

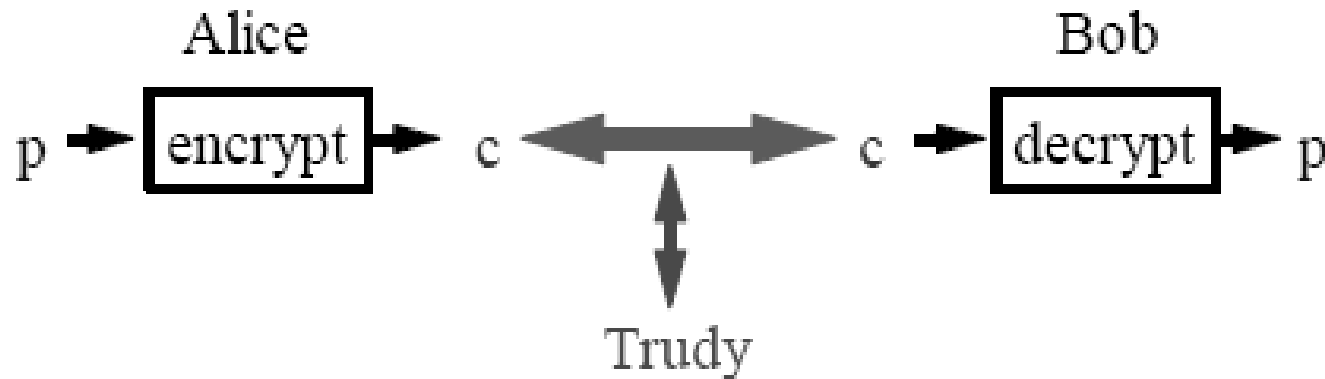
# **Cryptography**

**Symmetric cryptography**

# Reference scenario

---

- Alice needs to send a message to Bob
- The communication channel is insecure
  - Active attacks: modification
  - Passive attacks: eavesdropping
- Alice and Bob use a system to secure the communication
  - Cryptographic system



# Problems

---

- Which system to choose?
  - Cryptographic algorithms, e.g., encryption/decryption
- What is the complexity and security?
- How to agree on the keys used?

**How to ensure security of communication over insecure medium?**

# Cryptology

---

- Etymologically, **the art of secret writing**
- Cryptology = cryptography + cryptanalyse
  - Cryptography: secret writing
    - Study of mathematic techniques to enforce security properties
      - Confidentiality, integrity, authentication, non-repudiation
    - Converts data into unintelligible (random-looking) form
      - Must be reversible (can recover original data)
  - Cryptanalyse: **the art of revealing secret**
    - How to break a cryptographic system
- If cryptography is combined with compression:
  - What to do first?

# Cryptography vs. Steganography

---

- Steganography concerns **existence**
  - *covered writing*
  - hide existence of a message

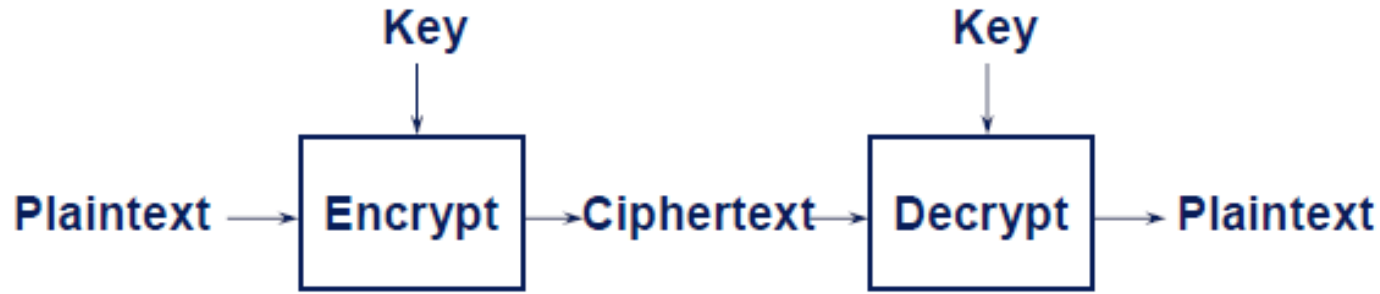
*A**pp**arently **n**eutral's **p**rotest **i**s **t**horoughly **d**iscounted **a**nd **i**gnored.  
**I**sman **h**ard **h**it. **B**lockade **i**ssue **a**ffects **p**retext **f**or **e**mbargo **o**n **b**y-  
**p**oducts, **e**jecting **s**uets **a**nd **v**egetable oils.*

**Pershing sails from NY June 1**

- Cryptography concerns content
  - *hidden writing*
  - hide meaning of a message

# Cryptographic system

---



- Plaintext: original message
- Ciphertext: coded message
- Cipher: algorithm transforming plaintext to ciphertext
- Key: info used in cipher known only to sender/receiver
- Encipher/encrypt: plaintext -> ciphertext
- Decipher/decrypt: ciphertext -> plaintext

# Cryptanalysis

---

- Objective: reveal the plaintext without knowing keys
- Difficulty depends on
  - Security of the encryption/decryption algorithms
  - Information disposed by the attacker
- **4 attack models**
  - Ciphertext only
  - Known plaintext
  - Chosen plaintext
  - Chosen ciphertext
  - Chosen text

# Attack models

---

- Ciphertext only
  - Attacker knows **only ciphertext**
  - Attacker intercepts some ciphertext
    - Breaking the system by analyzing intercepted ciphertext
  - Any algorithm vulnerable to this attack is completely insecure
- Known plaintext
  - Attacker has **some plaintext** and their **ciphertext**
- Chosen plaintext
  - Attacker can **choose arbitrary plaintext** to be encrypted and obtain the corresponding ciphertext
  - **Standard security level: resistance to chosen plaintext attack**
- Chosen ciphertext
  - Attacker can choose arbitrary **ciphertext** to be encrypted and obtain the corresponding **plaintext**
- Chosen text: chosen plaintext + chosen ciphertext



# Perfect vs. Computational Security

---

- Perfectly secure cipher
  - No matter how computer power is powerful, it cannot break the cipher
  - Ciphertext does not reveal any information about plaintext
    - Resilience against ciphertext only attack

And

- Plaintext does not reveal any information about ciphertext
    - Resilience against known/chosen plaintext attack

- Computationally secure cipher
  - The cost of breaking the cipher  $>$  the value of the encrypted info

And/or

- The time required to break the cipher  $>$  the useful lifetime of the info
- Ad hoc security (heuristic security)

# Secret Keys vs. Secret Algorithms

---

- Keep algorithms secret
  - Secret algorithms -> better security
    - Hard to keep secret if used widely
  - Every NATO and Warsaw Pact algorithm during Cold War
  - All digital cellular encryption
  - HD DVD, Blu-Ray
- Publish algorithms
  - Security depends on the secrecy of the keys
  - Public examination helps to find flaws
- Kerckhoffs principle
  - A cryptographic system should be secure even if everything about the system, **except the key**, is public knowledge.
  - Reformulated by Claude Shannon as **the enemy knows the system**
    - Shannon's maxim



# Kerckhoffs

---

- Auguste Kerckhoffs (1835-1903): Dutch-French cryptographer
  - Jean-Guillaume-Hubert-Victor-François-Alexandre-Auguste Kerckhoffs von Nieuwenhof
  - Dutch linguist of French nationality, got his Ph.D. in Germany, worked as professor in HEC Paris
- Major contribution: a practical, experience-based approach, including six design principles for military ciphers
  - **The system must be practically, if not mathematically, indecipherable.**
  - **It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience.**
  - Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents.
  - It must be applicable to telegraphic correspondence.
  - Apparatus and documents must be portable, and its usage and function must not require the concourse of several people.
  - It is necessary, given the circumstances that command its application, that the system be easy to use, requiring neither mental strain nor the knowledge of a long series of rules to observe.

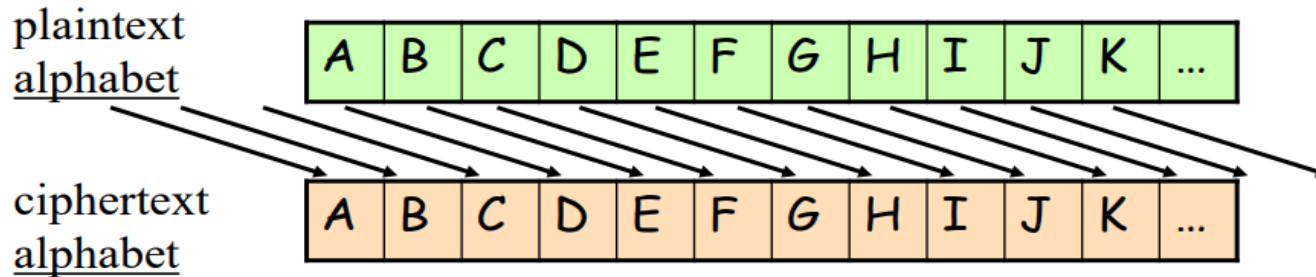
# History of cryptographic systems

---

- First generation: classic ciphers, paper & ink based
  - Substitution cipher
    - Mono-alphabetic
      - Caesar
    - Poly-alphabetic
      - Vigenere
  - Permutation cipher
- Second generation: use cryptographic engines
  - Mechanic and electro-mechanic
    - Enigma, Hagelin C38
- Third generation: modern cryptography
  - Based on advanced math/TCS
  - Information-theoretic security
  - Computational security

# Caesar cipher

- Substitution, mono-alphabetic cipher



Plaintext :    **THIS IS THE CAESAR CIPHER**  
Ciphertext :   **WKLV LV WKH FDHVDU FLSKHU**  
Key: k (=3)

Caesar has never changed the key!

- Test: find the plaintext of the ciphertext below
  - VHFXULWBDQGSULYDFB**

# Caesar cipher: formalism

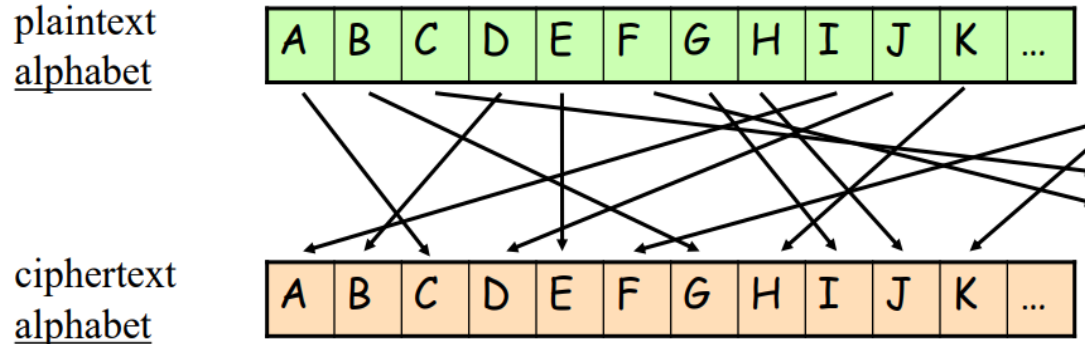
---

- $P=C=K=Z_{26}$
- Encryption:  $e_k(x) = x+k \bmod 26$
- Decryption: and  $d_k(y) = y-k \bmod 26$
- $x \in P, y \in C, k \in K$
  
- Dominates the art of secret writing in the first millennium A.D.
  - Julius Caesar ~60 BC
  - Phonetic substitution used in India even earlier
- Thought to be unbreakable

# Generalized Caesar cipher

---

- Substitution, mono-alphabetic cipher
  - Randomly map one letter in plaintext to ciphertext



- # combination
  - $26! = 2^{88}$
- Key length: 88 bits
  - Need to specify which permutation

# Generalized Caesar cipher: attack

- Vulnerable to known plaintext attack
- Ciphertext only
  - Attacker disposes ciphertext
- Technique: frequency analysis

UXGPOGZCFJZJTFADADAJEJNDZMZHBBGZGGKQGVVGXCDIWGX

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
3	2	2	4	1	2	8	1	1	4	1	0	1	1	1	1	1	0	0	1	1	2	1	3	0	5

A $\approx$ 8.2%	H $\approx$ 6.1%	O $\approx$ 7.5%	V $\approx$ 1.0%
B $\approx$ 1.5%	I $\approx$ 7.0%	P $\approx$ 1.9%	W $\approx$ 2.4%
C $\approx$ 2.8%	J $\approx$ 0.2%	Q $\approx$ 0.1%	X $\approx$ 0.2%
D $\approx$ 4.3%	K $\approx$ 0.8%	R $\approx$ 6.0%	Y $\approx$ 2.0%
E $\approx$ 12.7%	L $\approx$ 4.0%	S $\approx$ 6.3%	Z $\approx$ 0.1%
F $\approx$ 2.2%	M $\approx$ 2.4%	T $\approx$ 9.1%	
G $\approx$ 2.0%	N $\approx$ 6.7%	U $\approx$ 2.8%	

FREQUENCY  
ANALYSIS IS  
AMAZING NOW  
WE NEED  
BETTER CIPHER



# Frequency analysis

---

- Earliest known in a book by the ninth-century scientist al-Kindi
- Rediscovered or introduced in Europe during Renaissance
- Frequency analysis made substitution cipher insecure
- Improvements
  - Using numbers as ciphertext alphabet, some representing nothing are inserted randomly
  - Deliberately misspell words
    - **Thys haz thi ifekkt off diztaughting thi ballans off frikwenseas**
  - Homophonic substitution cipher
    - Each letter replaced by a variety of substitutes
- Make frequency analysis more difficult, but not impossible
  - **Mono-alphabetic cipher is still vulnerable to frequency analysis**

# From mono- to poly-alphabetic cipher

---

- Weaknesses of mono-alphabetic cipher
  - Each ciphertext letter corresponds to only one plaintext letter
  - Frequency attack easy to mount
- Idea for a stronger cipher
  - Use more than one cipher alphabet, and switch between them when encrypting different letters
  - 1460's by Alberti: The Alberti Cipher disk
  - Plaintext on inner, ciphertext on outer
  - Rotated to a new position periodically



# Leon Battista Alberti

- 1404-1472, Italian Renaissance humanist author, artist, **architect**, **poet**, priest, **linguist**, philosopher and *cryptographer*
  - **The Renaissance man**



# Vigenere Cipher

---

- Substitution, poly-alphabetic cipher
  - Use multiple mono-alphabetic substitution rules
- Example: key = (3 1 5)
  - Replace first letter in plaintext by letter+3, second by 1, third letter by 5
  - Repeat the above cycle

plaintext message

B	A	N	D	B	A	D
---	---	---	---	---	---	---

shift amount

3	1	5	3	1	5	3
---	---	---	---	---	---	---

ciphertext message

E	B	S	G	C	F	G
---	---	---	---	---	---	---



# Blaise de Vigenère

- 1523-1596, French diplomat, cryptographer, translator and alchemist



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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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


# Vigenere Cipher: formalism

---

- Encryption:
  - $e_k(p_1, \dots p_m) = (p_1+k_1, \dots p_m+k_m) \pmod{26}$
- Decryption:
  - $d_k(c_1, \dots c_m) = (c_1-k_1, \dots c_m-k_m) \pmod{26}$
- Can be regarded as mutiple Caesar ciphers
  - But masks frequency
  - Frenquency analysis more difficult, but still possible

# Vigenere Cipher: cryptanalysis

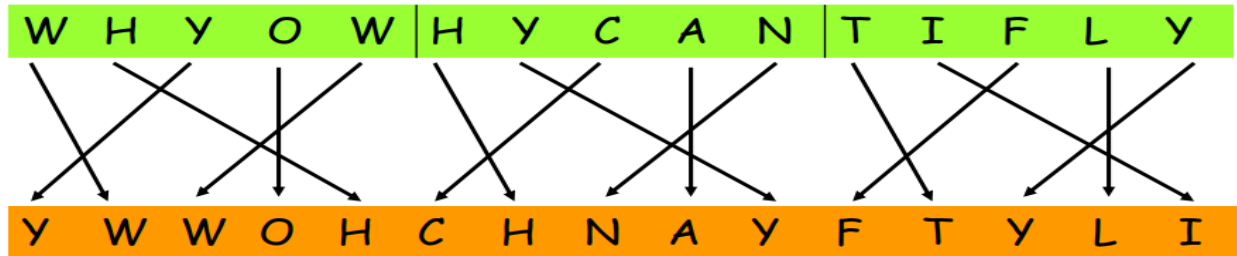
---

- Find the length of the key  $m$
  - Divide ciphertext into  $m$  Caesar ciphertexts
  - Method 1: exhaustive search
  - Method 2: Kasiski test
    - First described in 1863 by Friedrich Kasiski
      - Key:  
K I N G K I N G K I N G K I N G K I N G K I N G
      - Plaintext:  
t h e s u n a n d t h e m a n i n t h e m o o n
      - Ciphertext:  
D P R Y E V N T N B U K W I A O X B U K W W B T
-   
8 positions
- Frequency analysis made substitution cipher insecure

# Permutation cipher

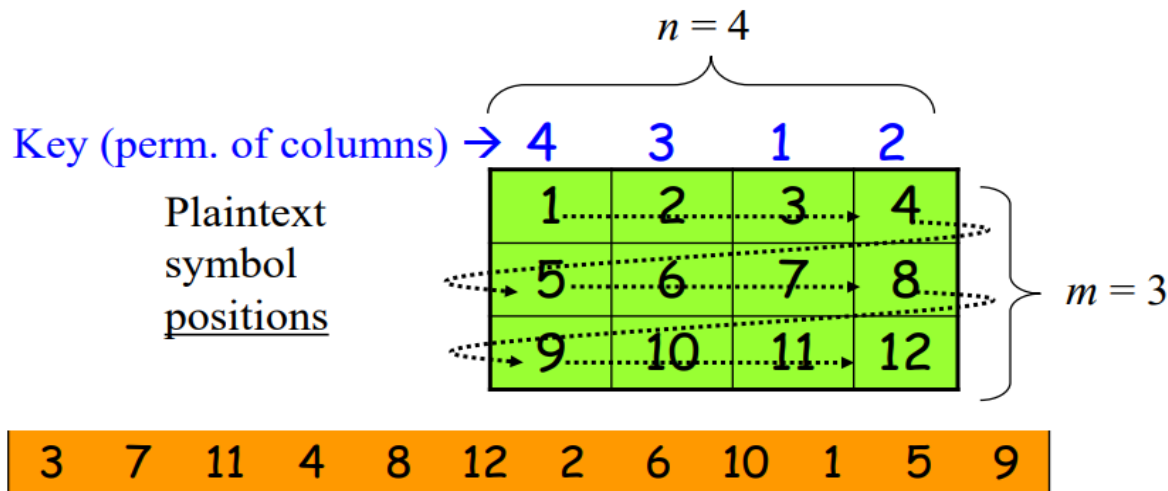
## 1D permutation

- Permute each 5-letter block in plaintext according  $\langle +1, +3, -2, 0, -2 \rangle$



## 2D permutation

- Arrange plaintext in  $n \times m$  blocks
- Permute columns in a block according to key





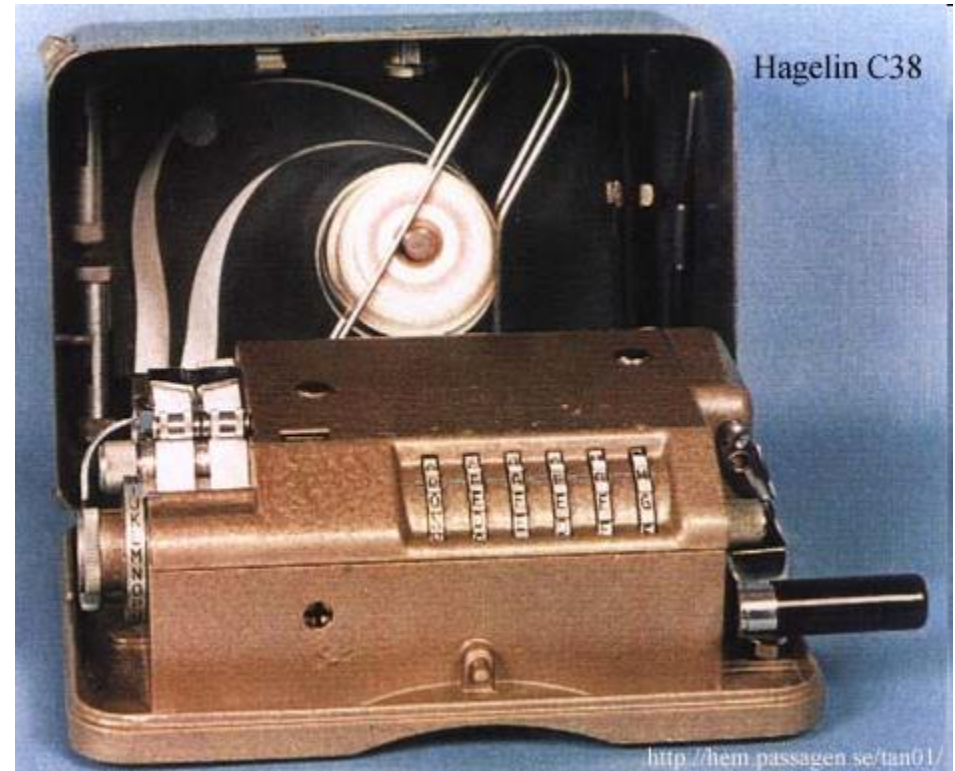
# Permutation example: Scytale

---

- First mentioned by a poet in Greece, 7th century BC
  - Key: diameter of Scytale



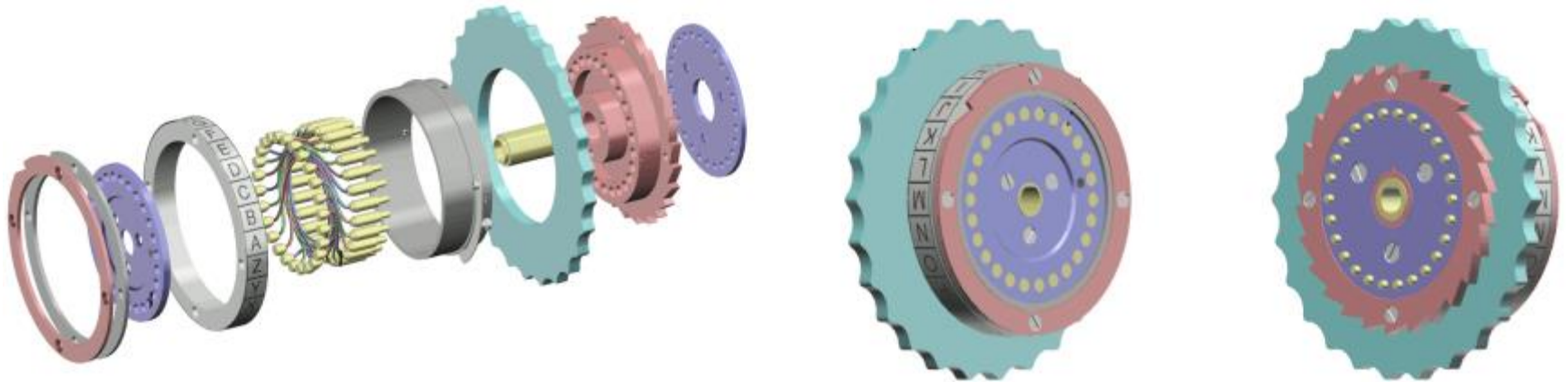
# 2G: (electro-)mechanic cipher



# Enigma

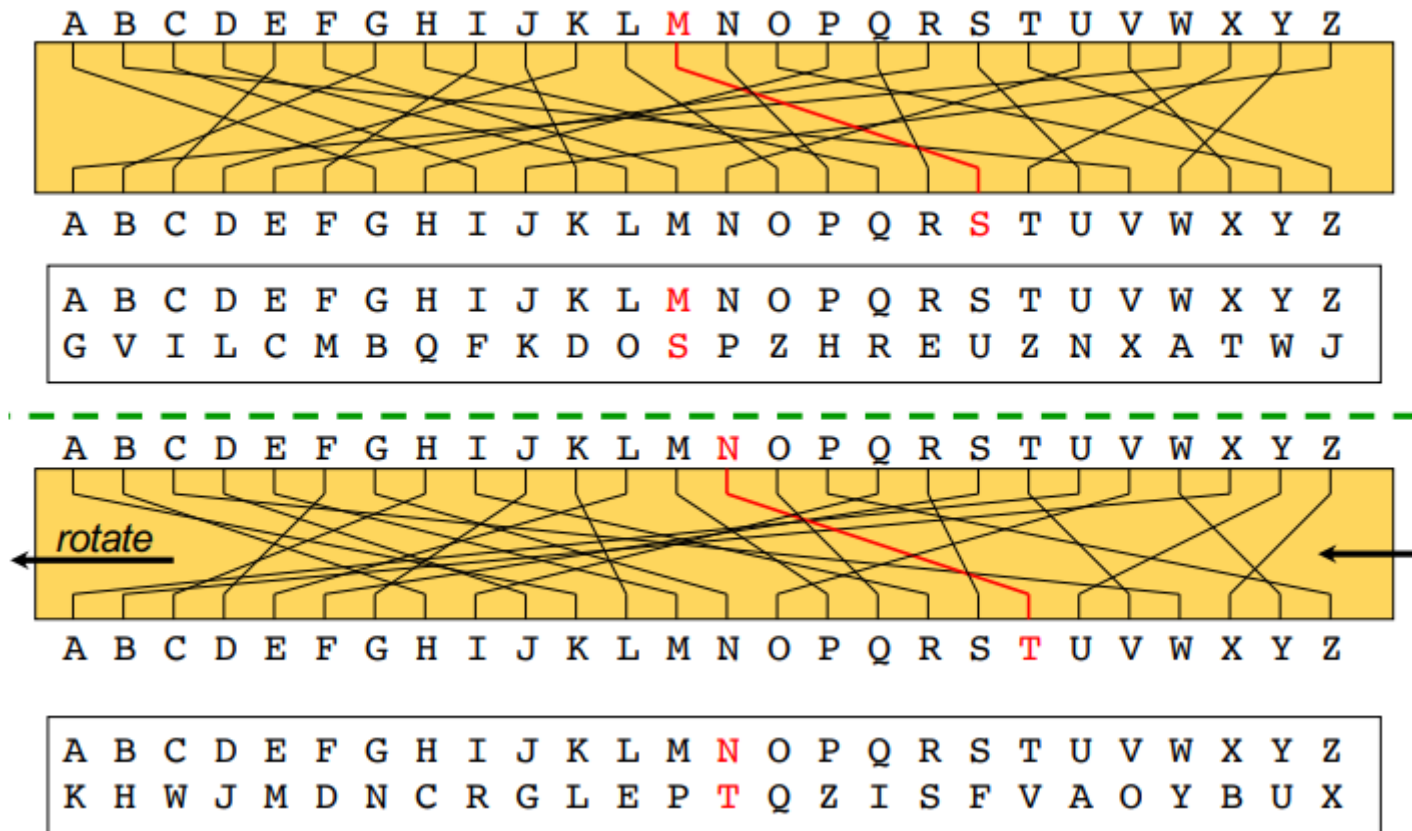
---

- Widely used in Germany during WWII
  - Rotor machine: enhanced Vigenère cipher
- Rotor machine:
  - Multiple rotating cylinders (rotors)
  - Each rotor implements a substitution cipher
  - Output of each rotor is fed into the next rotor



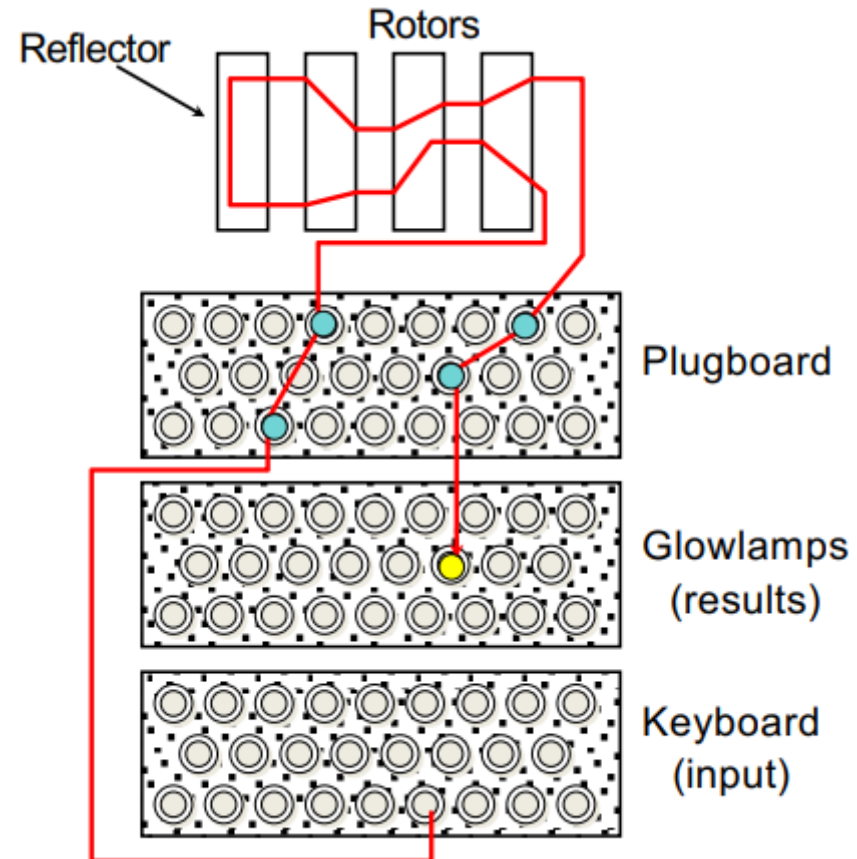
# Single cylinder

- After a letter typed, rotates one position
  - Polyalphabetic substitution cipher with period 26



# Multiple cylinders

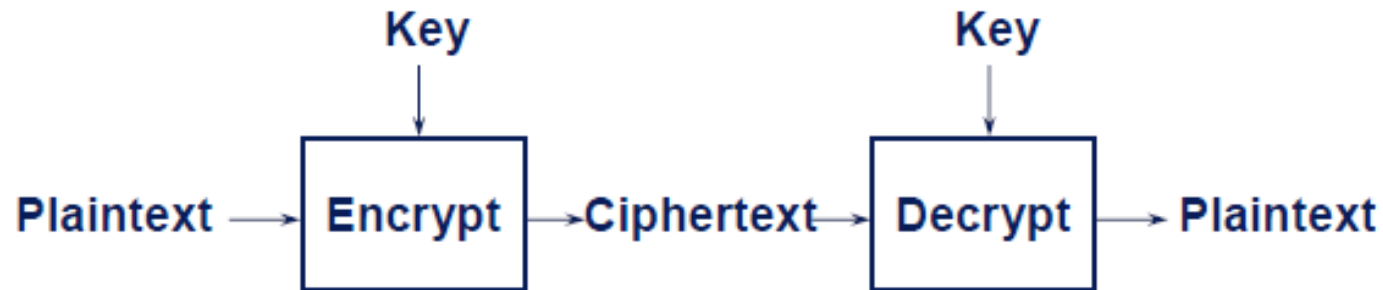
- Output of cylinder  $i$ 
  - $\rightarrow$  input of cylinder  $i+1$
  - Cylinder  $i+1$  advances 1 position after a period of cylinder  $i$
- Period of 3-cylinder rotor
  - $26^3 = 17576$
- Enigma
  - Input permuted before entering rotor
  - Output of last rotor reflected back
    - Make encryption symmetric
  - Initial rotor setting is secret
    - Depends on data
  - Broken by group at Bletchley Park
    - Alan Turing: Bombe, Colossus



# Third generation: modern ciphers

---

- Based on advanced math/TCS
- Symmetric (secret) key cryptography
  - Single key for both encryption and decryption
- Asymmetric (public) key cryptography
  - A pair of keys: (public, private)
  - One for encryption, the other for decryption
- Hash Algorithm

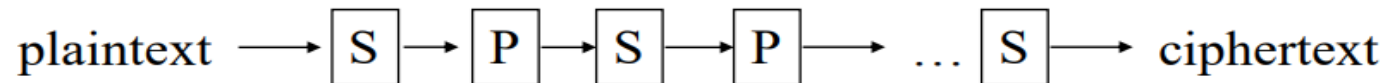


# Symmetric key crypto-systems

---



- A single key: secret key
- Technique: multiple applications of interleaved substitutions and permutations

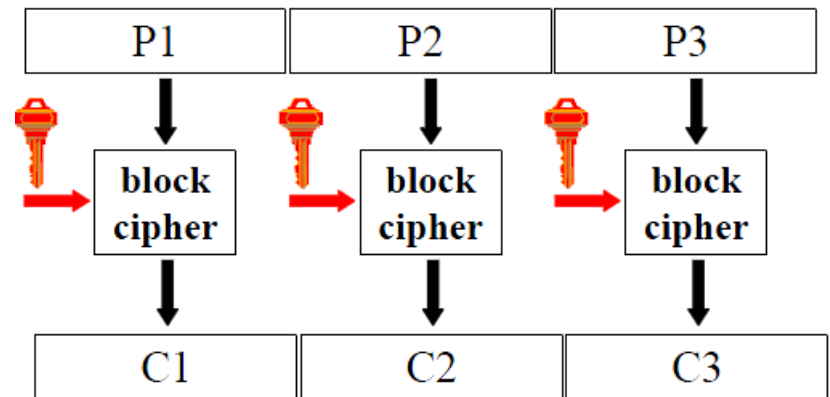
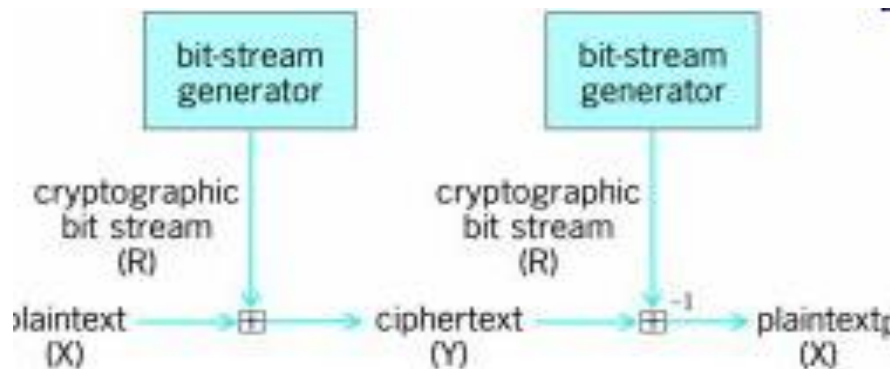


- ☺ : fast encryption and decryption
- ☹ : key exchange
- Usage : confidentiality service



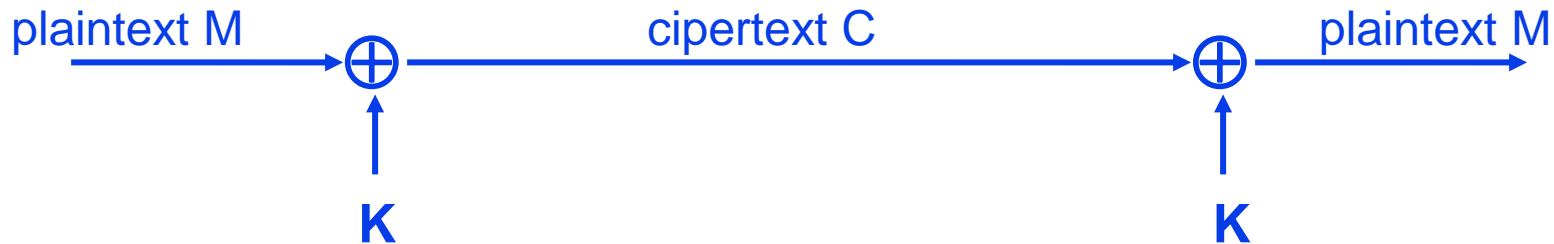
# Stream vs. block ciphers

- Stream cipher
  - Encrypt plaintext message one symbol (e.g., 1 bit) at a time
- Block cipher
  - Divide plaintext into blocks (e.g., 64 bits), treats block as a unit to process





# One-time-pad: stream cipher



- Created by Gilbert Vernam 1917
  - 1890-1960: US engineer, inventor of XOR
- M, C, K: same length
  - Encryption:  $C = M \oplus K$
  - Decryption:  $M' = C \oplus K = M \oplus K \oplus K = M$
  - Extended Vigenere cipher
- ☺ the only algorithm theoretically sure, if:
  - K is perfectly random
  - K is used only once:
    - if K is reused ?
- ☺ C contains no information about M
- ☹ generation and transportation of K



a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

# One-time-pad

---



- Teletypewriter used in US/Soviet Union hotline kept at US National Cryptological Museum

# Names connected with OTP

---

- Co-inventors of OTP
  - Joseph Mauborgne (1881-1971) became a Major General in US Army
  - Gilbert Sandford Vernam (1890 -1960) was AT&T Bell Labs engineer
- Security of OTP
  - Claude Elwood Shannon (1916 -2001), American mathematician and electronic engineer, father of information theory



# One-time-pad: test

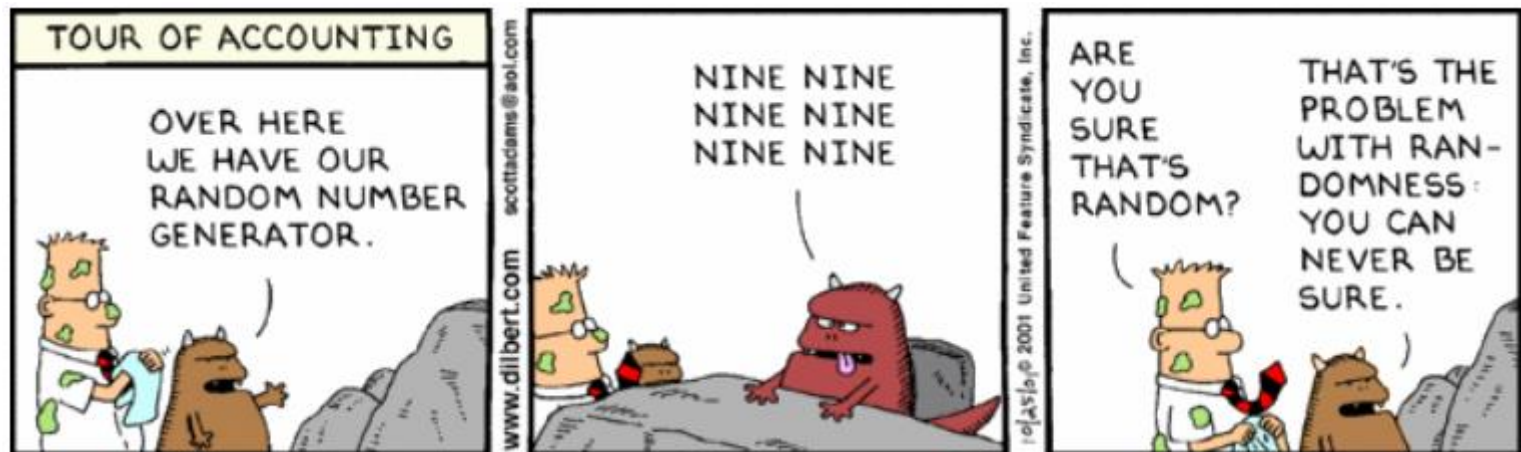
---

- Alice wants to send a message  $M$  to Bob
- Difficult to agree/exchange the secret key  $K$
- They think of the protocole below
  - Alice picks a random number  $K_1$  and sends  $S_1 = M \oplus K_1$  to Bob
  - Bob picks a random number  $K_2$  and sends  $S_2 = S_1 \oplus K_2$  to Alice
  - Alice sends  $S_3 = S_2 \oplus K_1$  to Bob
  - Bob obtains  $M$  :  $M = S_3 \oplus K_2$
- Is it secure? If not, give an attack

# One-time-pad: random numbers

---

- Keys need to be random
- Random number generation is challenging
  - External randomness: noise
  - Pseudo-random generators
- John vonNeumann
  - *Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.*



# OTP: proof of security (secrecy)

---

- OTP is the only secure protocol proved so far
- What is security (secrecy), formally?
  - Intuition: attacker cannot obtain any information about M
  - Perfect secrecy
- A system is perfectly secret if for any m, c
  - $\Pr(M=m) = \Pr(M=m|C=c)$
- OTP is perfectly secret
  - $\Pr(C=c|M=m) = \Pr(K=m \oplus c|M=m) = 1/2^l$
  - $\Pr(C=c|M=m) * \Pr(M=m) = \Pr(M=m|C=c) * \Pr(C=c)$

# DES: Data Encryption Standard

---

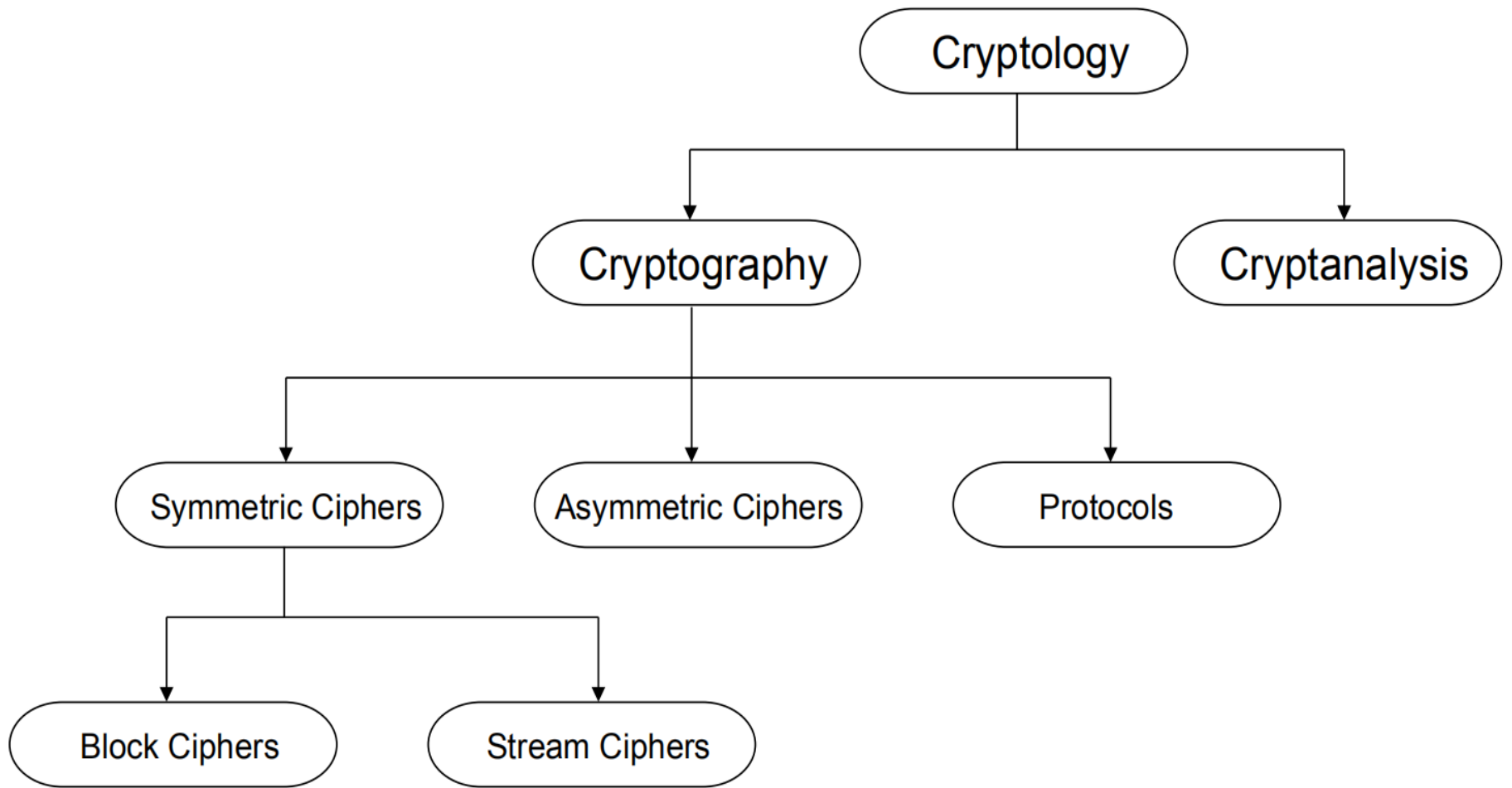
- Developed by IBM influenced by National Security Agency
- Standardized in 1977
- Block size: 64 bits
- Key size: 56 bits
- One of the most popular block cipher
- The best studied symmetric algorithm
- Nowadays considered insecure due to key length
  - But: 3DES yields very secure cipher, still widely used today
- Replaced by the Advanced Encryption Standard (AES) in 2000

# Design criteria

---

- High level of security
- Security must reside in key, not algorithm
- Not patented
- Efficient to implement in hardware
- Slow to execute in software





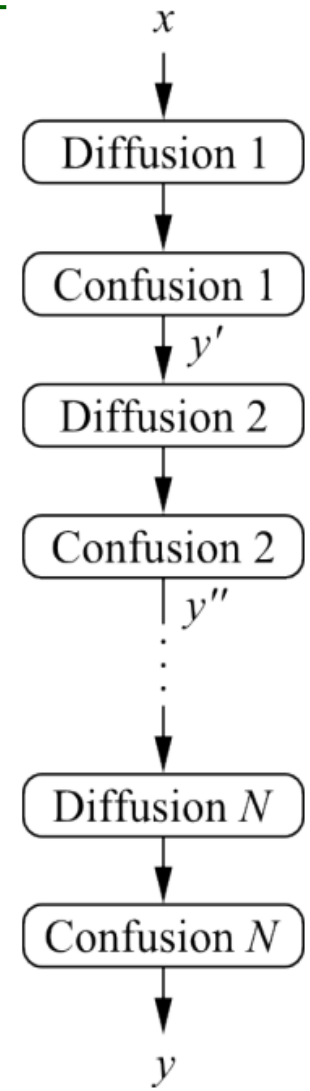
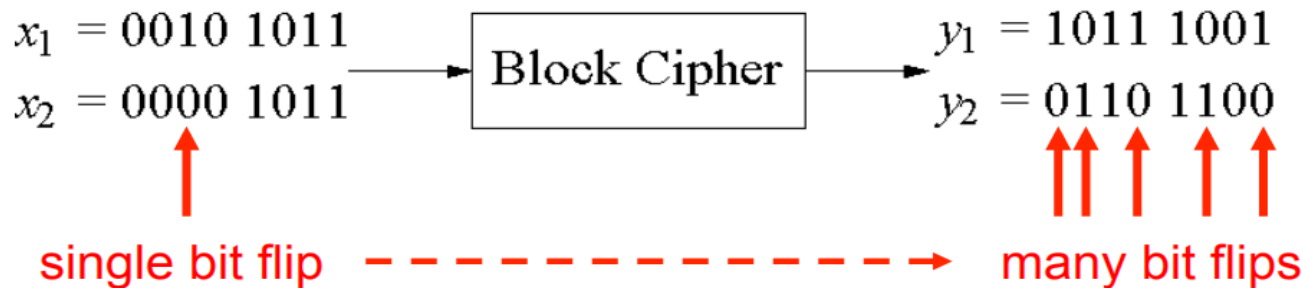
# Design rationale Confusion & Diffusion

---

- Claude Shannon:
  - There are two primitives with which strong encryption algorithms can be built: **Confusion and Diffusion**
- Confusion
  - Make relationship between (plaintext, key) and ciphertext output as complex (non-linear) as possible
  - Achieved by substitution
- Diffusion:
  - Spread influence of each input bit across many output bits
  - Achieved by permutation
- Confusion or diffusion alone is not enough
  - Concatenate confusion and diffusion
  - **Product ciphers**

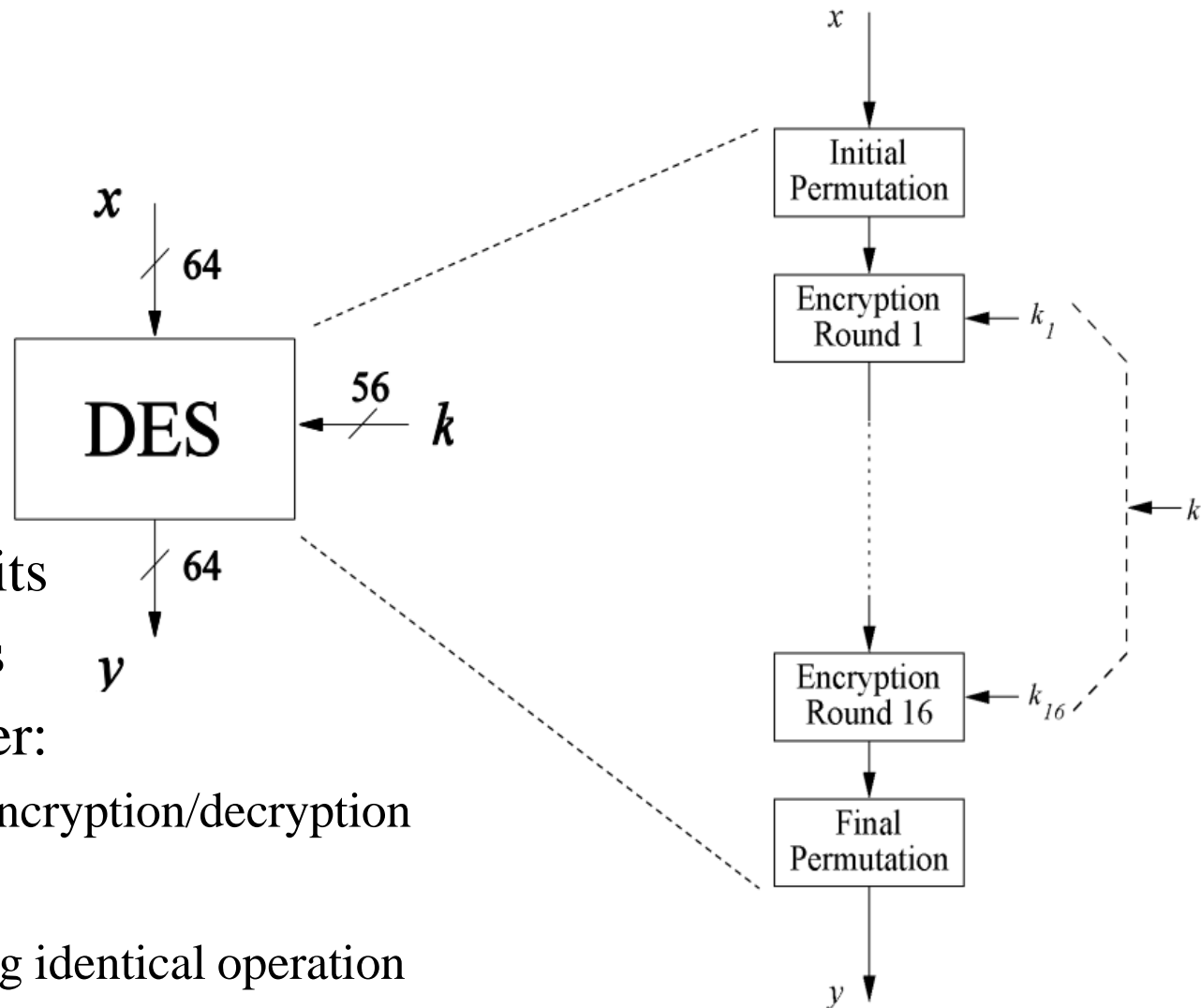
# Product ciphers

- Most of today's block ciphers are product ciphers
  - Consist of **rounds**
  - Changing 1 bit in plaintext results on average changing half of bits in ciphertext



# DES algorithm overview

- Block size: 64 bits
- Key size: 56 bits
- Symmetric cipher:
  - Same key for encryption/decryption
- 16 rounds
  - each performing identical operation
- Different sub-key in each round
  - derived from main key

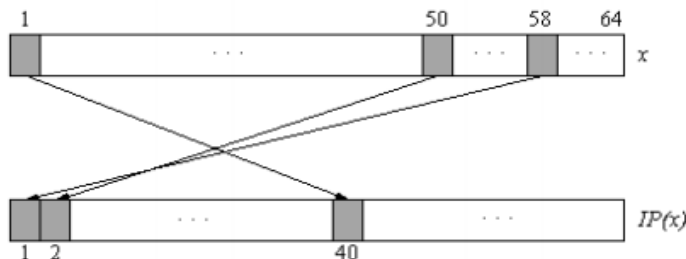


# Initial and Final Permutation

- Bitwise Permutations.
- Inverse operations.
- Described by tables  $IP$  and  $IP^{-1}$
- No security value: hardware consideration

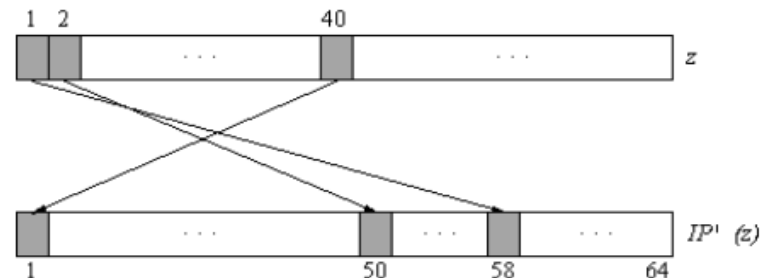
Initial Permutation

$IP$										
58	50	42	34	26	18	10	2			
60	52	44	36	28	20	12	4			
62	54	46	38	30	22	14	6			
64	56	48	40	32	24	16	8			
57	49	41	33	25	17	9	1			
59	51	43	35	27	19	11	3			
61	53	45	37	29	21	13	5			
63	55	47	39	31	23	15	7			



Final Permutation

$IP^{-1}$										
40	8	48	16	56	24	64	32			
39	7	47	15	55	23	63	31			
38	6	46	14	54	22	62	30			
37	5	45	13	53	21	61	29			
36	4	44	12	52	20	60	28			
35	3	43	11	51	19	59	27			
34	2	42	10	50	18	58	26			
33	1	41	9	49	17	57	25			

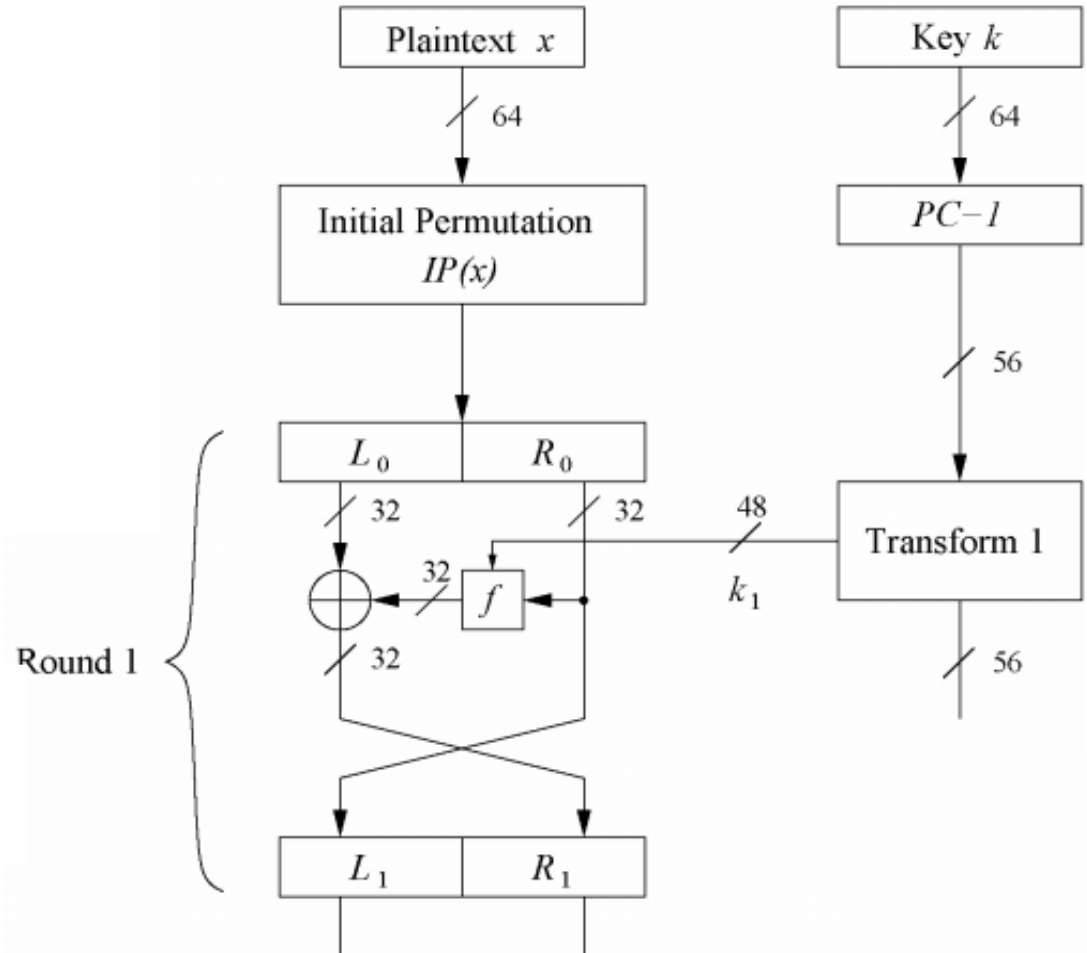


# Feistel Network

- An important template for block ciphers
- Can be used for both encryption and decryption
  - **Just inverse arrows**

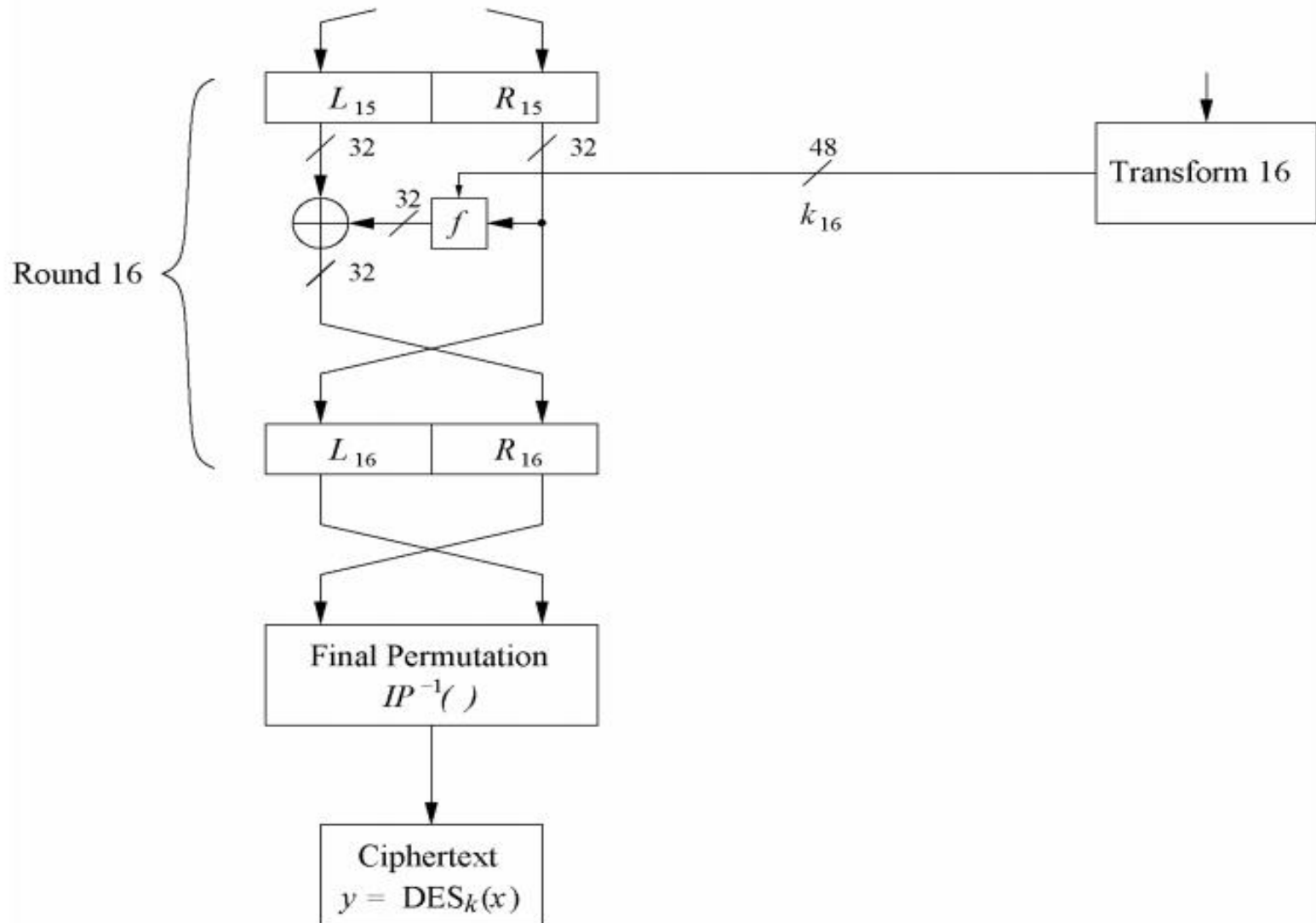
$$L_i = R_{i-1},$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, k_i)$$



# Feistel Network

- L and R swapped again at the end of the cipher

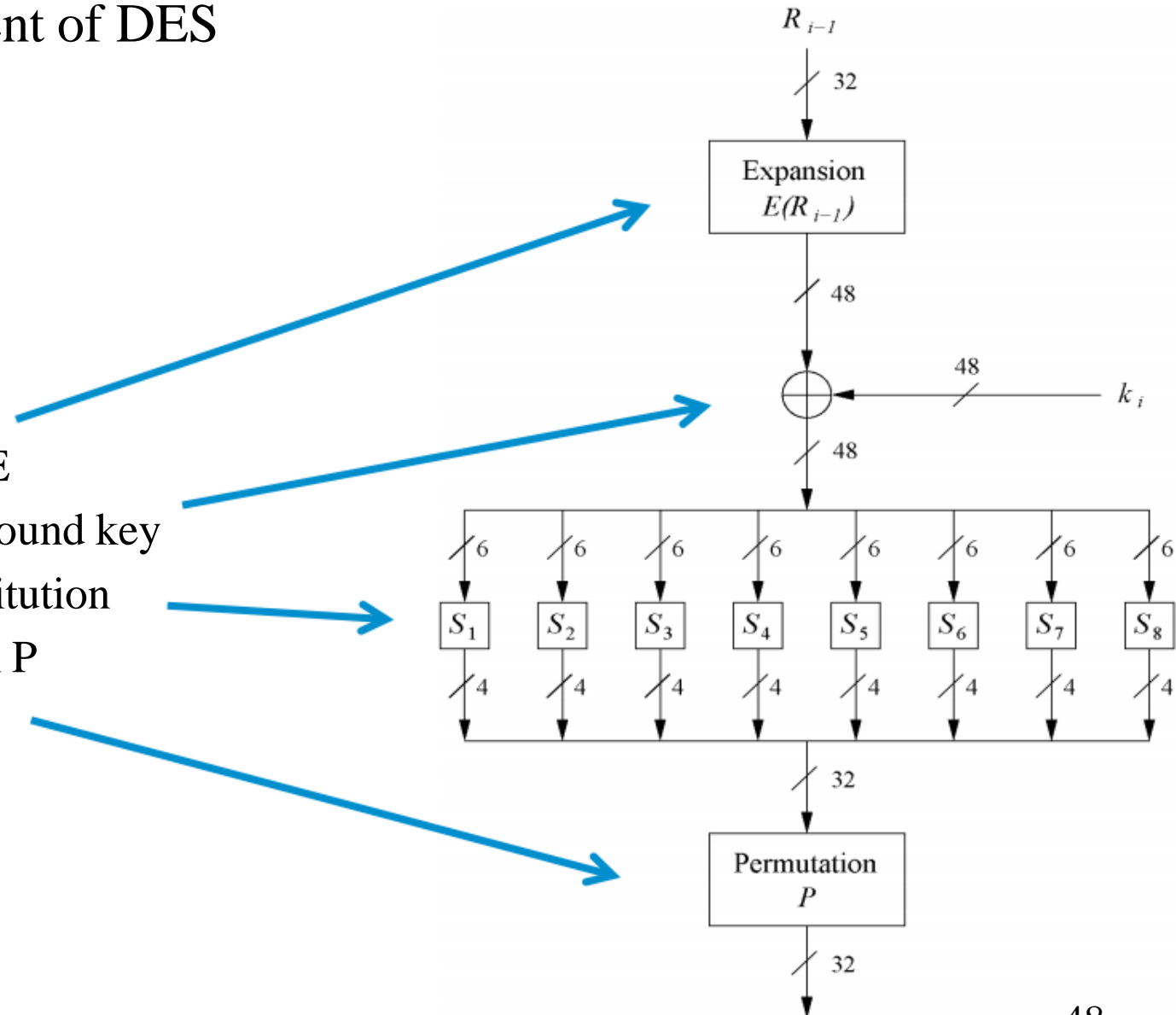


# The Scrambling function $f$

- Key component of DES

- 4 Steps:

- Expansion E
- XOR with round key
- S-box substitution
- Permutation P

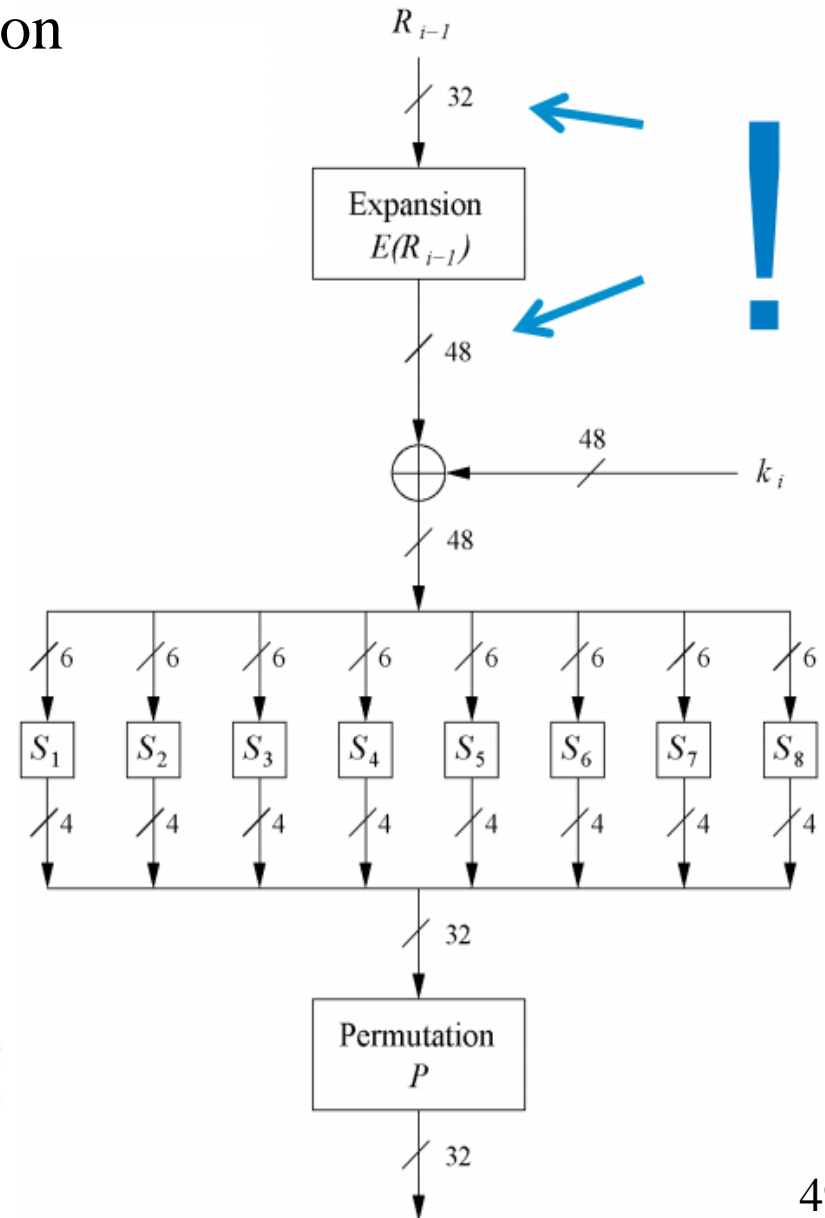
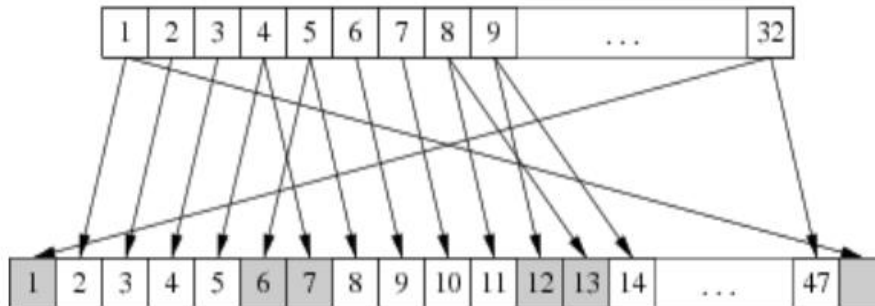




# The Expansion Function E

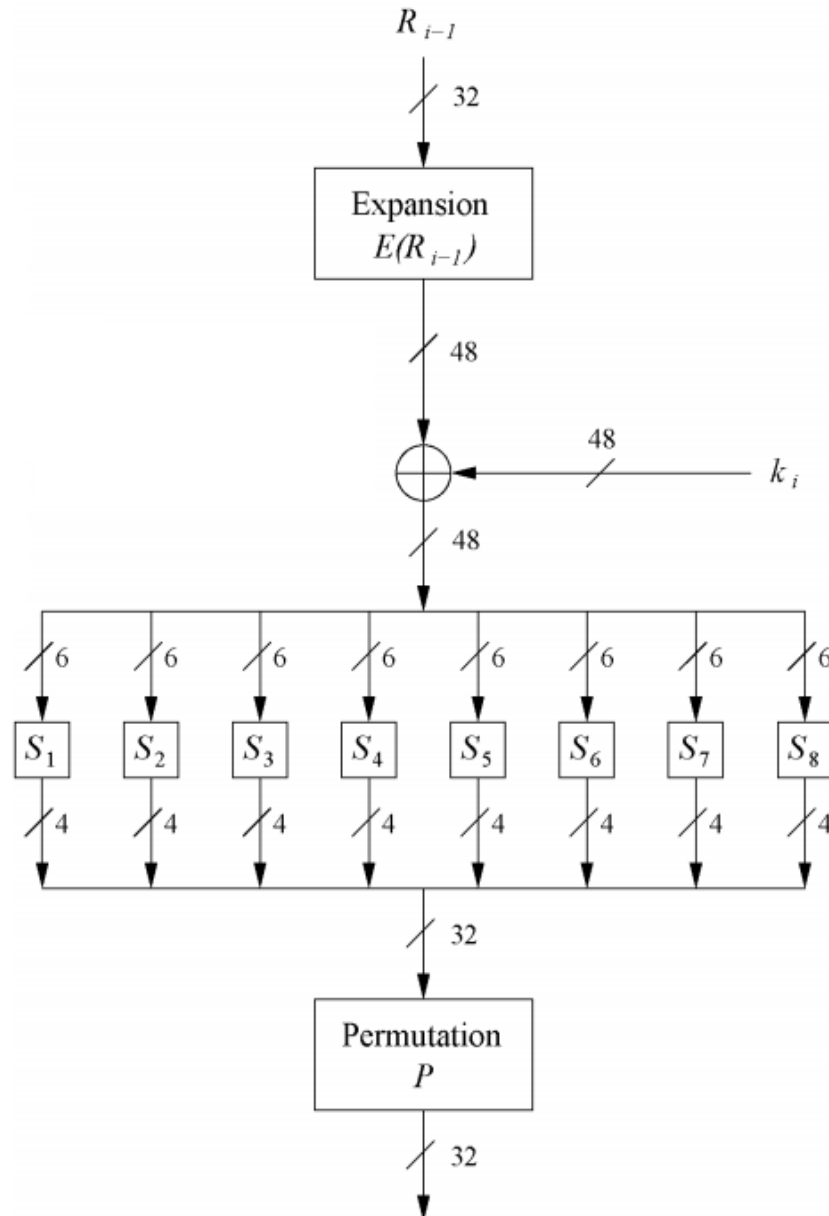
- Main purpose: increases diffusion

$E$										
32	1	2	3	4	5					
4	5	6	7	8	9					
8	9	10	11	12	13					
12	13	14	15	16	17					
16	17	18	19	20	21					
20	21	22	23	24	25					
24	25	26	27	28	29					
28	29	30	31	32	1					



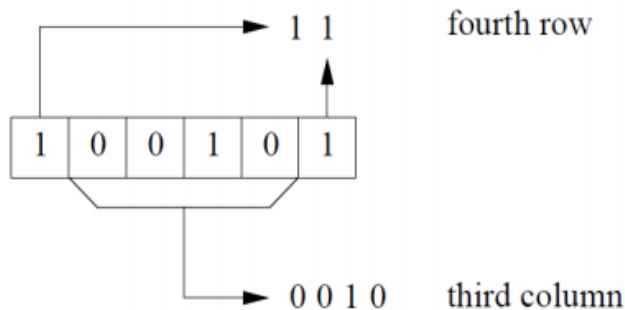
# XOR with Round Key

---

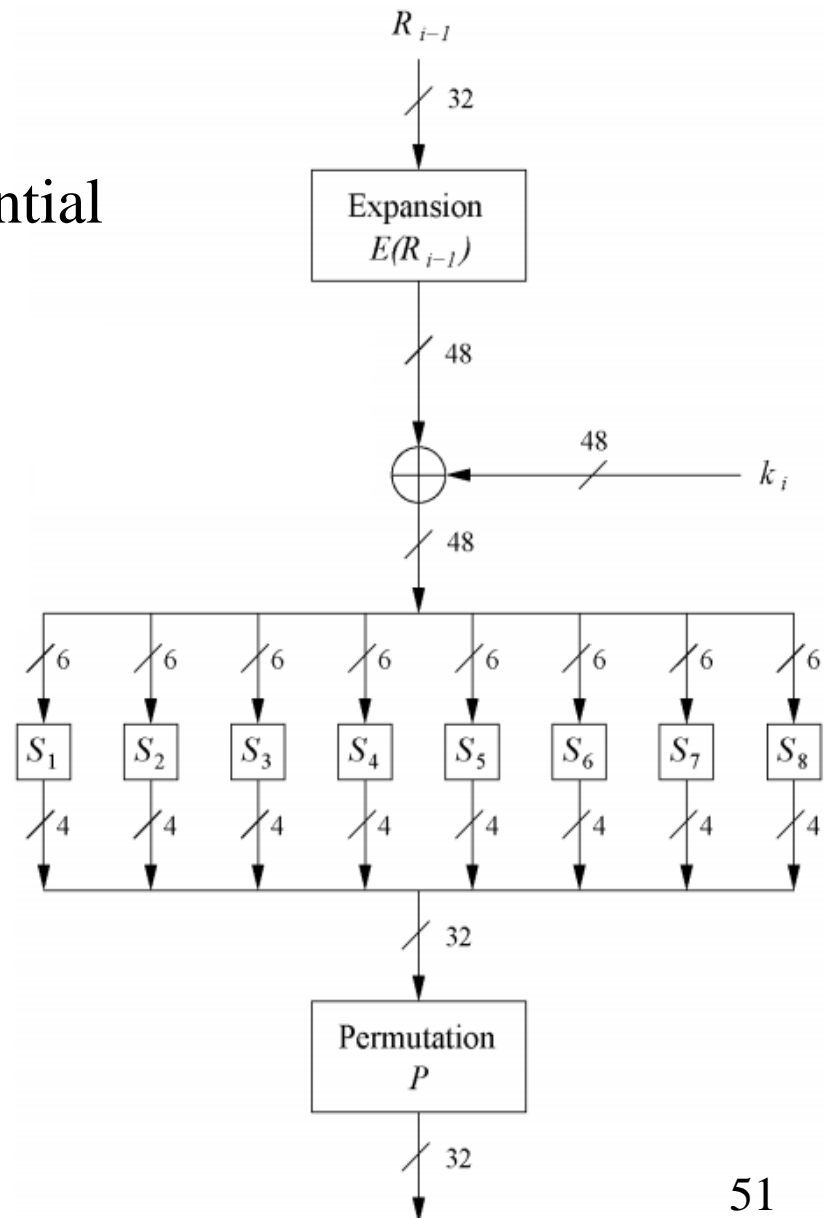


# The S-Box

- 8 substitution tables.
- 6 bits of input, 4 bits of output.
- Non-linear and resistant to differential cryptanalysis
- Crucial element for DES security!
  - The only non-linear part of DES



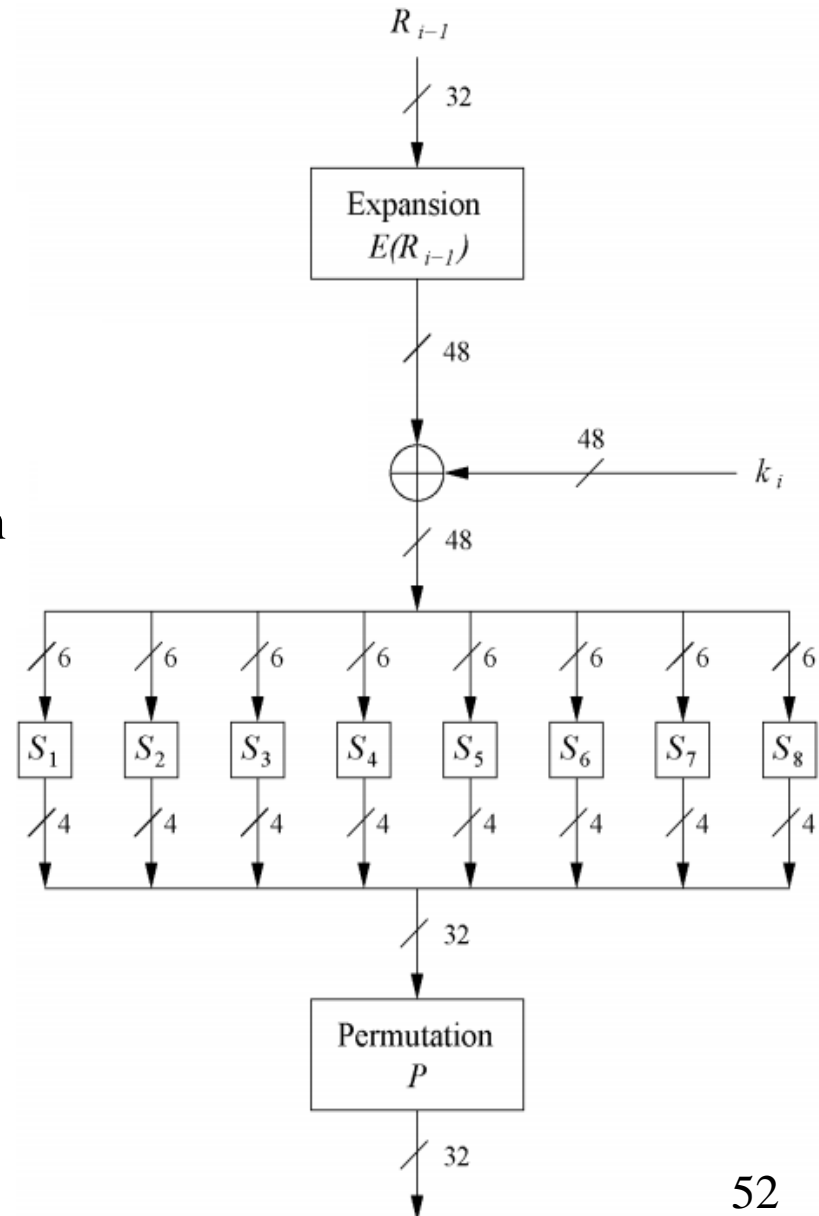
$S_1$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



# The permutation P

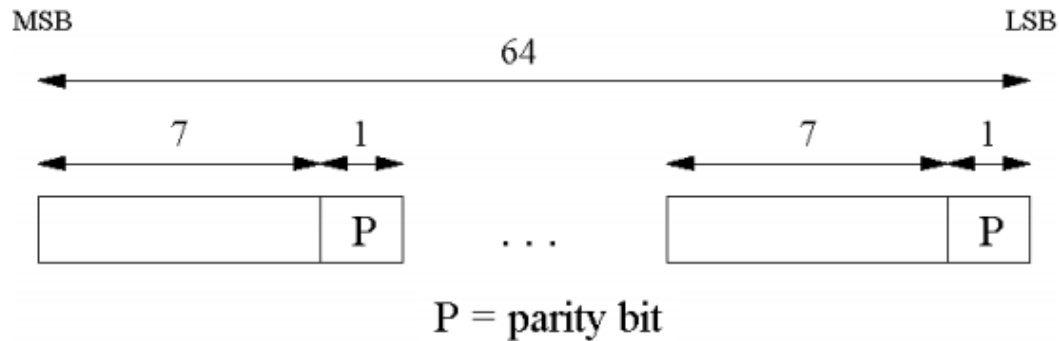
- Bitwise permutation
- Introduces diffusion
- Output of 1 S-Box affect several S-Boxes in next round
- Diffusion by E, S-Boxes and P guarantees that
  - after Round 5 every bit is a function of each key bit and plaintext bit.

$P$							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25



# Sub-key generation

- 16 round keys (subkeys)  $k_i$ 
  - 48 bits each
  - derived from original 56-bit key
- Input key size of the DES is 64 bit
  - 56 bit key + 8 bit parity



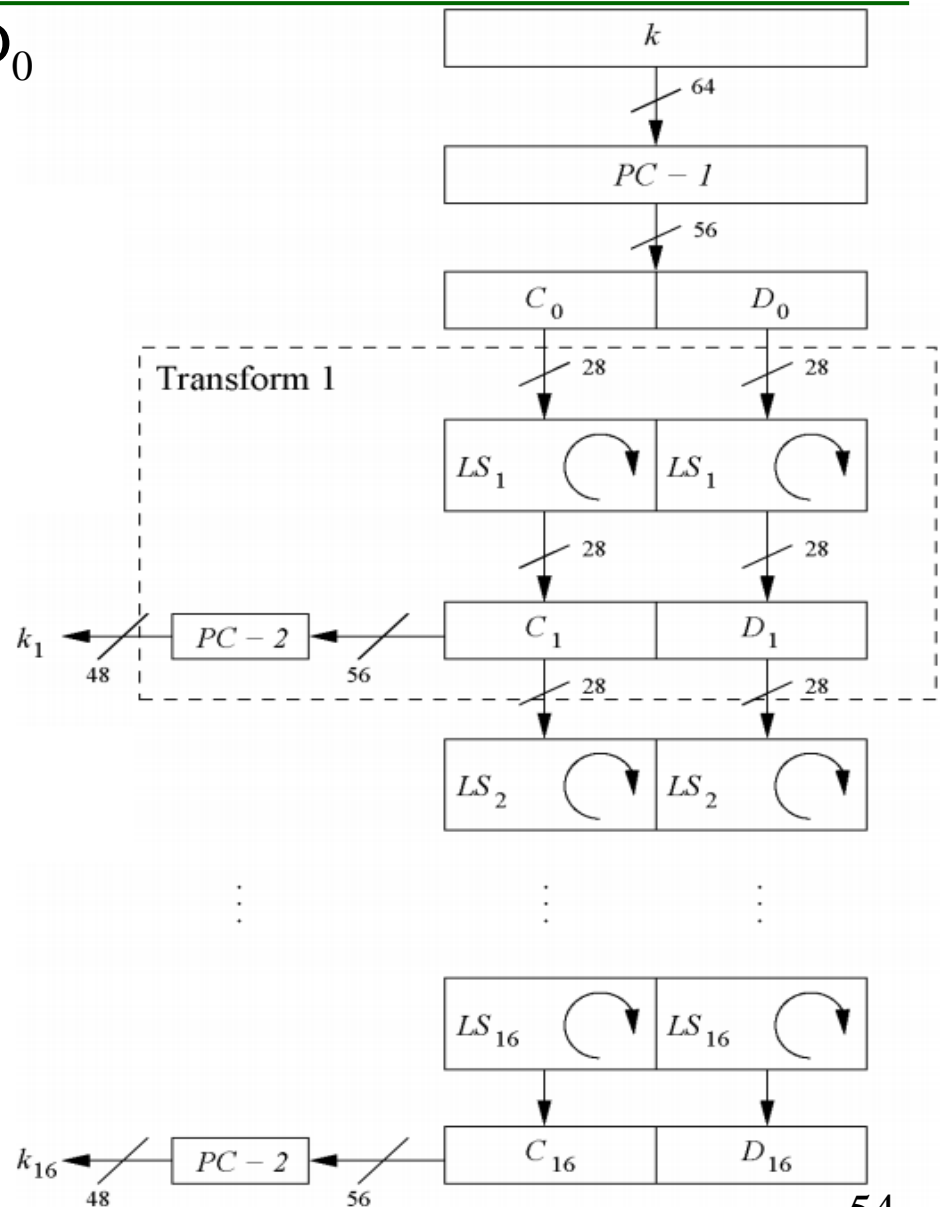
- Parity bits removed in permutation PC-1

<i>PC - 1</i>							
57	49	41	33	25	17	9	1
58	50	42	34	26	18	10	2
59	51	43	35	27	19	11	3
60	52	44	36	63	55	47	39
31	23	15	7	62	54	46	38
30	22	14	6	61	53	45	37
29	21	13	5	28	20	12	4

# Sub-key generation

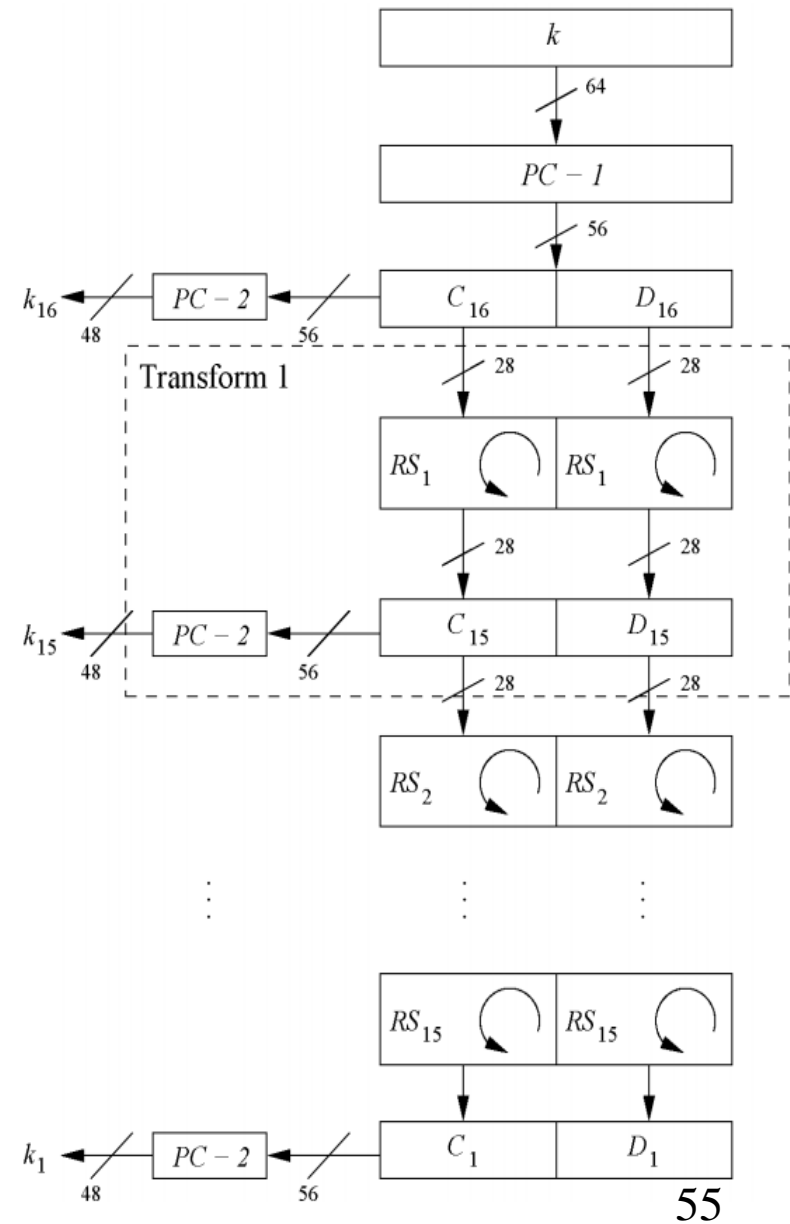
- Split key into 28-bit  $C_0$  and  $D_0$
- Rounds  $i=1, 2, 9, 16$ ,
  - $C_i$  and  $D_i$  rotated left by 1 bit
- Other rounds
  - rotated left by 2 bits
  - all together 28 bits
  - $C_{16}=C_0$  and  $D_{16}=D_0$
- Round key  $K_i$ 
  - permuted subset of  $C_i$  and  $D_i$

$PC - 2$							
14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32



# Decryption

- Feistel ciphers
  - only keyschedule is modified for decryption
- Generate same round keys in reverse order



# Implementation

---

- Operations
  - Permutation
  - Swapping
  - Substitution (S-box, table lookup)
  - Bit discard
  - Bit replication
  - Circular shift
  - XOR
- Hard to implement?
  - Hardware: easy
  - Software: hard



# Avalanche Effect

- DES has a strong avalanche effect
  - Small change in plaintext/key leads to big change in ciphertext

Round		$\delta$	Round		$\delta$
	02468aceeca86420 12468aceeca86420	1	9	c11bfc09887fbc6c 99f911532eed7d94	32
1	3cf03c0fbad22845 3cf03c0fbad32845	1	10	887fbc6c600f7e8b 2eed7d94d0f23094	34
2	bad2284599e9b723 bad3284539a9b7a3	5	11	600f7e8bf596506e d0f23094455da9c4	37
3	99e9b7230bae3b9e 39a9b7a3171cb8b3	18	12	f596506e738538b8 455da9c47f6e3cf3	31
4	0bae3b9e42415649 171cb8b3ccaca55e	34	13	738538b8c6a62c4e 7f6e3cf34bc1a8d9	29
5	4241564918b3fa41 ccaca55ed16c3653	37	14	c6a62c4e56b0bd75 4bc1a8d91e07d409	33
6	18b3fa419616fe23 d16c3653cf402c68	33	15	56b0bd7575e8fd8f 1e07d4091ce2e6dc	31
7	9616fe2367117cf2 cf402c682b2cefbcb	32	16	75e8fd8f25896490 1ce2e6dc365e5f59	32
8	67117cf2c11bfc09 2b2cefbcb99f91153	33	IP-1	da02ce3a89ecac3b 057cde97d7683f2a	32

# DES attacks

---

- 2 criticisms
  - Key space too small: 56 bits
  - S-box design criteria kept secret
    - Are there any hidden attack/backdoor, only known to NSA?
- So far there is no known analytical attack in realistic scenarios
- 1998: DeepCrack: 50h, 250k\$
- 2006: COPACOBANA (Cost-Optimized Parallel COde Breaker): 6.4 days, 10k\$

DeepCrack, 1998  
\$250,000



COPACOBANA, 2006  
\$10,000



# 2 DES

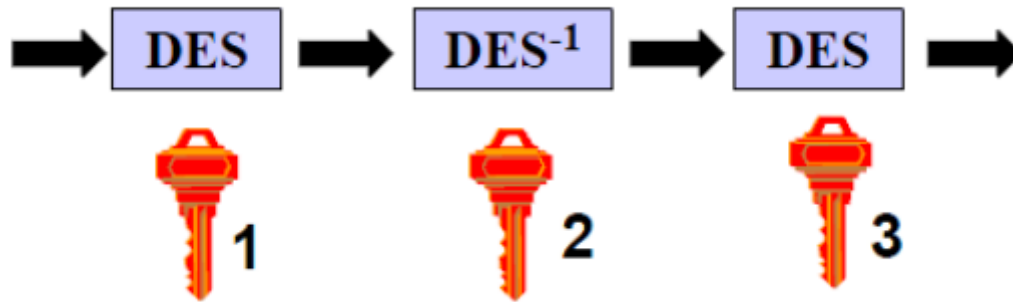
---



- Key length:  $56 \times 2 = 112$  bits
- Does 2DES “double” security? No!
- Meet-in-the-Middle Attack: suppose attacker disposes  $(P, C)$ 
  - Encrypt  $P$  with all  $2^{56}$  possible keys for  $K1$
  - Decrypt  $C$  with all  $2^{56}$  possible keys for  $K2$
  - Until  $E_{K1}(P) = D_{K2}(C)$
  - Complexity:  $O(2^{56})$
- Mathematically, DES is not a group
  - Caesar cipher is a group

# 3 DES

---



- 3 DES is used in practice
  - Effective key length 112 bits
    - Adequate for now
  - Reconsider MITM attack
- If  $k_1 = k_2$ , then becomes DES
  - Backward compatible
- If  $k_1 = k_3$ , then becomes 3 DES with 2 keys

# The “key” problems: weak keys

- Below are keys which, after the first key permutation, are:

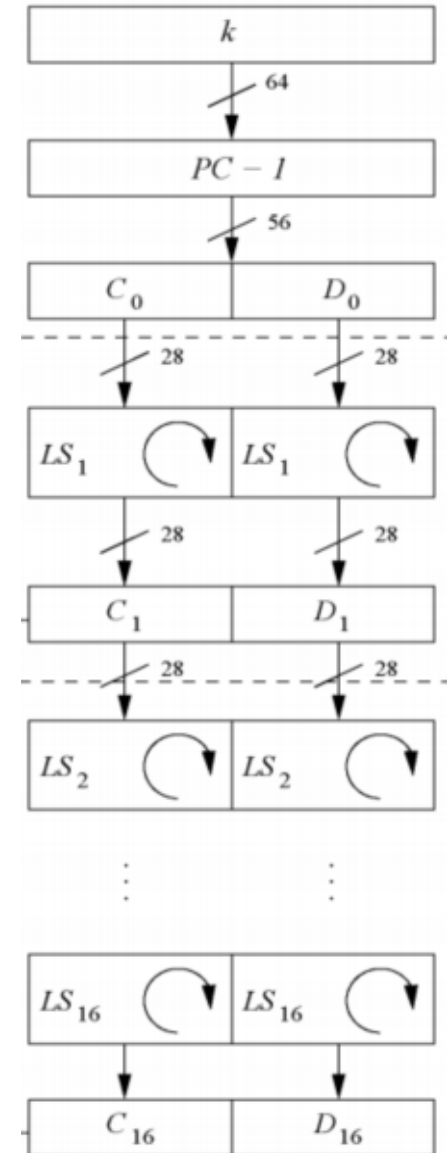
- 28 0's followed by 28 0's
- 28 1's followed by 28 1's
- 28 0's followed by 28 1's
- 28 1's followed by 28 0's

- Why they are weak

- Easy clue for brute force attacks.
- Sixteen identical subkeys.
- Encrypting twice produces the original plaintext

- Weak keys

- Alternating ones + zeros: 0x0101010101010101
- Alternating 'F' + 'E': 0xFEFEFEFEFEFEFEFEFE
- 0xE0E0E0E0F1F1F1F1
- 0x1F1F1F1F0E0E0E0E



# Semi-weak keys

---

- Below are keys which, after the first key permutation, are:
  - 1. 28 0's followed by alternating 0's and 1's
  - 2. 28 0's followed by alternating 1's and 0's
  - ...
  - 12. Alternating 1's and 0's followed by alternating 1's and 0's
- Why they are weak
  - For a semi-weak key pair  $(K_1, K_2)$ ,  $K_1(K_2(m)) = m$
- Semi-weak keys
  - 0x011F011F010E010E and 0x1F011F010E010E01
  - 0x01E001E001F101F1 and 0xE001E001F101F101
  - 0x01FE01FE01FE01FE and 0xFE01FE01FE01FE01
  - 0x1FE01FE00EF10EF1 and 0xE01FE01FF10EF10E
  - 0x1FFE1FFE0EFE0EFE and 0xFE1FFE1FFE0EFE0E
  - 0xE0FEE0FEF1FEF1FE and 0xFEE0FEE0FEF1FEF1

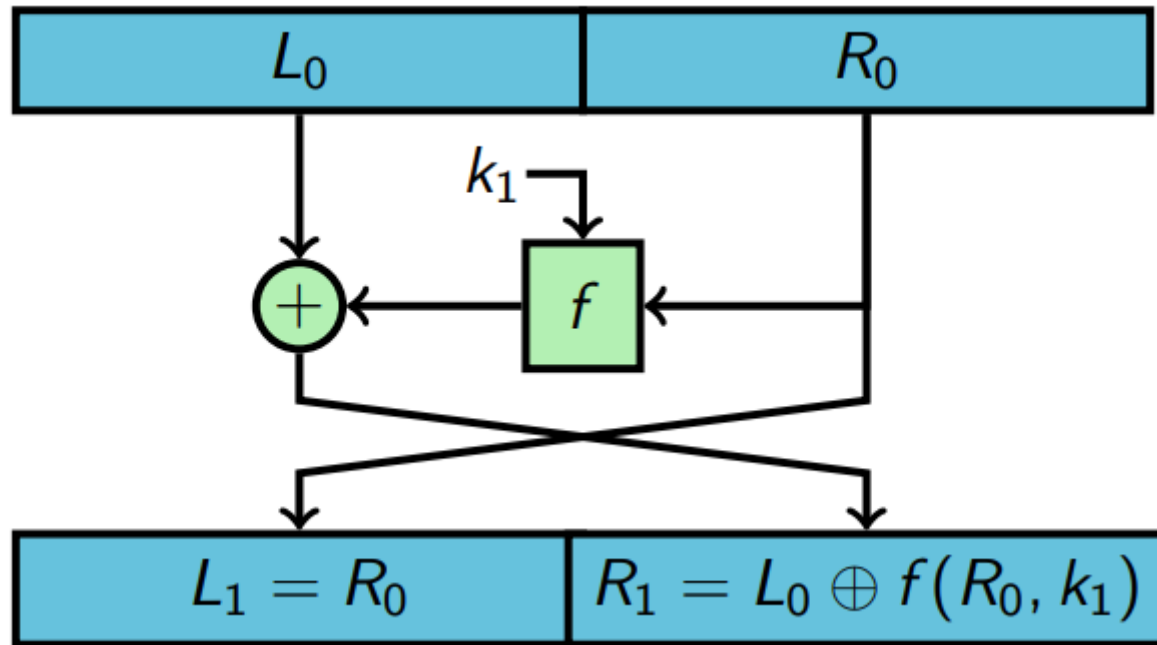
# Test

---

- How to distinguish a 2-round Feistel system with a random number generator?
- Prove  $DES_k(m) = \overline{DES_{\bar{k}}(\bar{m})}$
- Break a 1-round Feistel system, 2-round, 3-round.

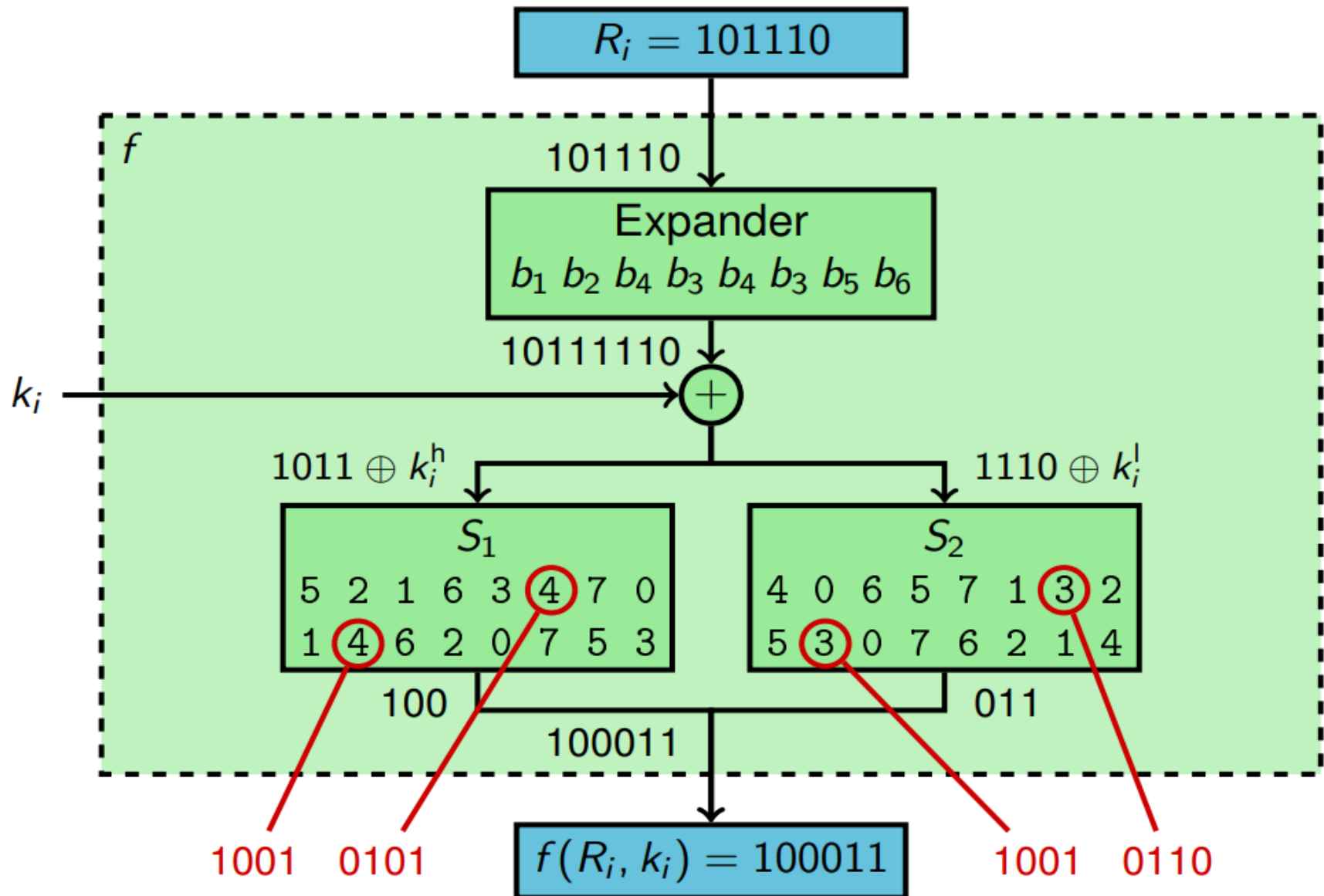
# Break 1-round Feistel system

---



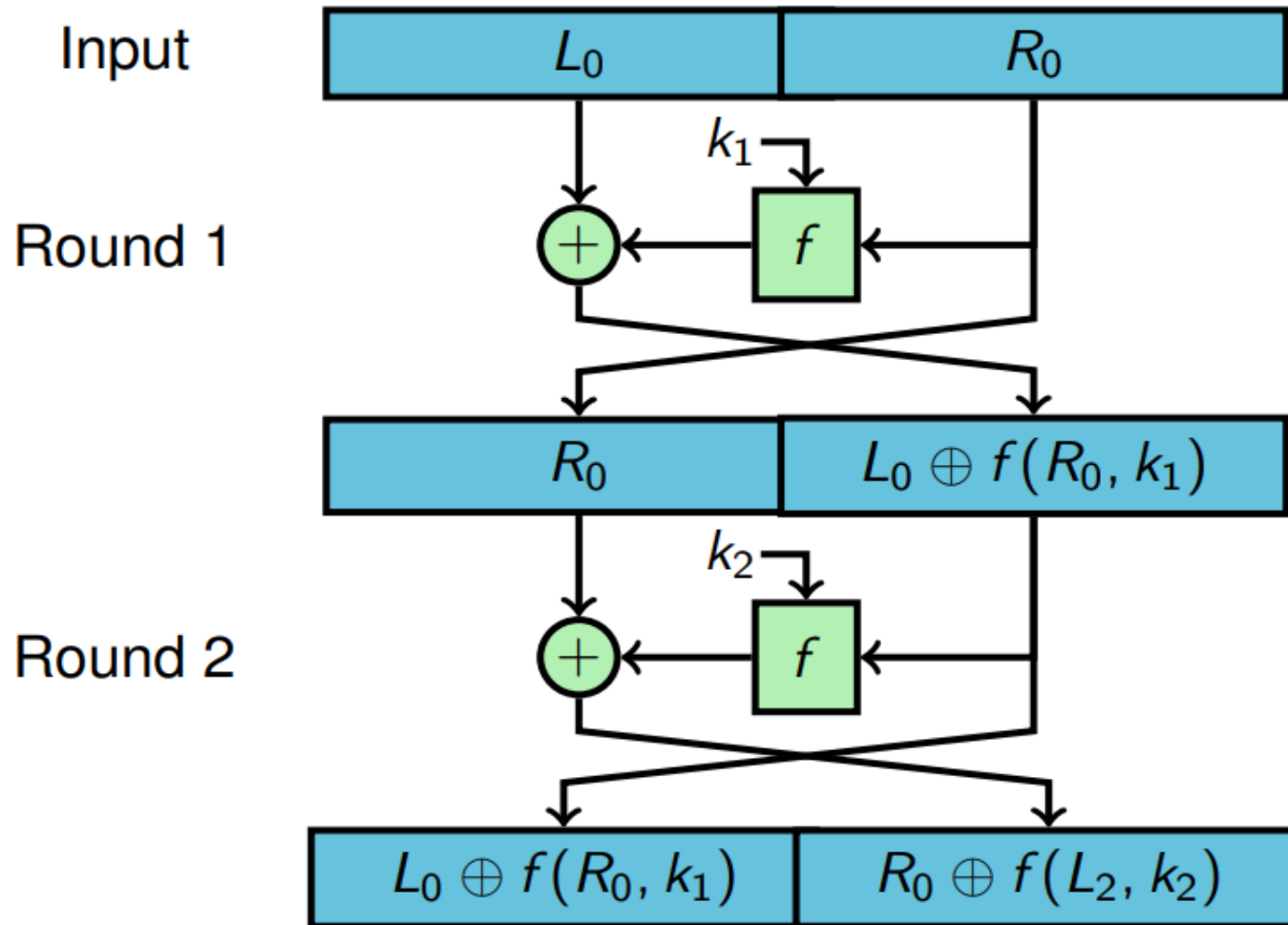


# Break 1-round Feistel system



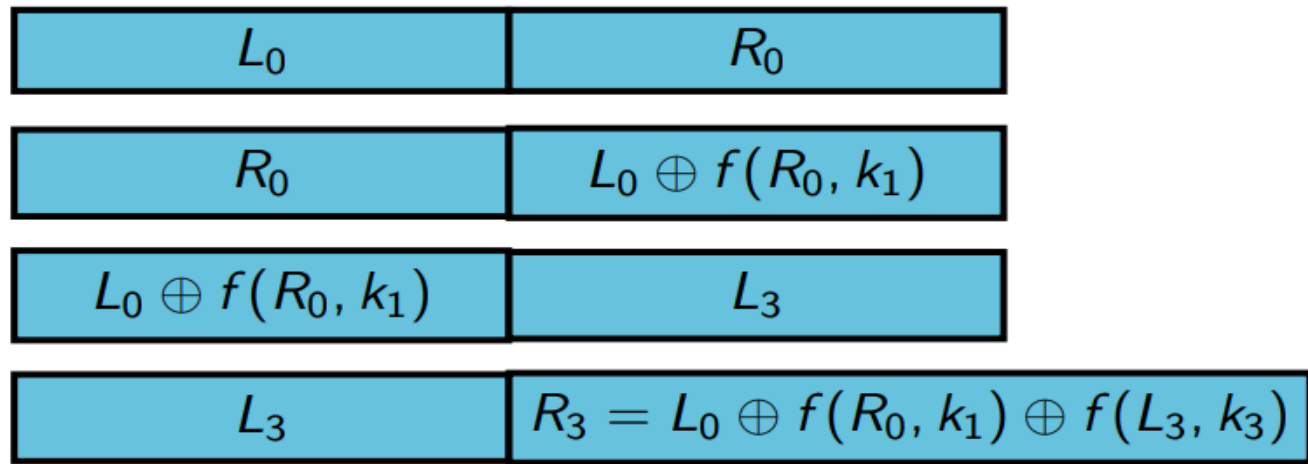
# Break 2-round Feistel system

---



# Break 3-round Feistel system

---

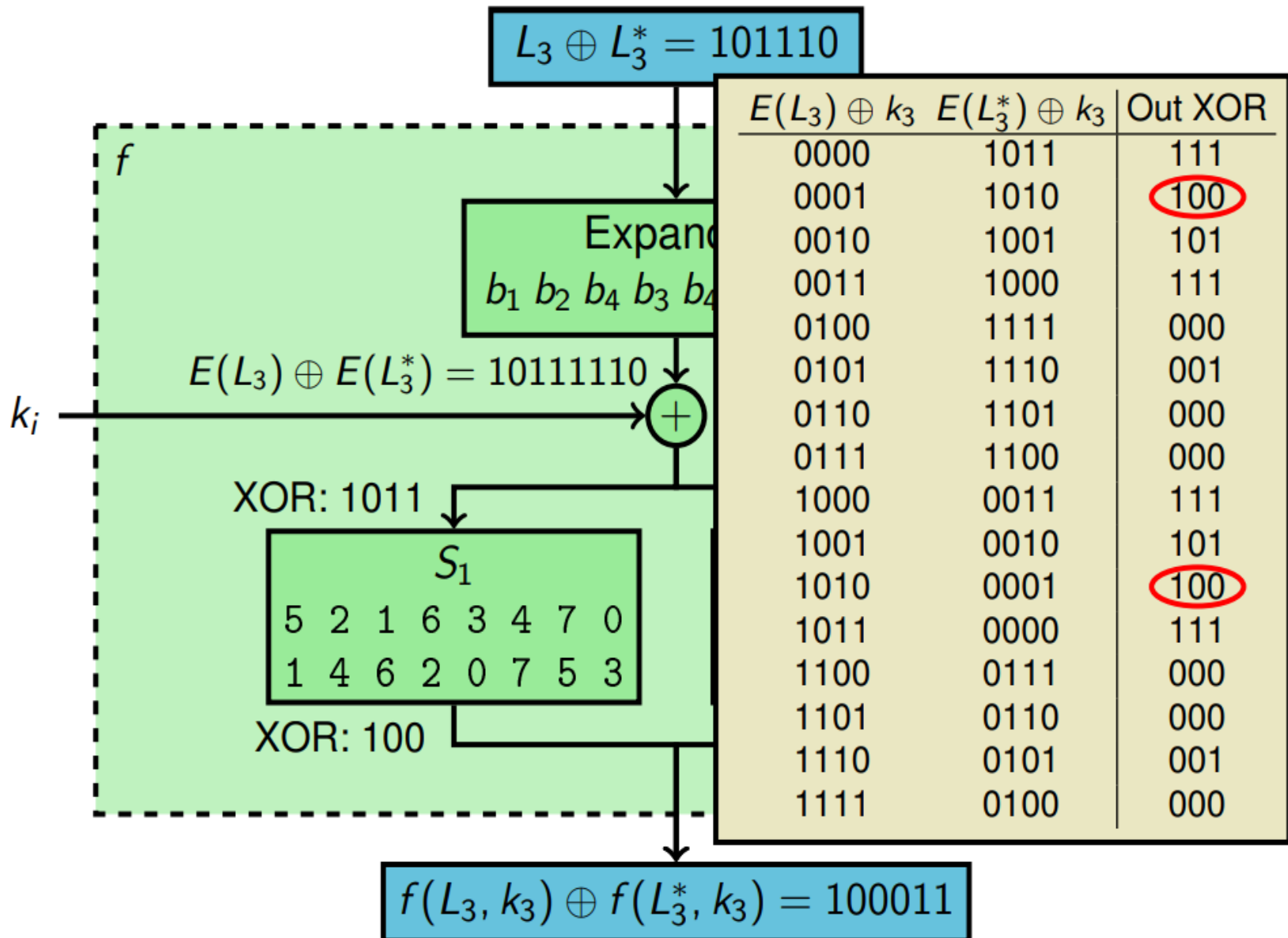


Perform two attacks for  $L_0 R_0$  and  $L_0^* R_0^*$  with  $R_0 = R_0^*$ .  
Then, the outputs have the relation

$$R_3 \oplus R_3^* = L_0 \oplus L_0^* \oplus f(L_3, k_3) \oplus f(L_3^*, k_3)$$

We have  $L_3 \oplus L_3^*$  and  $f(L_3, k_3) \oplus f(L_3^*, k_3)$

# Break 3-round Feistel system



# Advanced Encryption Standard (AES)

---

- Objective: replace DES
  - DES key size and block size too small
  - Can use Triple-DES, but slow
- US NIST issued call for ciphers in 1997
  - 15 candidates accepted in 1998
  - 5 shortlisted in 1999
  - Rijndael selected in 2000
  - Standard in 2001

# AES Requirements

---

- Symmetric key block cipher
- Block size: 128 bits
- Key length: 128/192/256 bits
- Stronger & faster than 3 DES
- Active life of 20-30 years
- Provide full specification & design details
- Both C & Java implementations
- NIST have released all submissions & unclassified analyses

# AES Evaluation Criteria

---

- Initial criteria:
  - Security: randomness, soundness, effort for practical cryptanalysis
  - Cost: computational efficiency, no licensing fee, small memory
  - Algorithm & implementation: flexibility, implementable in both software and hardware, simplicity
  
- Final criteria
  - General security: NIST relies on cryptanalysis by cryptologues
  - Ease of software & hardware implementation
  - Flexibility: encryption, decryption, keying
  - Implementation attacks
    - Timing attacks: an algorithm takes different time on different inputs
    - Power analysis: power consumed depends on instructions
      - multiplication > addition, writing 1 > 0

# Shortlisted algorithms

---

- MARS (IBM) - complex, fast, high security margin
- RC6 (USA) - very simple, very fast, low security margin
- Rijndael (Belgium) - clean, fast, good security margin
- Serpent (Euro) - slow, clean, very high security margin
- Twofish (USA) - complex, very fast, high security margin
  
- Contrast between algorithms with
  - Few complex rounds vs. many simple rounds
  - Refined existing ciphers vs. new proposals

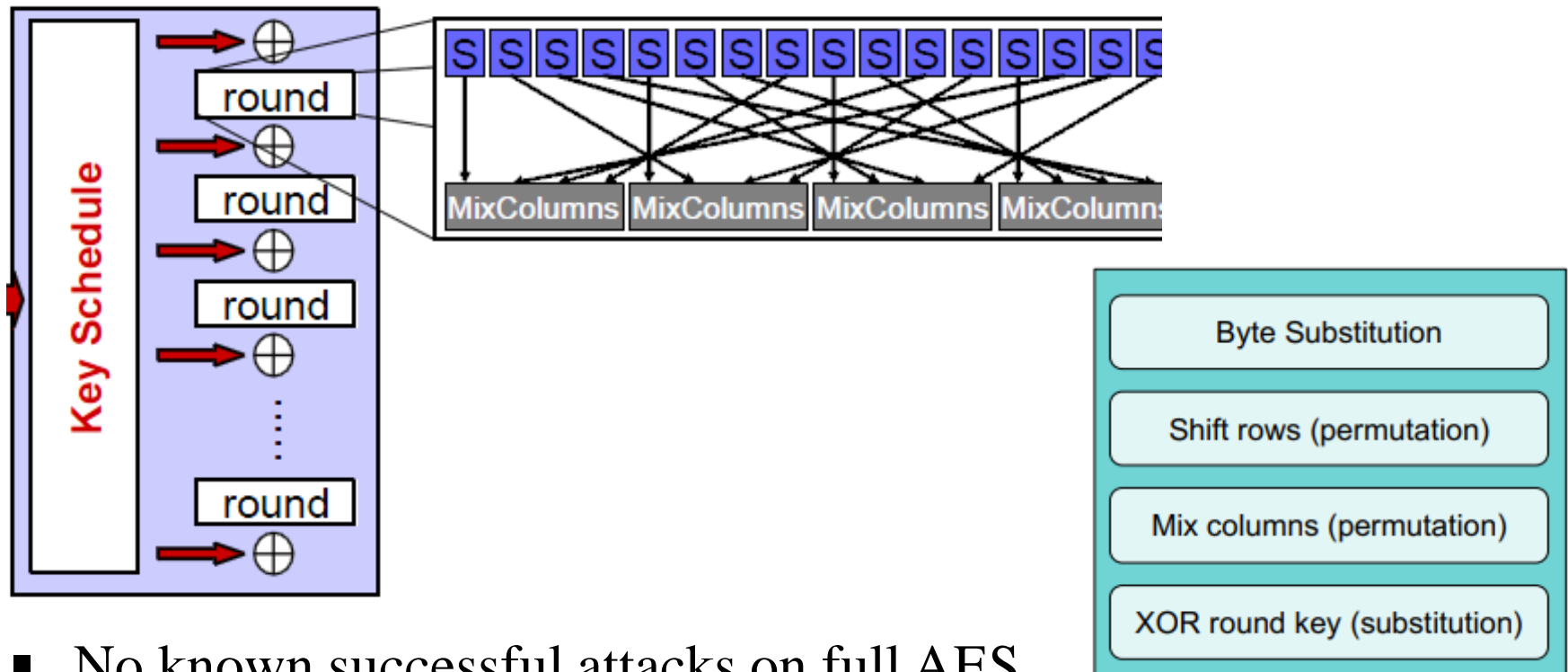


# The winner: Rijndael

---

- Designed by Rijmen-Daemen in Belgium
- An iterative rather than Feistel Cipher
  - Processes data as block of 4 columns of 4 bytes
- Designed to be:
  - Resistant against known attacks
  - Speed and code compactness on many CPUs
  - Design simplicity
- Some similarities to DES
  - rounds, round keys, alternate permutation+substitution
  - but not a Feistel cipher
- Block size 128 bits
- Key sizes 128, 192, or 256

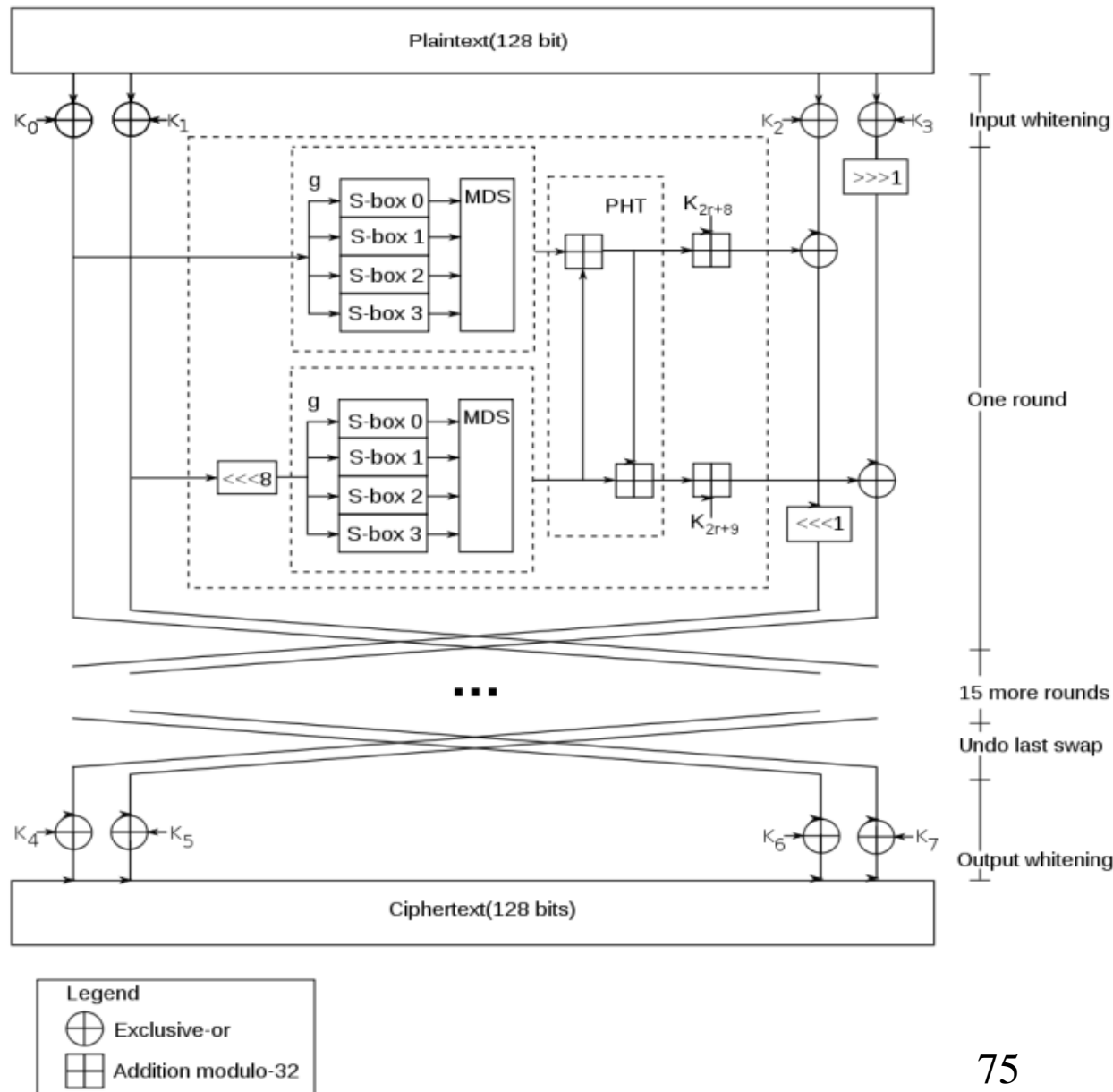
# AES structure



- No known successful attacks on full AES
  - Best attacks work on 7-9 rounds
- For brute force attack, AES-128 needs much more effort than DES

# Twofish: AES third place

- Feistel
- S-boxes depend on key
- Slower than AES



# Serpent: AES second place

- Constructed for security
  - not speed
- Feistel system, 32 rounds
- Four-bit S-boxes
  - $A_i = S(K_i \oplus P_i)$
- Adapted for parallel calculation
- Same speed as DES, 1/3 of AES

