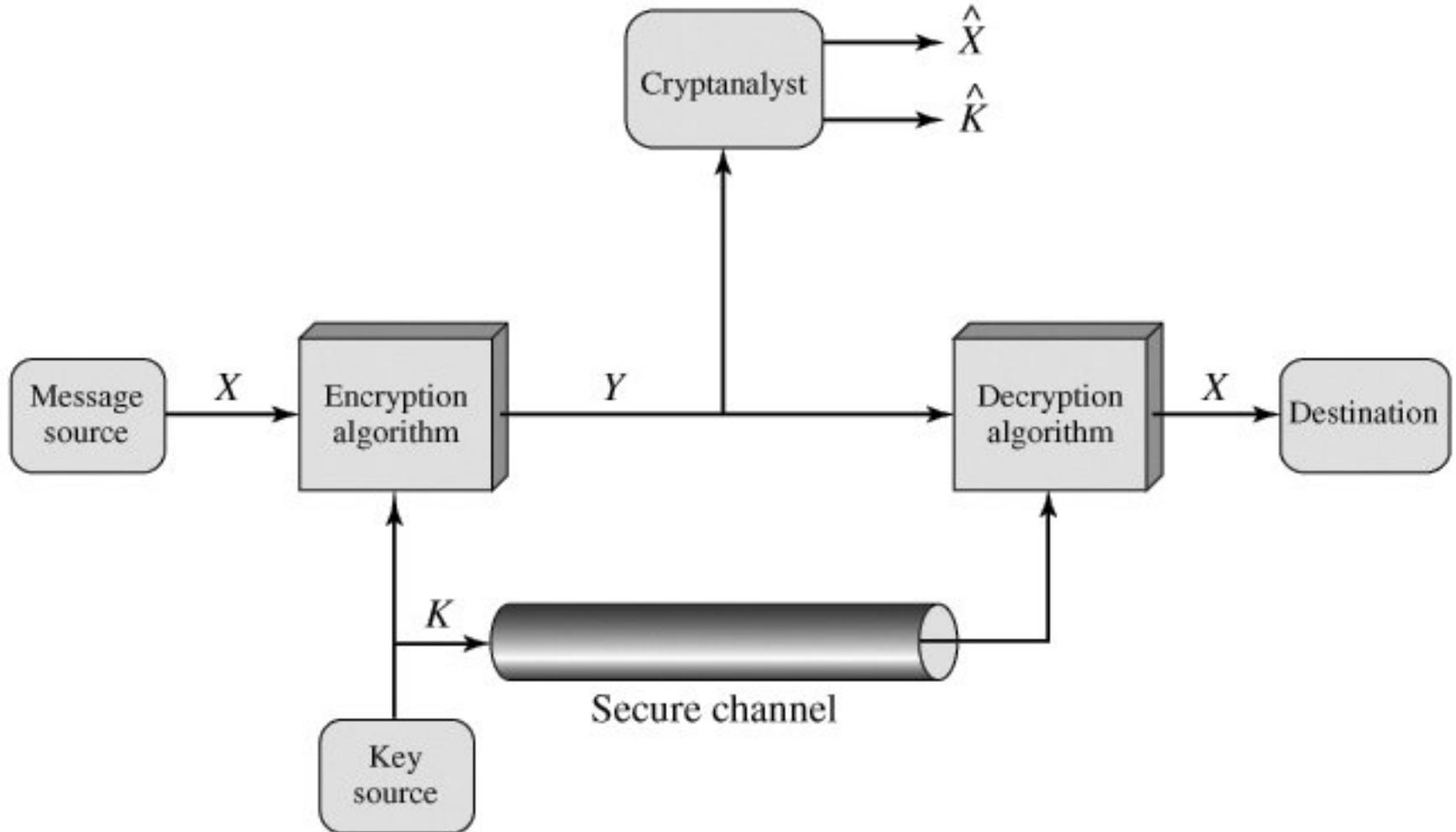


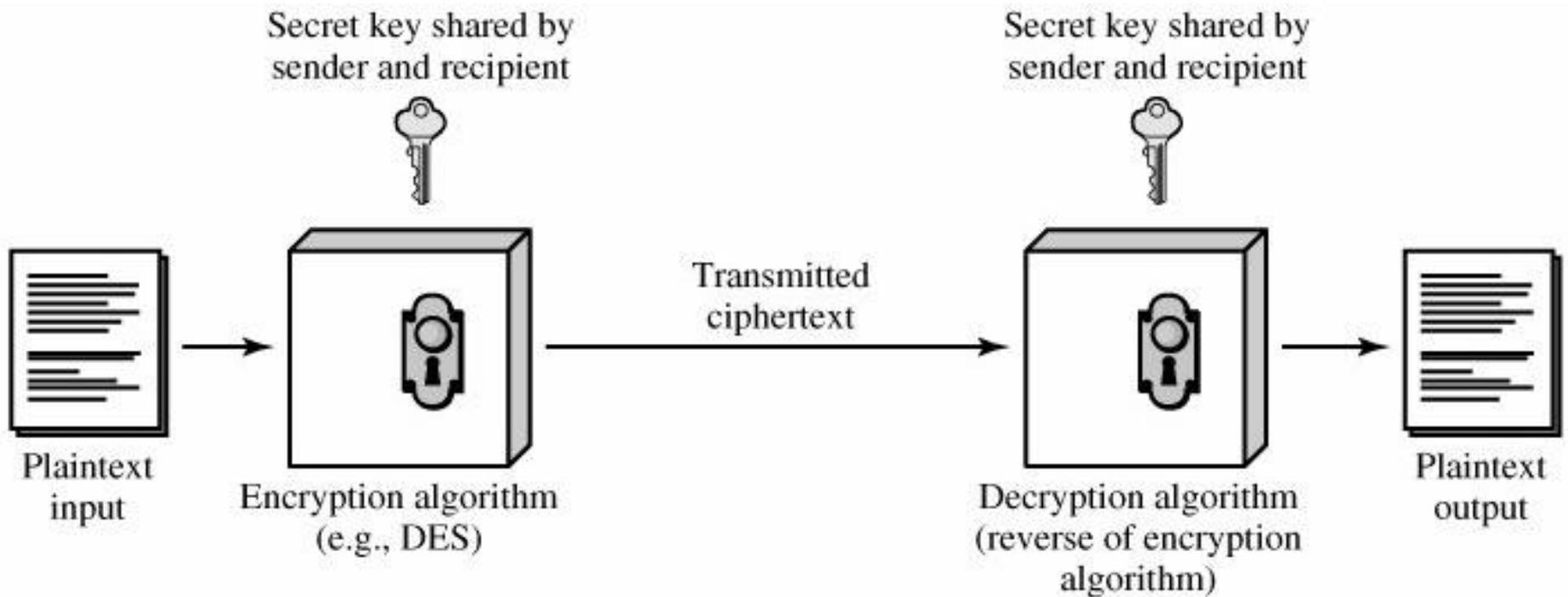
# CS ??? Computer Security Cryptography

Yasser F. O. Mohammad

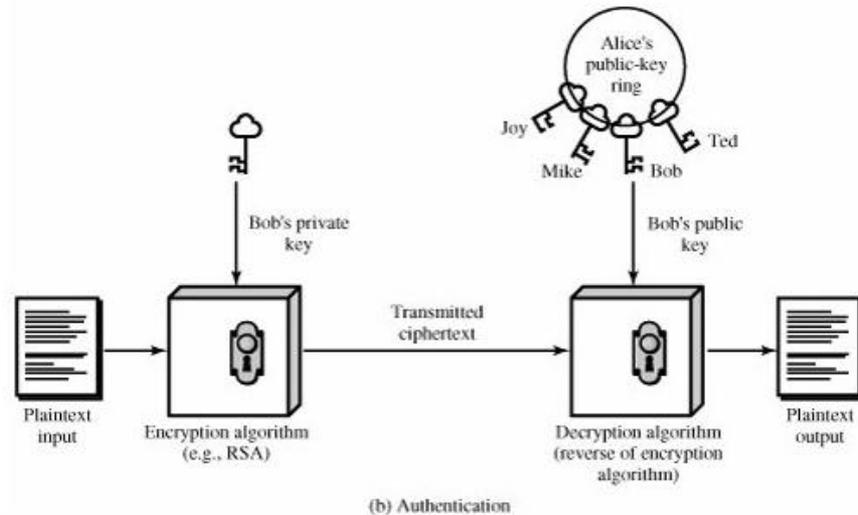
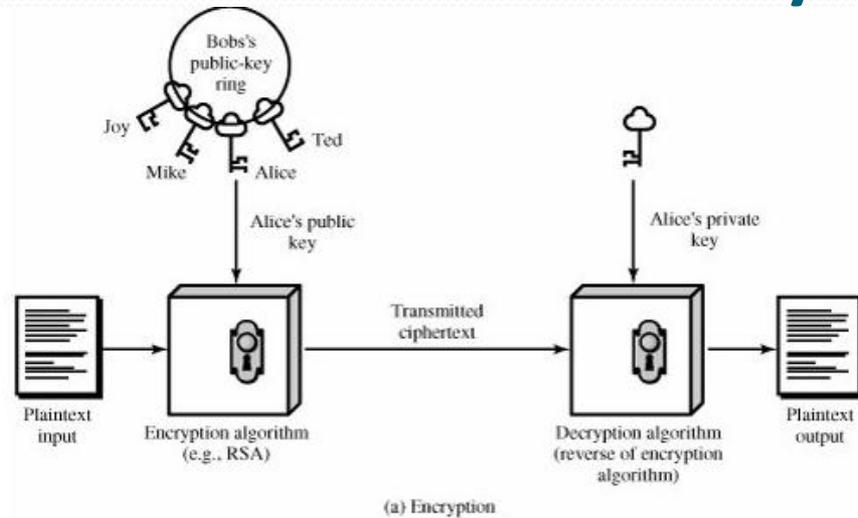
# REMINDER 1: Operational of Conventional Cryptosystem



# REMINDER 2: Shared Key Encryption



# REMINDER 3: Public Key Encryption



# REMINDER 4: Classical

## Cryptosystems

### Substitution Techniques

- Caesar Cipher

- Example

Plain : meet me after the toga party

cipher: PHHW PH DIWHU WKH WRJD SDUWB

- Substitution Table:

plain: abcdefghijklmnopqrstuvwxyz

cipher: DEFGHIJKLMNOPQRSTUVWXYZABC

- Formula

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

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

*How to do cryptanalysis???*

# One Time Pad

- Ultimate Security Algorithm

$$c_i = k_i \oplus p_i$$

- If  $k$  is truly random, then the code is unbreakable
- To encipher a text of  $n$  characters you need to securely distribute a key of  $n$  characters. Why don't we transfer the original plain text instead?

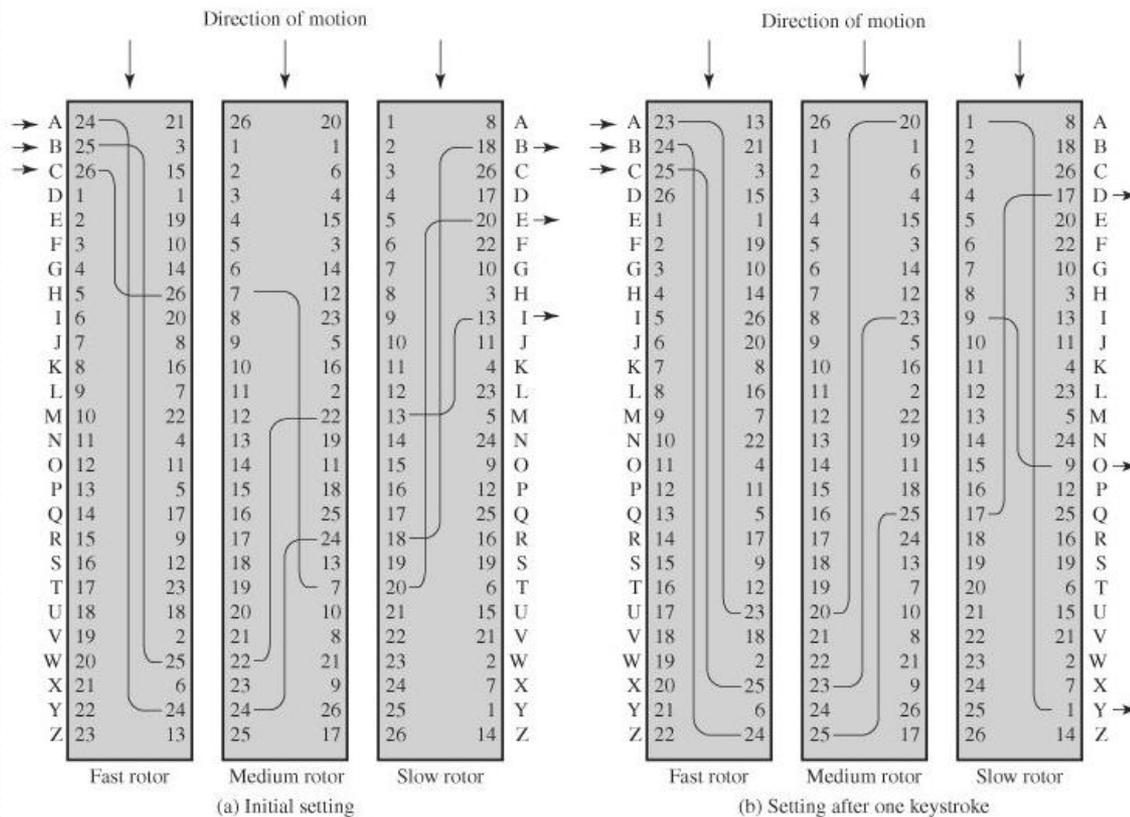
# Simple Transposition Cipher

- Put data in rows and read them in columns of arbitrary order
- Key: 4 3 1 2 5 6 7
- Input: t t n a a p t  
m t s u o a o  
d w c o i x k  
n l y p e t z

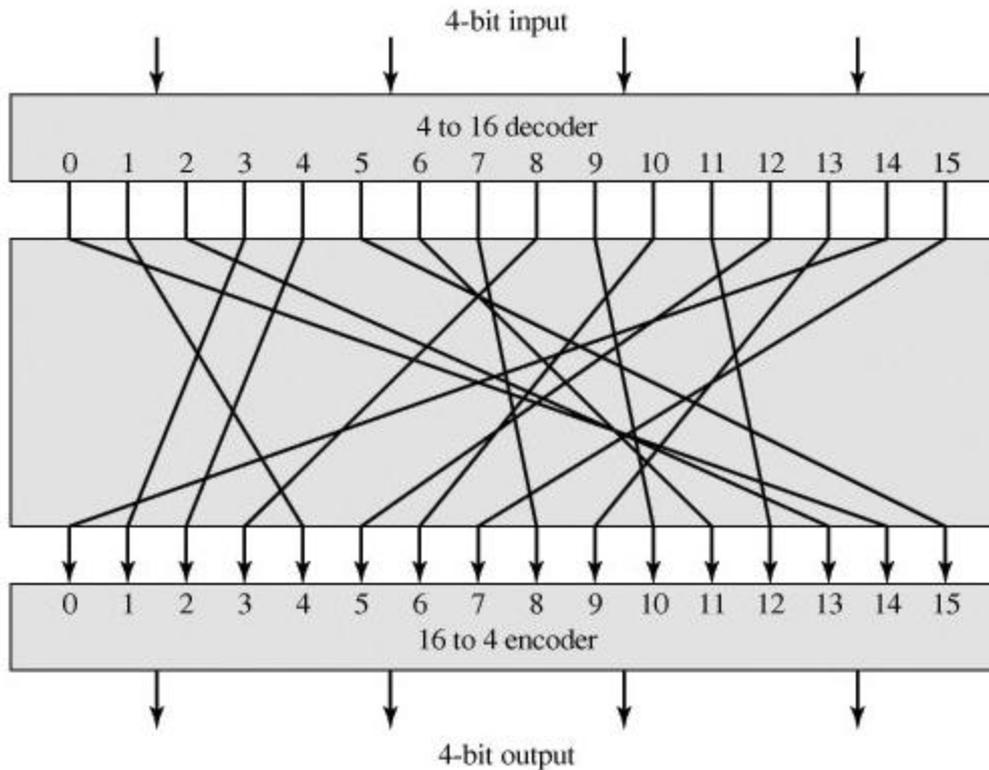
Output: NSCYAUOPTTWLTMDNAOIEPAXTTOKZ

# Rotor Machines

- Used by German (Engema) and Japanese (Purple) in WW II and was broken by Turing and others



# Ideal Block Cipher

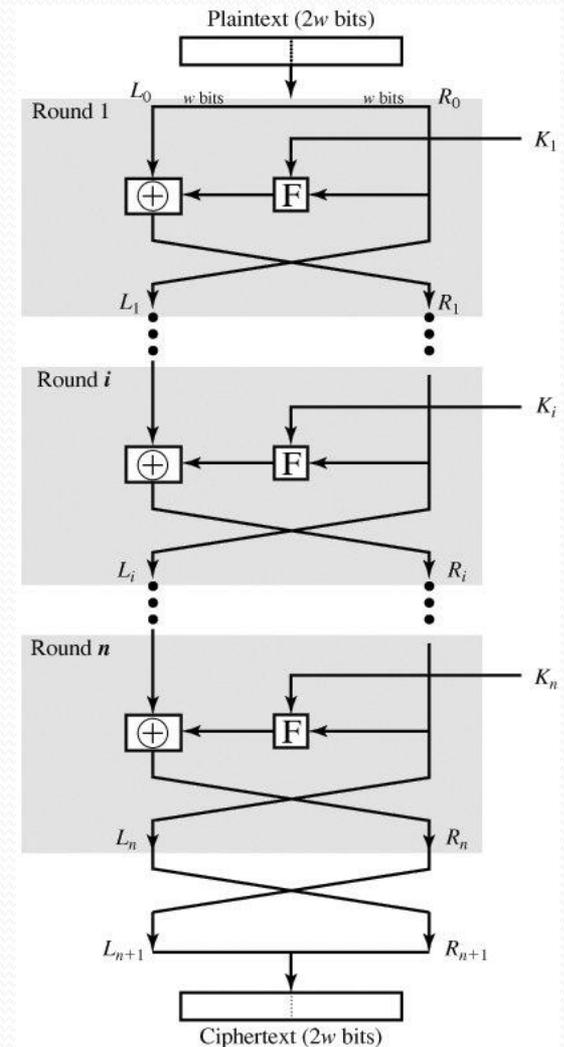


Plaintext	Ciphertext
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111

- Needs  $n \cdot 2^n$  key [in fact  $n \cdot (2^n - 1)$ ]
- Short Block  $\rightarrow$  Easily breakable
- Long Block  $\rightarrow$  Difficult to implement (storing the key)

# Fiestel Network

- Each round consists of:
  - Substitution on left half of text
  - Permutation of the two halves
- The substitution is controlled by the key of every round
- Factors of Security:
  - Block size
  - Key size
  - N. rounds
  - Subkey generation
  - Round Function
- Decryption = Encryption with reversed subkey order



# Example Block Ciphers

- DES (Data Encryption Standard)  $\approx$  DEA
  - 1977 and cracked in 1998 with 250,000\$ in 3 days
  - 64 bits block and 56 bits key
- 3DES
  - $C = E(k_3, D(k_2, E(k_1, M)))$ ,  $M = D(k_1, E(k_2, D(k_3, C)))$
  - Key length = 56, 112, 168
  - Not suitable for software
- AES (Advanced Encryption Standard)
  - 128 bits block and 128, 192, 256 bits key
  - Not a Feistel structure

# Other Examples of Block Ciphers

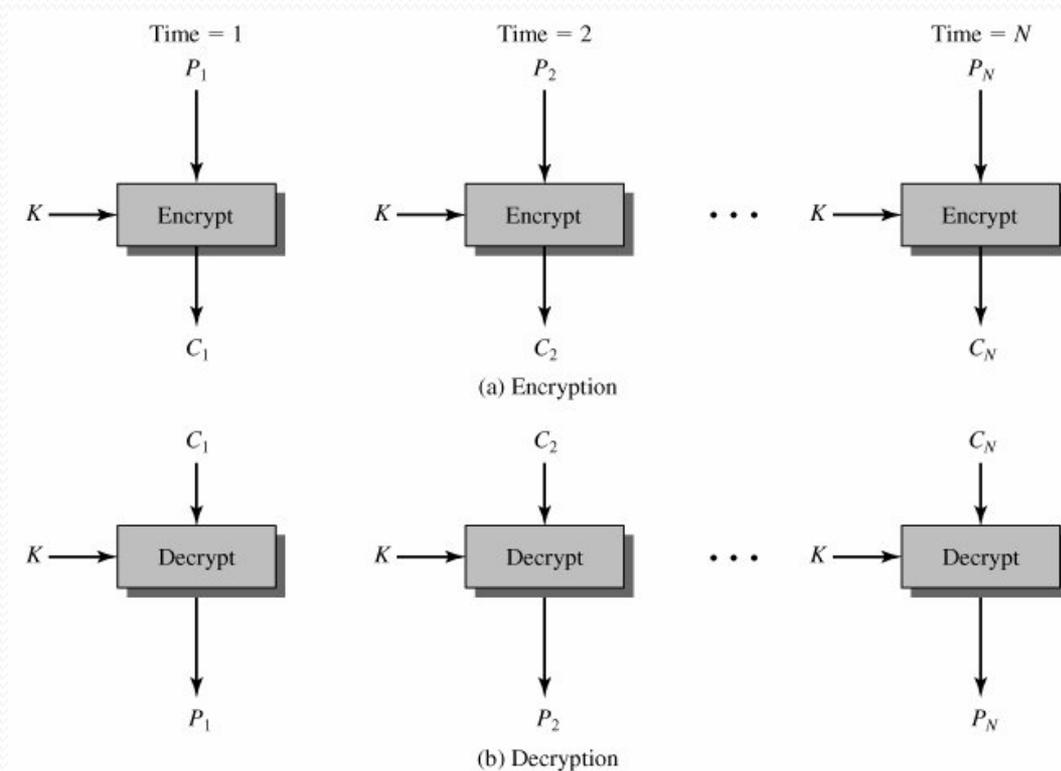
- IDEA (International Data Encryption Algorithm)
  - 128 bit key
  - Uses XOR, binary addition and multiplication
- Blowfish
  - 1993 by Bruce Schneier
  - Fast and easy to implement
  - Variable S-boxes
- RC<sub>5</sub>
  - 1994 By Ron Rivest
  - Suitable for hardware and software
  - Used by RSA security Inc.

# Uses of Shared-Key Ciphers

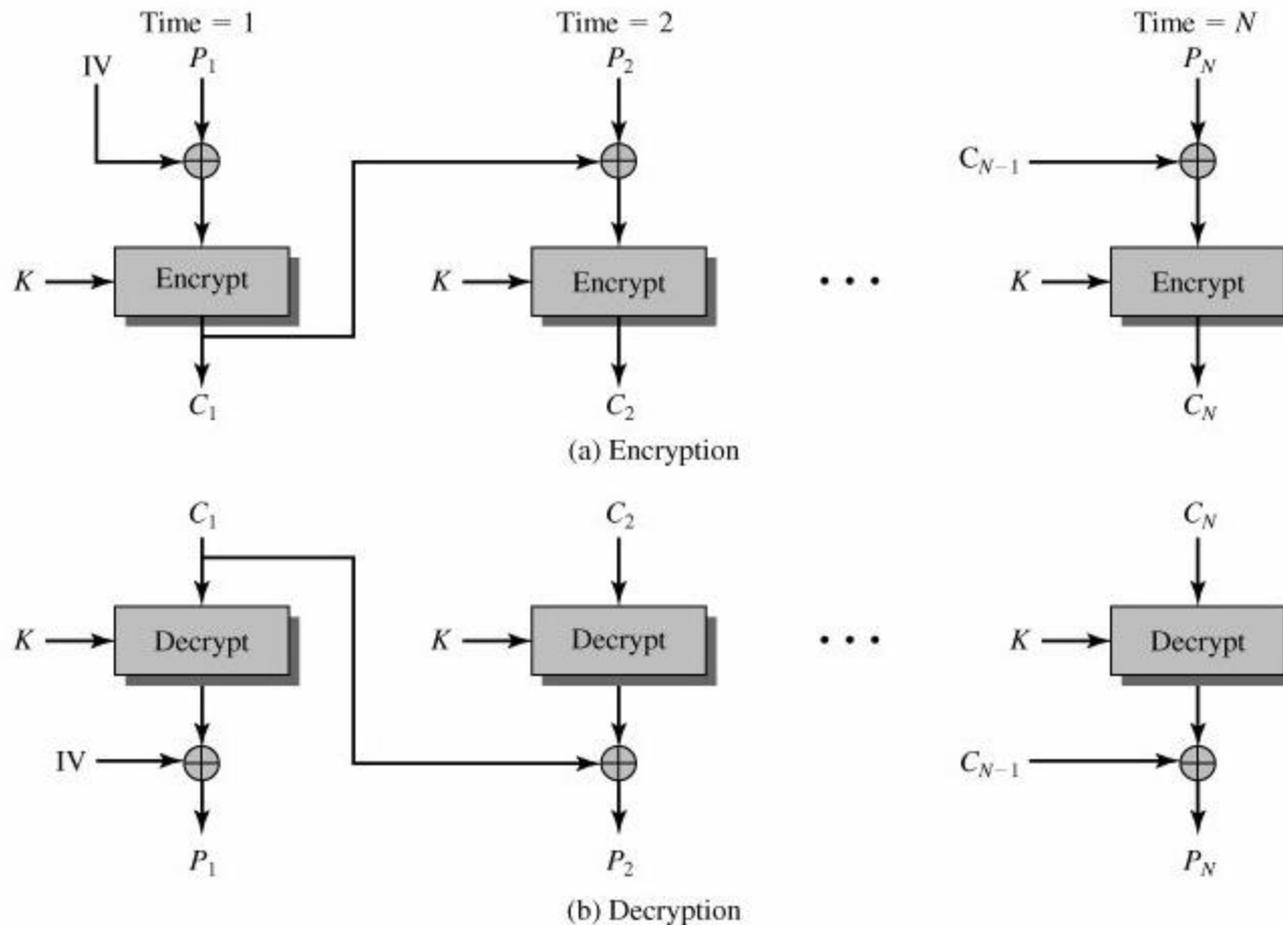
<b>Algorithm</b>	<b>Key Size (bits)</b>	<b>Block Size (bits)</b>	<b>Number of Rounds</b>	<b>Applications</b>
DES	56	64	16	SET, Kerberos
Triple DES	112 or 168	64	48	Financial key management, PGP, S/MIME
AES	128, 192, or 256	128	10, 12, or 14	Intended to replace DES and 3DES
IDEA	128	64	8	PGP
Blowfish	Variable to 448	64	16	Various software packages
RC5	Variable to 2048	64	Variable to 255	Various software packages

# ECB (Electronic Codebook)

- Just apply it to every block in succession
- Every plain text block has the same corresponding cipher

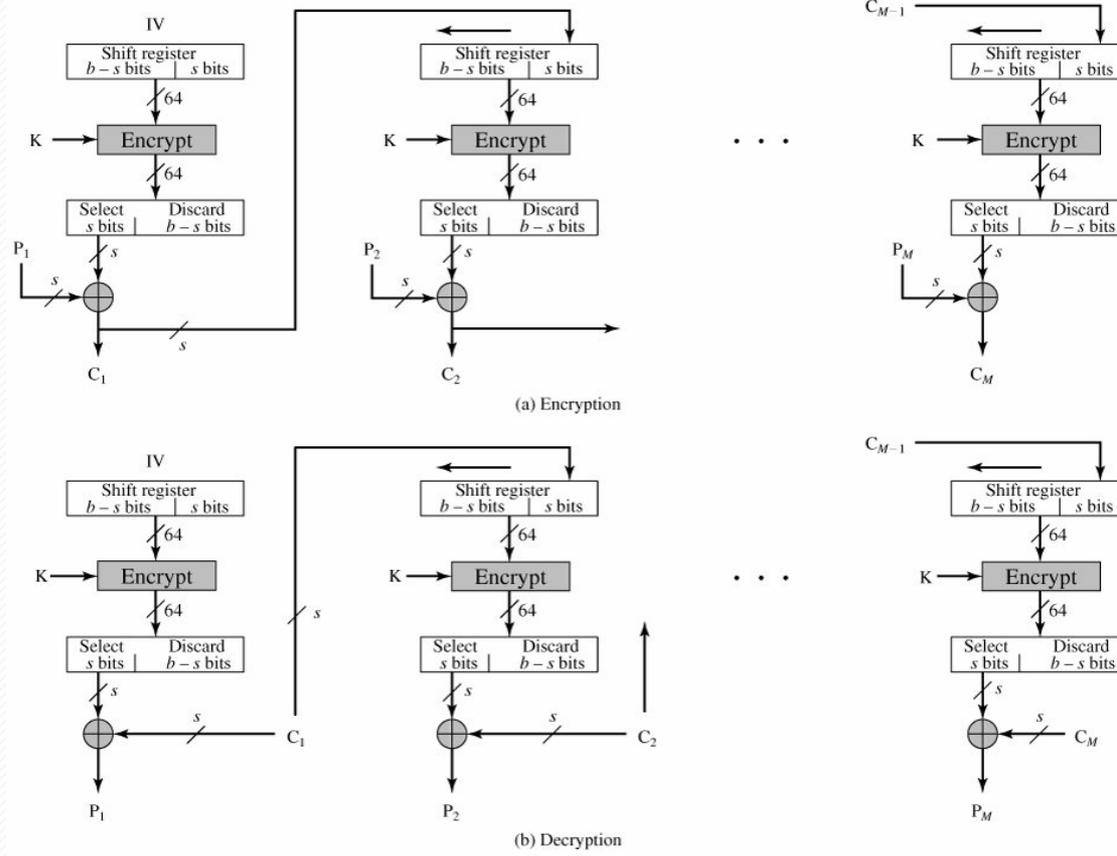


# CBC (Cipher Block Chaining Mode)



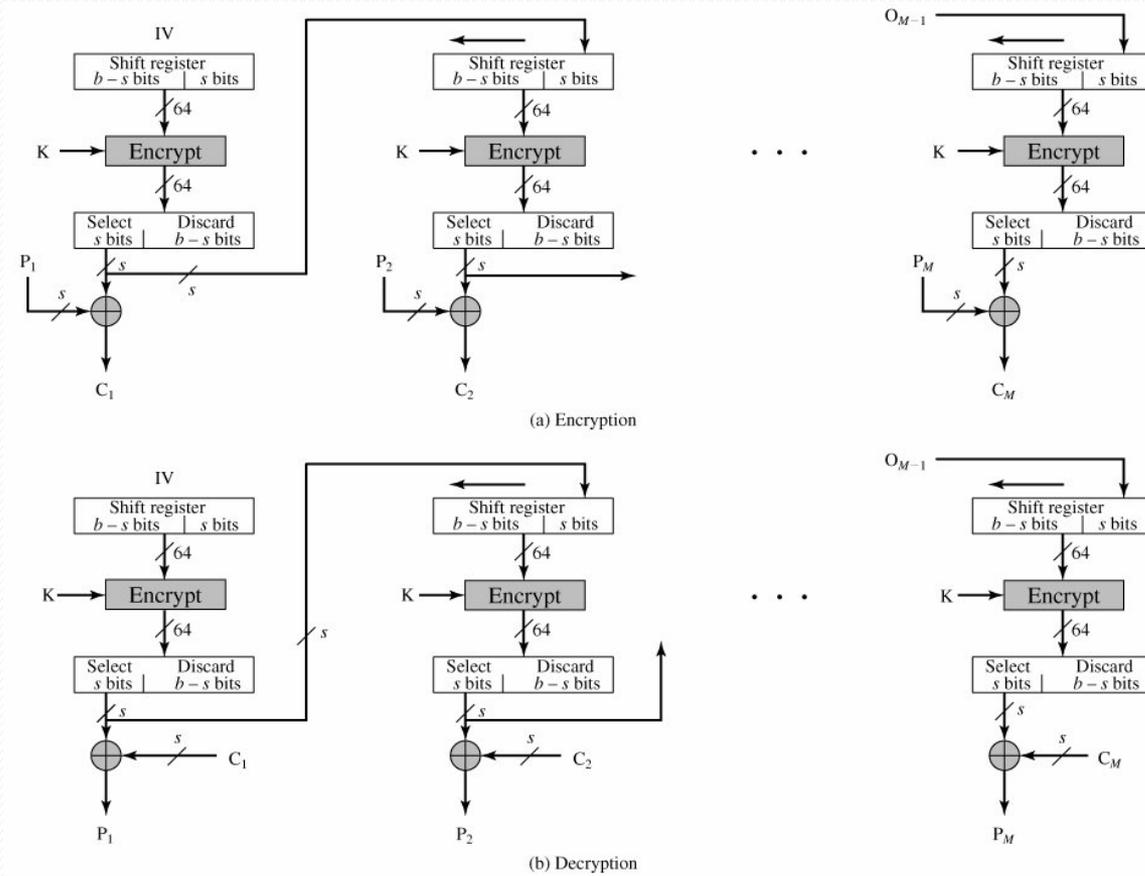
# CFB (Cipher Feedback Mode)

- Block cipher  $\rightarrow$  stream cipher

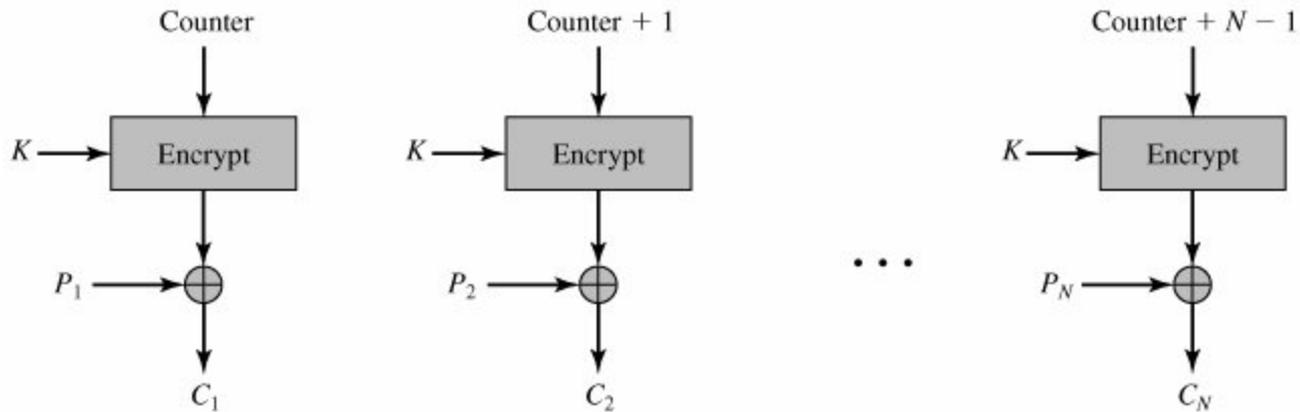


# OFB (Output Feedback Mode)

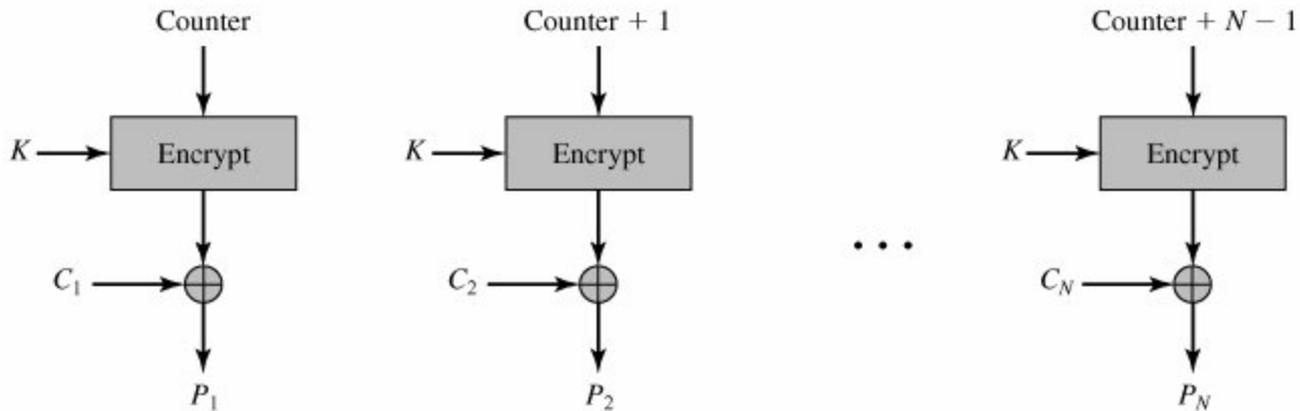
- Block cipher  $\rightarrow$  Stream cipher



# CTR (Counter Mode)

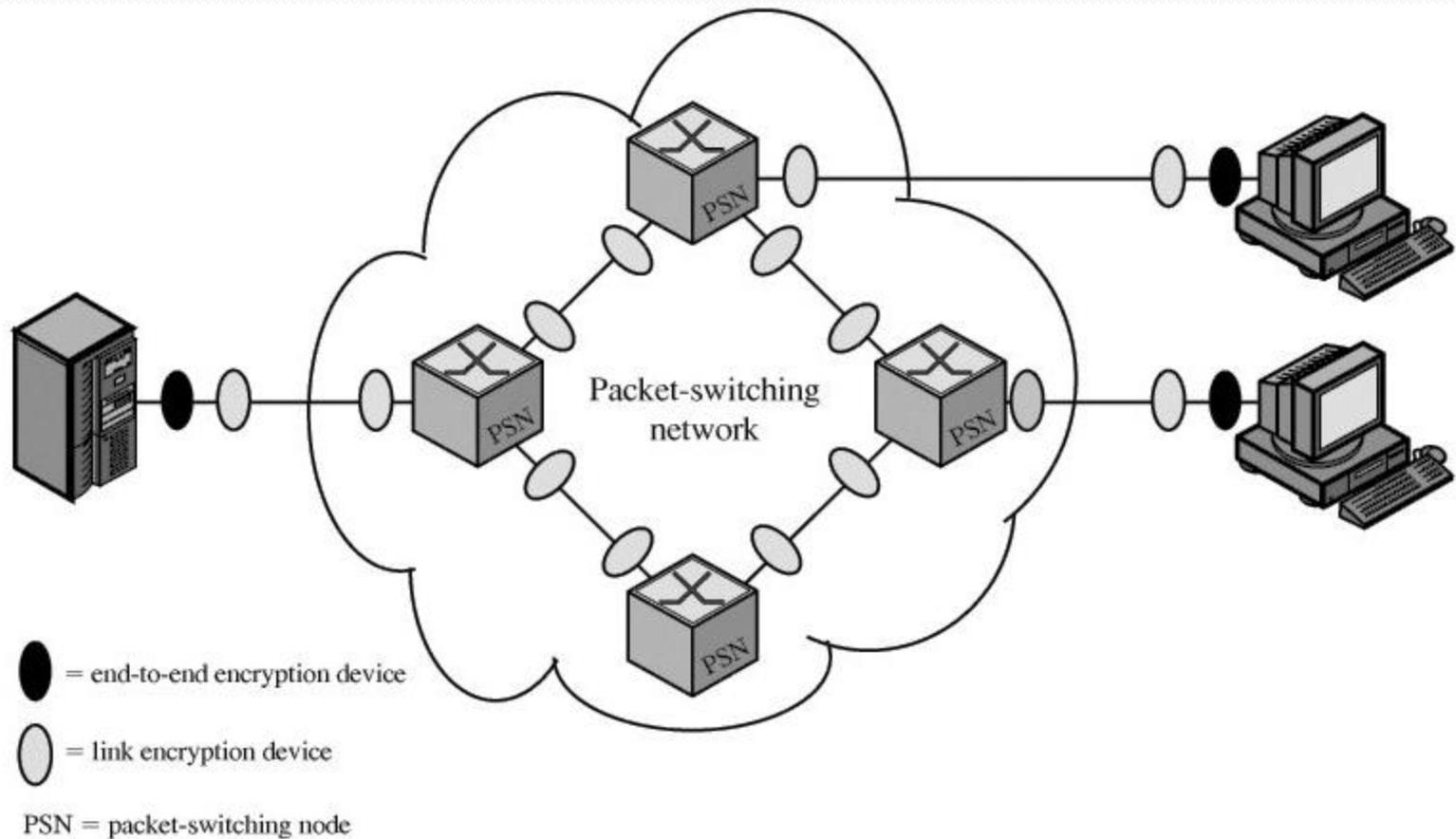


(a) Encryption

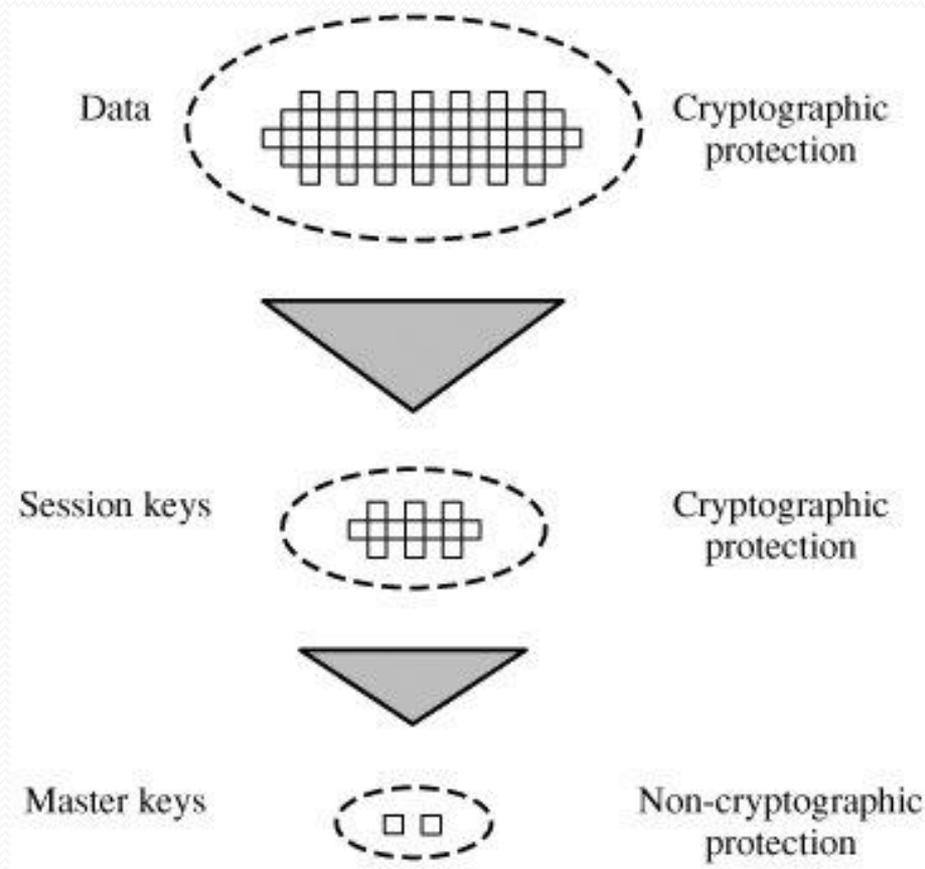


(b) Decryption

# Location of Encryption Function



# Key Hierarchy



# Key Distribution Center

1. Host sends packet requesting connection
2. Front end buffers packet; asks KDC for session key
3. KDC distributes session key to both front ends
4. Buffered packet transmitted

FEP = front-end processor  
KDC = key distribution center

