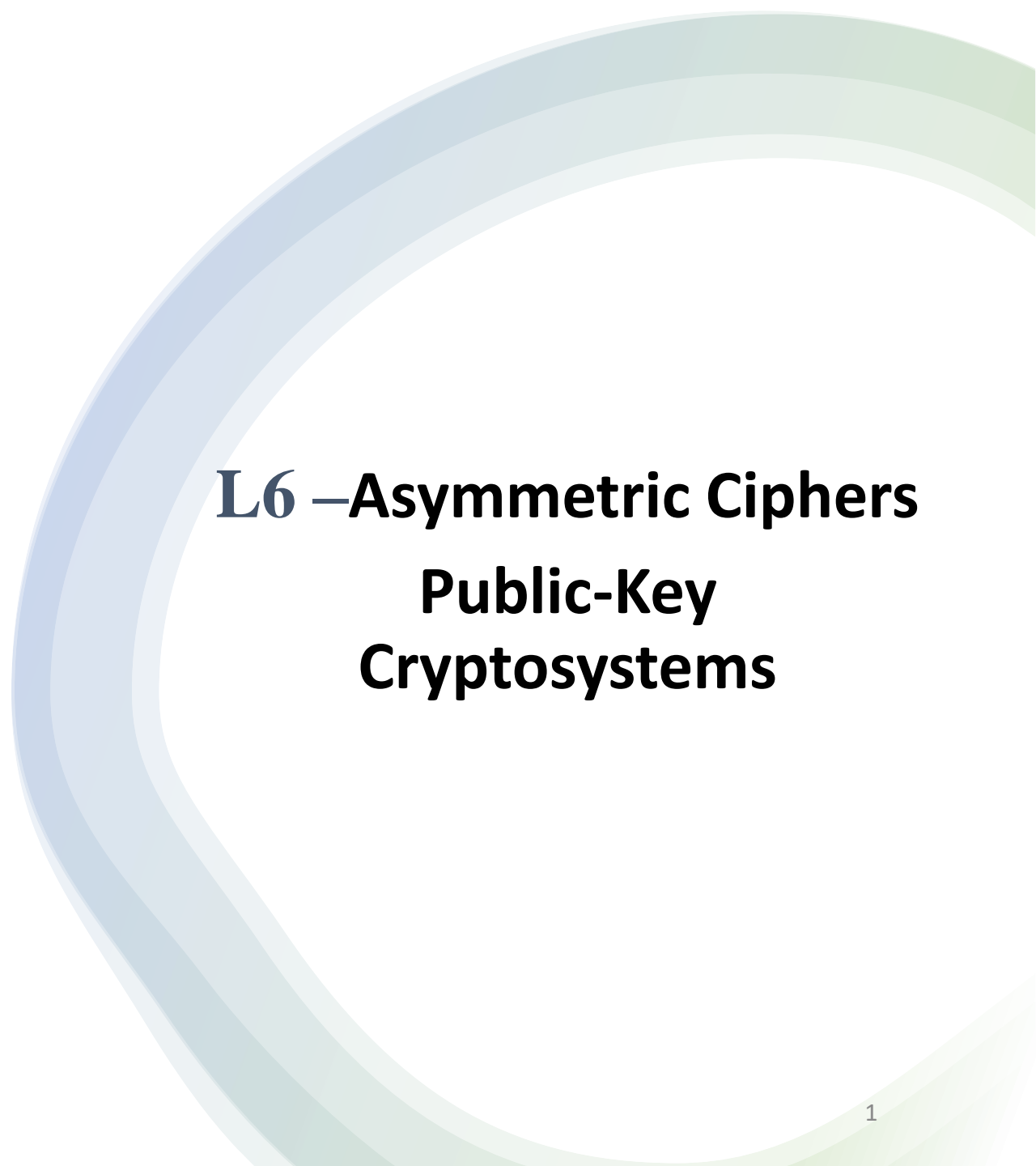




Data Security and Privacy

DSE 3258



L6 –Asymmetric Ciphers

Public-Key Cryptosystems

Public-Key Cryptosystems

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.

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.

Public-Key Cryptosystems

Algorithms rely on one key for encryption and a different but related key for decryption.

Important characteristics

- It is computationally infeasible to determine the decryption key with only knowledge of the cryptographic algorithm and the encryption key.
- Either of the two related keys can be used for encryption, with the other used for decryption. (exhibited in RSA)

Ingredients

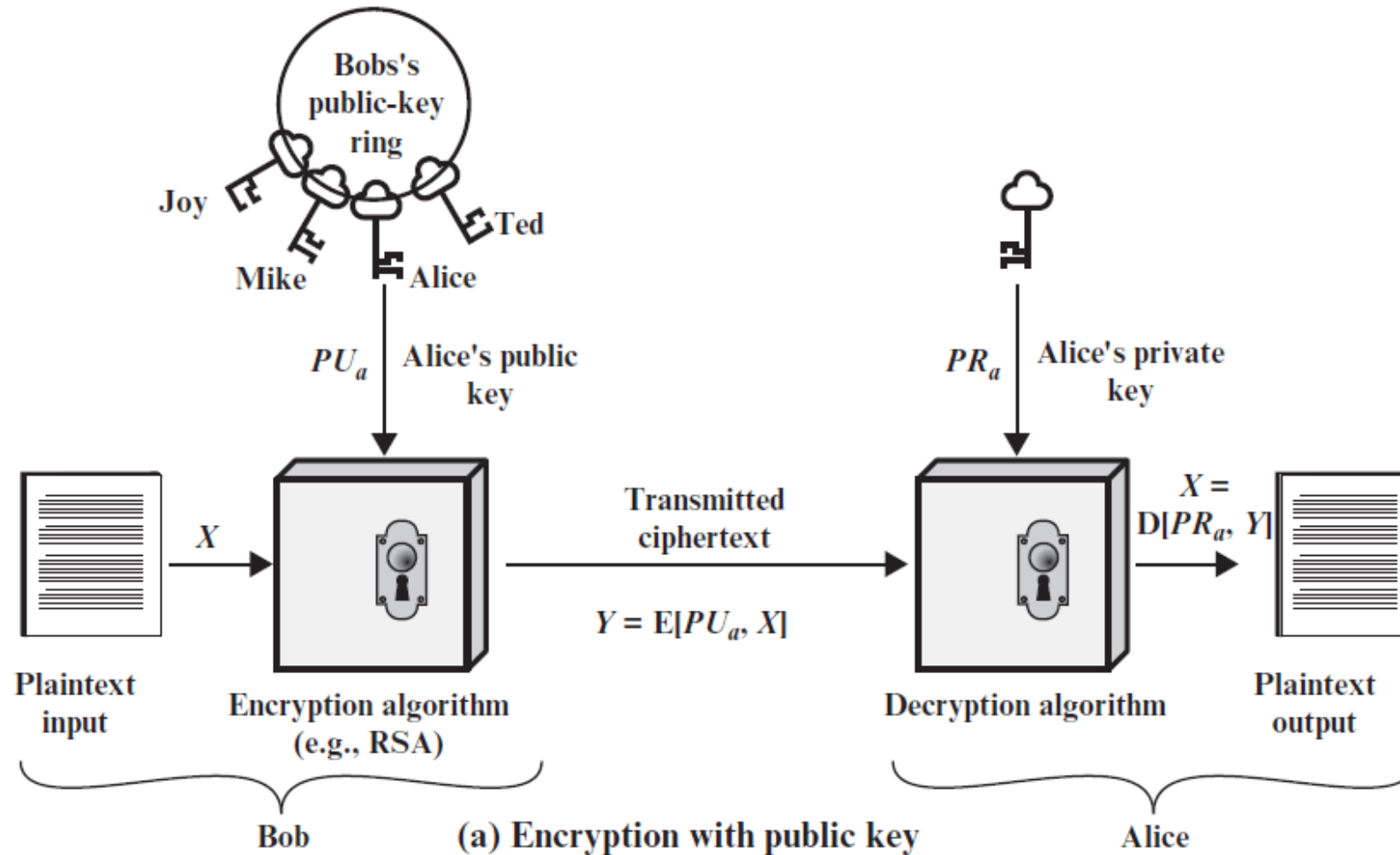
- Public-key encryption scheme has six ingredients
 - **Plaintext:** This is the readable message or data that is fed into the algorithm as input.
 - **Encryption algorithm:** The encryption algorithm performs various transformations on the plaintext.
 - **Public and private keys:** This is a pair of keys that have been selected so that if one is used for encryption, the other is used for decryption.
 - **Ciphertext:** This is the encrypted message produced as output. It depends on the plaintext and the key.
 - For a given message, two different keys will produce two different ciphertexts.
 - **Decryption algorithm:** This algorithm accepts the ciphertext and the matching key and produces the original plaintext.

The essential steps

The essential steps are the following:

1. Each user generates a pair of keys to be used for the encryption and decryption of messages.
2. Each user places one of the two keys in a public register or other accessible file. This is the public key. The companion key is kept private. Each user maintains a collection of public keys obtained from others.
3. If Bob wishes to send a confidential message to Alice, Bob encrypts the message using Alice's public key.
4. When Alice receives the message, she decrypts it using her private key. No other recipient can decrypt the message because only Alice knows Alice's private key.

Encryption with Public key



Encryption with Private key

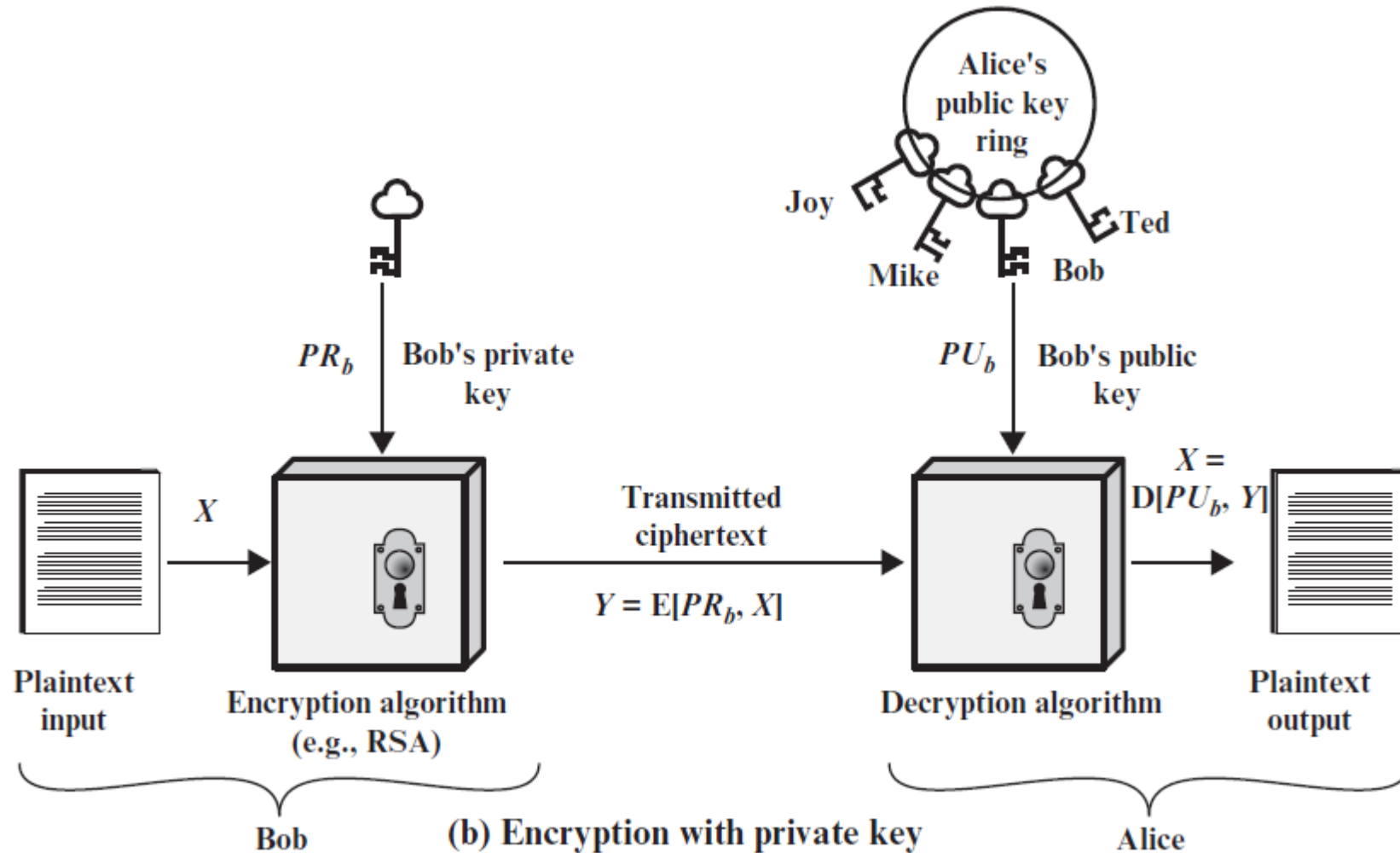


Table 9.2 Conventional and Public-Key Encryption

Conventional Encryption	Public-Key Encryption
<p><i>Needed to Work:</i></p> <ol style="list-style-type: none">1. The same algorithm with the same key is used for encryption and decryption.2. The sender and receiver must share the algorithm and the key. <p><i>Needed for Security:</i></p> <ol style="list-style-type: none">1. The key must be kept secret.2. It must be impossible or at least impractical to decipher a message if the key is kept secret.3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.	<p><i>Needed to Work:</i></p> <ol style="list-style-type: none">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.2. The sender and receiver must each have one of the matched pair of keys (not the same one). <p><i>Needed for Security:</i></p> <ol style="list-style-type: none">1. One of the two keys must be kept secret.2. It must be impossible or at least impractical to decipher a message if one of the keys is kept secret.3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.

Applications of PKC

1. Encryption / Decryption

Sender encrypts a message with the recipient's public key

2. Digital signature

sender signs a message with private key

3. Key exchange

two sides cooperate to exchange a session key

Requirements for public key cryptography

1. Pair of keys (public key KU_b , private key KR_b)
2. Easy to encrypt the message $C=E_{KU_b}(M)$
3. Easy to decrypt the ciphertext

$$M = D_{KR_b}(C)=D_{KR_b}[E_{KU_b}(M)]$$

4. Knowing KU_b , it is infeasible to determine KR_b
5. Knowing C & KU_b , it is infeasible to determine M
6. Either of 2 keys can be used for encryption

$$M = D_{KR_b}[E_{KU_b}(M)] = D_{KU_b}[E_{KR_b}(M)]$$

- An adversary, observing Y and having access to PU_b , but not having access to PR_b or X , must attempt to recover X and/or PR_b .
- It is assumed that the adversary does have knowledge of the encryption (E) and decryption (D) algorithms.
- If the adversary is interested only in this particular message, then the focus of effort is to recover X by generating a plaintext estimate X_n .
- Often, however, the adversary is interested in being able to read future messages as well, in which case an attempt is made to recover PR_b by generating an estimate PR_n .

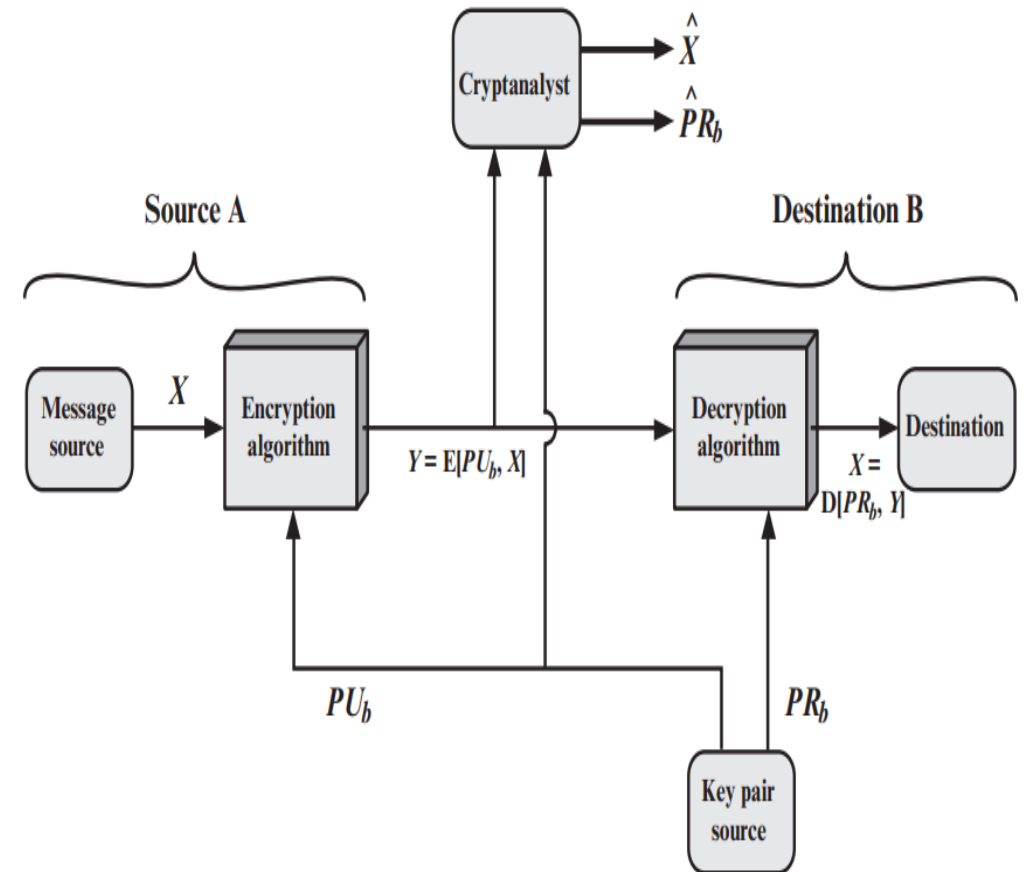


Figure 9.2 Public-Key Cryptosystem: Confidentiality

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

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

- In this case, A prepares a message to B and encrypts it using A's private key before transmitting it.
- B can decrypt the message using A's public key. Because the message was encrypted using A's private key, only A could have prepared the message.
- Therefore, the entire encrypted message serves as a digital signature.
- In addition, it is impossible to alter the message without access to A's private key, so the message is authenticated both in terms of source and in terms of data integrity

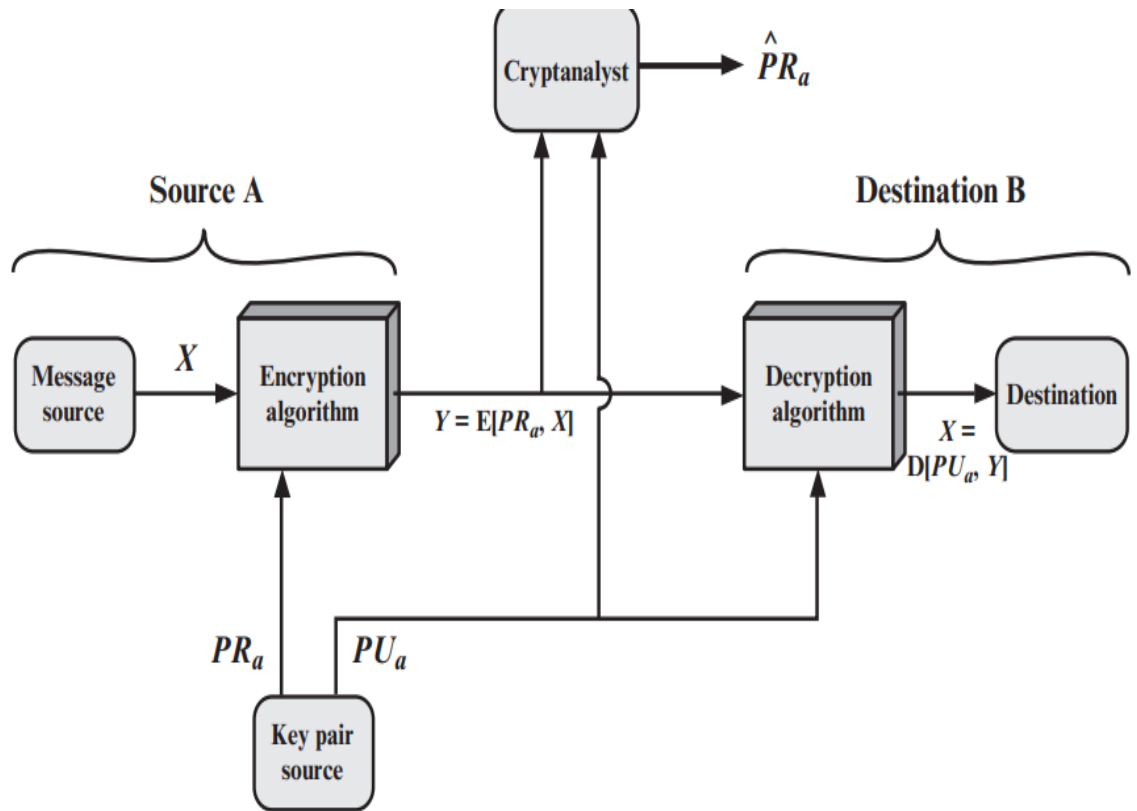


Figure 9.3 Public-Key Cryptosystem: Authentication

- **Drawback:**

- In the preceding scheme, the entire message is encrypted, which, although validating both author and contents, requires a great deal of storage.
- Each document must be kept in plaintext to be used for practical purposes.
- A copy also must be stored in ciphertext so that the origin and contents can be verified in case of a dispute.

- **Solution:**

- A more efficient way of achieving the same results is to encrypt a small block of bits that is a function of the document. Such a block, called an **authenticator**, must have the property that it is infeasible to change the document without changing the authenticator.

- It is, however, possible to provide both the authentication function and confidentiality by a double use of the public-key scheme

$$Z = E(PU_b, E(PR_a, X))$$

$$X = D(PU_a, D(PR_b, Z))$$

- In this case, we begin as before by encrypting a message, using the sender's private key.
- This provides the digital signature. Next, we encrypt again, using the receiver's public key.
- The final ciphertext can be decrypted only by the intended receiver, who alone has the matching private key. Thus, confidentiality is provided.
- The disadvantage of this approach is that the public-key algorithm, which is complex, must be exercised four times rather than two in each communication.

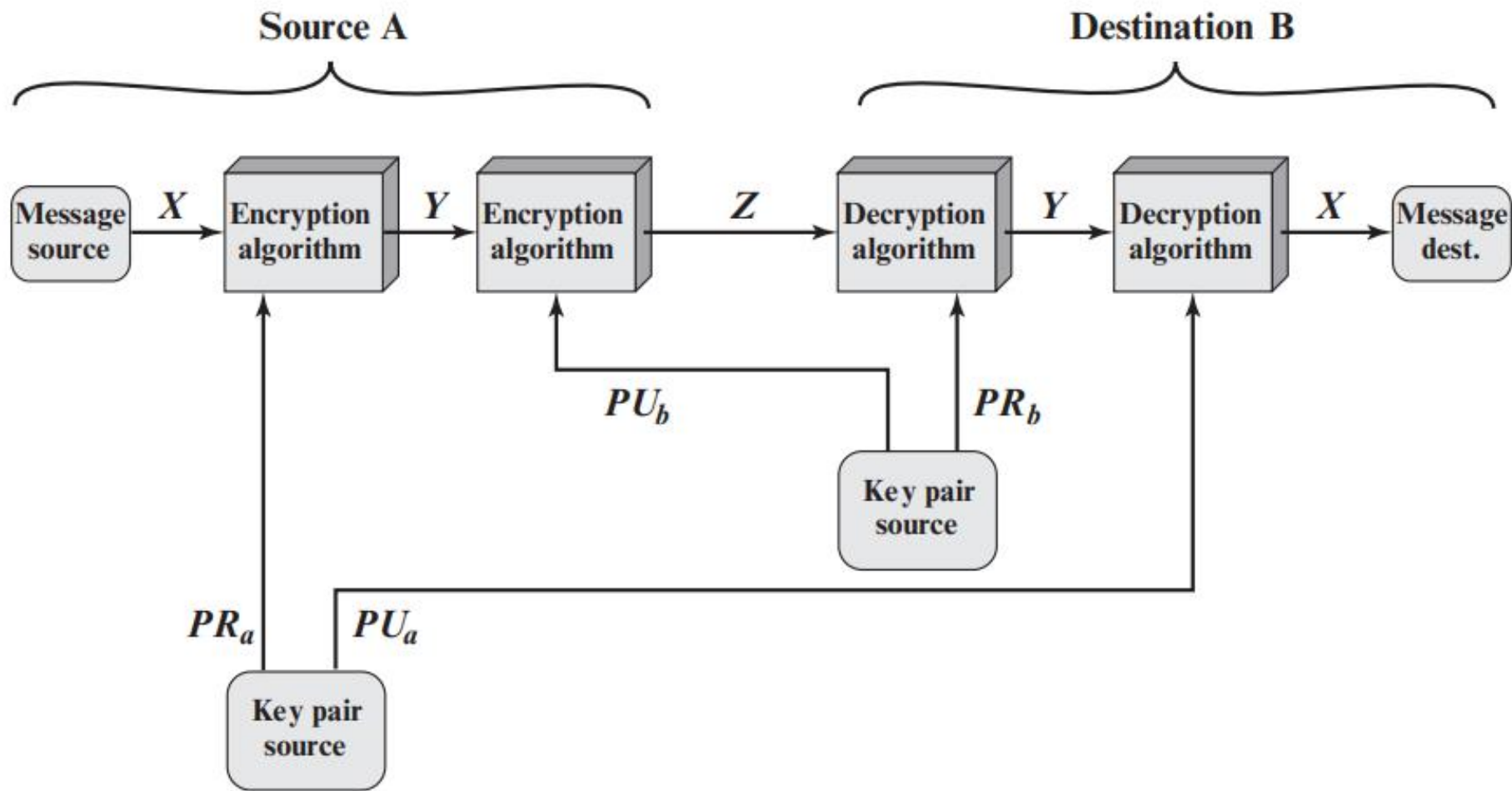


Figure 9.4 Public-Key Cryptosystem: Authentication and Secrecy

Applications of PKC

1. Encryption / Decryption

The sender encrypts a message with the recipient's public key, and the recipient decrypts the message with the recipient's private key.

2. Digital signature

The sender "signs" a message with its private key. Signing is achieved by a cryptographic algorithm applied to the message or to a small block of data that is a function of the message

3. Key exchange

Two sides cooperate to exchange a session key, which is a secret key for symmetric encryption generated for use for a particular transaction (or session) and valid for a short period of time. Several different approaches are possible, involving the private key(s) of one or both parties; t

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

Requirements for Public-Key Cryptography

1. It is computationally easy for a party B to generate a key pair (public key PUb, private key PRb).
2. It is computationally easy for a sender A, knowing the public key and the message to be encrypted, M, to generate the corresponding ciphertext: $C = E(\text{PUb}, M)$
3. It is computationally easy for the receiver B to decrypt the resulting ciphertext using the private key to recover the original message: $M = D(\text{PRb}, C) = D[\text{PRb}, E(\text{PUb}, M)]$
4. It is computationally infeasible for an adversary, knowing the public key, PUb, to determine the private key, PRb.
5. It is computationally infeasible for an adversary, knowing the public key, PUb, and a ciphertext, C, to recover the original message, M.

A sixth requirement that, although useful, is not necessary for all public-key applications:

6. The two keys can be applied in either order: $M = D[\text{PUb}, E(\text{PRb}, M)] = D[\text{PRb}, E(\text{PUb}, M)]$

Rivest-Shamir-Adleman (RSA) Algorithm

- The RSA scheme 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. That is, n is less than 2^{1024} .

- 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

Rivest-Shamir-Adleman (RSA) Algorithm

- each user generates a public/private key pair by:
- selecting two **large primes at random: p, q**
- computing their system **modulus $n=p.q$**
 - note **$\phi(n)=(p-1)(q-1)$** [Euler's totient function]
- 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**
 - **$e.d = 1 \bmod \phi(n)$ and $0 \leq d \leq n$**
- publish their public encryption key: $PU = \{e, n\}$
- keep secret private decryption key: $PR = \{d, n\}$

RSA Example:

1. Select primes: $p=17$ & $q=11$
2. Calculate $n = pq = 17 \times 11 = 187$
3. Calculate $\phi(n) = (p-1)(q-1) = 16 \times 10 = 160$
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 $23 \times 7 = 161 = 10 \times 160 + 1$
6. Publish public key $PU = \{7, 187\}$
7. Keep secret private key $PR = \{23, 187\}$

RSA Example:

- sample RSA encryption/decryption is:
- given message $M = 88$ (see that $88 < 187$)
- encryption:
$$C = 88^7 \bmod 187 = 11$$
- decryption:
$$M = 11^{23} \bmod 187 = 88$$

$$88^7 \bmod 187 = [(88^4 \bmod 187) \times (88^2 \bmod 187) \times (88^1 \bmod 187)] \bmod 187$$

$$88^1 \bmod 187 = 88$$

$$88^2 \bmod 187 = 7744 \bmod 187 = 77$$

$$88^4 \bmod 187 = 59,969,536 \bmod 187 = 132$$

$$88^7 \bmod 187 = (88 \times 77 \times 132) \bmod 187 = 894,432 \bmod 187 = 11$$

For decryption, we calculate $M = 11^{23} \bmod 187$:

$$11^{23} \bmod 187 = [(11^1 \bmod 187) \times (11^2 \bmod 187) \times (11^4 \bmod 187) \times (11^8 \bmod 187) \times (11^8 \bmod 187)] \bmod 187$$

$$11^1 \bmod 187 = 11$$

$$11^2 \bmod 187 = 121$$

$$11^4 \bmod 187 = 14,641 \bmod 187 = 55$$

$$11^8 \bmod 187 = 214,358,881 \bmod 187 = 33$$

$$\begin{aligned} 11^{23} \bmod 187 &= (11 \times 121 \times 55 \times 33 \times 33) \bmod 187 \\ &= 79,720,245 \bmod 187 = 88 \end{aligned}$$

Question: P and Q are two prime numbers. P=7, and Q=17. Take public key E=5. If plain text value is 6, then what will be cipher text value according to RSA algorithm? Again calculate plain text value from cipher text.

Solution:

1. Two prime numbers P=7, Q=17

2. $n = P * Q = 17 * 7 = 119$ **n = 119**

3. $\Phi(n) = (P-1) * (Q-1) = (17-1) * (7-1) = 16 * 6 = 96$ **$\Phi(n) = 96$**

4. Public key E = 5. **E = 5**

5. Calculate d = 77. $d = ((\Phi(n) * i) + 1) / e$ **d = 77**

$$d = ((96*1)+1) / 5 = 19.4$$

$$d = ((96*2)+1) / 5 = 38.6$$

$$d = ((96*3)+1) / 5 = 57.8$$

$$d = ((96*4)+1) / 5 = 77 \text{ (Stop finding d because getting integer value)}$$

6. Public key = {e, n} = {5, 119}, private key = {d, n} = {77, 119}.

7. Plain text PT = 6, $CT = PT^E \bmod n = 6^5 \bmod 119 = 41$. **Cipher Text = 41**

8. Cipher text CT = 41, $PT = CT^d \bmod n = 41^{77} \bmod 119 = 6$. **Plain Text = 6**

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:	$M = C^d \pmod n$

Figure 9.5 The RSA Algorithm

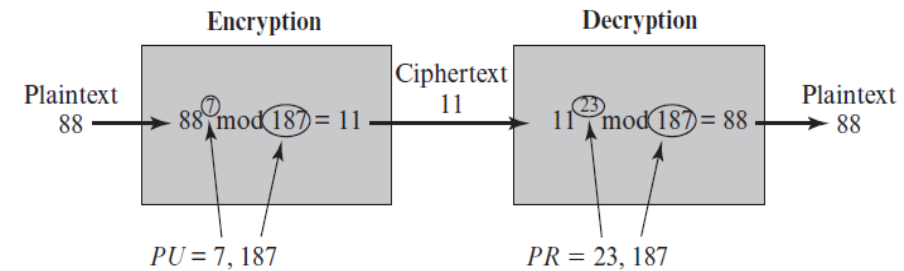


Figure 9.6 Example of RSA Algorithm

SOLVE

9.3 In a public-key system using RSA, you intercept the ciphertext $C = 20$ sent to a user whose public key is $e = 13$, $n = 77$. What is the plaintext M ?