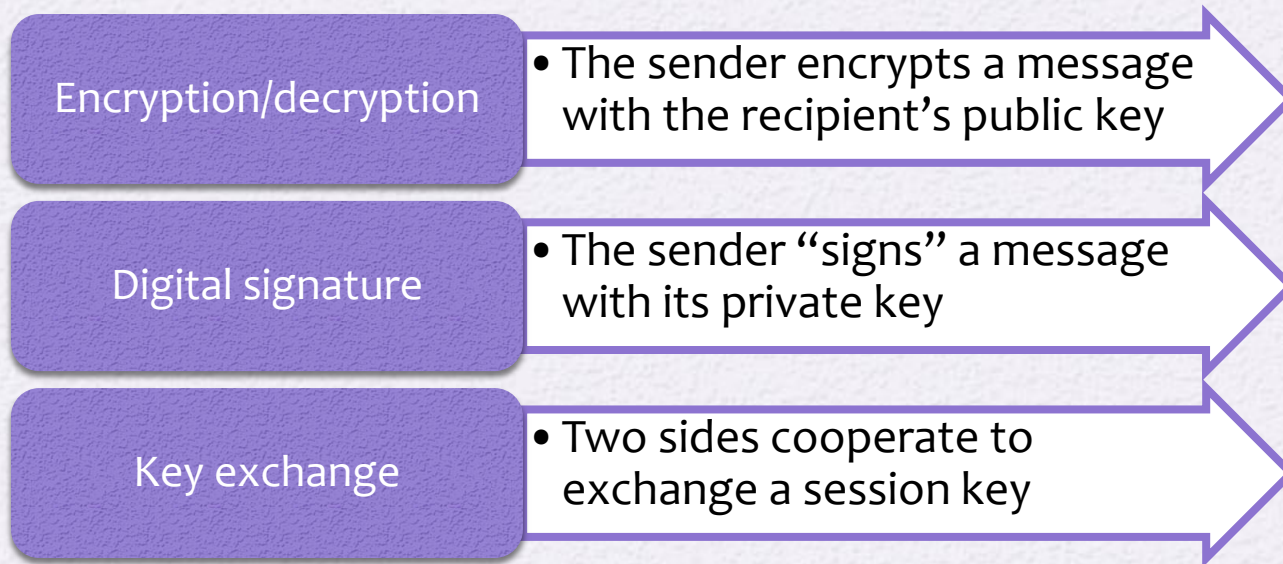


Figure 9.4 Public-Key Cryptosystem: Authentication and Secrecy

# Applications for Public-Key Cryptosystems

- Public-key cryptosystems can be classified into three categories:



- Some algorithms are suitable for all three applications, whereas others can be used only for one or two

# Table 9.3

## Applications for Public-Key Cryptosystems

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

Table 9.3 Applications for Public-Key Cryptosystems

# Public-Key Requirements

- Conditions that these algorithms must fulfill:
  - It is computationally easy for a party B to generate a pair (public-key  $PU_b$ , private key  $PR_b$ )
  - It is computationally easy for a sender A, knowing the public key and the message to be encrypted, to generate the corresponding ciphertext
  - It is computationally easy for the receiver B to decrypt the resulting ciphertext using the private key to recover the original message
  - It is computationally infeasible for an adversary, knowing the public key, to determine the private key
  - It is computationally infeasible for an adversary, knowing the public key and a ciphertext, to recover the original message
  - The two keys can be applied in either order

# Public-Key Requirements

- Need a trap-door one-way function
  - A one-way function is one that maps a domain into a range such that every function value has a unique inverse, with the condition that the calculation of the function is easy, whereas the calculation of the inverse is infeasible
    - $Y = f(X)$  easy
    - $X = f^{-1}(Y)$  infeasible
- A trap-door one-way function is a family of invertible functions  $f_k$ , such that
  - $Y = f_k(X)$  easy, if  $k$  and  $X$  are known
  - $X = f_k^{-1}(Y)$  easy, if  $k$  and  $Y$  are known
  - $X = f_k^{-1}(Y)$  infeasible, if  $Y$  known but  $k$  not known
- A practical public-key scheme depends on a suitable trap-door one-way function

# Rivest-Shamir-Adleman (RSA) Scheme

- Developed in 1977 at MIT by Ron Rivest, Adi Shamir & Len Adleman
- Most widely used general-purpose approach to public-key encryption
- Is a cipher in which the plaintext and ciphertext are integers between 0 and  $n - 1$  for some  $n$ 
  - A typical size for  $n$  is 1024 bits, or 309 decimal digits

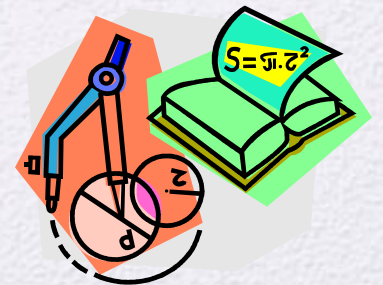


# RSA Algorithm

- RSA makes use of an expression with exponentials
- Plaintext is encrypted in blocks with each block having a binary value less than some number  $n$
- Encryption and decryption are of the following form, for some plaintext block  $M$  and ciphertext block  $C$ 
$$C = M^e \bmod n$$
$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$
- Both sender and receiver must know the value of  $n$
- The sender knows the value of  $e$ , and only the receiver knows the value of  $d$
- This is a public-key encryption algorithm with a public key of  $PU=\{e,n\}$  and a private key of  $PR=\{d,n\}$

# Algorithm Requirements

- For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:
  1. It is possible to find values of  $e, d, n$  such that  $M^{ed} \bmod n = M$  for all  $M < n$
  2. It is relatively easy to calculate  $M^e \bmod n$  and  $C^d \bmod n$  for all values of  $M < n$
  3. It is infeasible to determine  $d$  given  $e$  and  $n$



<b>Key Generation by Alice</b>	
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 $e$	$\text{gcd}(\phi(n), e) = 1; 1 < e < \phi(n)$
Calculate $d$	$d = e^{-1} \pmod{\phi(n)}$
Public key	$PU = \{e, n\}$
Private key	$PR = \{d, n\}$

<b>Encryption by Bob with Alice's Public Key</b>	
Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod{n}$

<b>Decryption by Alice with Alice's Private Key</b>	
Ciphertext:	$C$
Plaintext:	$M = C^d \pmod{n}$

**Figure 9.5 The RSA Algorithm**

# Example of RSA Algorithm

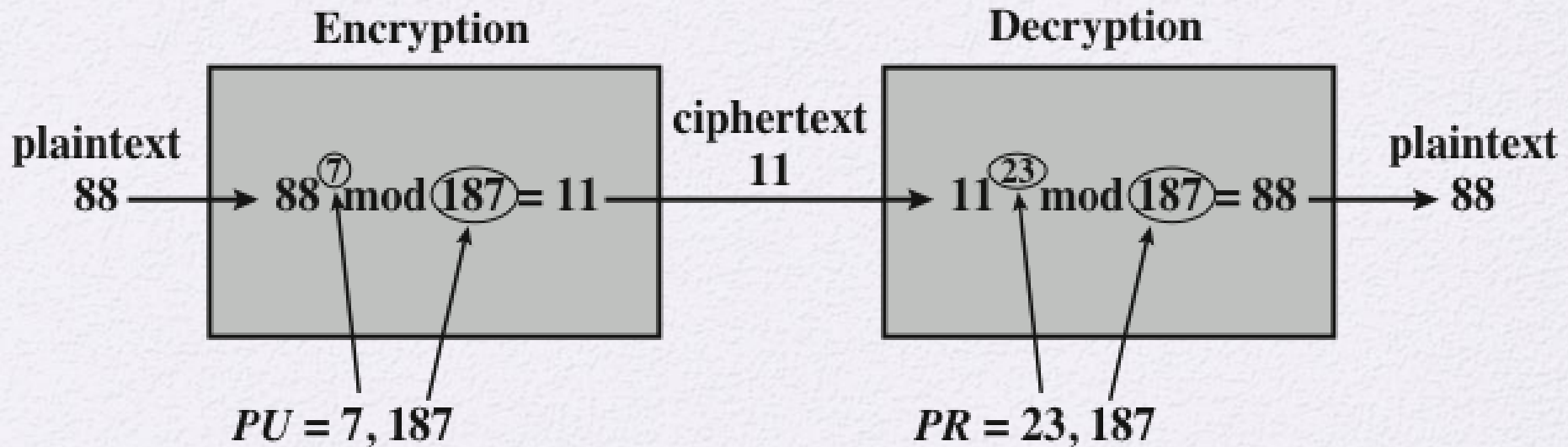
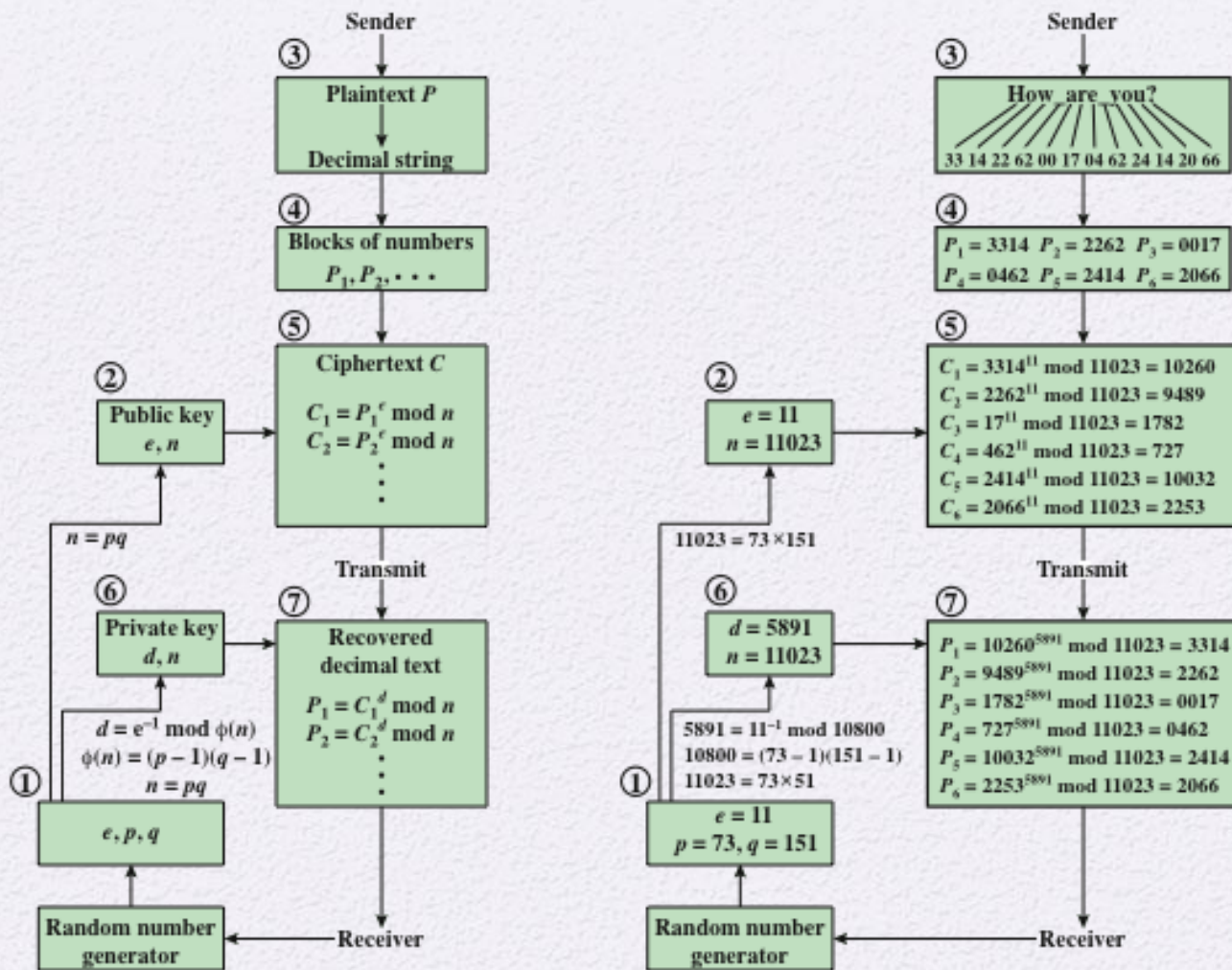


Figure 9.6 Example of RSA Algorithm



(a) General approach

(b) Example

Figure 9.7 RSA Processing of Multiple Blocks

# Exponentiation in Modular Arithmetic

- Both encryption and decryption in RSA involve raising an integer to an integer power, mod  $n$
- Can make use of a property of modular arithmetic:

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

- With RSA you are dealing with potentially large exponents so efficiency of exponentiation is a consideration

```

c ← 0; f ← 1
for i ← k downto 0
    do   c ← 2 × c
        f ← (f × f) mod n
    if  bi = 1
        then c ← c + 1
            f ← (f × a) mod n
return f

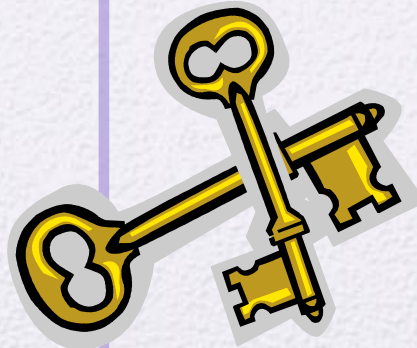
```

Note: The integer  $b$  is expressed as a binary number  $b_k b_{k-1} \dots b_0$

**Figure 9.8** Algorithm for Computing  $a^b \bmod n$

# Key Generation

- Before the application of the public-key cryptosystem each participant must generate a pair of keys:
  - Determine two prime numbers  $p$  and  $q$
  - Select either  $e$  or  $d$  and calculate the other
- Because the value of  $n = pq$  will be known to any potential adversary, primes must be chosen from a sufficiently large set
  - The method used for finding large primes must be reasonably efficient





# Procedure for Picking a Prime Number


- Pick an odd integer  $n$  at random
- Pick an integer  $a < n$  at random
- Perform the probabilistic primality test with  $a$  as a parameter. If  $n$  fails the test, reject the value  $n$  and go to step 1
- If  $n$  has passed a sufficient number of tests, accept  $n$ ; otherwise, go to step 2



# The Security of RSA



# Summary

- Public-key cryptosystems
  - Applications for public-key cryptosystems
  - Requirements for public-key cryptography
  - Public-key cryptanalysis
- 
- The RSA algorithm
    - Description of the algorithm
    - Computational aspects
    - Security of RSA