

# Chương 5: Network Security

## Cryptography

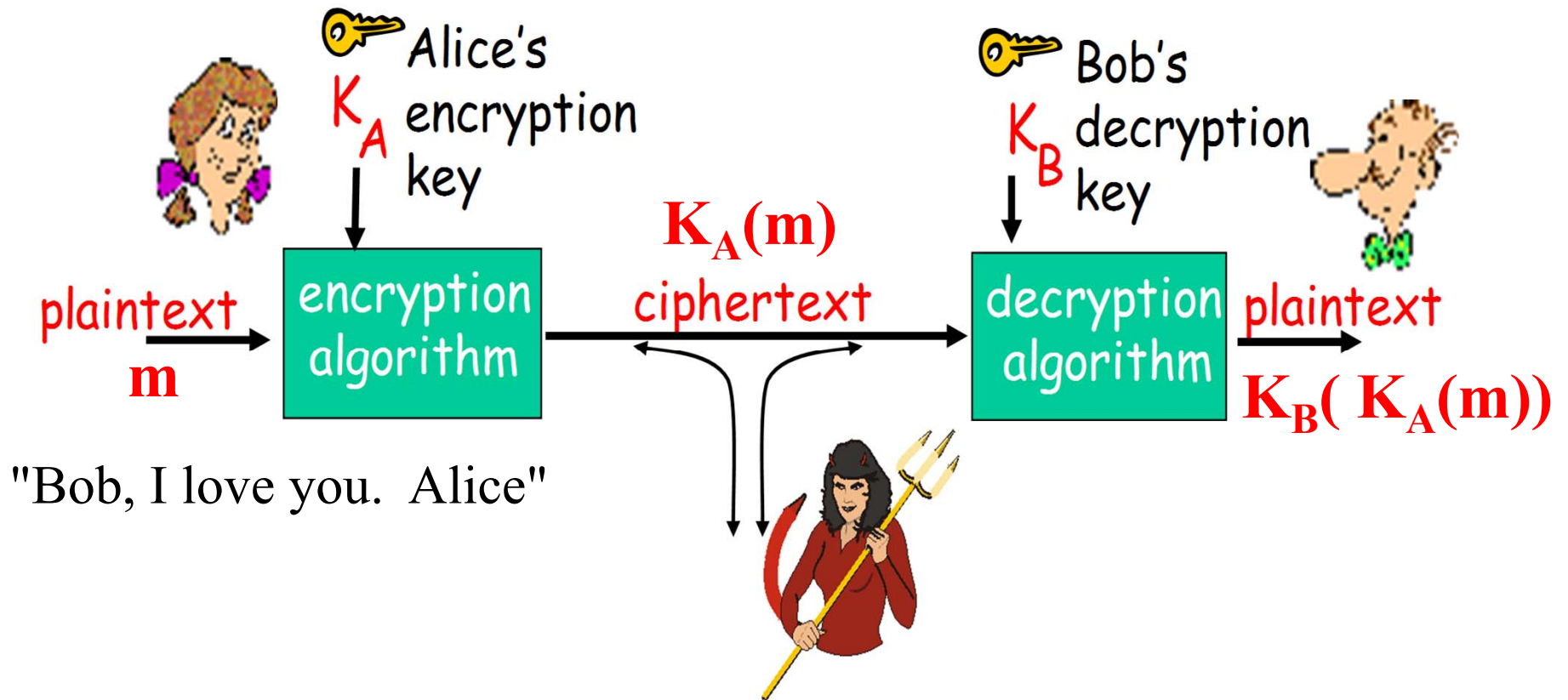
**TS. Trần Quang Vinh**  
**BM. Kỹ thuật Thông tin**  
**Viện Điện tử - Viễn thông**  
**Đại học Bách Khoa Hà Nội**  
**vinhtq@mail.hut.edu.vn**



# Mã hóa (Cryptography)

## ❑ KHÁI NIỆM

$$K_B(K_A(m)) = m$$

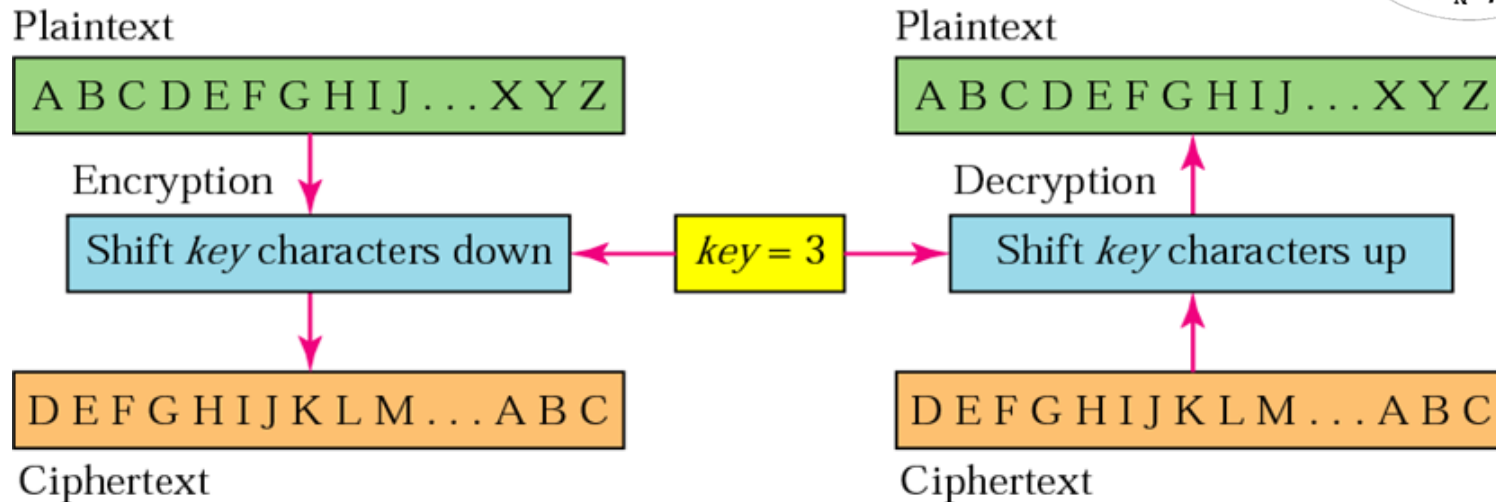


# Mật mã Xê-da (Caesar cipher)



## ❑ Classic Caesar Cipher

- Thay thế các chữ cái trong bảng chữ cái bằng cách dịch đi k chữ cái



E.g.: Plaintext: bob. i love you. alice  
ciphertext: nkn. s gktc wky. mgsbc

Key: the mapping from the set of 26 letters to the set of 26 letters

# Mật mã Xê-da (Caesar cipher)

## ❑ Mono-Alphabetic Cipher

- Thay thế các chữ cái trong bảng chữ cái bằng một chữ cái bất kỳ sao cho mỗi chữ cái đó có một chữ cái thay thế duy nhất và ngược lại
- KEY: 26! ( $\sim 10^{26}$ )
- Thám mã: phân tích tần xuất  $\leftarrow$  dễ dàng

### CIPHER ALPHABET

A = B	H = A	O = O	V = L
B = V	I = D	P = Y	W = P
C = G	J = Z	Q = F	X = U
D = Q	K = C	R = J	Y = I
E = K	L = W	S = X	Z = R
F = M	M = S	T = H	
G = N	N = E	U = T	

Figure 1

# Mật mã Xê-da (Caesar cipher)

## ❑ Polyalphabetic encryption (Blaise de Vigenere, 500 năm trước)

- Mục tiêu: chống thám mã bằng phân tích tần xuất
- Cách thức: Sử dụng nhiều mã Monoalphabetic
- Kết quả: Trong cùng một thông điệp, mỗi chữ cái ở một vị trí khác nhau có thể được mã hóa một cách khác nhau

### Vigenère cipher

<http://sharkysoft.com/vigenere/>

A polyalphabetic cipher

	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
A	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
B	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	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	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

Key = KING

The sun and the man in the moon  
Dpr yev ntn buk wia ox buk wwbt

4 possible ways to spell the word "the"

K - DPR

I - BUK

N - GNO

G - ZRM

K	I	N	G	K	I	N	G	K	I	N	G	K	I	N	G
T	H	E	S	U	N	A	N	D	T	H	E	M	A	N	I
D	P	R	Y	E	V	N	T	N	B	U	K	W	I	A	O
X	B	U	K	W	W	B	T								

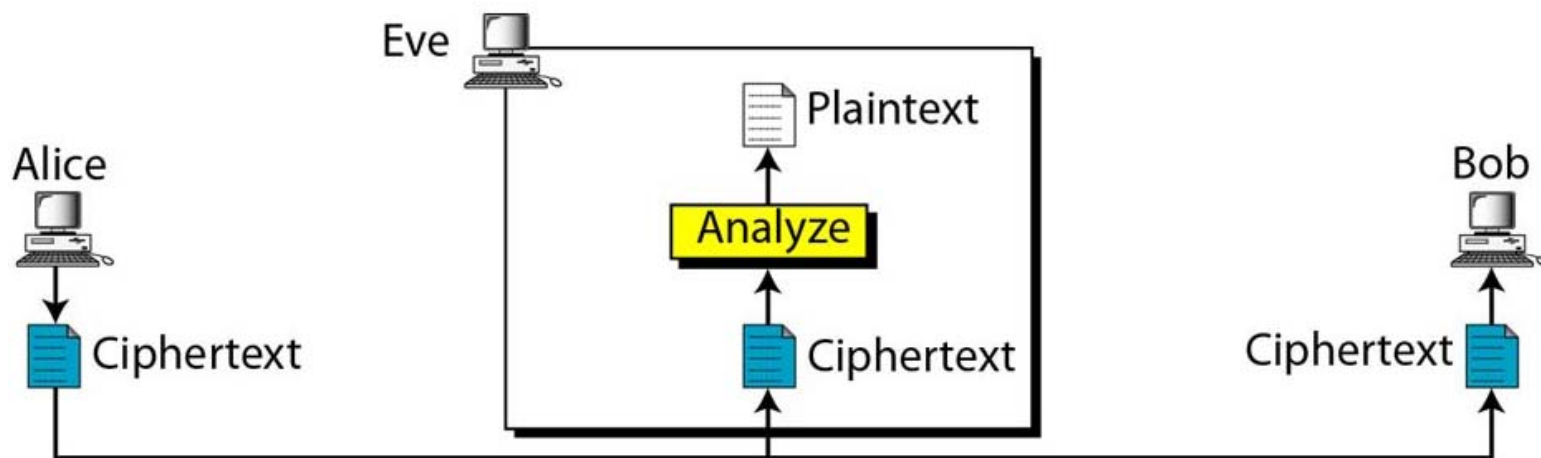
[http://www.simonsingh.net/The\\_Black\\_Chamber/vigenere\\_cracking\\_tool.html](http://www.simonsingh.net/The_Black_Chamber/vigenere_cracking_tool.html)

# Các phương pháp thám mã

## ❑ Tấn công chỉ biết bản mã

- Ciphertext + algorithm → Key and the plaintext
- Brute-Force attack
- Statistical attack
- Pattern attack

### *Ciphertext-Only Attack*

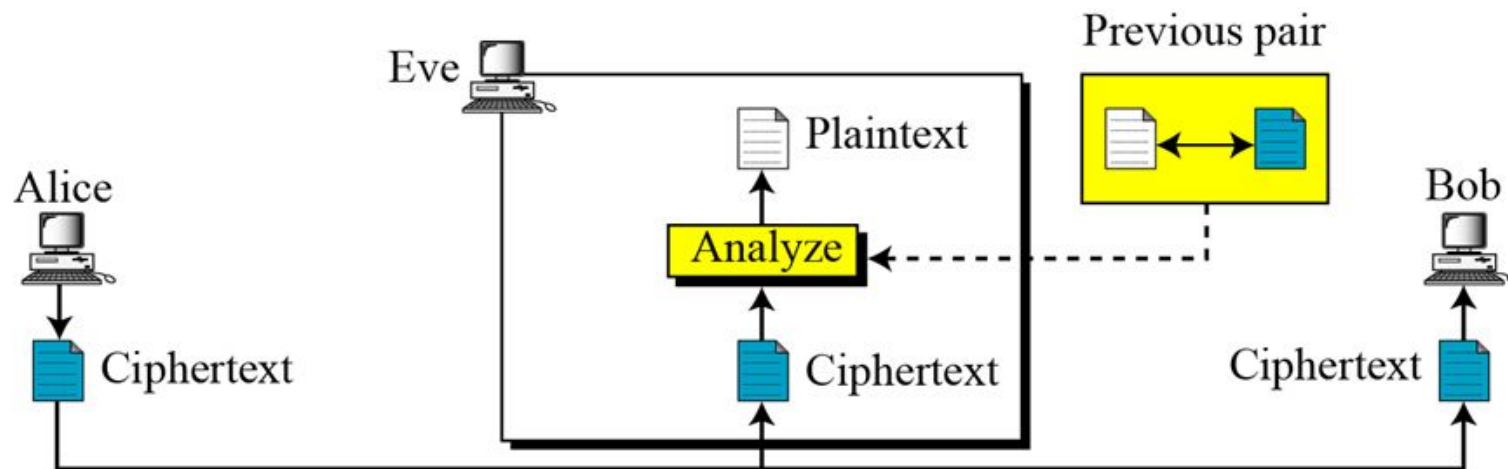


# Các phương pháp thám mã

## ❑ Tấn công nhận biết bản rõ

- Eve has access to some plaintext/ciphertext pairs in addition to the intercepted ciphertext
- Eve uses the relationship b/w the previous pair to analyze the current ciphertext

### *Known-Plaintext Attack*

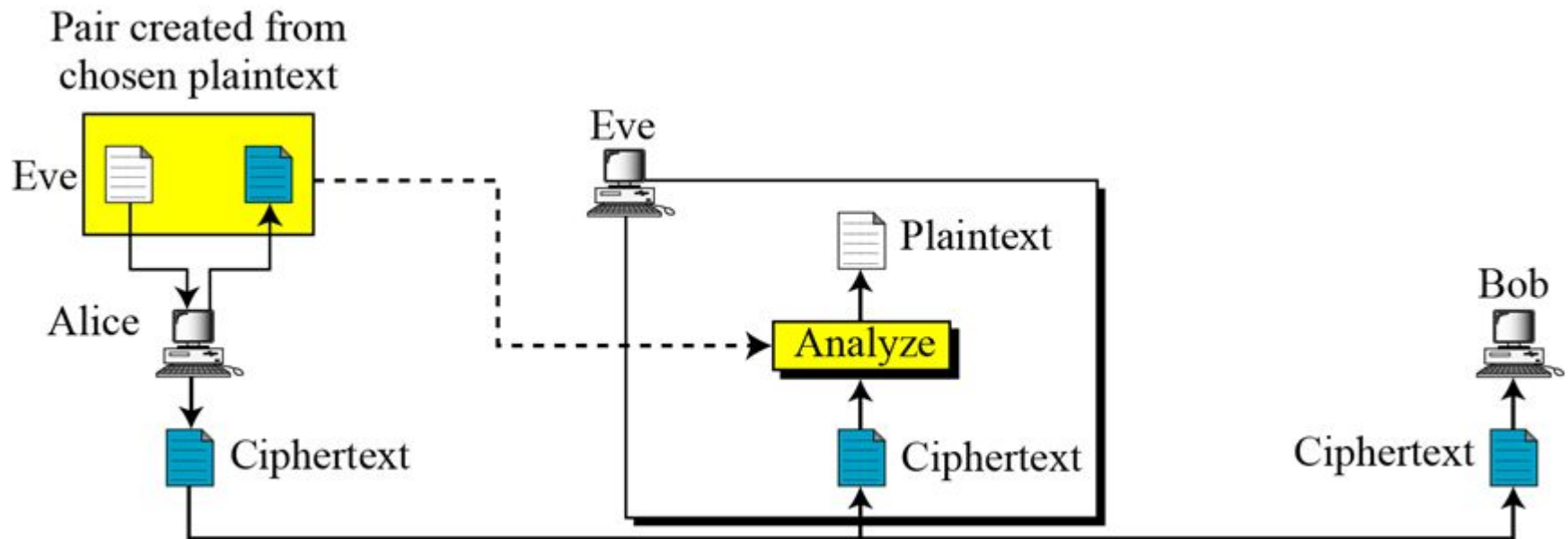


# Các phương pháp thám mã

## ❑ Tấn công lựa chọn bản rõ

- plaintext/ciphertext pairs have been chosen by attacker

### *Chosen-Plaintext Attack*





# REVIEW

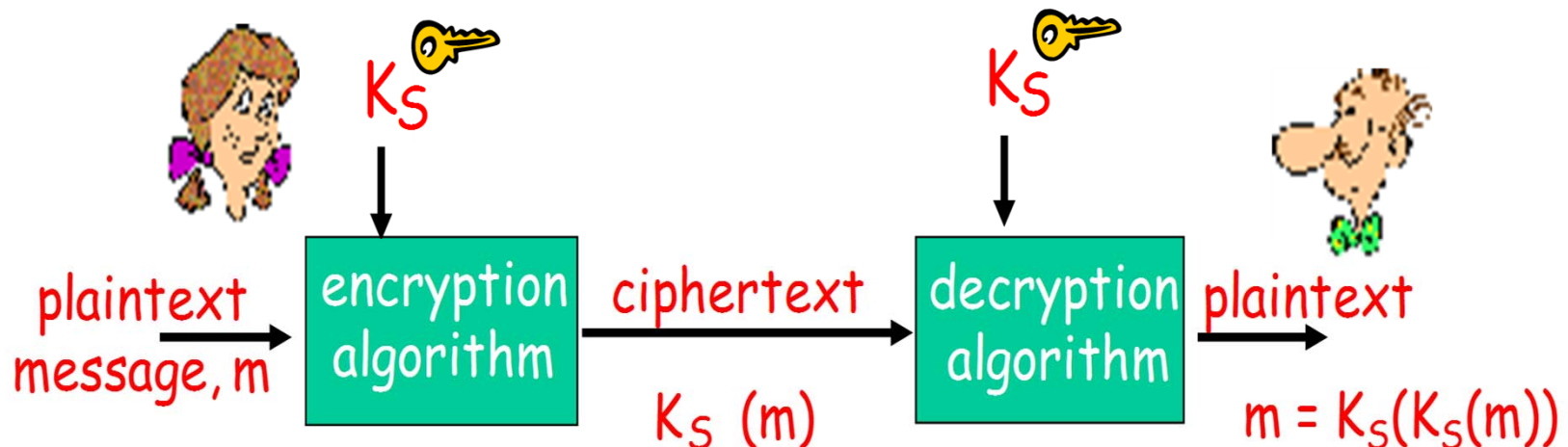
---

- **Plaintext** - original message
- **Ciphertext** - coded message
- **Cipher** - algorithm for transforming plaintext to ciphertext
- **Key** - info used in cipher known only to sender/receiver
- **Encipher (encrypt)** - converting plaintext to ciphertext
- **Decipher (decrypt)** - recovering ciphertext from plaintext
- **Cryptography** - study of encryption principles/methods
- **Cryptanalysis (code breaking)** - study of principles/methods of deciphering ciphertext without knowing key
- **Cryptology** - field of both cryptography and cryptanalysis

# Mã hóa khóa đối xứng

## ❑ THÀNH PHẦN

- Plaintext: original message or data fed into the algorithm as input
- Encryption algorithm: performs substitutions and transformations
- Secret key: exact substitutions and transformations performed by the algorithm depend on the key
- Ciphertext: scrambled message produced as output
- Decryption algorithm: ciphertext + secret key  $\rightarrow$  original plaintext



# Mã hóa khóa đối xứng

---

## ❑ ĐẶC ĐIỂM

- Yêu cầu:
  - Thuật toán mã hóa mạnh
  - Duy trì độ mật của khóa
- Độ mật của mã hóa đối xứng phụ thuộc vào độ mật của khóa, không phụ thuộc vào độ mật của thuật toán

## ❑ PHÂN LOẠI

- Mã hóa khối (block ciphers)
  - PGP (Pretty Good Privacy): bảo mật e-mail
  - SSL (Secure Sockets Layer): bảo mật kết nối TCP, và
  - IPsec để bảo mật lớp giao vận
- Mã hóa dòng (stream ciphers – mã hóa từng byte)

# Mã hóa khóa đối xứng

---

## ❑ **HỆ MÃ AN TOÁN**

- Chi phí phá vỡ hệ mã vượt quá giá trị của thông tin được mã hóa.
- Thời gian cần thiết để phá vỡ hệ mã vượt quá tuổi thọ hữu ích của thông tin

## ❑ **Brute Force attack**

- Thử tất cả các khả năng cho đến khi thành công
- Cần có sự hiểu biết về bản rõ mong muốn (nhận dạng bản rõ)



# Block Ciphers

## ❑ NGUYÊN TẮC

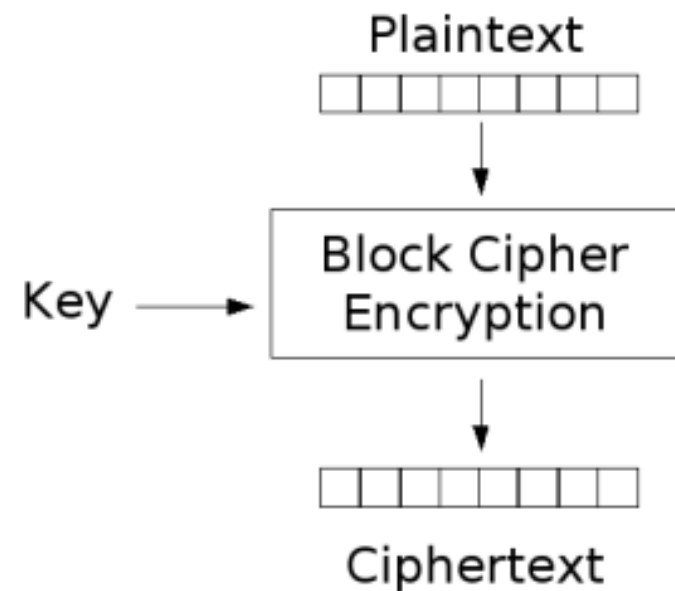
- Bản tin cần mã hóa được chia thành các khối k bit
- Mỗi khối được lập mã một cách độc lập

## ❑ Ví dụ:

<u>input</u>	<u>output</u>	<u>input</u>	<u>output</u>
000	110	100	011
001	111	101	010
010	101	110	000
011	100	111	001

**Bản rõ: 010 110 001 111**

**Bản mã: 101 000 111 001**



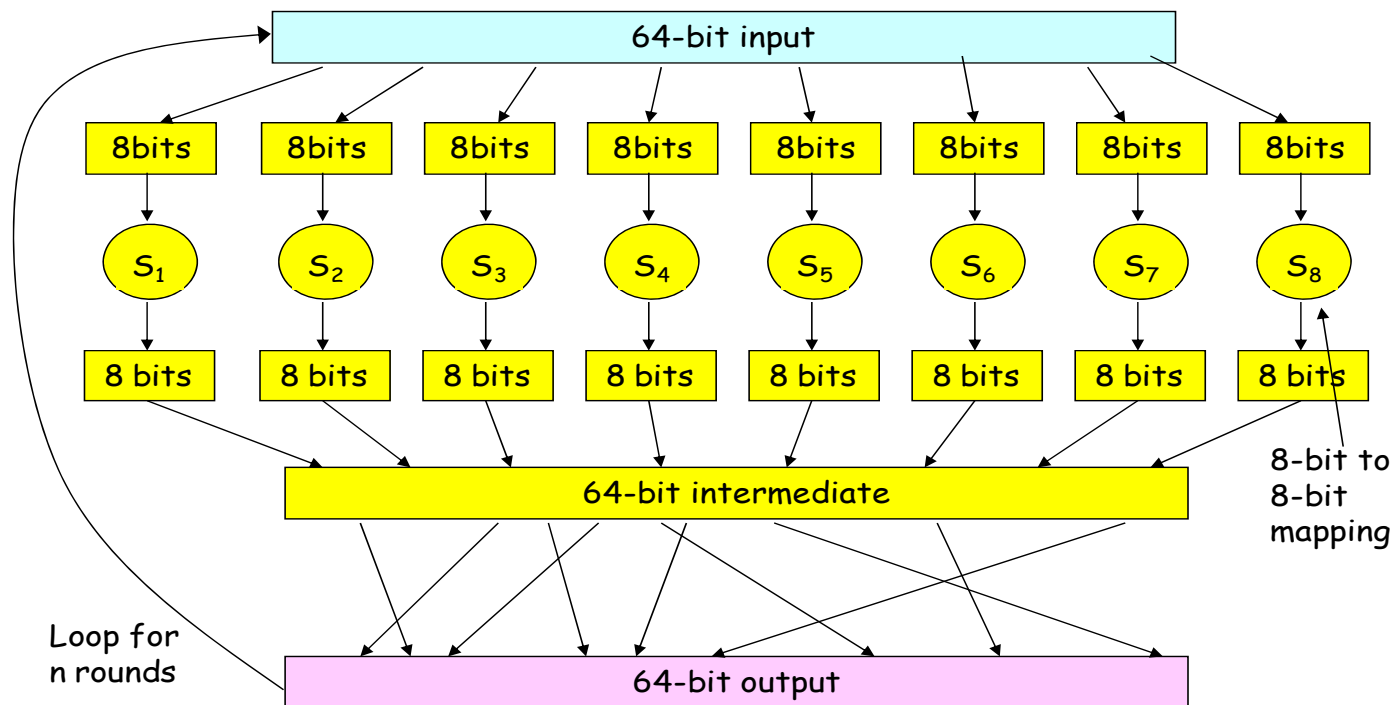
## Nhận xét:

- $2^3=8$  giá trị đầu vào có thể, nên ta sẽ có  $8!=40,320$  cách hoán vị
- Cần duy trì bảng ánh xạ lớn:  $2^k$ ,  $k=64$  là  $2^{64}$  giá trị với mỗi giá trị là một số 64 bit, key size:  $2^{64}$ !

# Block Ciphers

## ❑ KEY SIZE

- Thay vì sử dụng bảng ánh xạ đầy đủ, block cipher thường sử dụng các hàm đặc biệt để tạo ra các bảng hoán vị ngẫu nhiên
- Ứng dụng: DES (khối 64 bit với khóa 56 bit) và AES (khối 128 bit với khóa 128, 192, hoặc 256 bit)



# Symmetric Block encryption algorithms

---

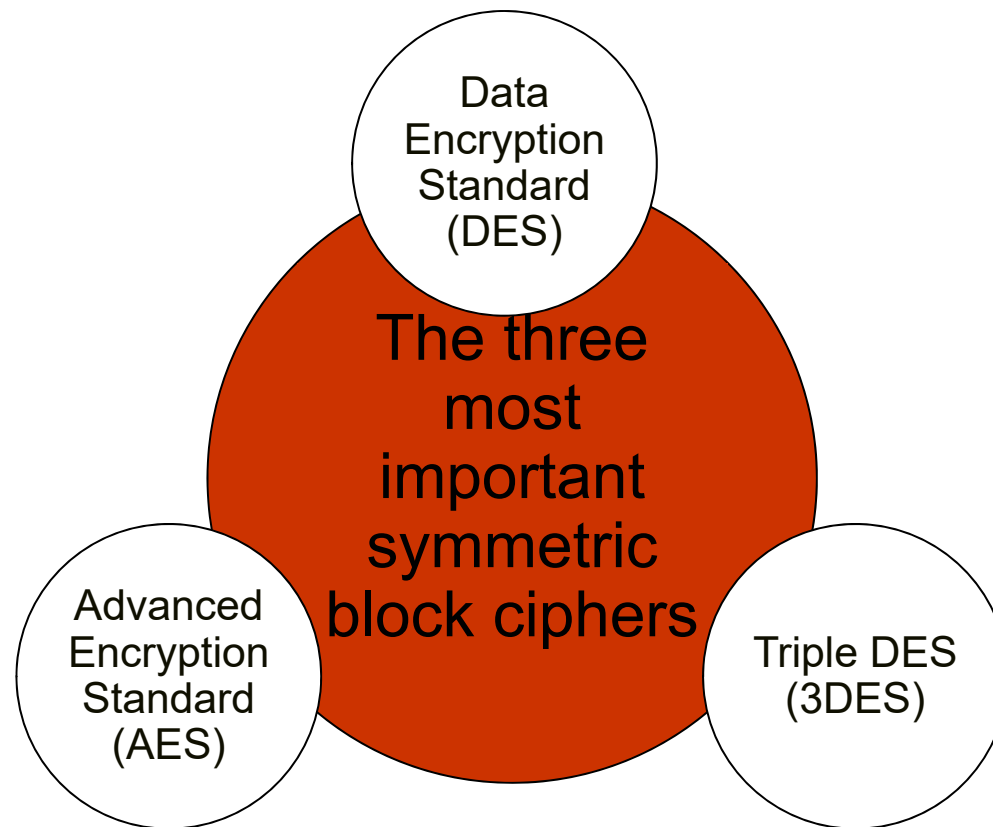
## Cipher block modes

ECB

**CBC**

CFB

CTR



# Block Ciphers

## ❑ THUẬT TOÁN

- Tạo các số ngẫu nhiên 64 bit  $r(i)$  cho mỗi khối bản rõ  $m(i)$
- Tính  $c(i) = K_S(m(i) \oplus r(i))$  //  $\oplus$ : phép XOR,  $K_S$  là khóa
- Truyền đi  $c(i)$  và  $r(i)$  với  $i=1,2,\dots$
- Tại phía thu:  $m(i) = K_S(c(i)) \oplus r(i)$

## ❑ Ví dụ: Xét mã khối 3 bit với khóa như bảng sau.

- Bản rõ: 010 010 010

-  $r(1)=001, r(2)=111, r(3)=100$

$m(i)=$  010 010 010

$r(i)=$  001 111 100

$m(i) \oplus r(i)=$  011 101 110

$c(i)=K_S(m(i) \oplus r(i))=$  100 010 000

<u>input</u>	<u>output</u>
000	110
001	111
010	101
011	100

<u>input</u>	<u>output</u>
100	011
101	010
110	000
111	001



# Block Ciphers

## ❑ MÃ HÓA

- Tạo chuỗi ngẫu nhiên k bit (Initialization Vector - IV) ký hiệu là  $r(0)$ .  
Gửi đi  $r(0)$  ở dạng bản rõ
- Với khối bản rõ đầu tiên  $m(1)$ , tính  $m(1) \oplus c(0)$  và mã hóa với khóa  $K_s$

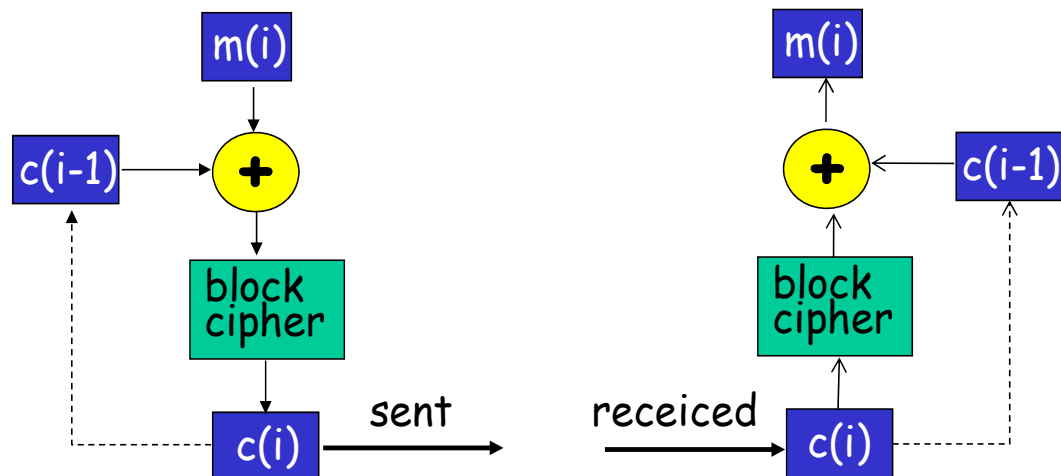
$$c(1) = K_s(m(1) \oplus c(0))$$

Gửi đi khối mã  $c(1)$

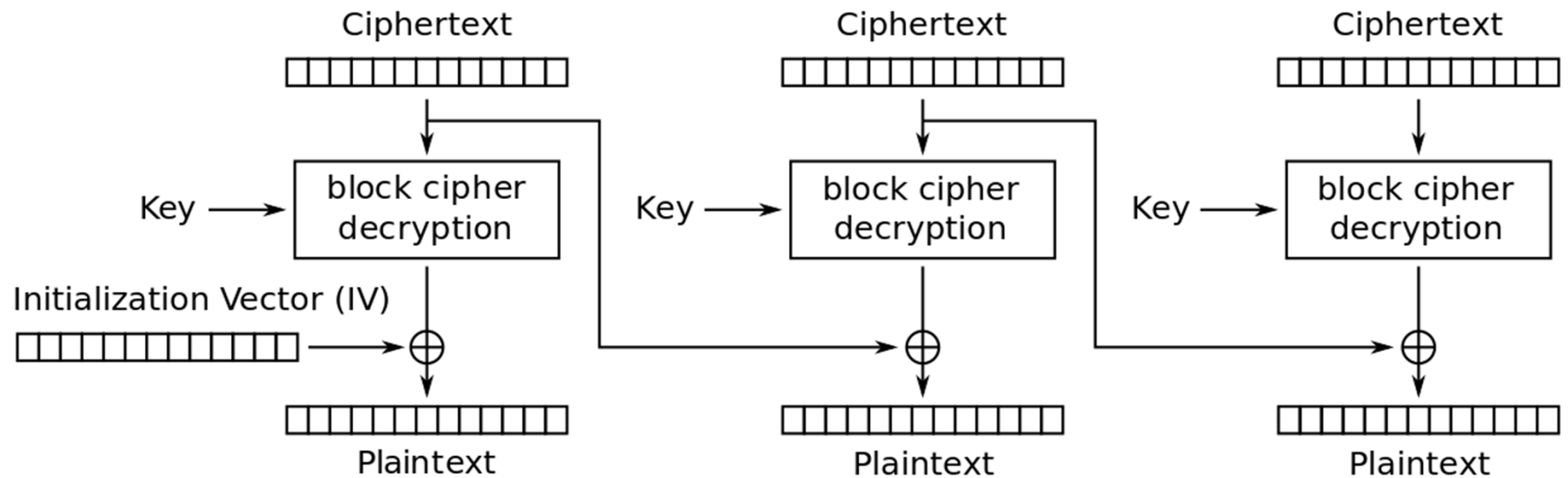
- Với khối thứ  $i$ , tạo khối mã tương ứng:  $c(i) = K_s(m(i) \oplus c(i-1))$

## ❑ GIẢI MÃ

- $m(1) = K_s(c(1)) \oplus c(0)$
- $m(i) = K_s(c(i)) \oplus c(i-1)$



# Cipher Block Chaining (CBC)

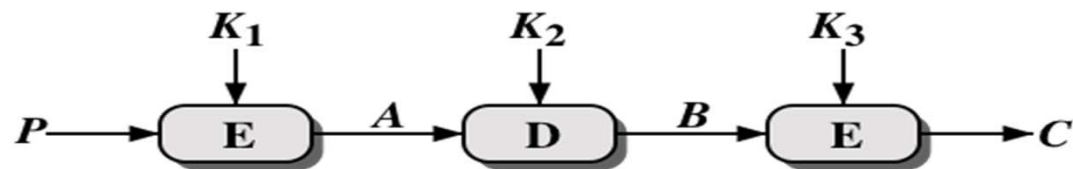


Cipher Block Chaining (CBC) mode decryption

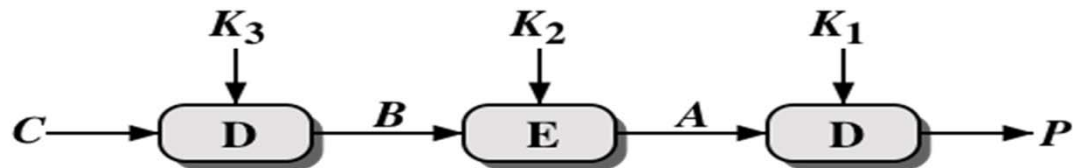
# Symmetric Block encryption algorithms

## ❑ DES: Data Encryption Standard

- Tiêu chuẩn mã hóa của Mỹ [NIST 1977]
  - Khóa đối xứng 56-bit
  - Sử dụng mã hóa khối (block 64 bit) với kỹ thuật CBC, Số lần trộn: 16
- Tính bảo mật của DES:
  - Phá mã DES 56 bit trong thời gian ít hơn 1 ngày
  - Tăng tính bảo mật của DES bằng cách mã hóa 3 lần với 3 khóa khác nhau (Triple-DES) (actually encrypt, decrypt, encrypt)



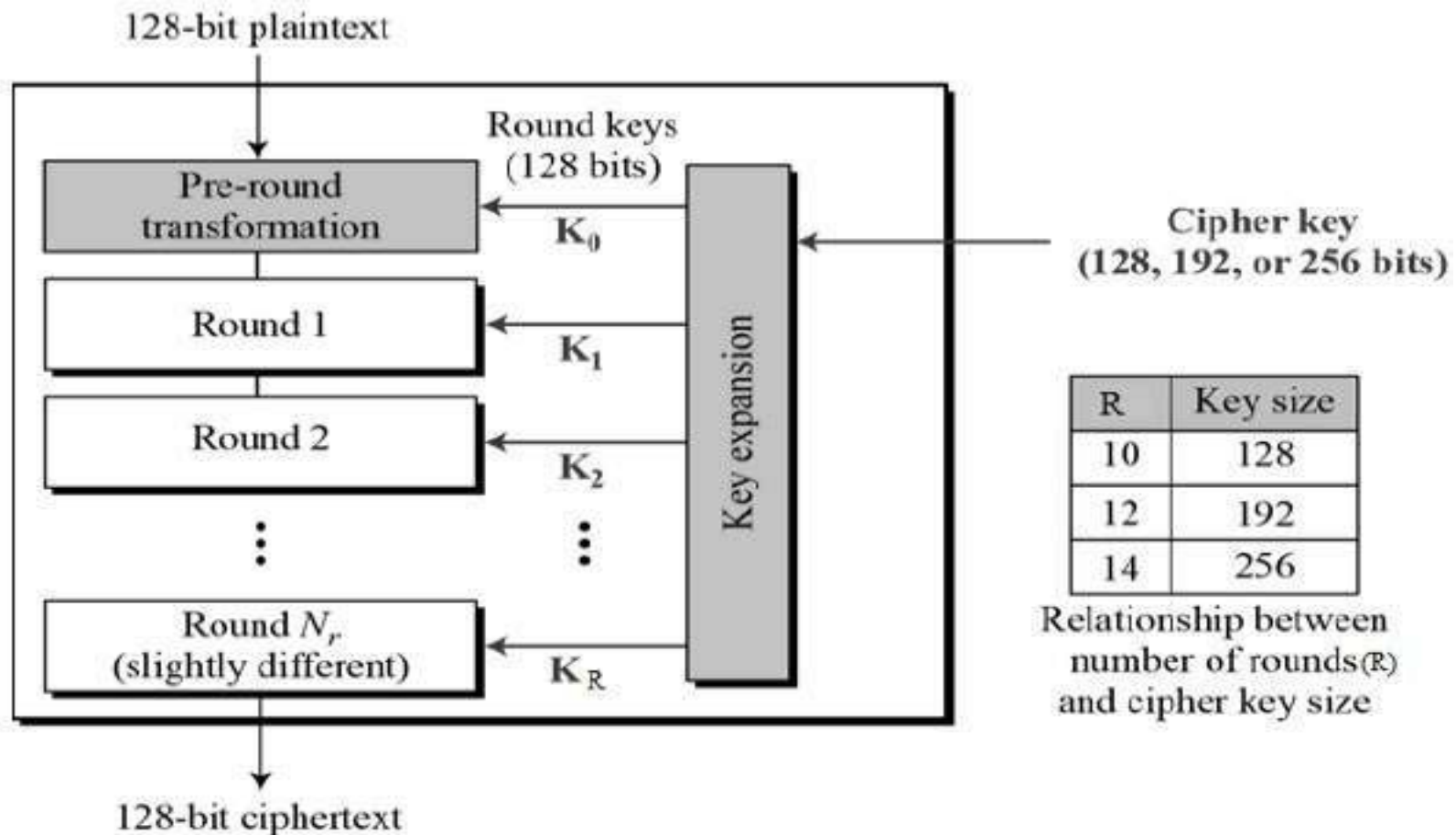
(a) Encryption



(b) Decryption

# AES: Advanced Encryption Standard

- ❑ **Chuẩn khóa đối xứng mới, được đề xuất 1997 thay thế cho DES**
- ❑ **Mã hóa khối 128 bit; Khóa: 128, 192, hoặc 256 bit**
  - Brute force decryption taking 1 sec on DES, takes 149 trillion years for AES

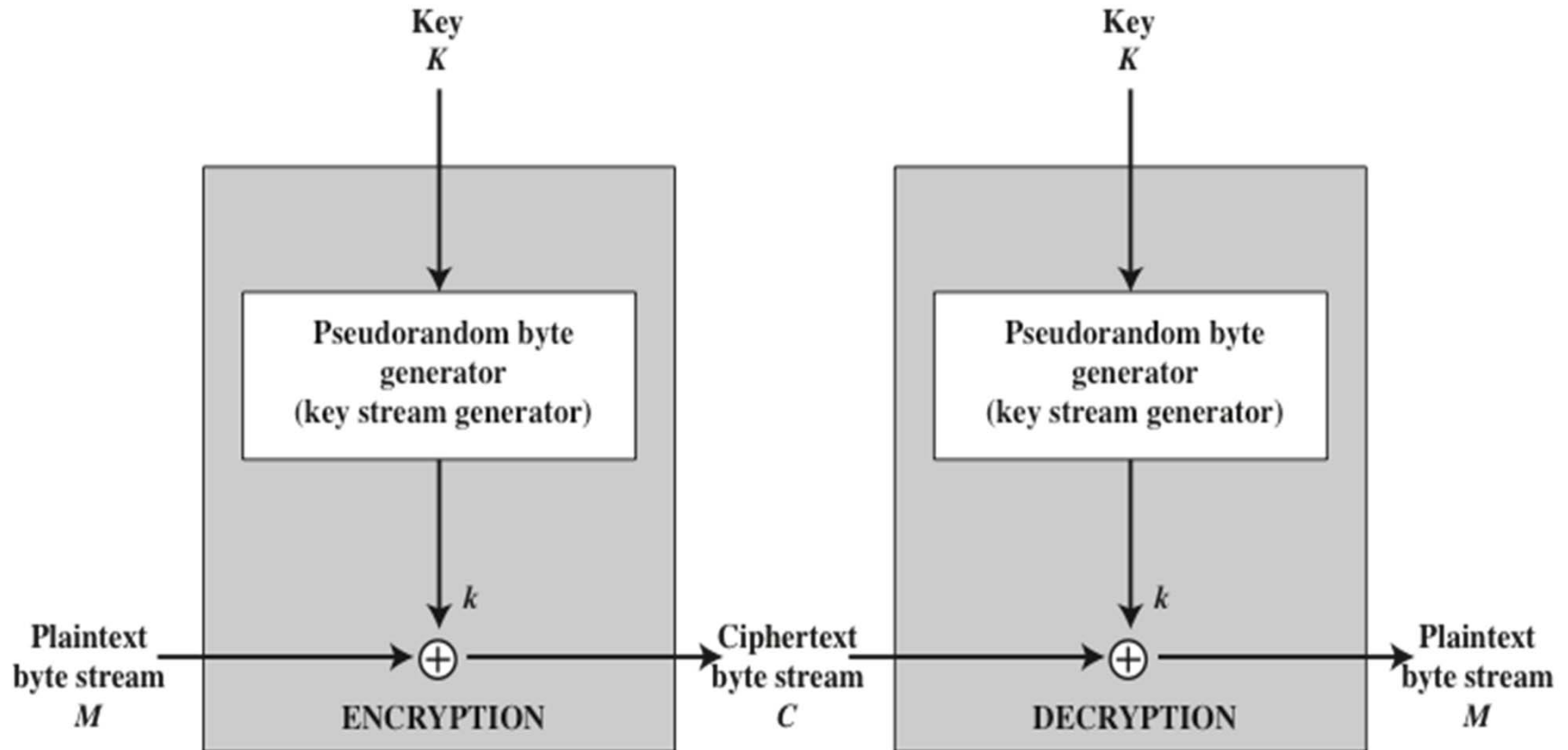


# Average Time Required for Exhaustive Key Search

---

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at $10^9$ decryptions/s	Time Required at $10^{13}$ decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	$2^{55}$ ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	$2^{127}$ ns = $5.3 \times 10^{21}$ years	$5.3 \times 10^{17}$ years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	$2^{167}$ ns = $5.8 \times 10^{33}$ years	$5.8 \times 10^{29}$ years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	$2^{191}$ ns = $9.8 \times 10^{40}$ years	$9.8 \times 10^{36}$ years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	$2^{255}$ ns = $1.8 \times 10^{60}$ years	$1.8 \times 10^{56}$ years

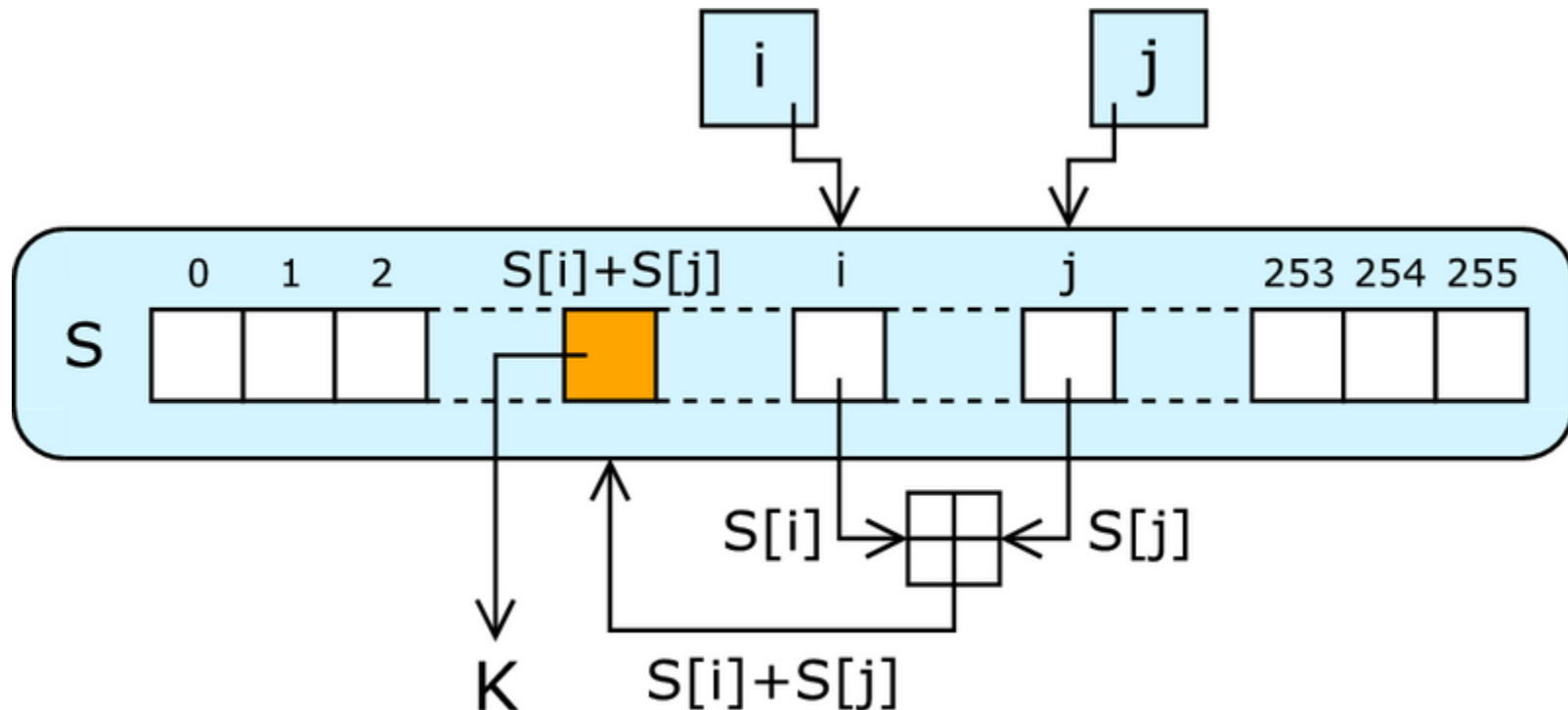
# Stream Ciphers



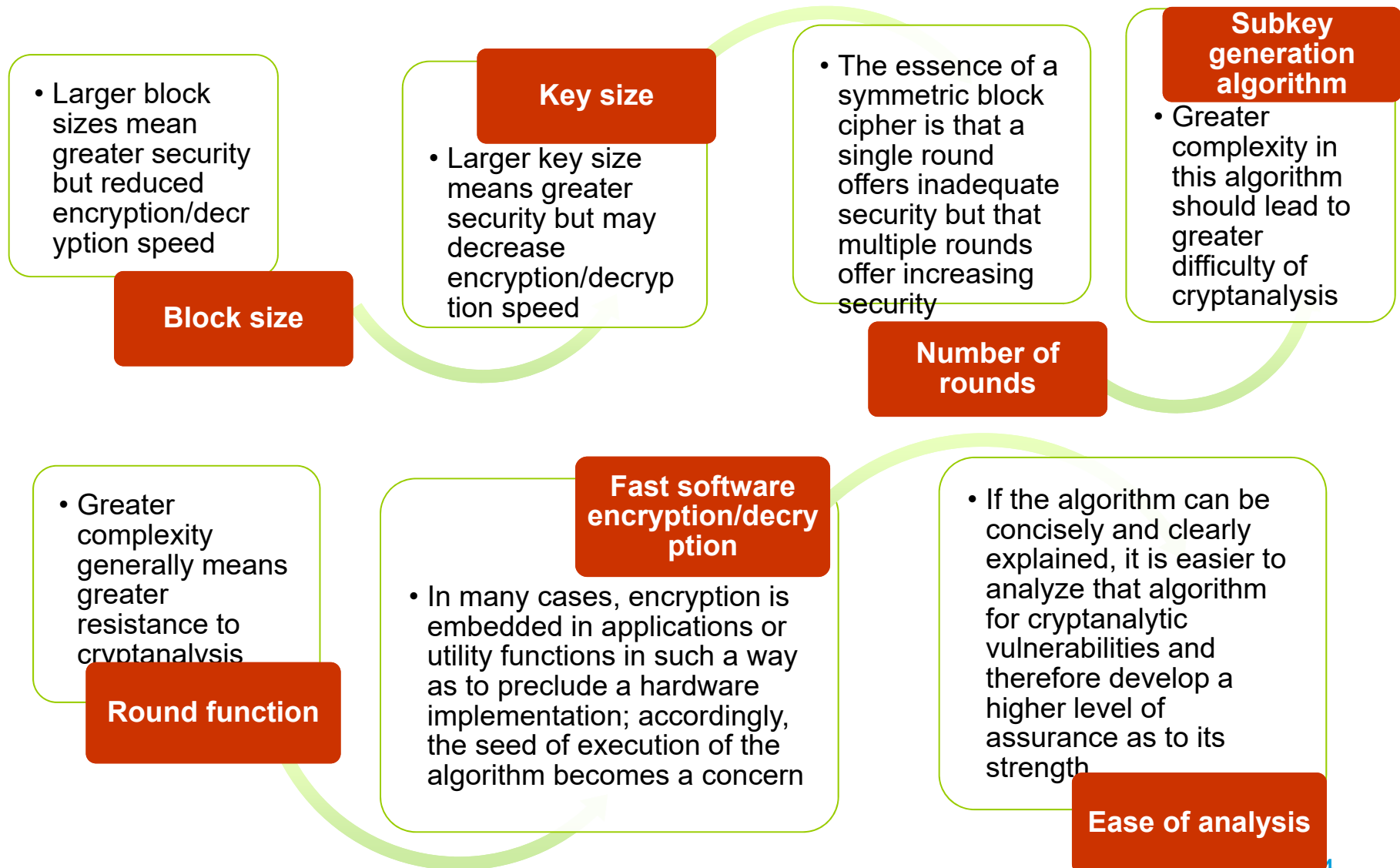
# RC4 algorithm

## ❑ Mã luồng (stream cipher) designed in 1987 by Ron Rivest

- Kích thước khóa thay đổi
- Thuật toán mã hóa dựa trên sự hoán vị ngẫu nhiên (random permutation)
- Được sử dụng trong Secure Sockets Layer/Transport Layer Security (SSL/TLS)
- Also used in Wired Equivalent Privacy (WEP) protocol and the newer WiFi Protected Access (WPA) protocol that are part of the IEEE 802.11 wireless LAN standard



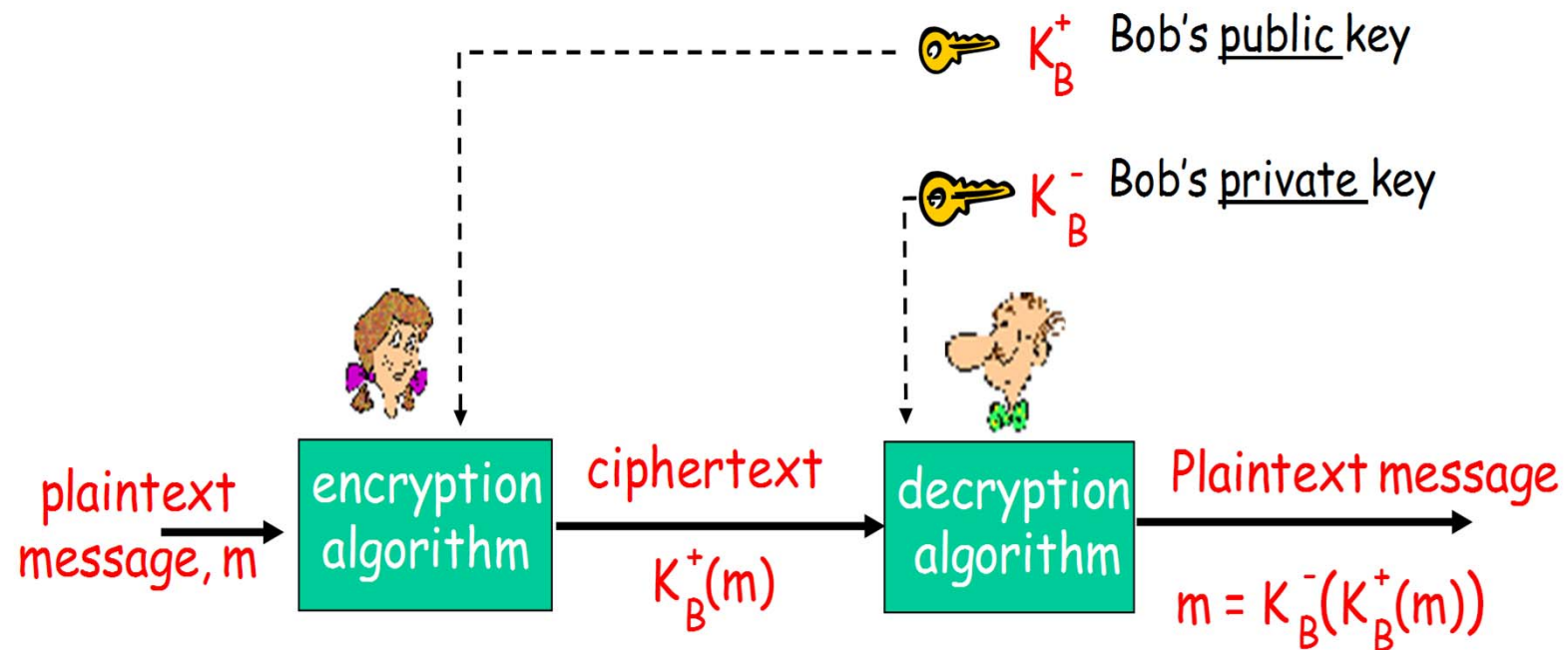
# Feistel Cipher Design Elements





# Public Key Cryptography

## ❑ NGUYÊN TẮC



$$K_B^-(K_B^+(m)) = K_B^+(K_B^-(m)) = m$$

# Giải thuật RSA

---

## ❑ TẠO KHÓA

- (1) Chọn 2 số nguyên tố lớn  $p$  và  $q$  với  $p \neq q$ , ngẫu nhiên và độc lập
- (2) Tính:  $n = pq$
- (3) Tính: giá trị  $z = (p-1)(q-1)$ .
- (4) Chọn một số nguyên tố  $e$ , với  $e < n$ , không có thừa số chung với  $z$ .
- (5) Tính:  $d$  sao cho  $ed \bmod z = 1$  (hay  $ed-1$  chia hết cho  $z$ ).

Khóa công khai  $(n,e)$  hay  $K^+B$ :

- $n$ , tích của hai số nguyên tố ban đầu
- $e$ , số mũ công khai (cũng gọi là số mũ mã hóa)

Khóa bí mật  $(n,d)$  hay  $K^-B$ :

- $n$ , xuất hiện cả trong khóa công khai và khóa bí mật
- $d$ , số mũ bí mật (cũng gọi là số mũ giải mã)

# Giải thuật RSA

## ❑ MÃ HÓA

$$c = m^e \bmod n$$

## ❑ GIẢI MÃ

$$m = c^d \bmod n$$

Ví dụ:

Bob chọn  $p=5$ ,  $q=7$ . Do đó  $n=35$ ,  $z=24$ .

Chọn  $e=5$ , tính được  $d=29$  (với  $ed-1=144$  chia hết cho  $z=24$ ).

Bản rõ:  $M = 0000\ 1100 \rightarrow 12 = m$

Encrypting 8-bit messages.

encrypt:	<u>bit pattern</u>	<u>m</u>	<u>m<sup>e</sup></u>	<u>c = m<sup>e</sup> mod n</u>
	00001100	12	248832	17

decrypt:	<u>c</u>	<u>c<sup>d</sup></u>	<u>m = c<sup>d</sup> mod n</u>
	17	481968572106750915091411825223071697	12

# Giải thuật RSA

---

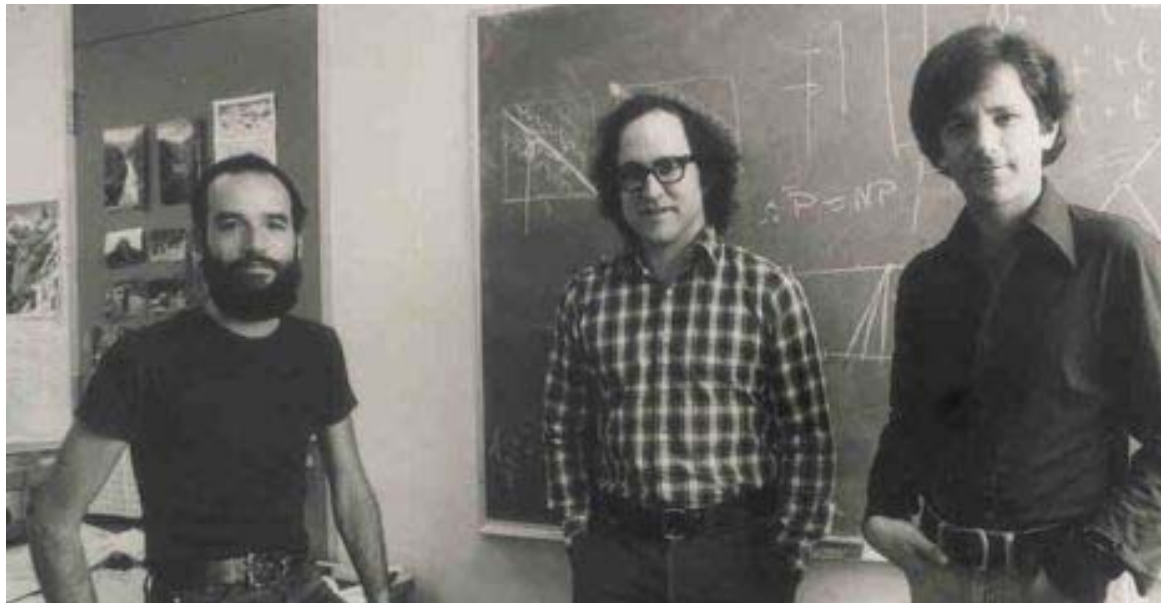
## ❑ CHUYỂN ĐỔI BẢN RÕ

- Chuyển đổi từ  $M$  sang  $m$  sao cho không có giá trị nào của  $M$  tạo ra văn bản mã không an toàn
- Nếu  $m = 0$  hoặc  $m = 1$  sẽ tạo ra các bản mã có giá trị là 0 và 1 tương ứng
- Khi mã hóa với số mũ nhỏ (chẳng hạn  $e = 3$ ) và  $m$  cũng có giá trị nhỏ, giá trị  $c$  cũng nhận giá trị nhỏ (so với  $n$ ). Như vậy phép môđun không có tác dụng và có thể dễ dàng tìm được  $m$  bằng cách khai căn bậc  $e$  của  $c$  (bỏ qua môđun)

# Giải thuật RSA

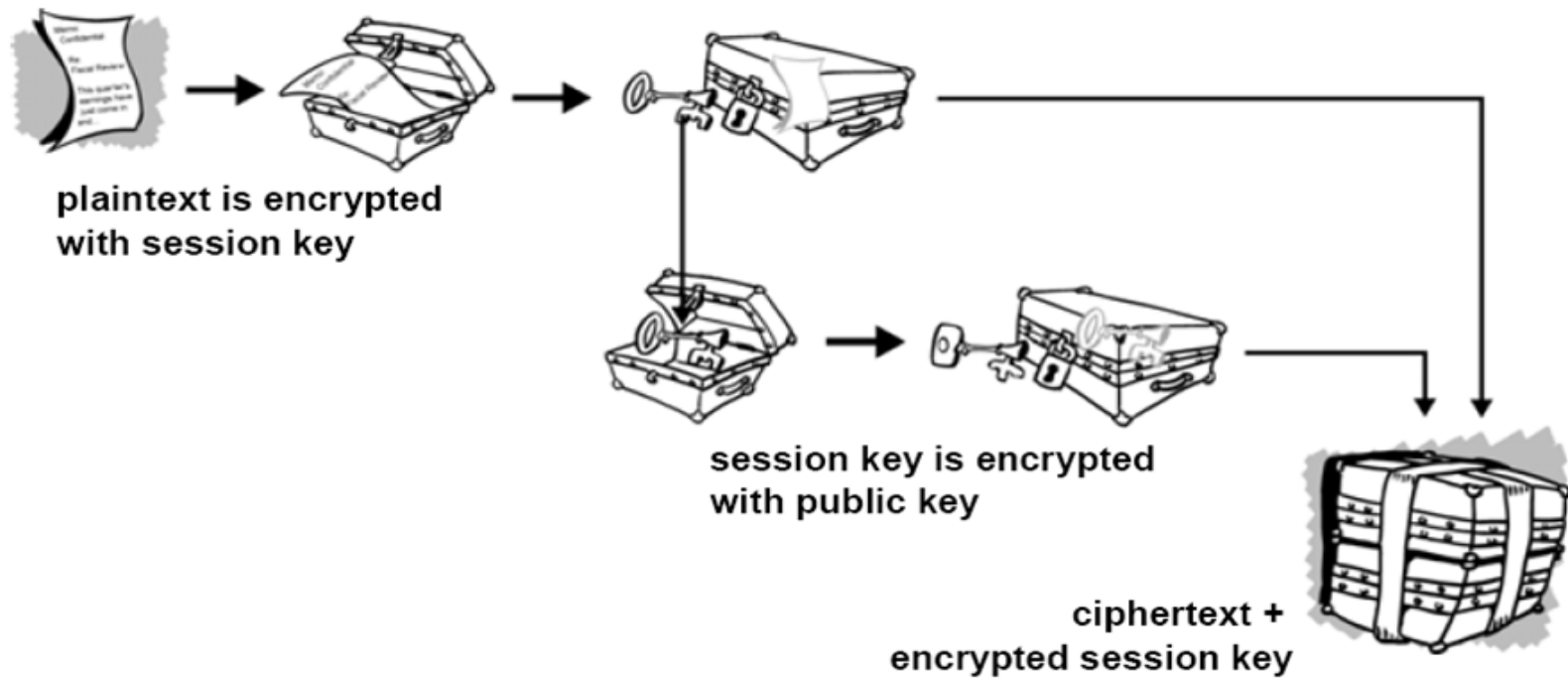
## ❑ ĐỘ BẢO MẬT CỦA RSA

- Độ an toàn của hệ thống RSA dựa trên 2 vấn đề của toán học:
- bài toán phân tích ra thừa số nguyên tố các số nguyên lớn và
- bài toán RSA (là bài toán tính căn bậc  $e$  môđun  $n$ , với  $n$  là hợp số)



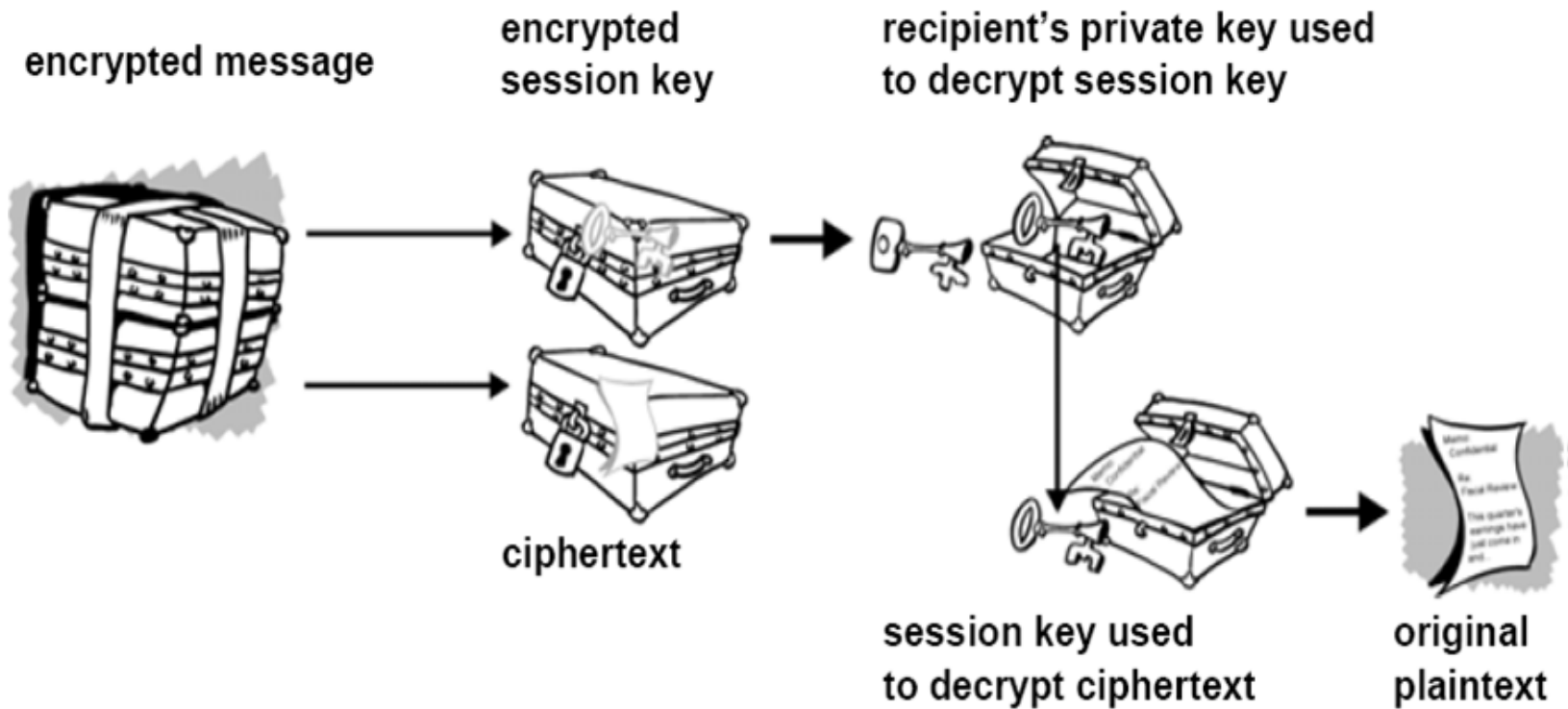
# Session Keys

## ❑ LẬP MÃ



# Session Keys

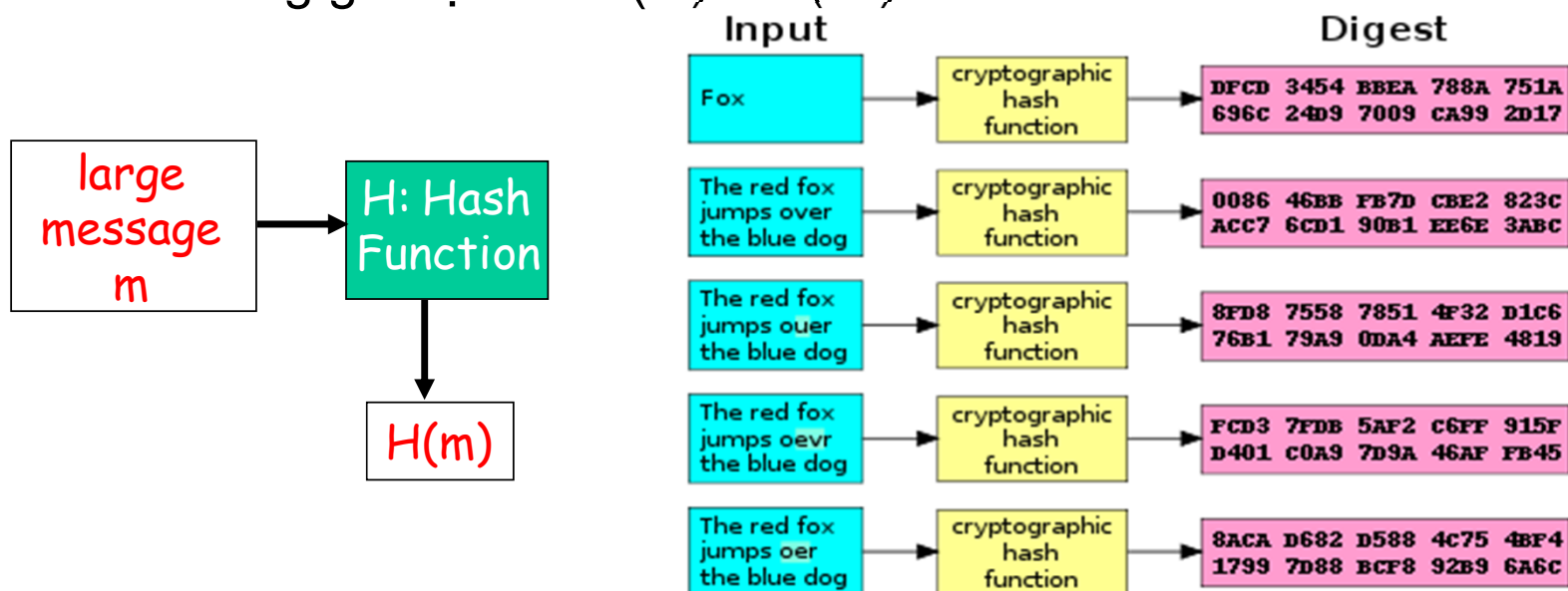
## ❑ GIẢI MÃ



# Hàm băm mật (Cryptographic Hash Function)

## ❑ Đặc điểm

- Hàm băm  $H(.)$  là một hàm nhận giá trị đầu vào là một chuỗi  $m$  có độ dài bất kỳ, đầu ra là một chuỗi  $H(m)$  có độ dài cố định
- Dễ tính toán
- Không thể đảo ngược: Không thể xác định  $m$  từ  $H(m)$
- Không thể thay thế: không thể tìm được hai chuỗi  $m$  và  $m'$  khác nhau mà có cùng giá trị băm  $H(m) = H(m')$ .





# Hàm băm mật (Cryptographic Hash Function)

## ❑ SO SÁNH VỚI INTERNET CHECKSUM

<u>message</u>	<u>ASCII format</u>
----------------	---------------------

I O U 1	49 4F 55 31
---------	-------------

0 0 . 9	30 30 2E 39
---------	-------------

9 B O B	39 42 D2 42
---------	-------------

B2 C1 D2 AC
-------------

<u>message</u>	<u>ASCII format</u>
----------------	---------------------

I O U <u>9</u>	49 4F 55 <u>39</u>
----------------	--------------------

0 0 . <u>1</u>	30 30 2E <u>31</u>
----------------	--------------------

9 B O B	39 42 D2 42
---------	-------------

B2 C1 D2 AC
-------------

different messages  
but identical checksums!

# Hàm băm mật (Cryptographic Hash Function)

---

## ❑ CÁC GIẢI THUẬT BĂM

- MD5 (Message-Digest Algorithm 5): giải thuật được sử dụng rộng rãi với giá trị băm dài 128 bit.
  - MD5 là chuẩn của Internet (RFC 1321), thiết kế bởi Ronald Rivest năm 1991
- SHA-1 (Secure Hash Algorithm 1) được sử dụng rộng rãi và là chuẩn của Mỹ (NIST), sử dụng giá trị băm có độ dài 160 bit.

# Authentication Key

---

## ❑ VẤN ĐỀ TOÀN VỆ BẢN TIN

- Alice tạo bản tin  $m$  và tính giá trị băm  $h=H(m)$  của  $m$
- Thêm  $h$  vào  $m$  được bản tin mở rộng  $(m, h)$  và gửi đi cho Bob
- Bob nhận được  $(m, h)$  và tính  $H(m)$  từ  $m$  đã nhận. Nếu  $H(m)=h$  thì Bob kết luận bản tin  $m$  không bị thay đổi

## ❑ ĐIỂM YẾU

- Hacker có thể tạo một bản tin giả  $m'$ , (khẳng định mình là Alice),
- Tính  $H(m')$  và gửi cho Bob  $(m', H(m'))$ .
- Khi nhận được bản tin, Bob check như bước 3 và không thể phát hiện bản tin này là giả

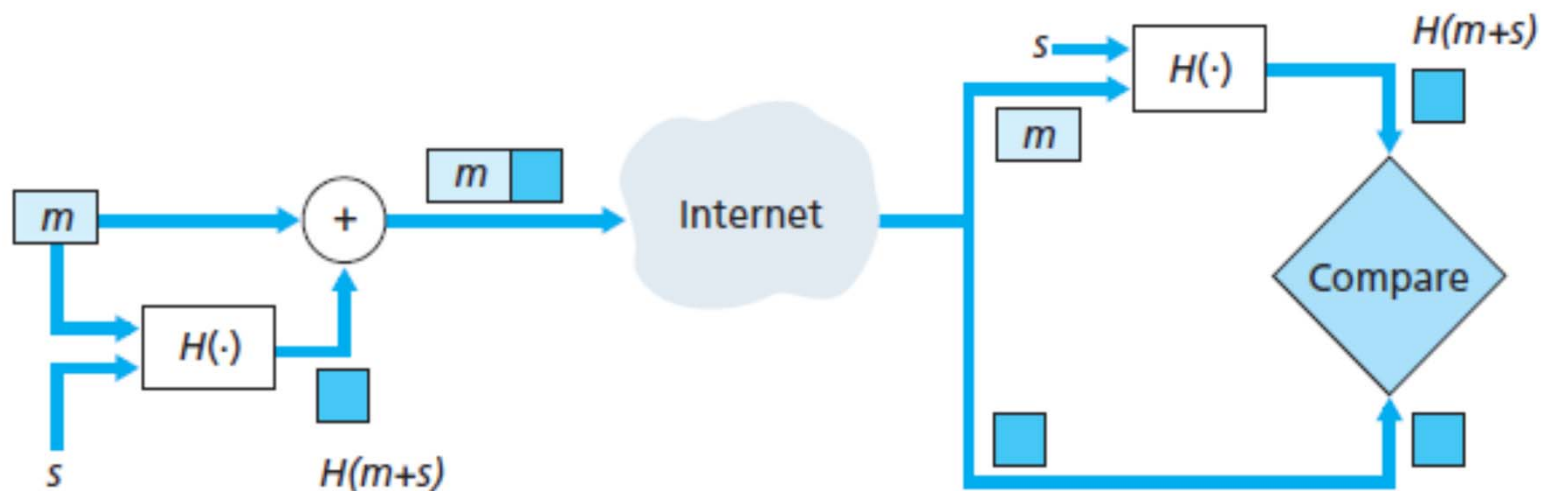
## ❑ GIẢI PHÁP ?

- Authentication Key

# Authentication Key

## ❑ HOẠT ĐỘNG

- Cần chia sẻ một bí mật  $s$  (là một chuỗi bit) được gọi là khóa nhận thực (authentication key)
- MAC (message authentication code) =  $H(m+s)$



Key:

$m$  = Message

$s$  = Shared secret


# Chữ ký số (Digital Signature)

- ❑ Chữ ký số cho phép người nhận kiểm tra tính xác thực (verifiable) và tính nguyên vẹn của văn bản và tính không thể giả mạo (nonforgeable) của người ký văn bản hay người tạo ra văn bản.
- ❑ Dựa vào tính chất của RSA

$$K_B^-(K_B^+(m)) = K_B^+(K_B^-(m)) = m$$

Bob's message,  $m$

Dear Alice  
Oh, how I have missed you. I think of you all the time! ... (blah blah blah)  
Bob

  $K_B^-$  Bob's private key

Public key  
encryption  
algorithm

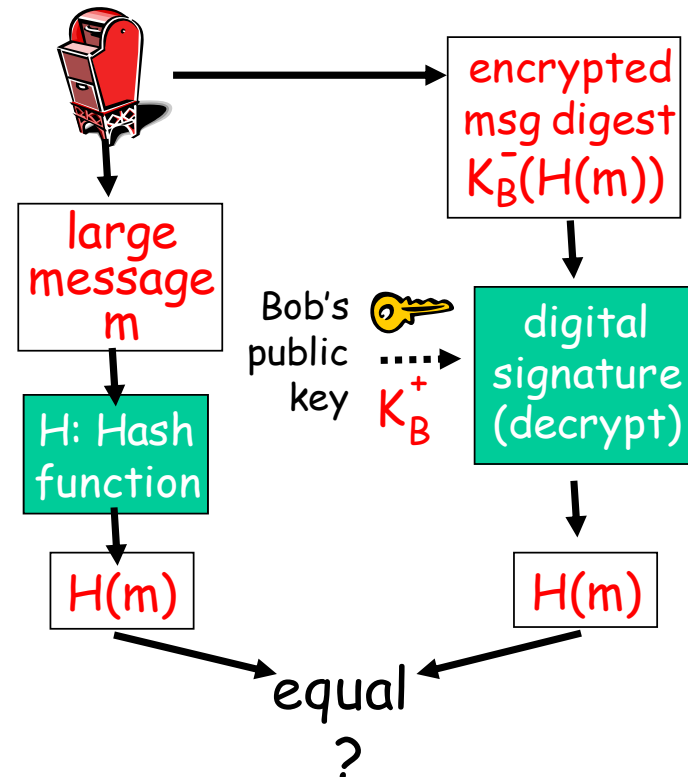
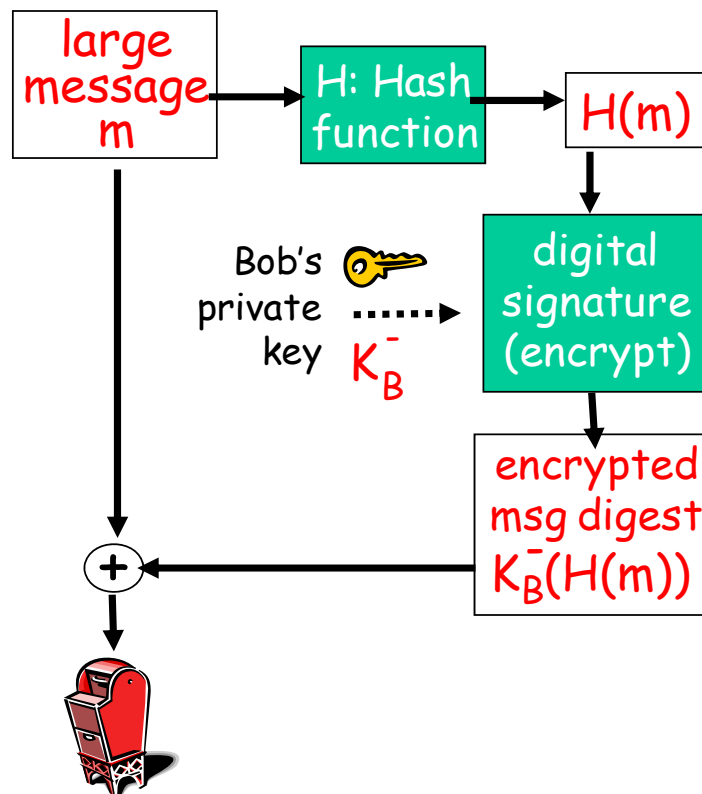
$K_B^-(m)$

Bob's message,  
 $m$ , signed  
(encrypted) with  
his private key

# Chữ ký số (Digital Signature)

## ❑ KẾT HỢP VỚI HÀM BĂM

Bob sends digitally signed message:



## Một số hướng đi tương lai của mật mã

---

- ❑ Bảo mật trong điện toán đám mây (cloud computing)
- ❑ Mở rộng mô hình mã hóa: cho đối tượng nhóm và cho việc giải mã bộ phận
- ❑ An toàn trước các tấn công vật lý
- ❑ An toàn trước sự tấn công của máy tính lượng tử