

# MẬT MÃ ƯD TRONG ATTT

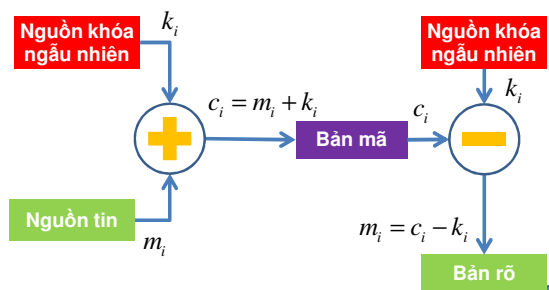
Bài 02. Mật mã đối xứng

- 1 Cấu trúc, đặc điểm của mã dòng
- 2 Mã dòng RC4
- 3 Cấu trúc, đặc điểm của mã khối
- 4 Chế độ hoạt động của mã khối

- 1 **Cấu trúc, đặc điểm của mã dòng**
- 2 Mã dòng RC4
- 3 Cấu trúc, đặc điểm của mã khối
- 4 Chế độ hoạt động của mã khối

## Khái niệm

### Hệ mật Vernam (One-Time Pad)



## Ví dụ hệ mật Vernam

### MÃ HÓA

Bộ kí tự: chữ cái latin  
 Khóa ngẫu nhiên: **PWKAX**  
 Thông điệp: **HELLO**

Rõ	H (7)	E (4)	L (11)	L (11)	O (14)
Khóa	P (15)	W (22)	K (10)	A (0)	X (23)
Mã	W (22)	A (0)	V (21)	? (?)	? (?)

5

## Ví dụ hệ mật Vernam

### GIẢI MÃ

Bộ kí tự: chữ cái latin  
 Khóa ngẫu nhiên: **PWKAX**  
 Bản mã: **WAVLL**

Mã	W (22)	A (0)	V (21)	L (11)	L (11)
Khóa	P (15)	W (22)	K (10)	A (0)	X (23)
Rõ	H (7)	E (4)	L (11)	L (11)	O (14)

6

### Đặc trưng của hệ mật Vernam



Có độ mật hoàn thiện

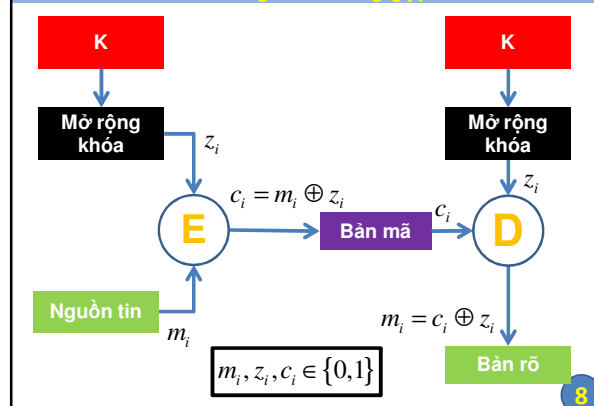
$$P(m/c) = P(m)$$



Kích thước khóa bằng kích thước bản rõ

7

### Mã dòng – Thường gặp



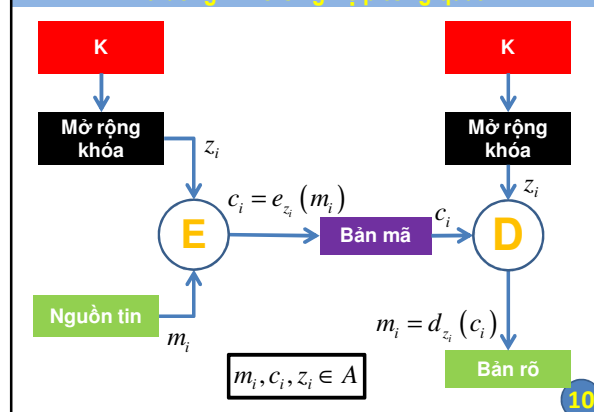
8

### Định nghĩa mã dòng

- Một hệ mật mã dòng là một hệ mật đối xứng, trong đó các kí tự rõ được kết hợp với một dòng kí tự khóa giả ngẫu nhiên.
- Trong mã dòng, từng kí tự rõ được mã hóa riêng rẽ bởi một kí tự tương ứng trong dòng khóa để cho ra một kí tự mã.

9

### Mã dòng – Trường hợp tổng quát



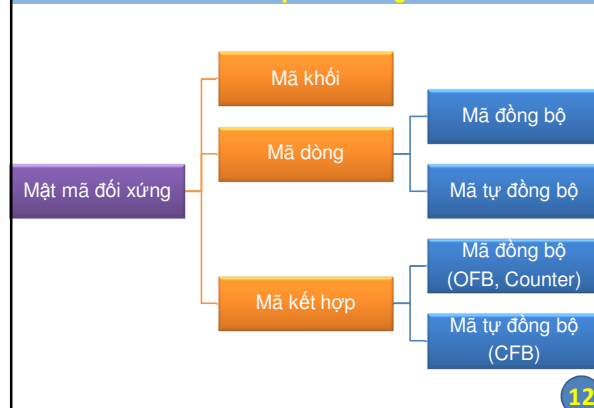
10

### Mã dòng

- **Khởi «mở rộng khóa»**
  - Là bộ sinh số giả ngẫu nhiên (PRNG)
  - Là quan trọng nhất
  - Quyết định độ an toàn của mã dòng
- **Phân loại**
  - $z_i$  chỉ phụ thuộc  $K$ : «mã dòng **đồng bộ**»
  - $z_i$  phụ thuộc  $c_{i-n}, c_{i-n+1}, \dots, c_{i-1}$ : «mã dòng **tự đồng bộ**»

11

### Khái niệm mã dòng



12

**Đặc điểm của mã dòng**

- Mỗi phần tử đầu vào được mã hóa bởi một phần tử riêng biệt của dòng khóa
- Kết quả biến đổi một phần tử đầu vào phụ thuộc vào vị trí của phần tử trong chuỗi
- Tốc độ cao, có thể mã hóa/giải mã gần với thời gian thực
- Có thể được cài đặt hiệu quả bằng phần cứng và/hoặc phần mềm

13

**Ứng dụng của mã dòng**

- Mã dòng có tốc độ cao do cơ chế sinh dòng khóa và mã hóa khá đơn giản so với mã khối.
- Mã dòng có thể mã hóa lượng dữ liệu bất kỳ, không cần phải chờ đợi kích thước đầu vào đạt đến giá trị nhất định như mã khối.

Mã hóa dữ liệu yêu cầu độ trễ thấp: voice, video conference

Mã hóa dữ liệu từ nguồn sinh liên tục, tốc độ không ổn định

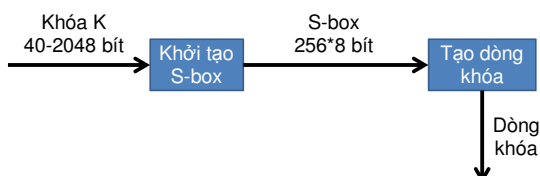
14

- 1 Cấu trúc, đặc điểm của mã dòng
- 2 **Mã dòng RC4**
- 3 Cấu trúc, đặc điểm của mã khối
- 4 Chế độ hoạt động của mã khối

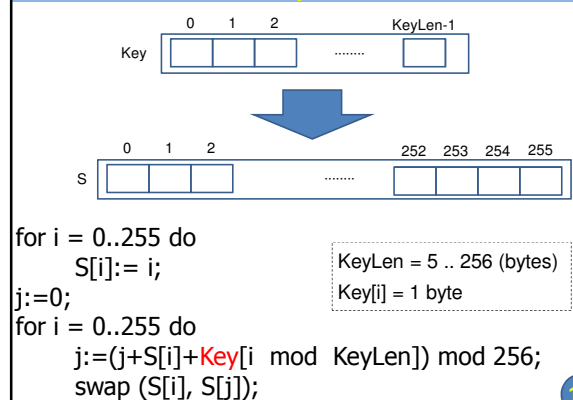
**Thông tin chung về RC4**

- ❑ RC4 được thiết kế để đạt hiệu năng cao khi cài đặt bằng phần mềm
- ❑ Xây dựng bởi Ron Rivest năm 1987 nhưng đến năm 1994 mới được tiết lộ
- ❑ Được ứng dụng rộng rãi
- ❑ Kích thước khóa: 40-2048 bit [4]

16

**Sơ đồ chung của RC4**

17

**Khởi tạo S-box**

18

### Tạo dòng khóa

```

i:=0; j:=0;
while GeneratingOutput:
  i:=(i+1) mod 256;
  j:= (j+S[i]) mod 256;
  swap (S[i], S[j]);
  z:= S[(S[i] + S[j]) mod 256];
  Output z; //dòng khóa được sinh từng byte
end while

```

19

### Thông tin thêm về RC4

- ❑Thuật toán đơn giản, rõ ràng
- ❑Kích thước từ có thể thay đổi (ví dụ, có thể sử dụng 4 bit thay vì 8 bit)
- ❑Dùng 1 khóa để mã 2 thông điệp???

20

### Thông tin thêm về RC4

❑**Ứng dụng RC4**

- WEP
- BitTorrent protocol encryption
- Microsoft Point-to-Point Encryption
- Opera Mini
- Secure Sockets Layer\*
- Secure shell\*
- Remote Desktop Protocol
- Kerberos\*
- SASL Mechanism Digest-MD5\*
- PDF
- Skype

21

- 1 Cấu trúc, đặc điểm của mã dòng
- 2 Mã dòng RC4
- 3 Cấu trúc, đặc điểm của mã khối**
- 4 Chế độ hoạt động của mã khối

### Nguyên lý thiết kế mã khối

#### Nguyên tắc Kerckhoffs

- "A cryptosystem should be secure even if everything about the system, except the key, is public knowledge"
- "The enemy knows the system" (Shannon)

#### Nguyên lý «Khuếch tán và xáo trộn»

- Khuếch tán
- Xáo trộn
- [http://en.wikipedia.org/wiki/Confusion\\_and\\_diffusion](http://en.wikipedia.org/wiki/Confusion_and_diffusion)

#### Sử dụng hàm hợp

- Hàm mã hóa phức tạp = Hàm hợp của nhiều hàm mã hóa đơn giản không giao hoán

23

### Nguyên lý thiết kế mã khối

#### Cấu trúc lưới Feistel

(Feistel network)

#### Cấu trúc SPN

(Substitution-permutation network)

24

### Cấu trúc lưới Feistel (Feistel network)

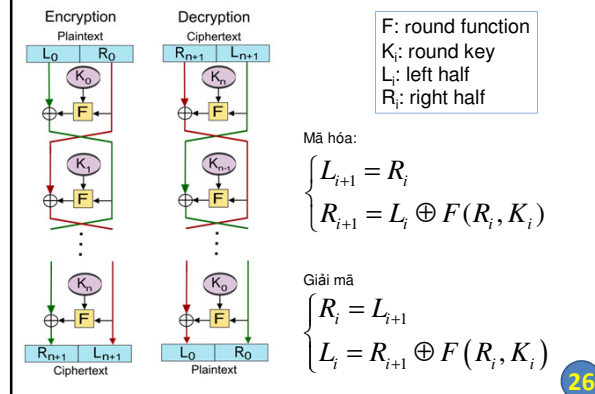
#### Horst Feistel

- Born in Germany
- January 30, 1915
- Died November 14, 1990
- Moved to US in 1934



25

### Cấu trúc lưới Feistel



26

### Đặc điểm của mạng Feistel

1. Tại mỗi vòng, chỉ một nửa khối được mã hóa  
→ cần nhiều vòng → giảm hiệu năng
2. Việc mã hóa và giải mã là giống hệt nhau, chỉ khác ở trật tự sử dụng khóa vòng → chỉ cần 1 hàm/1 mạch điện tử để thực hiện cả mã hóa và giải mã
3. Hàm  $F()$  không cần phải có hàm ngược  $F^{-1}()$

27

### Các hệ mật sử dụng mạng Feistel

Blowfish	Lucifer
Camellia	MARS
CAST-128	MAGENTA
DES	MISTY1
FEAL	RC5
GOST 28147-89	Simon
ICE	TEA
KASUMI	Triple DES
LOKI97	Twofish
	XTEA

28

### Ứng dụng khác của mạng Feistel

- ❑ Một số hệ mật sử dụng biến thể của mạng Feistel (CAST-256, CLEFIA, MacGuffin, RC2, RC6, Skipjack, SMS4)
- ❑ Một số hệ mật không có cấu trúc mạng Feistel nhưng chứa mạng Feistel trong thành phần của nó (MISTY1, Threefish)
- ❑ Mạng Feistel còn được sử dụng cho mục đích khác với xây dựng mã khối, ví dụ, sử dụng trong lược đồ OAEP (Optimal Asymmetric Encryption Padding)

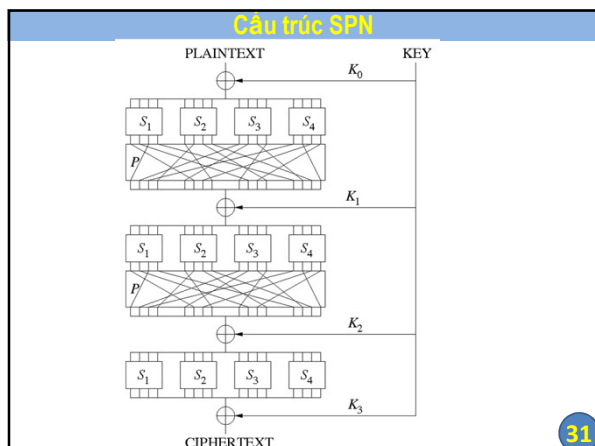
29

### Nguyên lý thiết kế mã khối

#### Cấu trúc SPN (Substitution-permutation network)

Rijndael, Square, Shark, BKSQ

30



- 1 Cấu trúc, đặc điểm của mã dòng
- 2 Mã dòng RC4
- 3 Cấu trúc, đặc điểm của mã khối
- 4 **Chế độ hoạt động của mã khối**

**Chế độ hoạt động của mã khối**

## Confidentiality modes

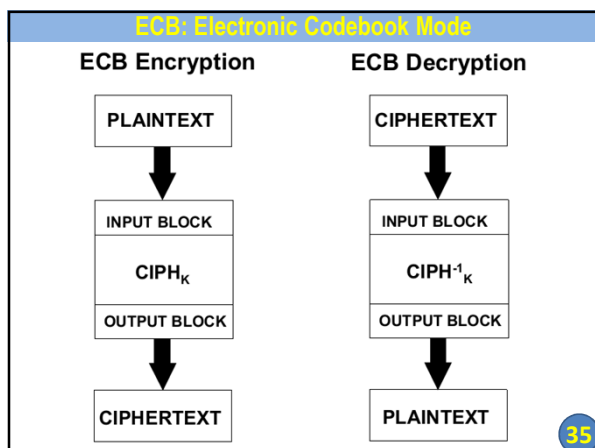
- ECB, CBC, OFB, CFB, CTR...

## Authenticated Encryption modes

- CCM, GCM, CWC, EAX...

33

- Chế độ hoạt động của mã khối**
1. ECB: Electronic Codebook Mode
  2. CBC: Cipher Block Chaining Mode
  3. OFB: Output Feedback Mode
  4. CFB: Cipher Feedback Mode
  5. CTR: Counter Mode
- 34



**ECB: Electronic Codebook Mode**

**ECB: Encrypt**

$$C_j = CIPH_K(P_j) \quad j = 1..n$$

**ECB: Decrypt**

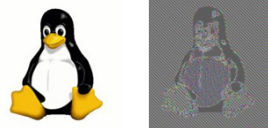
$$P_j = CIPH_K^{-1}(C_j) \quad j = 1..n$$

36

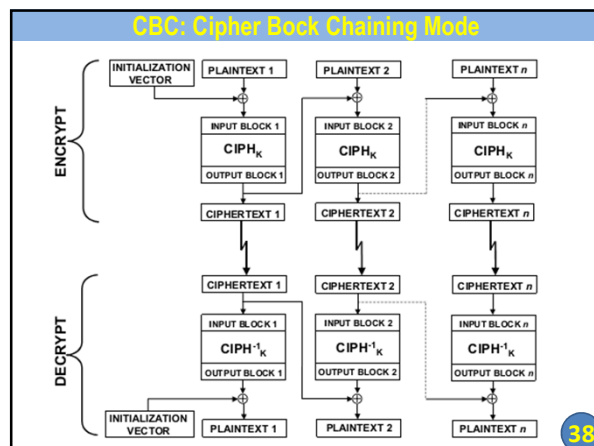
### ECB: Electronic Codebook Mode

**ECB: Đặc điểm**

- Đầu vào giống nhau dẫn đến đầu ra giống nhau
- Một bit lỗi trong khối mã dẫn đến mất một khối tương ứng khi giải mã
- Có thể xử lý các khối song song



37



### CBC: Cipher Block Chaining Mode

**CBC: Encrypt**

$$C_1 = CIPH_K(P_1 \oplus IV);$$

$$C_j = CIPH_K(P_j \oplus C_{j-1}) \quad \text{for } j = 2 \dots n.$$

**CBC: Decrypt**

$$P_1 = CIPH^{-1}_K(C_1) \oplus IV;$$

$$P_j = CIPH^{-1}_K(C_j) \oplus C_{j-1} \quad \text{for } j = 2 \dots n$$

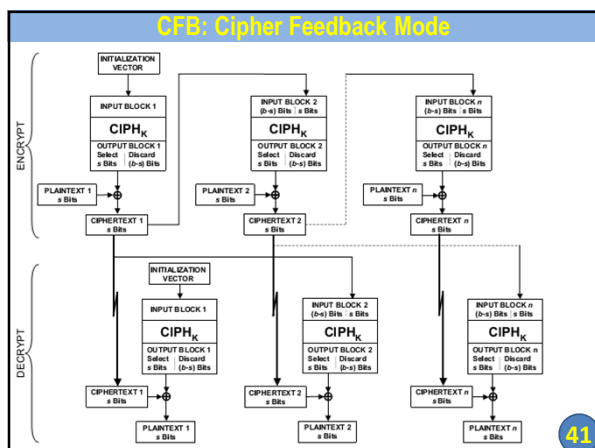
39

### CBC: Cipher Block Chaining Mode

**CBC: Đặc điểm**

- Khi các khối rõ giống nhau thì các khối mã vẫn khác nhau
- Một bit lỗi trong khối mã thứ  $j$  dẫn đến mất khối  $j$  và  $j+1$  khi giải mã
- Không thể mã hóa song song các khối, chỉ có thể giải mã song song

40



### CFB: Cipher Feedback Mode

**CFB: Encrypt**

$$I_1 = IV$$

$$I_j = LSB_{b-s}(I_{j-1}) \parallel C_{j-1}^\#$$

$$O_j = CIPH_K(I_j)$$

$$C_j^\# = P_j^\# \oplus MSB_s(O_j)$$

**CFB: Decrypt**

$$I_1 = IV$$

$$I_j = LSB_{b-s}(I_{j-1}) \parallel C_{j-1}^\# \quad j = 2..n$$

$$O_j = CIPH_K(I_j) \quad j = 1..n$$

$$P_j^\# = C_j^\# \oplus MSB_s(O_j) \quad j = 1..n-1$$

42

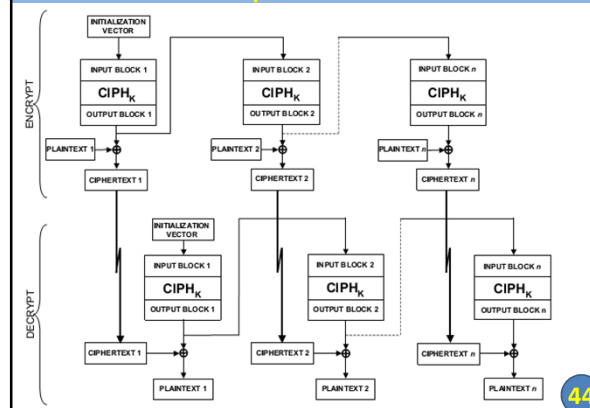
## CFB: Cipher Feedback Mode

## CFB: Đặc điểm

- Khi bản rõ là giống nhau thì bản mã vẫn khác nhau
- Một bit lỗi trong bản mã dẫn đến mất  $\lceil b/s \rceil$  khối (s bit) khi giải mã
- Là mã dòng tự đồng bộ, chịu được lỗi mất hoặc thêm ký tự

43

## OFB: Output Feedback Mode



44

## OFB: Output Feedback Mode

## OFB: Encrypt

$$\begin{aligned}
 I_1 &= IV & I_1 &= IV \\
 I_j &= O_{j-1} & I_j &= O_{j-1} & j &= 2..n \\
 O_j &= CIPH_K(I_j) & O_j &= CIPH_K(I_j) & j &= 1..n \\
 C_j &= P_j \oplus O_j & P_j &= C_j \oplus O_j & j &= 1..n-1 \\
 C_n^\# &= P_n \oplus MSB_u(O_n) & P_n^\# &= C_n \oplus MSB_u(O_n)
 \end{aligned}$$

## OFB: Decrypt

45

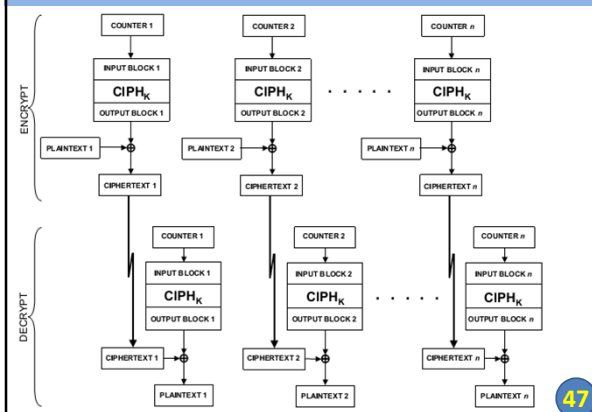
## OFB: Output Feedback Mode

## OFB: Đặc điểm

- Khi bản rõ là giống nhau thì bản mã vẫn khác nhau
- Một bit lỗi trong bản mã dẫn đến một bit lỗi tương ứng khi giải mã
- Bản rõ không chặn kích thước khối vẫn không cần đệm
- Là mã dòng đồng bộ

46

## CTR: Counter Mode



47

## CTR: Counter Mode

## CTR: Encrypt

$$\begin{aligned}
 O_j &= CIPH_K(T_j) & O_j &= CIPH_K(T_j) & j &= 1..n \\
 C_j &= P_j \oplus O_j & P_j &= C_j \oplus O_j & j &= 1..n-1 \\
 C_n^\# &= P_n \oplus MSB_u(O_n) & P_n^\# &= P_n \oplus MSB_u(O_n)
 \end{aligned}$$

## CTR: Decrypt

48



**CTR: Counter Mode****CTR: Đặc điểm**

- Khi bản rõ là giống nhau thì bản mã vẫn khác nhau
- Một bit lỗi trong bản mã dẫn đến một bit lỗi tương ứng khi giải mã
- Bản rõ không chặn kích thước khối vẫn không cần đệm
- Là mã dòng đồng bộ
- Có thể cài đặt song song

49