BAHRIA UNIVERSITY
Discovering Knowledge

Cryptography

Information Security (CSC-407)

Fall 2024 (BSE-7A & 7B)

Basic Terminologies

- **Plaintext:** the original intelligible message.
- **Ciphertext:** the coded unintelligible message.
- **Enciphering\Encryption:** the process of converting plaintext to ciphertext.
- **Deciphering\Decryption:** the process of restoring plaintext from ciphertext.

Basic Terminologies (Cont.)

- **Cryptography:** the study of encryption.
- **Cryptanalysis:** techniques used for deciphering a message without any knowledge of the enciphering details, such as the keys used to perform encryption.
- **Cryptology:** The field of science that encompasses cryptography and cryptanalysis together.

Cryptographic Systems

- Cryptographic systems are characterized by three dimensions:

Type of operations used for converting plaintext to ciphertext	Number of keys used	The way in which plaintext is processed
Substitution	Symmetric, single-key, secret-key, conventional encryption	Block cipher
Transposition	Asymmetric, two-key, or public-key encryption	Stream cipher

Cryptographic Systems (Cont.)

Type of operations used for transforming plaintext to ciphertext:

- All encryption algorithms are based on two general principles:

  a. **Substitution,** in which each element in the plaintext (bit, letter, group of bits or letters) is mapped into another element.

  b. **Transposition**, in which elements in the plaintext are rearranged.

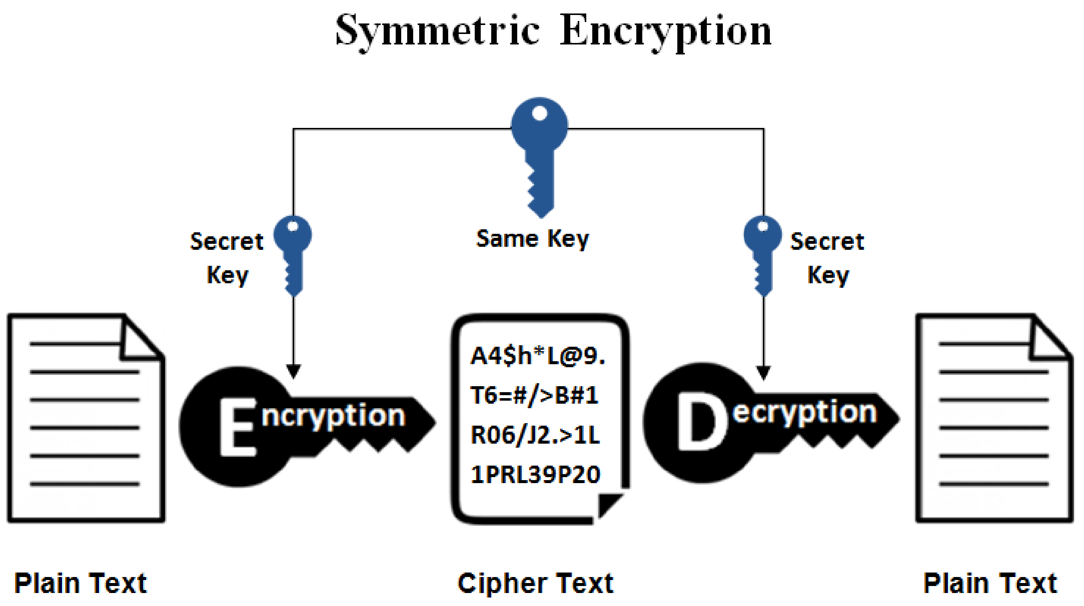
Cryptographic Systems (Cont.)

The way in which the plaintext is processed:

- **Block cipher** processes the input one block of elements at a time, producing an output block for each input block.
- **Stream cipher** processes the input elements continuously, producing output one element at a time, as it goes along.

Cryptographic Algorithms

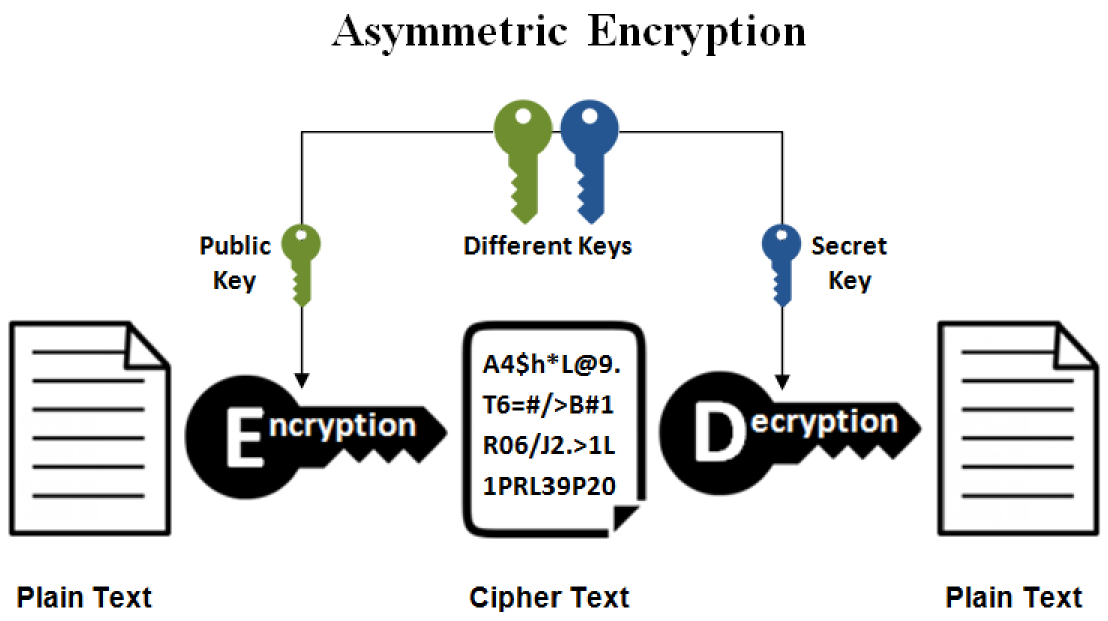
- Cryptographic algorithms can be grouped into:
  1. **Symmetric-key Algorithms:** cryptography algorithms that use the **same cryptographic keys** for both encryption and decryption.



Cryptographic Algorithms (Cont.)

- Cryptographic algorithms can be grouped into:

  2. **Asymmetric-key Algorithms:** cryptography algorithms that uses **pairs of keys**, i.e. public keys and private keys, to encrypt and decrypt data.

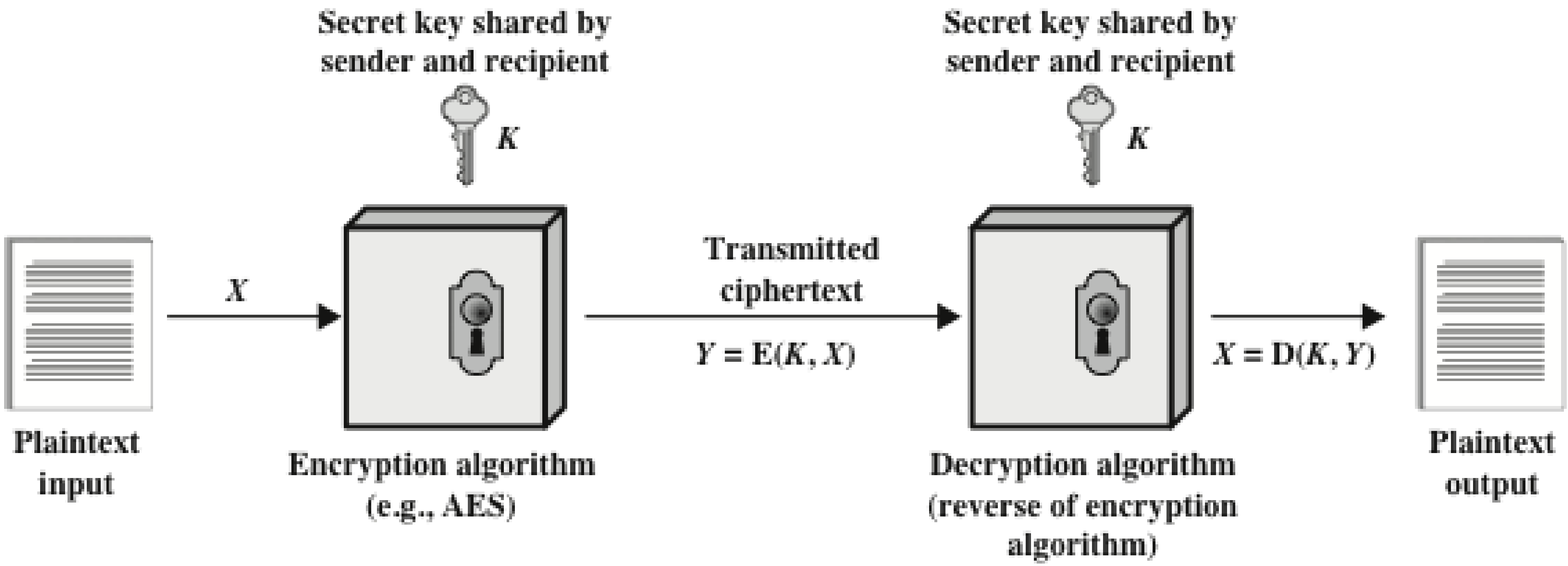


Cryptographic Algorithms (Cont.)

- **Symmetric Encryption** is used to conceal the contents of blocks or streams of data of any size, including messages, files, encryption keys and passwords.
- **Asymmetric Encryption** is used to conceal small blocks of data, such as encryption keys and hash function values, which are used in digital signatures.

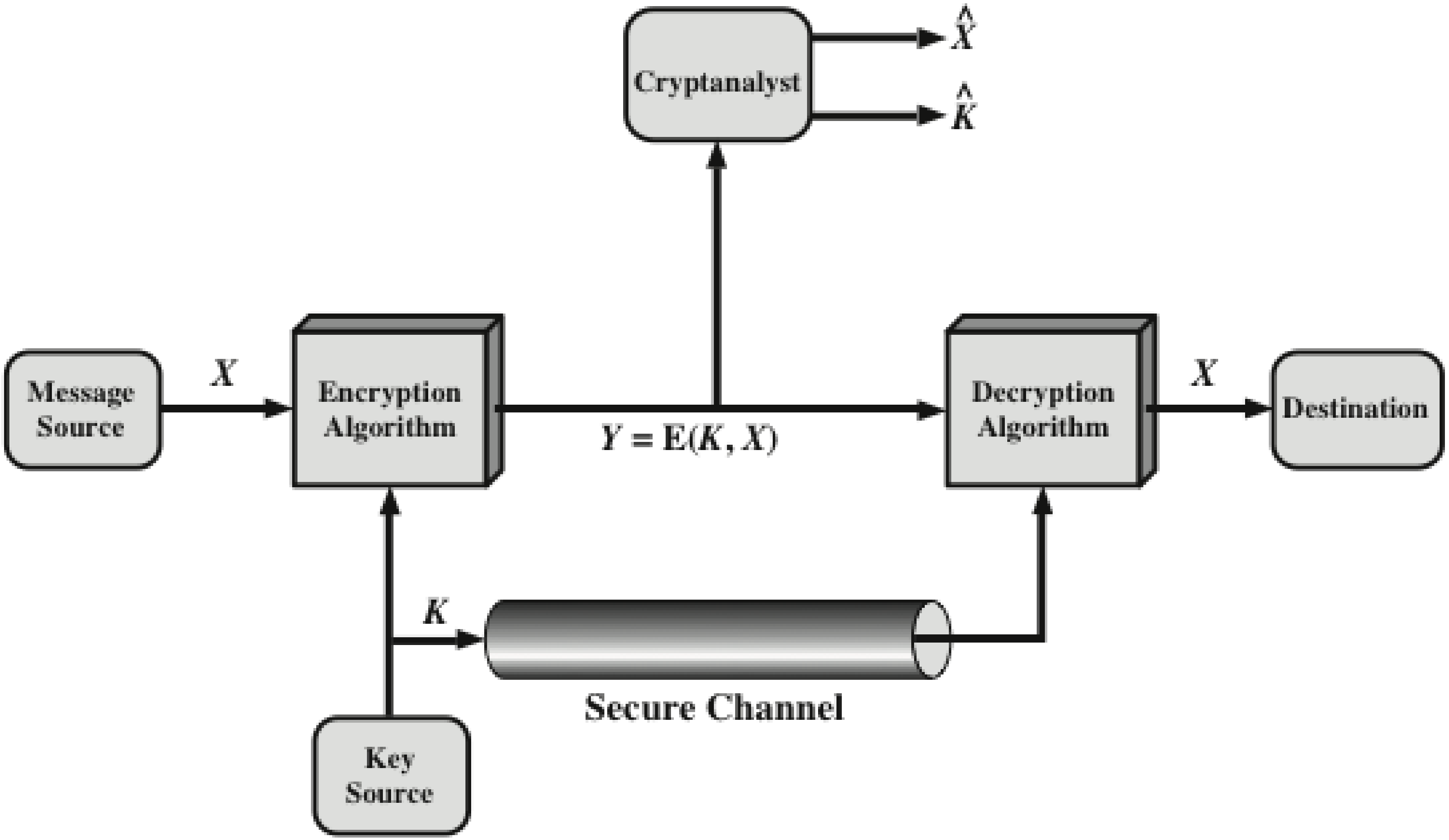
Symmetric Ciphers
(Substitution, Transposition)

Symmetric Encryption



A general model for the symmetric encryption.

Symmetric Encryption (Cont.)



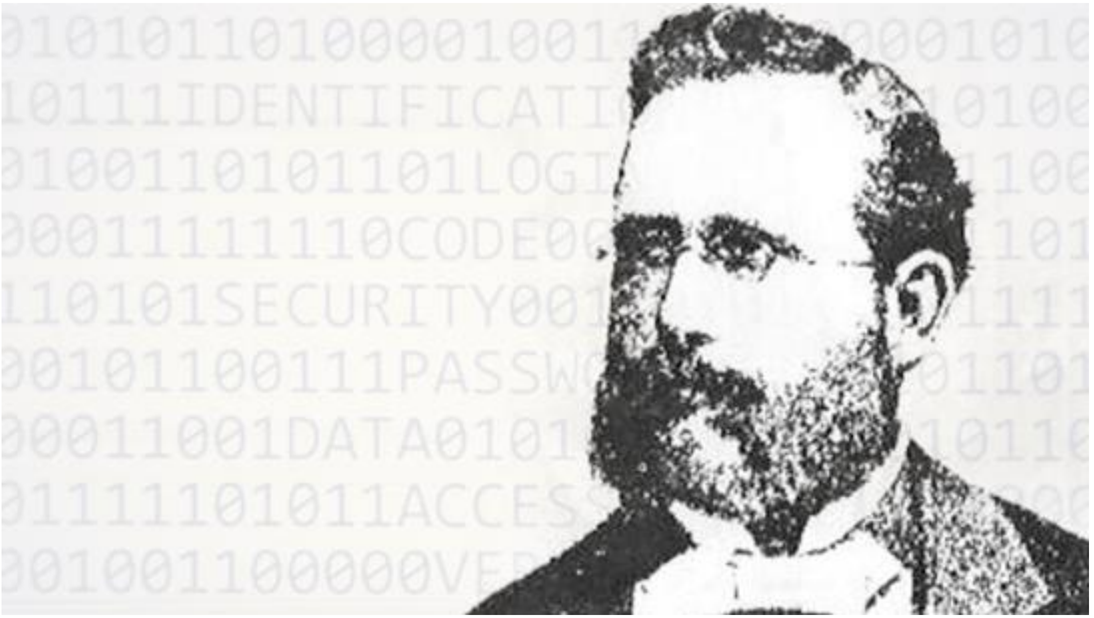
Symmetric Encryption (Cont.)

Properties of secret key in symmetric encryption:

- The **key** is input to **encryption algorithm** along with **plaintext**.
- The **key** is a value independent of the **plaintext** and the **algorithm**.
- The **algorithm** will produce a different output depending on the specific **key** being used. Hence, for a given message, two different **keys** will produce two different **ciphertexts**.
- The **encryption algorithm** performs various substitutions and transpositions on the **plaintext**, where the exact substitutions and transpositions depends on the **key**.

Symmetric Encryption (Cont.)

- **Kerckhoff's principle:** one should always assume that the adversary knows the **encryption/decryption algorithm**. The resistance of the cipher to attack must be based only on the secrecy of the key.



Symmetric Encryption (Cont.)

- It is "impractical" to decrypt a message on the basis of ciphertext plus knowledge of encryption/decryption algorithm.
- There is no need to keep the algorithm secret; but only keep the key secret. ***This feature makes symmetric key feasible for widespread use. Hence, manufacturers can and have developed low-cost chip implementations of data encryption algorithms***.
- With the use of symmetric encryption, the principal security problem is maintaining the secrecy of the key.

Cryptanalysis and Brute-Force

- There are two general approaches for attacking a conventional encryption scheme:
  - **Cryptanalysis,** rely on nature of the algorithm plus some general characteristics of plaintext or plaintext–ciphertext pairs. This attack attempts to deduce a specific plaintext or the key being used.
  - **Brute-force attack**, the attacker tries every possible key on a piece of ciphertext until an intelligible translation is obtained. ***On average, half of all possible keys must be tried to achieve success!***

Cryptanalysis and Brute-Force (Cont.)

- In brute-force attack, the attack is proportional to **key size**.

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/µs	Time required at 10^6 decryptions/µs
32	$2^{32} = 4.3 \times 10^9$	35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	5.9×10^{36} years	5.9×10^{30} years

Substitution Technique

Substitution Technique

- Techniques in which the letters of plaintext are replaced by other letters, numbers or symbols.
- If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns.
- Substitution ciphers can be categorized as either **monoalphabetic ciphers** or **polyalphabetic ciphers**.

Monoalphabetic Ciphers

- In **monoalphabetic** substitution, the relationship between a symbol in the plaintext to a symbol in the ciphertext is always **one-to-one**.
- **Example:** the following shows a plaintext and its corresponding ciphertext. The cipher is **monoalphabetic** because both ***l'*s** are encrypted as ***O'*s**.

Plaintext: hello **Ciphertext:** KHOOR

Caesar Cipher

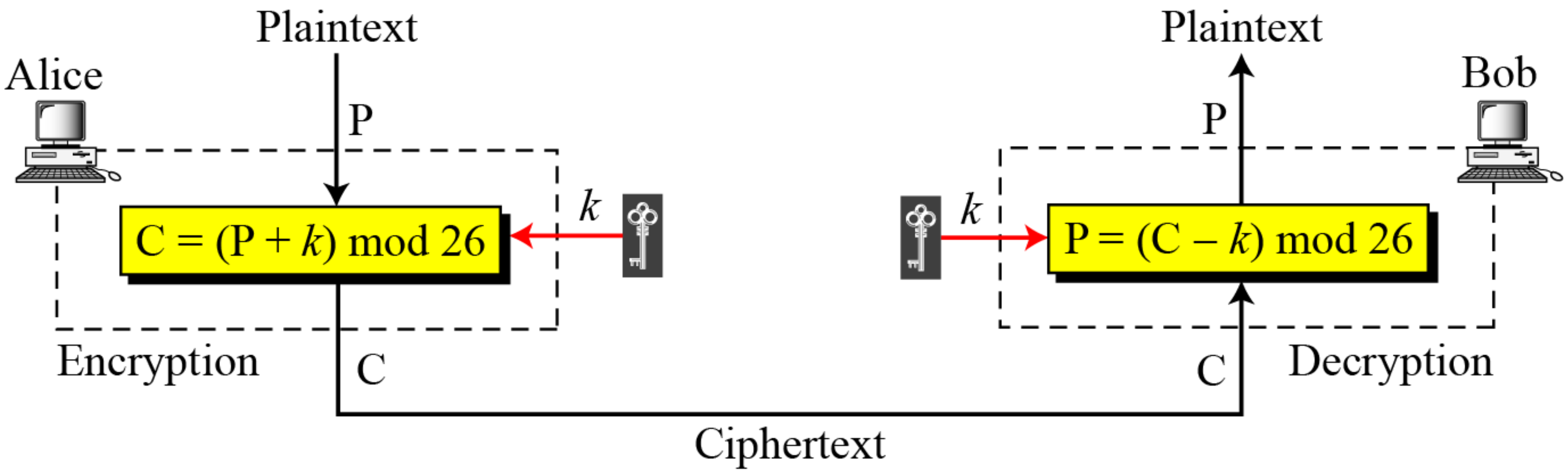
- **Caesar cipher** is a **monoalphabetic** cipher that involves replacing each letter of alphabet with that standing **3 places** further down the alphabet, where alphabet is wrapped around so the letter following **Z** is **A**.

```
plain:  meet me after the toga party
cipher: PHHW PH DIWHU WKH WRJD SDUWB
```

- The shift may be of any amount, so the general **Caesar cipher** is called an **additive cipher**.

Additive Cipher

Plaintext →	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext →	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25



Additive Cipher (Cont.)

- The plaintext, ciphertext and key are integers in $\mathbf{Z}_{26}$.
- The general algorithm is:

$$C = E(k, P) = (P + k) \bmod 26$$

- The decryption algorithm is simply:

$$P = D(k, C) = (C - k) \bmod 26$$

- Where, k takes on a value in the range 1 to 25).
- With $k = 0$, ciphertext is same as plaintext. Hence, only 25 keys are useful.

Additive Cipher (Cont.)

- **Example 01:** use the additive cipher with $k = 15$ to encrypt the message “hello”.
- **Solution:**

Plaintext: h → 07	Encryption: (07 + 15) mod 26	Ciphertext: 22 → W
Plaintext: e → 04	Encryption: (04 + 15) mod 26	Ciphertext: 19 → T
Plaintext: l → 11	Encryption: (11 + 15) mod 26	Ciphertext: 00 → A
Plaintext: l → 11	Encryption: (11 + 15) mod 26	Ciphertext: 00 → A
Plaintext: o → 14	Encryption: (14 + 15) mod 26	Ciphertext: 03 → D

Additive Cipher (Cont.)

- **Example 02:** use the additive cipher with k = 15 to decrypt the message “WTAAD”.
- **Solution:**

Ciphertext: W → 22	Decryption: (22 – 15) mod 26	Plaintext: 07 → h
Ciphertext: T → 19	Decryption: (19 – 15) mod 26	Plaintext: 04 → e
Ciphertext: A → 00	Decryption: (00 – 15) mod 26	Plaintext: 11 → l
Ciphertext: A → 00	Decryption: (00 – 15) mod 26	Plaintext: 11 → l
Ciphertext: D → 03	Decryption: (03 – 15) mod 26	Plaintext: 14 → o

Additive Cipher (Cont.)

- If it is known that a given ciphertext is an Additive cipher, then brute-force cryptanalysis is easily performed.
- ***Simply try all the 25 possible keys.***
- In this example, the plaintext leaps out as occupying the third line.
- *Note:* with only 25 possible keys, the Caesar cipher is far from secure.

KEY	PHHW PH DIWHU WKH WRJD SDUWB
1	oggv og chvgt vjg vqic rctva
2	nffu nf bgufs uif uphb qbsuz
3	meet me after the toga party
4	ldds ld zesdq sgd snfz ozqsx
5	kccr kc ydrcp rfc rmey nyprw
6	jbbq jb xcqbo qeb qldx mxoqv
7	iaap ia wbpan pda pkcw lwnpu
8	hzzo hz vaozm ocz ojbv kvmot
9	gyyn gy uznyl nby niau julns
10	fxxm fx tymxk max mhzt itkmr
11	ewwl ew sxlwj lzw lgys hsjlq
12	dvvk dv rwkvi kyv kfxr grikp
13	cuuj cu qvjuh jxu jewq fqhjo
14	btti bt puitg iwt idvp epgin
15	assh as othsf hvs hcuo dofhm
16	zrrg zr nsgre gur gbtn cnegl
17	yqqf yq mrfqd ftq fasm bmdfk
18	xppe xp lqepc esp ezrl alcej
19	wood wo kpdob dro dyqk zkbdi
20	vnnc vn jocna cqn cxpj yjach
21	ummb um inbmz bpm bwoi xizbg
22	tlla tl hmaly aol avnh whyaf
23	skkz sk glzkx znk zumg vgxze
24	rjjy rj fkyjw ymj ytlf ufwyd
25	qiix qi ejxiv xli xske tevxc

Exploiting Monoalphabetic Ciphers

- **Cryptanalysis** with frequency of characters in English:

Letter	*Frequency*	*Letter*	*Frequency*	*Letter*	*Frequency*	*Letter*	*Frequency*
E	12.7	H	6.1	W	2.3	K	0.08
T	9.1	R	6.0	F	2.2	J	0.02
A	8.2	D	4.3	G	2.0	Q	0.01
O	7.5	L	4.0	Y	2.0	X	0.01
I	7.0	C	2.8	P	1.9	Z	0.01
N	6.7	U	2.8	B	1.5		
S	6.3	M	2.4	V	1.0		

Exploiting Monoalphabetic Ciphers (Cont.)

- **Example:** the attacker has intercepted the following ciphertext. Using a **statistical attack**, find the plaintext.

XLILSYWIMWRSAJSVWEPIJSVJSYVQMPPMSRHSPPEVWMXMWASVX-LQSVILY-VVCFIJSVIXLIWIPPIVVIGIMZIWQSVISJJIVW

- **Solution:** after tabulating the frequency of letters in this ciphertext, we get: **I =14**, **V =13**, **S =12**, and so on. The most common character is **I** with 14 occurrences. **This means key = 4**.

the house is now for sale for four million dollars it is worth more hurry before the seller receives more offers

Exploiting Monoalphabetic Ciphers

- **Monoaphabetic ciphers** are easy to break because they reflect **frequency** of the original alphabet.
- However, a countermeasure is to provide multiple substitutes for a single letter.

Polyalphabetic Ciphers

- In polyalphabetic substitution, each occurrence of a character may have a different substitute.
- The relationship between a character in plaintext to a character in ciphertext is **one-to-many**.
- **E.g.** **“a”** could be enciphered as **“D”** at beginning of text, but as **“N”** at the middle.
- **Benefit of polyalphabetic ciphers:**
  - Hides letter frequency of the language.
  - Cannot use frequency statistics to break the ciphertext.

Polyalphabetic Ciphers (Cont.)

- We need to make each **ciphertext character** dependent on both corresponding **plaintext character(s)** and **position** of plaintext character(s) in the message.
- The key should be a stream of subkeys, i.e. $\boldsymbol{k = \{k_1, k_2, k_3, \dots\}}$.
- $\boldsymbol{k_i}$ is used to encipher the $\boldsymbol{i}$**th** character in **plaintext** to create the $\boldsymbol{i}$**th** character in **ciphertext**.

Hill Cipher

- The plainttext is divided into **equal-size blocks** that are encrypted one at a time, where m is the size of the block.
- Each character in a block contributes to the encryption of other characters in the block.
- Hill cipher algorithm takes m successive plaintext letters and **substitutes** by m ciphertext letters.
- The key is a square matrix of size $m \times m$.

Hill Cipher (Cont.)

- Key in Hill cipher:

$$K = \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1m} \\ k_{21} & k_{22} & \cdots & k_{2m} \\ \vdots & \vdots & & \vdots \\ k_{m1} & k_{m2} & \cdots & k_{mm} \end{bmatrix}$$

- The key matrix needs to have a **multiplicative inverse**, where not all square matrices do in Z_{26}.

Hill Cipher (Cont.)

- Substitution is determined by *m* linear equations.
- If we call *m* characters in plaintext block $P_1, P_2, \ldots, P_m$, the corresponding characters in ciphertext block are $C_1, C_2, \ldots, C_m$.
- ***Each ciphertext character depends on all plaintext characters***.

$$
\begin{aligned}
C_1 &= P_1 k_{11} + P_2 k_{21} + \cdots + P_m k_{m1} \\
C_2 &= P_1 k_{12} + P_2 k_{22} + \cdots + P_m k_{m2} \\
&\cdots \\
C_m &= P_1 k_{1m} + P_2 k_{2m} + \cdots + P_m k_{mm}
\end{aligned}
$$

Hill Cipher (Cont.)

- **E.g.** for $m = 3$, the system can be described as:

$$c_1 = (k_{11}p_1 + k_{21}p_2 + k_{31}p_3) \bmod 26$$
$$c_2 = (k_{12}p_1 + k_{22}p_2 + k_{32}p_3) \bmod 26$$
$$c_3 = (k_{13}p_1 + k_{23}p_2 + k_{33}p_3) \bmod 26$$

- This can be expressed in terms of row vectors and matrices as:

$$(c_1\ c_2\ c_3) = (p_1\ p_2\ p_3)\begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \bmod 26$$

$$\mathbf{C} = \mathbf{PK} \bmod 26$$

Hill Cipher (Cont.)

- **Example 01:** consider the plaintext **"paymoremoney"** and use the encryption key for encryption purpose through Hill cipher, where $m = 3$.

$$\mathbf{K} = \begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix}$$

Solution:

- The first three letters of the plaintext are represented by the vector (15 0 24).
- Then (15 0 24)**K** = (303 303 531) mod 26 = (17 17 11) = RRL.

Hill Cipher (Cont.)

Solution (Cont.):

- Continuing this way, the ciphertext for the entire plaintext is RRLMWBKASPDH.
- Decryption requires using the inverse of the matrix **K**.

$$\mathbf{K}^{-1} = \begin{pmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 17 \end{pmatrix}$$

- It can be seen that if the matrix $\mathbf{K}^{-1}$ is applied to the ciphertext, then the plaintext is recovered.

Hill Cipher (Cont.)

- **Example 02:** the plaintext "code is ready" can make a 3 × 4 matrix, where *m* = 4, when adding extra bogus character "z" to the last block along with removing spaces.
- Ciphertext is:

  "OHKNIHGLLISS"

- *Note:* (mod 26) is applied.

$$\overset{C}{\begin{bmatrix} 14 & 07 & 10 & 13 \\ 08 & 07 & 06 & 7 \\ 5 & 08 & 18 & 18 \end{bmatrix}} = \overset{P}{\begin{bmatrix} 02 & 14 & 03 & 04 \\ 08 & 18 & 17 & 04 \\ 00 & 03 & 24 & 25 \end{bmatrix}} \overset{K}{\begin{bmatrix} 09 & 07 & 11 & 13 \\ 04 & 07 & 05 & 06 \\ 02 & 21 & 14 & 09 \\ 03 & 23 & 21 & 08 \end{bmatrix}}$$

a. Encryption

$$\overset{P}{\begin{bmatrix} 02 & 14 & 03 & 04 \\ 08 & 18 & 17 & 04 \\ 00 & 03 & 24 & 25 \end{bmatrix}} = \overset{C}{\begin{bmatrix} 14 & 07 & 10 & 13 \\ 08 & 07 & 06 & 7 \\ 5 & 08 & 18 & 18 \end{bmatrix}} \overset{K^{-1}}{\begin{bmatrix} 02 & 15 & 22 & 03 \\ 15 & 00 & 19 & 03 \\ 09 & 09 & 03 & 11 \\ 17 & 00 & 04 & 07 \end{bmatrix}}$$

b. Decryption

Hill Cipher (Cont.)

Hill Cipher Benefits:

- It completely hides single-letter frequencies.
- The use of a larger matrix hides more frequency information.
- A Hill cipher hides not only single-letter but also two-letter frequency information.
- Hill cipher is strong against a **ciphertext-only** attack.

Transposition Technique

Transposition Ciphers

- Transposition Ciphers does not substitute one symbol for another, rather it changes location of the symbols.
- Transposition Ciphers performs some sort of ***permutation*** on the plaintext letters i.e. reorders (transposes) the symbols.
- Has two types of ciphering:
    1. Keyless transposition ciphers
    2. Keyed transposition ciphers

Keyless Transposition Ciphers

- Simple transposition ciphers, which were used in the past, are keyless. **E.g.**:
  - ***Text is written into table column by column and then transmitted row by row.***
  - ***Text is written into table row by row and then transmitted column by column.***

Keyless Transposition Ciphers (Cont.)

- **Example:** sender and receiver can agree on the number of columns. Sender writes the same plaintext, row by row, in a table of four columns.

m	e	e	t
m	e	a	t
t	h	e	p
a	r	k	

- The ciphertext is "MMTAEEHREAEKTTP".

Keyed Transposition Ciphers

- In keyed ciphers, the plaintext is divided into groups of predetermined size, **called blocks**, and then a **key** is used to permute the characters in each block separately.
- **Example:** plaintext message "Enemy attacks tonight".
- **Solution:** arrange the text in blocks of size 5 characters each. Followed by sending each character, within a block, in the sequence defined by key.

e	n	e	m	y	a	t	t	a	c	k	s	t	o	n	i	g	h	t	z

Keyed Transposition Ciphers (Cont.)

- **Example (Cont.):**
- The use key is

Encryption ↓

3	1	4	5	2
1	2	3	4	5

↑ Decryption

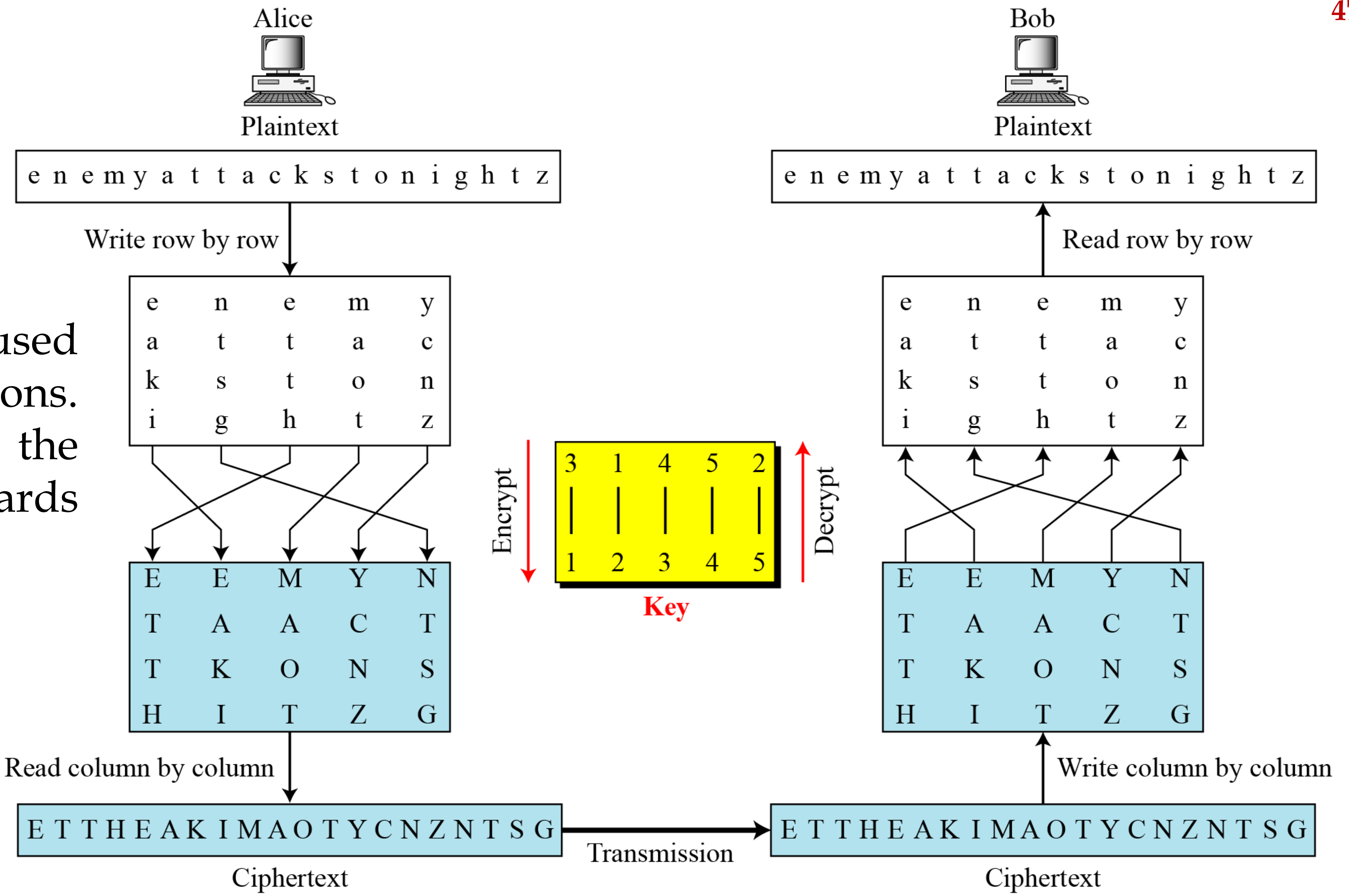
- The permutation yields

E	E	M	Y	N	T	A	A	C	T	T	K	O	N	S	H	I	T	Z	G

Columnar Transposition Ciphers

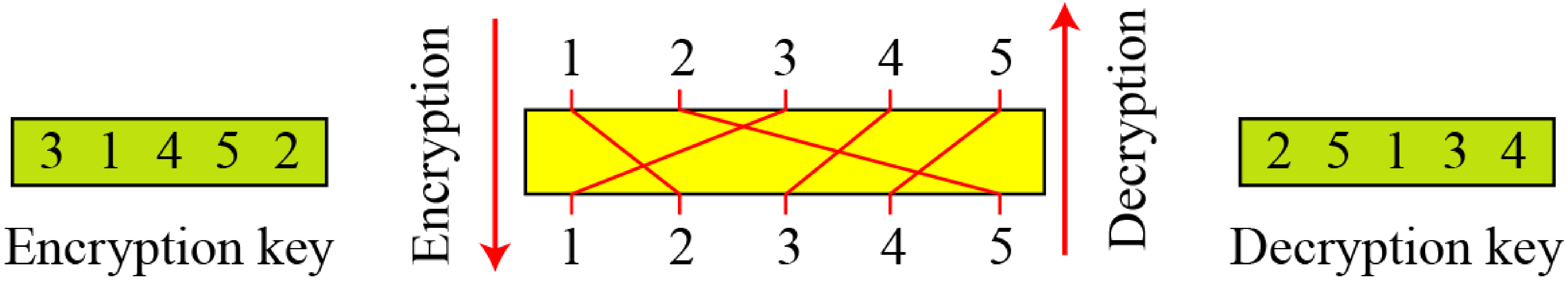
- To achieve better scrambling, we can combine the two approaches (Keyless & Keyed).
- Below is an *example* in which the following steps are followed:
  1. Text is written into table row by row.
  2. Permutation is done by reordering the columns, i.e. according to the provided key.
  3. A new table is read column by column.
- ***Note:*** the 1st and 3rd steps provide a keyless reordering, while the second step provides a keyed reordering.

A **single key** is used in two directions. Downwards for the encryption; upwards for decryption.



Columnar Transposition Ciphers (Cont.)

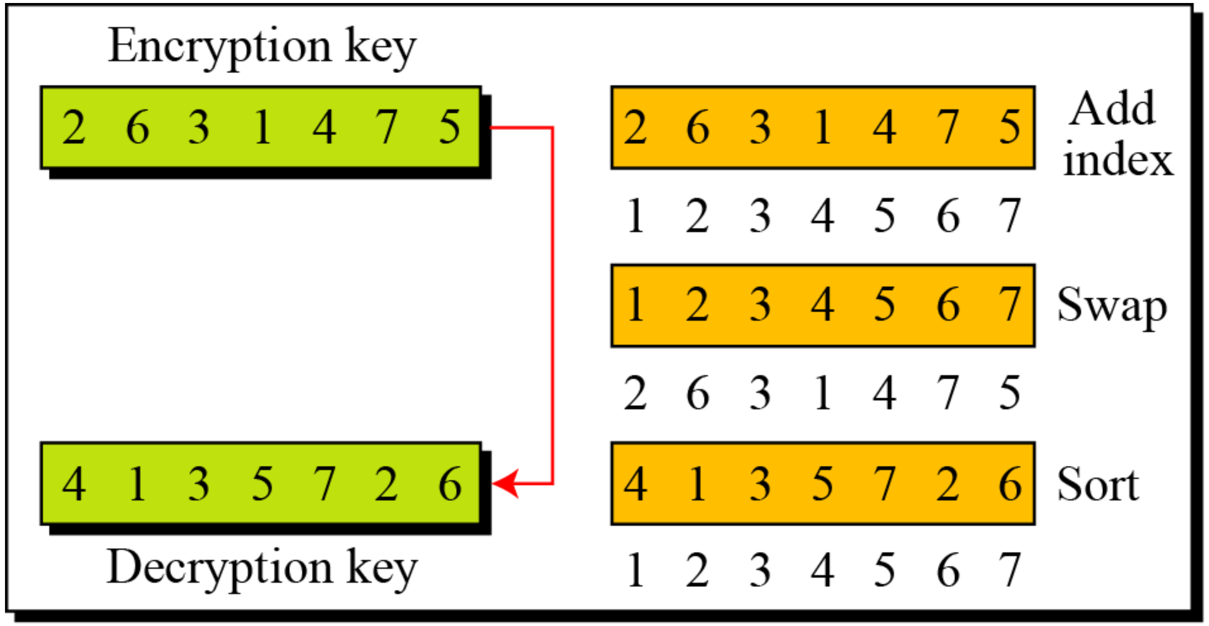
- It is customary to create two keys:



- In encryption key (3 1 4 5 2), the first entry shows that column 3 in source becomes column 1 in destination.
- In decryption key (2 5 1 3 4), the first entry shows that column 2 in source becomes column 1 in destination.

Columnar Transposition Ciphers (Cont.)

- It is possible to create the decryption key if the encryption key is provided as given below:

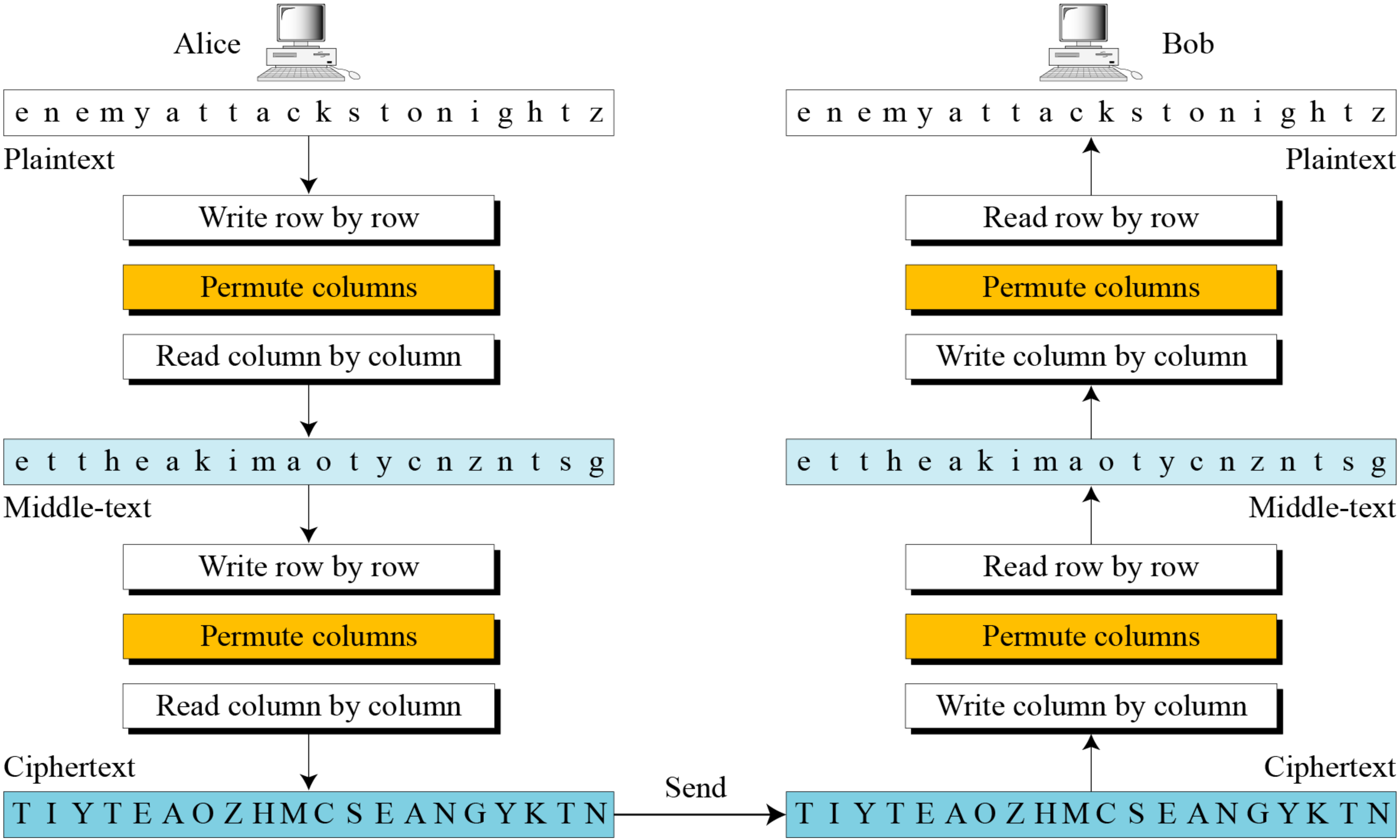


a. Manual process

```
Given: EncKey [index]
index ← 1
while (index ≤ Column)
  {
    DecKey[EncKey[index]] ← index
    index ← index + 1
  }
Return : DecKey [index]
```

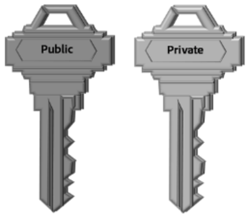
b. Algorithm

Double Transposition Ciphers



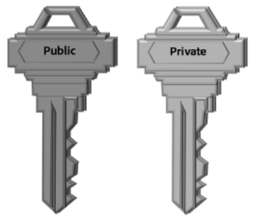
Asymmetric Ciphers (Public-Key Cryptography)

Overview



- Public-key cryptography is **asymmetric**, i.e. involving use of ***two separate keys***.
- **Asymmetric Keys:** two "related" keys, a **public** and **private key**, used to perform **complementary operations**.
- There is nothing about **symmetric** or **asymmetric** encryption that makes one superior to another w.r.t. ***cryptanalysis***.
- Computational overhead of public-key encryption exists w.r.t. key management.

Public-Key Cryptosystems



- The concept of **public-key cryptography** evolved from an attempt to solve two of the most difficult problems in symmetric encryption:

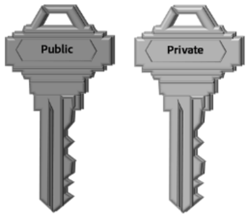
Key distribution

- How to have secure communications in general without having to trust a Key Distribution Center with your key

Digital signatures

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

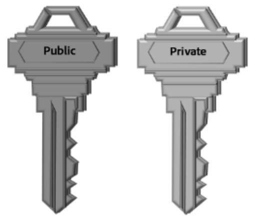
Public-Key Cryptosystems (Cont.)



Mechanism in public-key cryptosystem:

- Each user generates a **pair of keys** to be used for encryption and decryption of messages.
- Each user places one of the two keys in a **public register** ***(this is the public key)***, while the companion key is kept private.
- All participants have access to public keys, hence each user has a collection of public keys obtained from others.

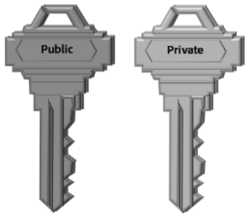
Public-Key Cryptosystems (Cont.)

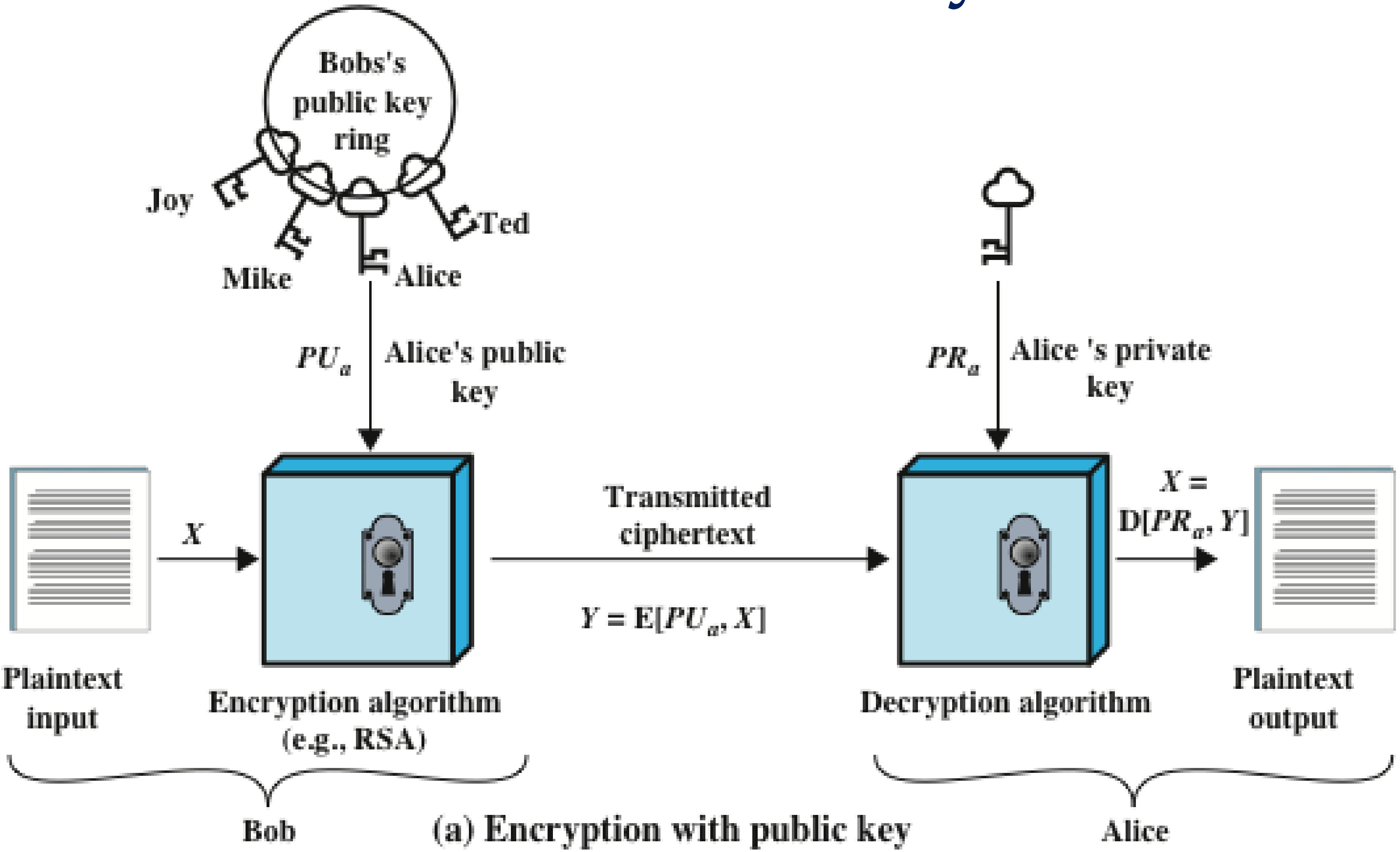


Mechanism in public-key cryptosystem (Cont.):

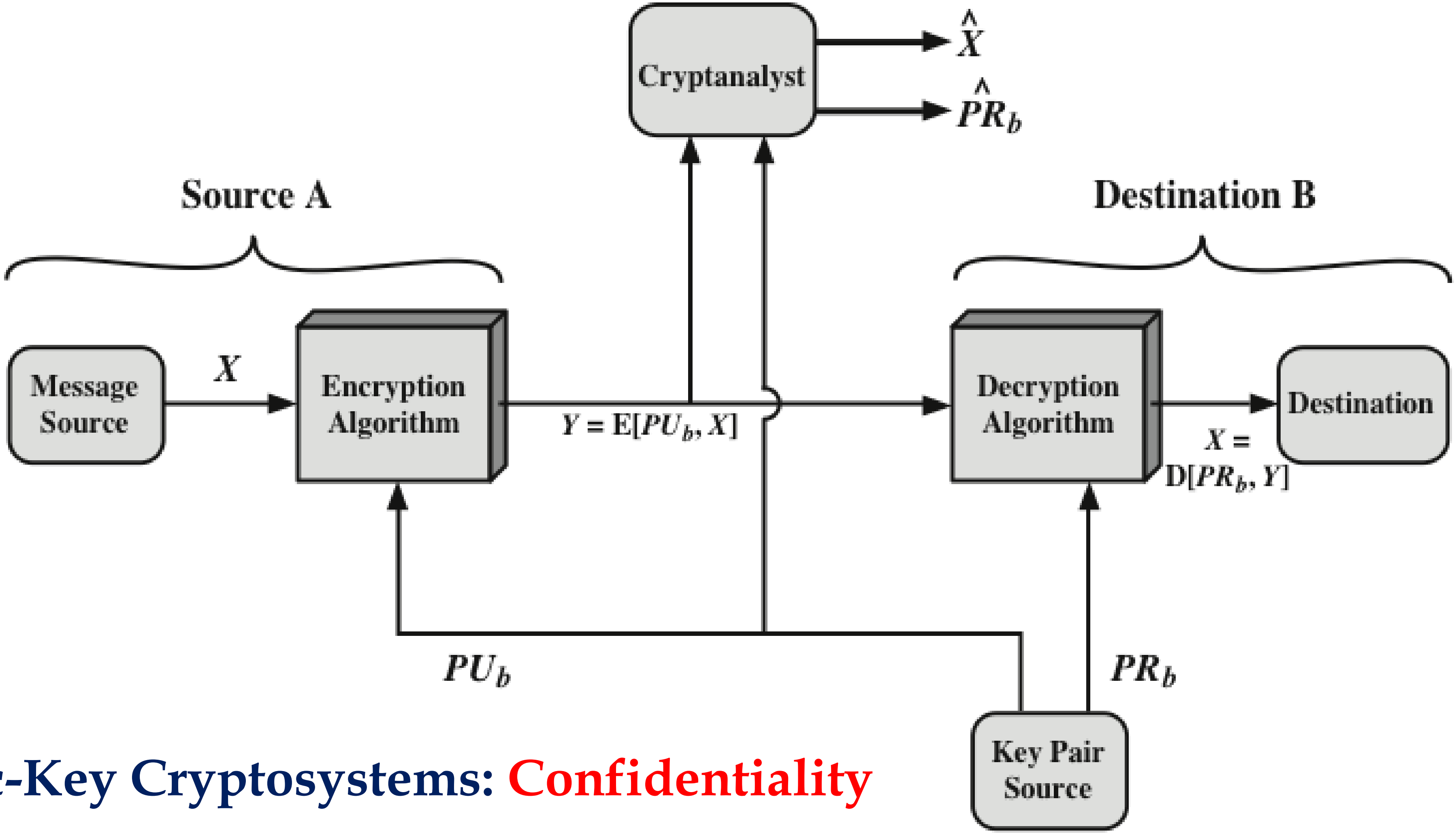
- Private keys are generated locally by each participant and therefore need never be distributed.
- At any time, a system can change its private key and publish the companion public key to replace its old public key.

Public-Key Cryptosystems: Confidentiality



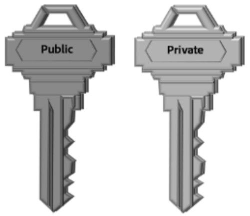


Public-Key Cryptosystems: Confidentiality (Cont.)



Public-Key Cryptosystems: Confidentiality

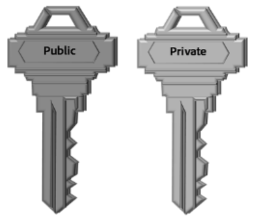
Public-Key Cryptosystems: Confidentiality (Cont.)



Example:

- If **Bob** wishes to send a confidential message to **Alice**, **Bob** encrypts the message using **Alice's public key**.
- When **Alice** receives the message, she decrypts it using her **private key**, where no other recipient can decrypt the message because only **Alice** knows **Alice's private key**.

Public-Key Cryptosystems: Authentication

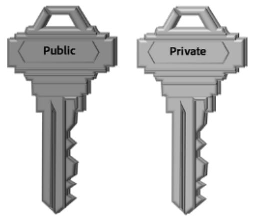


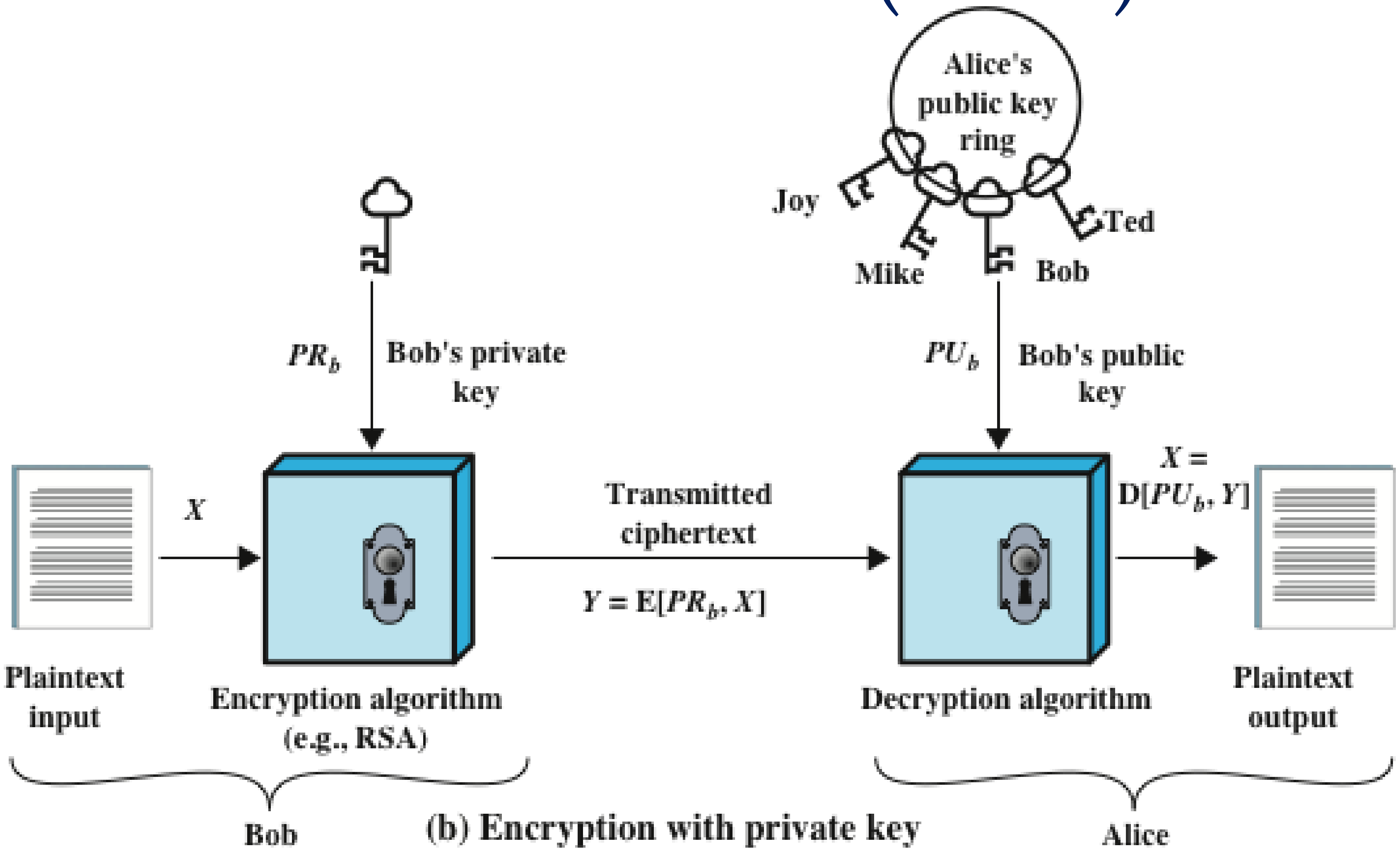
- Since either of the **two related keys** can be used for encryption with the other used for decryption, the public-key encryption can also be used to provide **authentication**.

Example:

- **A** prepares a message to **B** and encrypts it using **A's** private key.
- **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 not anyone else*.
- Hence, entire encrypted message serves as a **digital signature**.

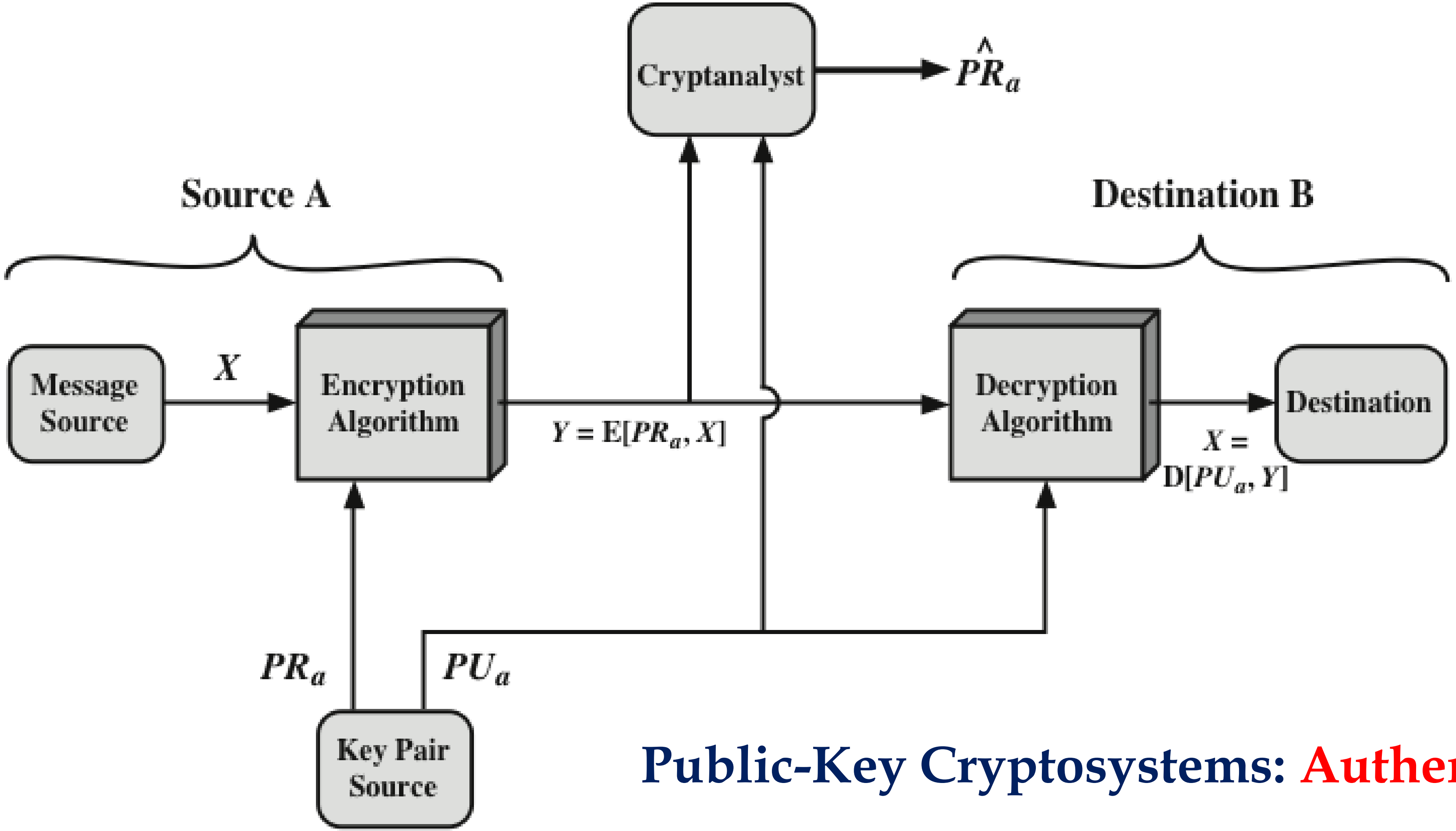
Public-Key Cryptosystems: Authentication (Cont.)





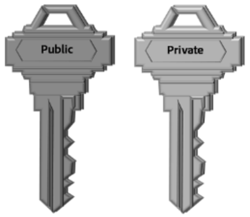
(b) Encryption with private key

Public-Key Cryptosystems: Authentication (Cont.)



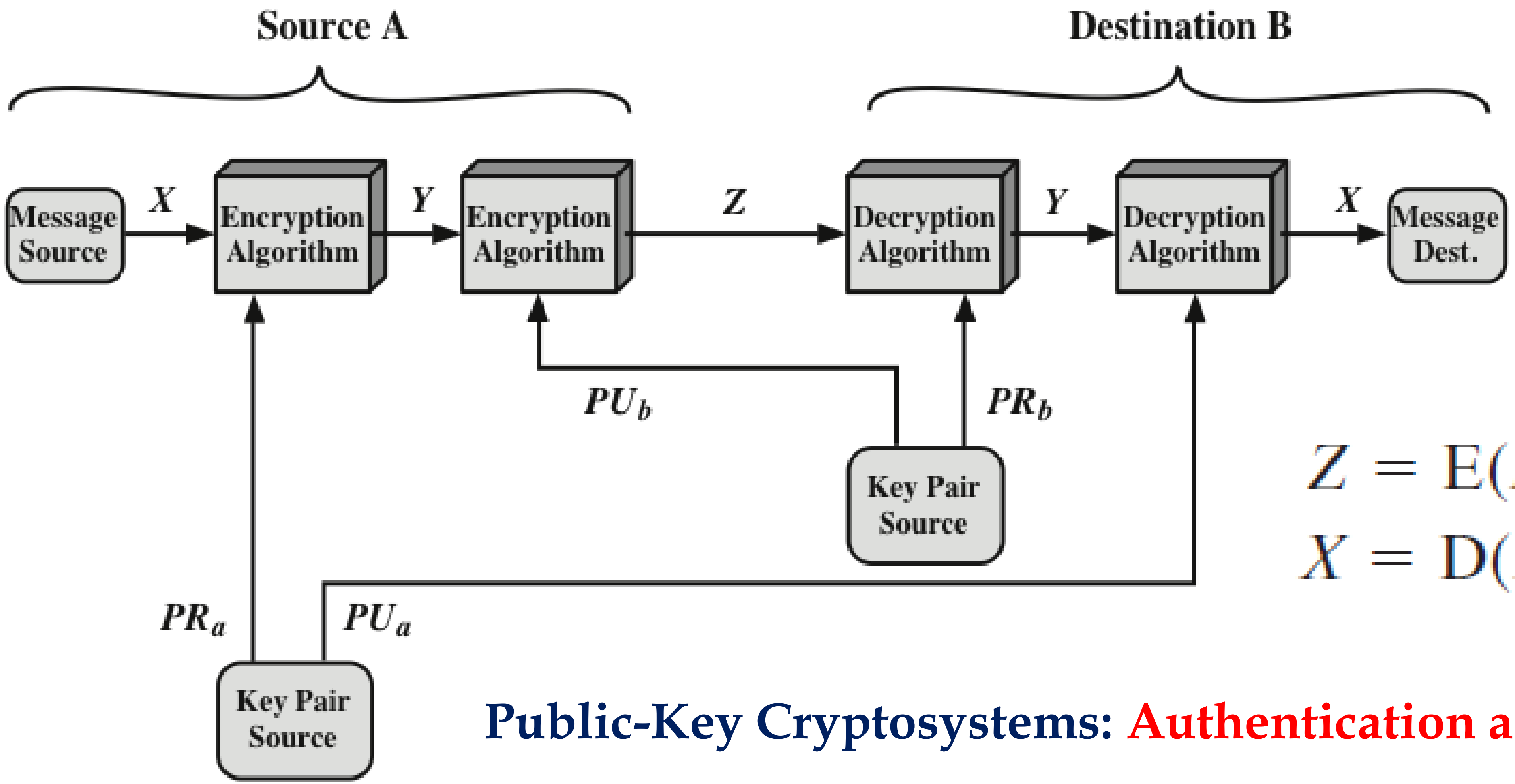
Public-Key Cryptosystems: Authentication

Public-Key Cryptosystems: Confidentiality and Authentication



- The encryption process, using the private key for encryption, does not provide **confidentiality**.
- The message being sent is safe from **authentication** issues but not from **eavesdropping**.
- There is no protection of **confidentiality** because any observer can decrypt the message by using the sender’s public key.
- It is possible to provide both **authentication** and **confidentiality** by a **double use of the public-key scheme**.

Public-Key Cryptosystems: Confidentiality and Authentication (Cont.)



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

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

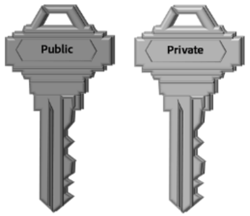
Public-Key Cryptosystems: Authentication and Confidentiality

Public-Key Cryptosystems: Confidentiality and Authentication (Cont.)

Working mechanism:

- We begin by encrypting a message using **sender's private key**. Hence, providing **digital signature**.
- Next, we encrypt again using the **receiver's public key**. Hence, generating the **final ciphertext**. This **final ciphertext** can be decrypted only by the receiver who has the **matching private key**. Thus, **confidentiality** is achieved.
- The receiver decrypts the received **ciphertext** first by its **own private key**. Followed by decrypting the result with the **sender's public key**. By that, the **plaintext** is obtained.

Application of Public-Key Cryptosystems



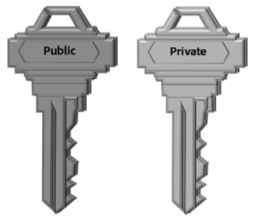
- Depending on the application, sender uses either the **sender's private key** or **receiver's public key**.
- Broadly, we can classify the use of public-key cryptosystems into three categories:

Encryption / Decryption
- **The sender encrypts a message with the recipient's public key**

Digital signature
- **The sender "signs" a message with its private key**

Key exchange
- **Two sides cooperate to exchange a session key, which is a secret key for symmetric encryption**

Application of Public-Key Cryptosystems (Cont.)



- Some algorithms are suitable for all three applications, whereas others can be used only for one or two of these applications.
- Table below indicates the applications supported by the algorithms.

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

Thank You!