

# Basics of Cryptography

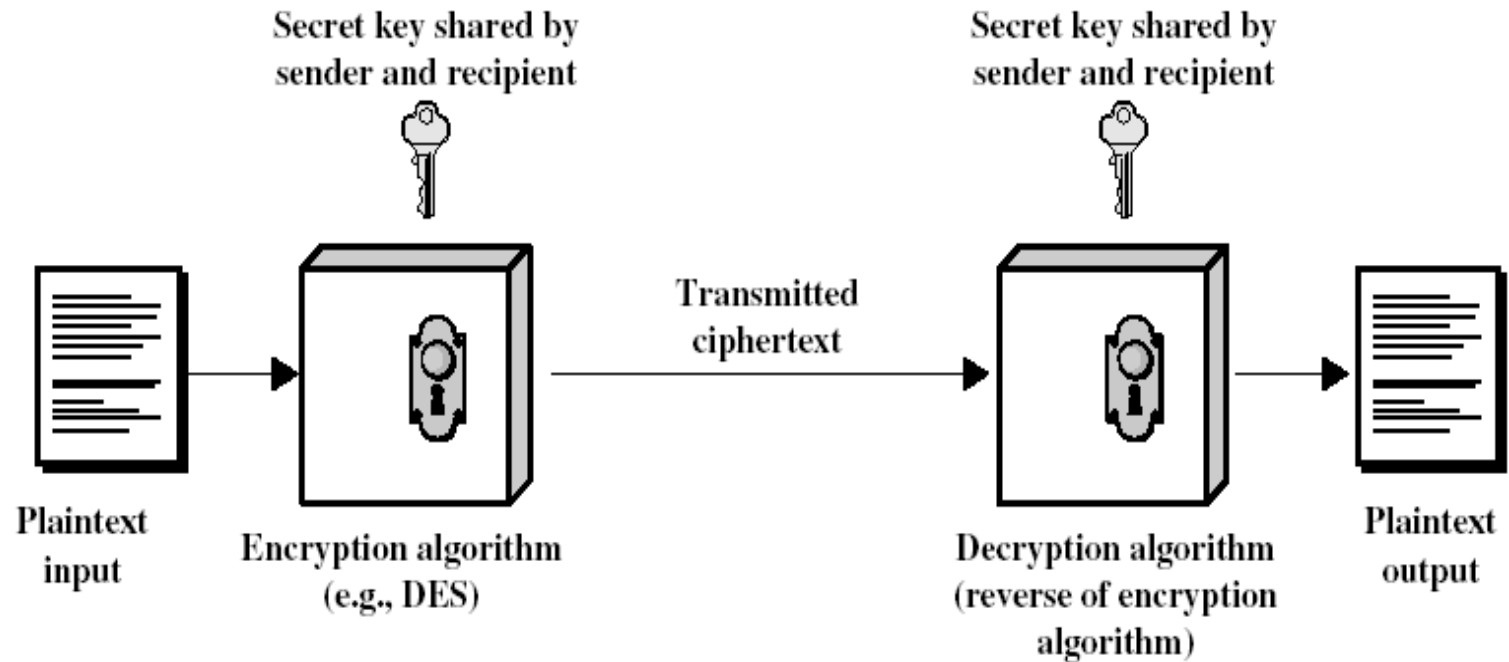
# Terminology

- **Plaintext** - the original message
- **Ciphertext** - the coded message
- **Cipher** - algorithm for transforming plaintext to ciphertext
- **key** - info used in cipher known only to sender/receiver
- **Cryptanalysis (codebreaking)** - the study of principles/ methods of deciphering ciphertext *without* knowing key
- **Cryptology** - the field of both cryptography and cryptanalysis

# More Terminology

- Symmetric Encryption
  - Both Sender/Receiver use the same algorithms/keys for encryption/decryption
- Asymmetric Encryption
  - Sender/receiver can employ different keys

# Symmetric Encryption Model



# Encryption Basics

- Gen() algorithm for generating keys
- Encryption algorithm to convert plaintext into ciphertext
  - $E(M, \text{keys}) = C$
- Decryption algorithm to convert ciphertext to plaintext
  - $D(C, \text{keys}) = M$

# Some early ciphers

- Substitution (eg., Caesar cipher)
- Security is hard: tfdvsjuz jt ibse
- $C = (M+k) \bmod 26$
- $M = (C-k) \bmod 26$
- Only 26 possibilities with English Alphabet
- Brute Force search can decrypt

# Monoalphabetic cipher

- Instead of plain rotation, use random letter substitution
- Key is 26 letters long

Plain:    abcdefghijklmnopqrstuvwxyz

Cipher:  DKVQFIBJWPESCXHTMYAUOLRGZN

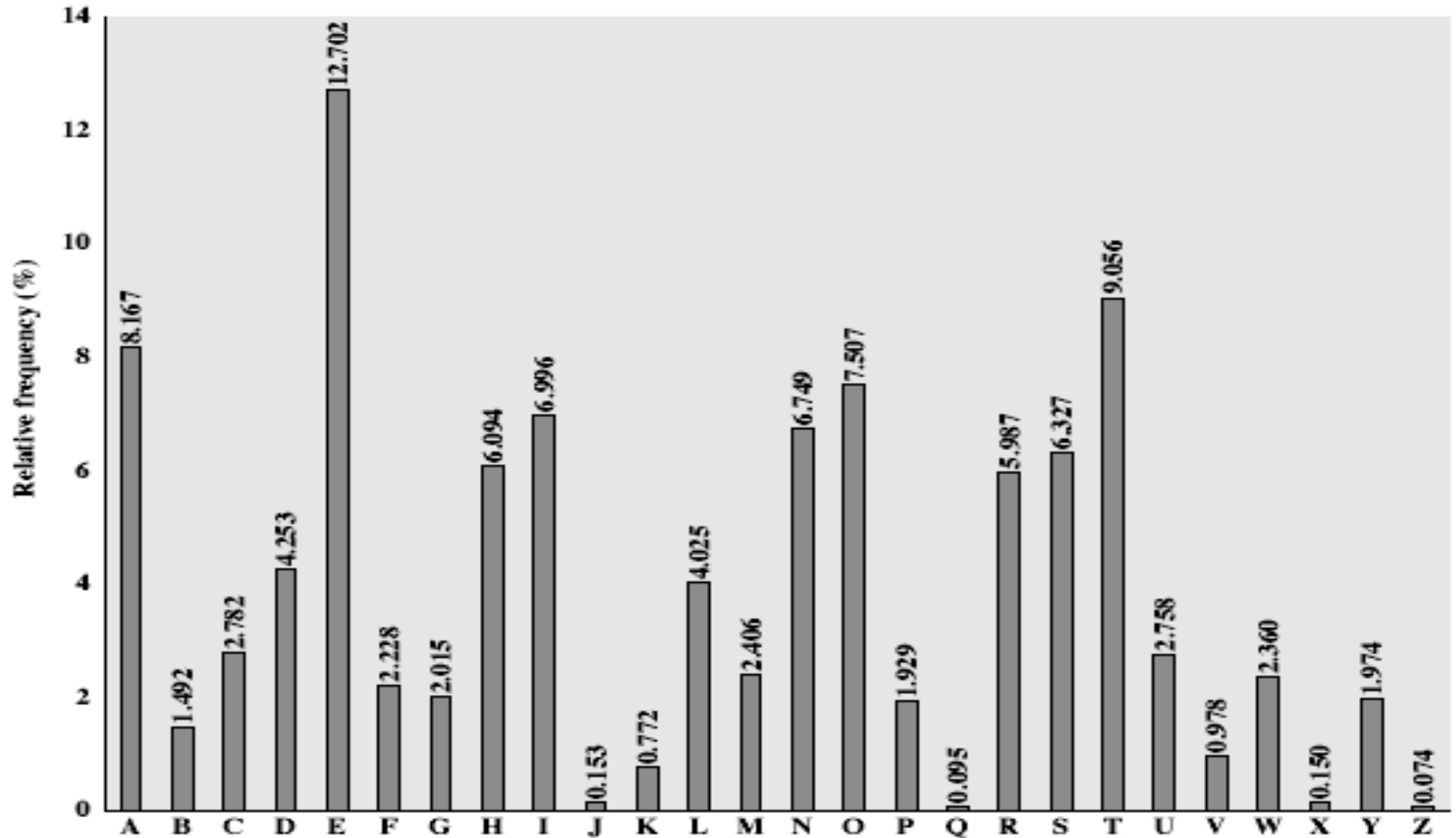
Plaintext:    security is hard

Ciphertext:  AFVOYWUZ  WA  JDYQ

# Monoalphabetic cipher

- 26! Combinations
- Difficult to decrypt?
- Not really!
- Language gives lots of hints
  - Single letters are I or A
  - Most common letter E
- Use Lang. characteristics to break

# English letter Frequencies



# Breaking substitution ciphers

- Use Letter frequencies of ciphertext
- Compare to plaintext frequencies
- These don't change -enable analysis
- Use common two-letter words etc.

# Measures of ciphers

- Shannon Secrecy
- $\Pr (M = m | E(K,m) = c) = \Pr (M = m)$ 
  - Probability of guessing the plaintext knowing the ciphertext = probability of guessing plaintext without knowing ciphertext

# Perfect Secrecy

- $\Pr( E(K, m) = c ) = \Pr (E(K, m') = c)$
- Probability of any message giving a ciphertext is the same

# Block vs. Stream ciphers

- Block ciphers encrypt block at a time
- Message is broken into blocks and encrypted
- Stream ciphers process a bit or byte at a time during encryption/decryption

# Shannon and ciphers

- Claude Shannon introduced idea of substitution-permutation (S-P) networks (1949)
  - the basis of modern block ciphers
- S-P networks are based on the two primitive cryptographic operations:
  - *substitution* (S-box)
  - *permutation* (P-box)
- Provide *confusion* and *diffusion* of message

# Confusion and Diffusion

- Cipher needs to completely obscure statistical properties of original message
- Shannon suggested confusion & diffusion
- **Diffusion** – dissipates statistical structure of plaintext over bulk of ciphertext
- **Confusion** – makes relationship between ciphertext and key as complex as possible

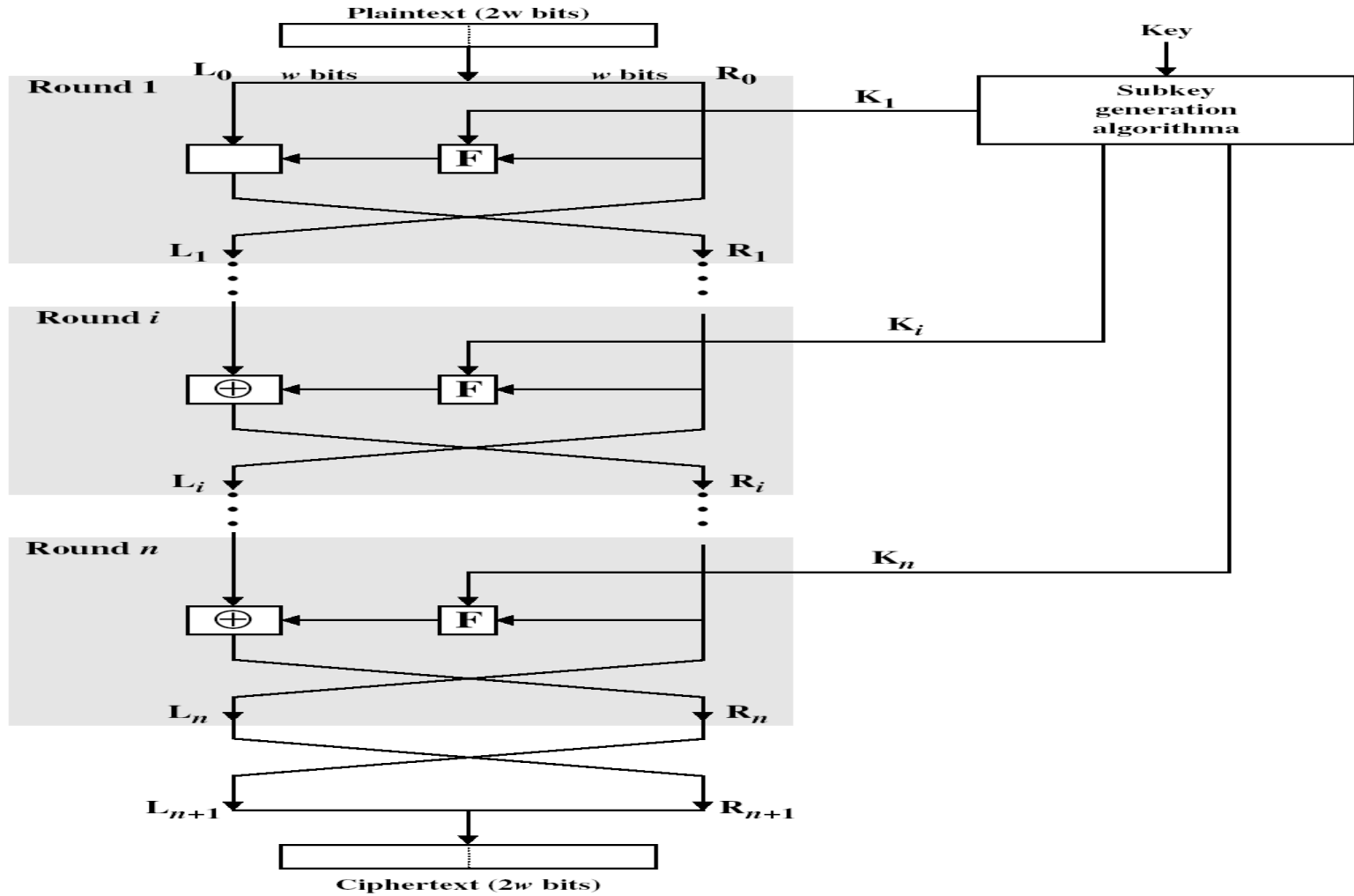
# Shannon's one-time pad

- Choose a key as long as the message
- $E(M, k) = k \text{ XOR } M = C$
- $D(C, k) = k \text{ XOR } C = M$
- Choose  $k$  randomly (uniformly distributed in  $\{0,1\}^l$ ),  $l =$  message length
- One-time pad has perfect secrecy
  - $\Pr(m \text{ xor } k = c) = \Pr(m' \text{ xor } k = c) = 2^{-l}$

# One time pad

- Each key works only once
- Works with fixed length messages
- Key length = message length
- Not very practical

# Fiestel Cipher



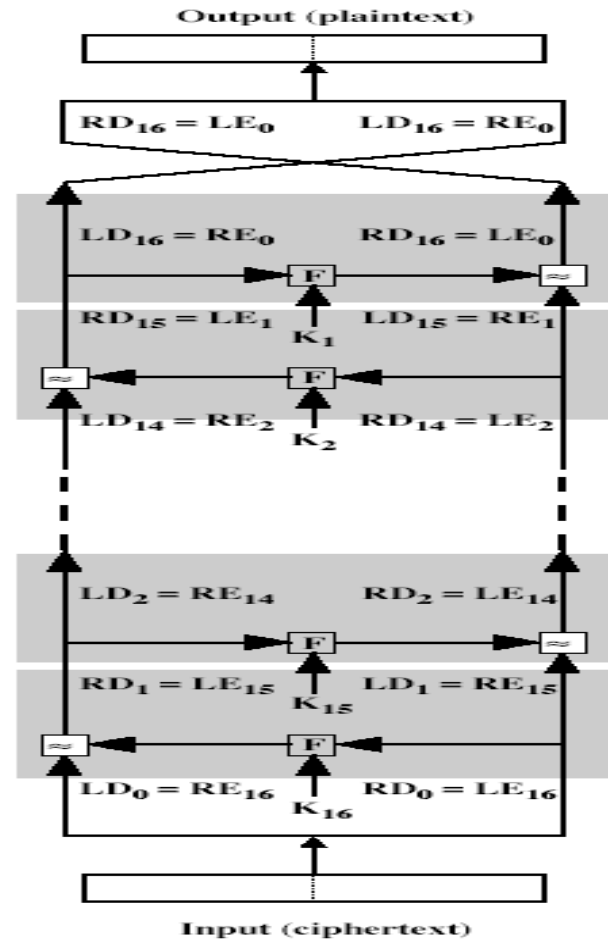
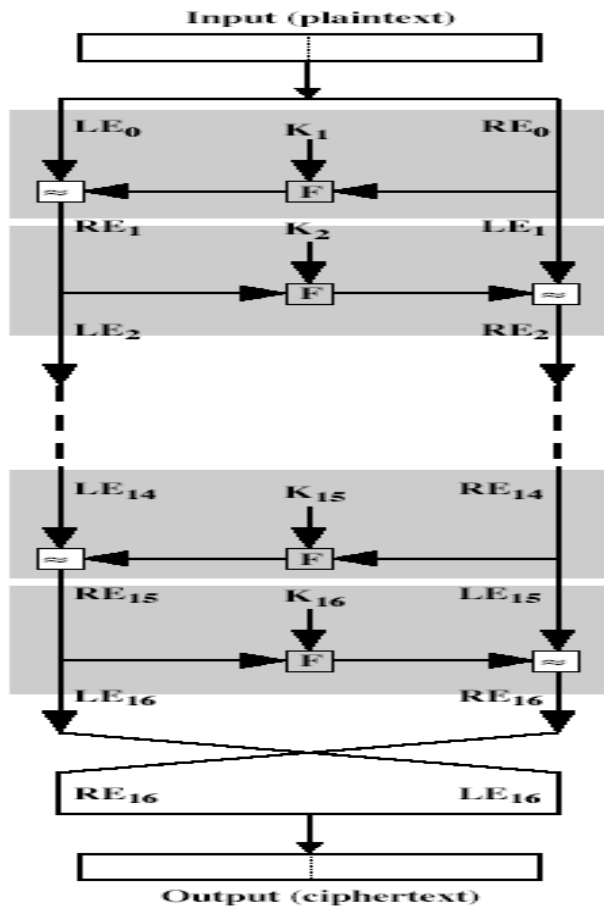
# Fiestel Cipher (IBM, 70s)

- Partitions input block into two halves
  - Employs multiple rounds of processing
  - Performs a substitution on left data half based on a fn. of right half & subkey
  - Employs permutation swapping halves
- Implements Shannon's substitution-permutation network concept

# Cipher parameters

- **Block size**
  - increasing size improves security, but slows cipher
- **key size**
  - increasing size improves security, makes exhaustive key searching harder, but may slow cipher
- **Number of rounds**
  - increasing number improves security, but slows cipher
- **Subkey generation**
  - greater complexity can make analysis harder, but slows cipher
- **Round function**
  - greater complexity can make analysis harder, but slows cipher

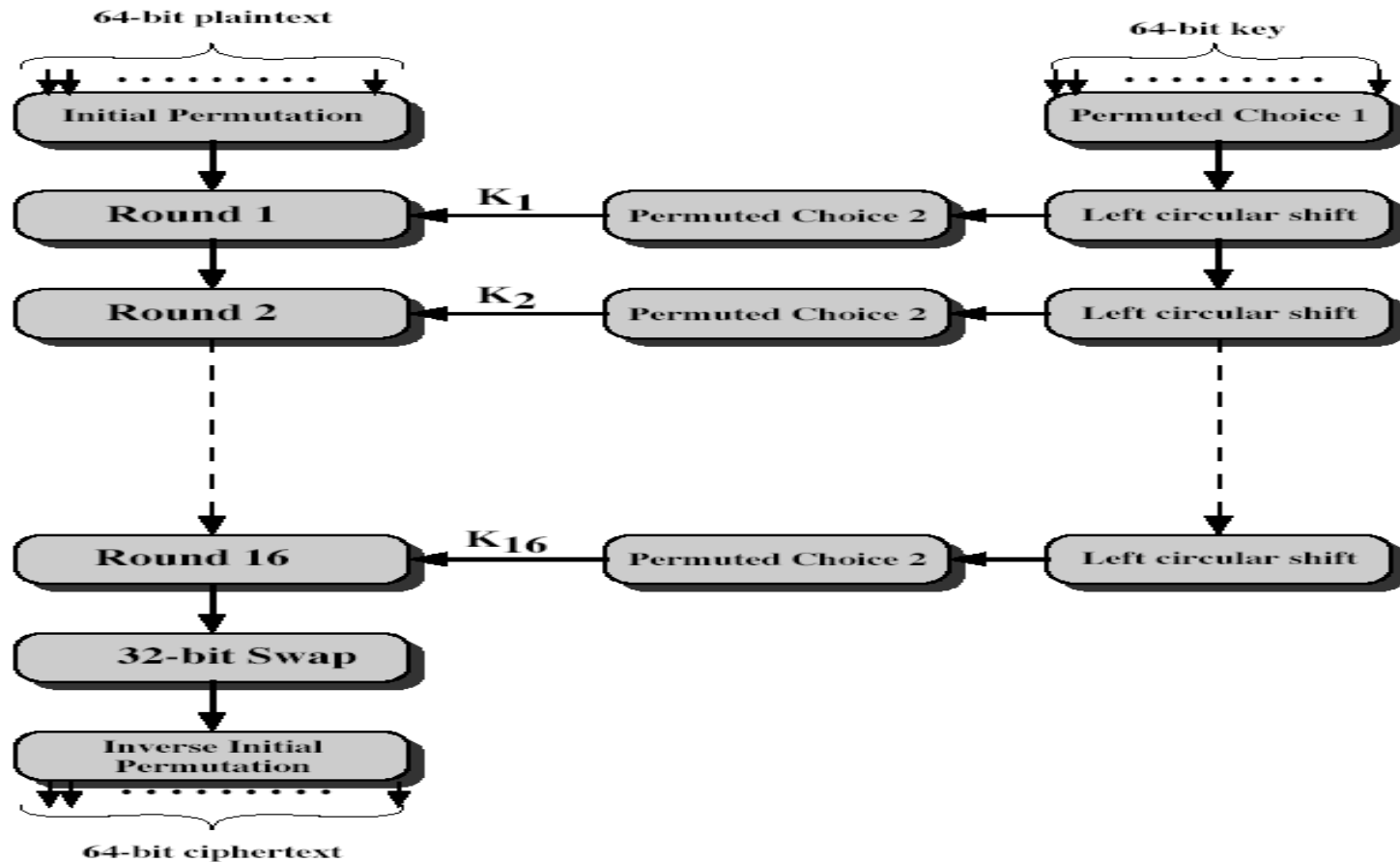
# Decryption



# DES cipher

- Data Encryption Standard
- Most widely used block cipher in world
- Adopted in 1977 by NIST as a standard
- Encrypts 64-bit data using 56-bit key
- Based on IBM's Lucifer cipher (128-bit key)

# DES Encryption



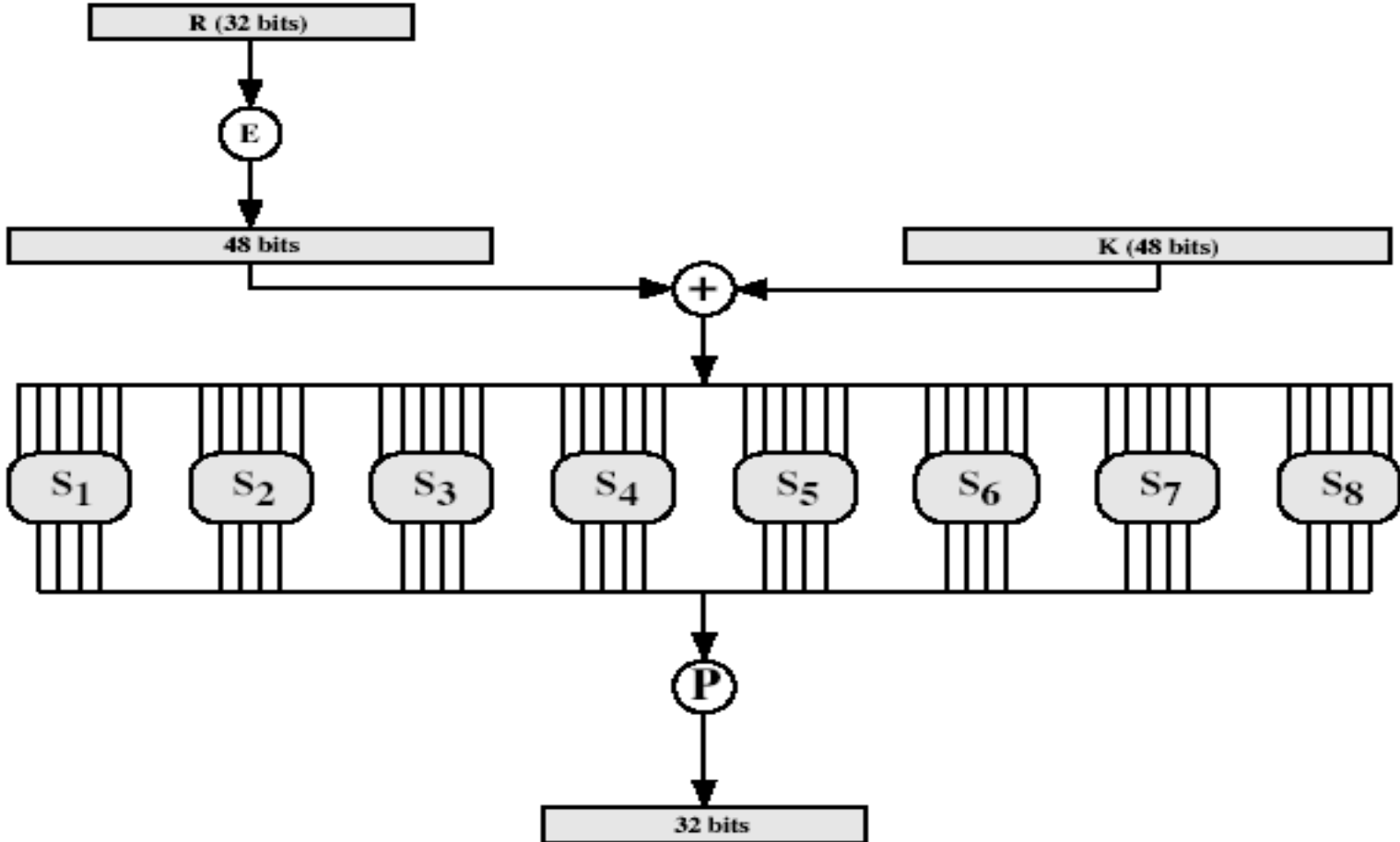
# Initial permutation

- First step of the data computation
- IP reorders the input data bits
- Even bits to LH half, odd bits to RH half
- Quite regular in structure
  - easy to build h/w

# DES Rounds

- Uses two 32-bit L & R halves
- Similat to Feistel cipher can describe as:  
$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \text{ xor } F(R_{i-1}, K_i)$$
- Takes 32-bit R half and 48-bit subkey and:
  - expands R to 48-bits using perm E
  - adds to subkey
  - passes through 8 S-boxes to get 32-bit result
  - finally permutes this using 32-bit perm P

# DES Round



# Strength of DES

- 56-bit keys have  $2^{56} = 7.2 \times 10^{16}$  values
- Brute force search requires lot of work
- But, possible
  - in 1997 on Internet in a few months
  - in 1998 on dedicated h/w in a few days
  - in 1999 above combined in 22hrs!
- Must be able to recognize plaintext
- Alternatives to DES being considered

# References

- [1] Network Security Essentials, Applications and Standards, 2<sup>nd</sup> edition by William Stallings -Chapter2