Wolkite University

Department of Information Systems

INFORMATION SYSTEM SECURITY

CHAPTER FOUR

Security Techniques: Cryptography

Wendosen Z.(MSc in Computer Networking)

Outline

- Basic Security techniques
- Cryptography
- Symmetric and asymmetric encryption
- Cryptanalytic Attacks
- Caesar cipher
- Block vs Stream Ciphers
- Substitution-Permutation Ciphers
- Cryptographic Algorithms
- Symmetric Block Cipher Algorithms
  - DES (Data Encryption Standard)
  - Double DES
  - 3DES (Triple DES)
- DES strength/weakness

Information Security Techniques

- Internet Cryptography Techniques
- Transport Layer Security
- Application Layer Security
- Server Proxies and Firewalls
- Access controls
- Intrusion Detection System

Basic Cryptography Techniques

- Ciphering/Deciphering (keys)
- Authentication
- Digital Signature
- Digital Certification

Purpose of cryptography

- Secure stored information - regardless if access obtained
- Secure transmitted information - regardless if transmission has been monitored

Services Provided by Cryptography

- Confidentiality
  - provides privacy for messages and stored data by hiding
- Message Integrity
  - provides assurance to all parties that a message remains unchanged
- Non-repudiation
  - Can prove a document came from the intended user
- Authentication
  - identifies the origin of a message
  - verifies the identity of person using a computer system

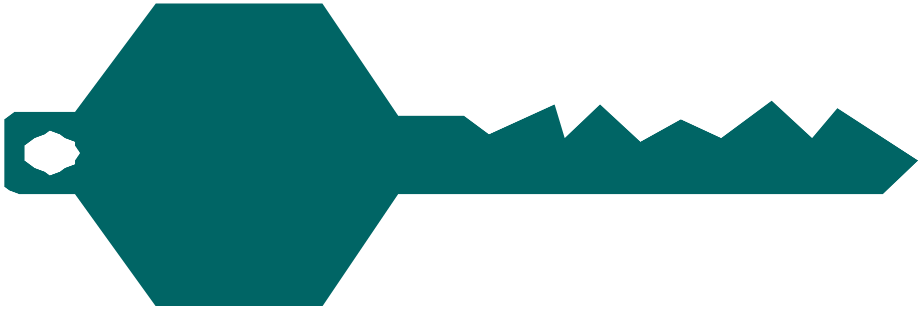
Cryptography

- Cryptography has five components:
  - Plaintext: This is what you want to encrypt.
  - Ciphertext: The encrypted output.
  - Enciphering or encryption: The process by which plaintext is converted into ciphertext.
  - Encryption algorithm: The sequence of data processing steps that go into transforming plaintext into ciphertext.
  - Secret Key: is used to set some or all of the various parameters used by the encryption algorithm.
  - Deciphering or decryption: Recovering plaintext from ciphertext.
  - Decryption algorithm: The sequence of data processing steps that go into transforming ciphertext back into plaintext.

Keys

- A key can be thought of as simply a collection of bits
- The more bits, the stronger the key
- Keys are tied to specific encryption algorithms
- Lengths vary depending on the encryption algorithm
  - e.g. 128 bits is long for some algorithms, but short for others

1 0 1 1 1 1 0 1 1
1 0 1 1 0 0 1 0 1

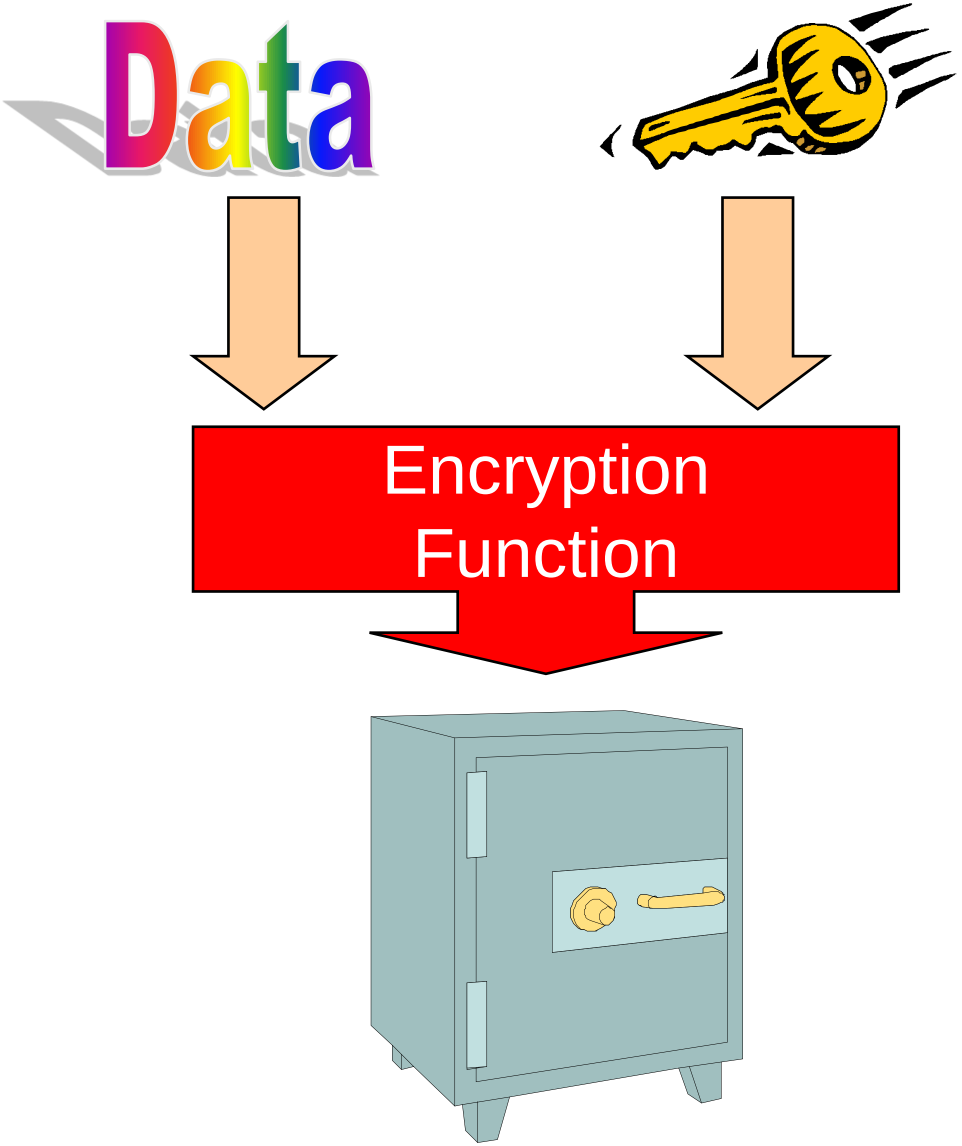


Cryptography

- Encryption Overview
  - Plain text is converted to cipher text by making use of an algorithm and key.
    - Algorithm is publicly known
    - Key is held private
  - **Three Main Categories**
    - Secret Key
      - single key is used to encrypt and decrypt information
    - Public/Private Key
      - two keys are used: one for encryption (public key) and one for decryption (private key)
    - One-way Function
      - information is encrypted to produce a “digest” of the original information that can be used later to prove its authenticity

Encryption

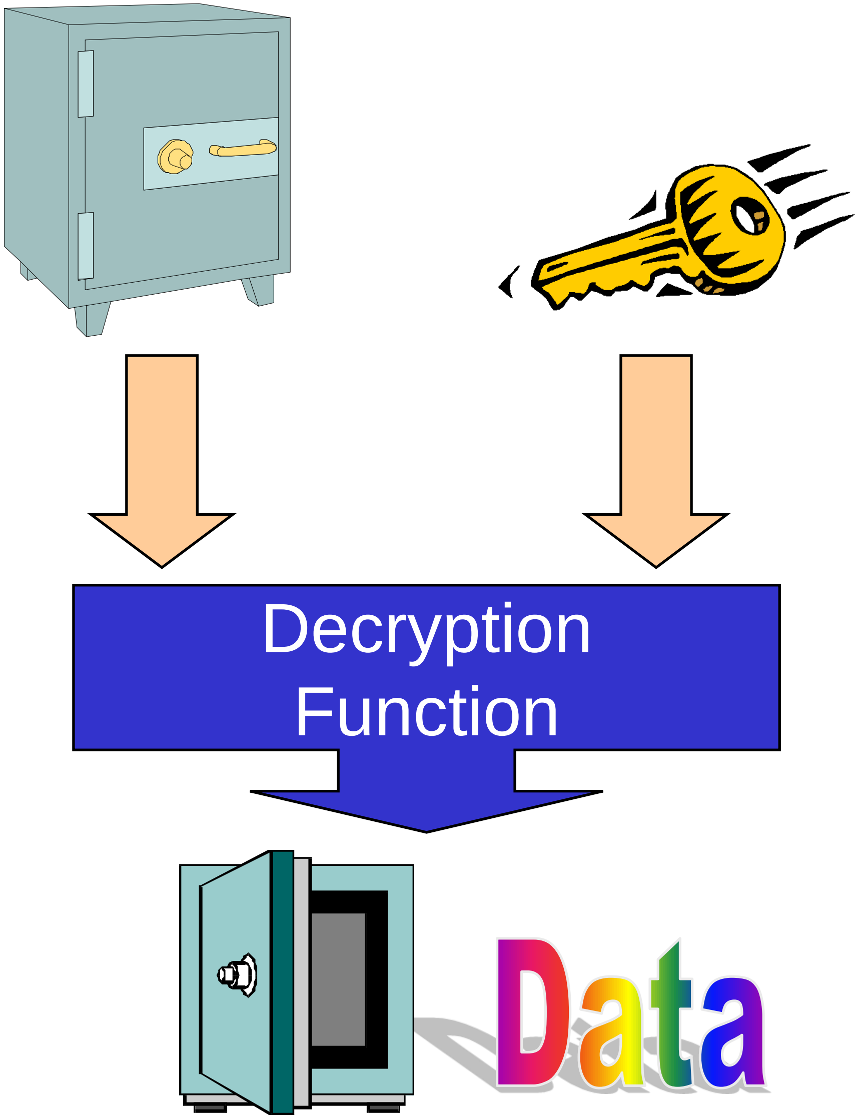
- Encryption is the process of taking some data and a key and feeding it into a function and getting encrypted data out
- Encrypted data is, in principle, unreadable unless decrypted



Encryption Function

Decryption

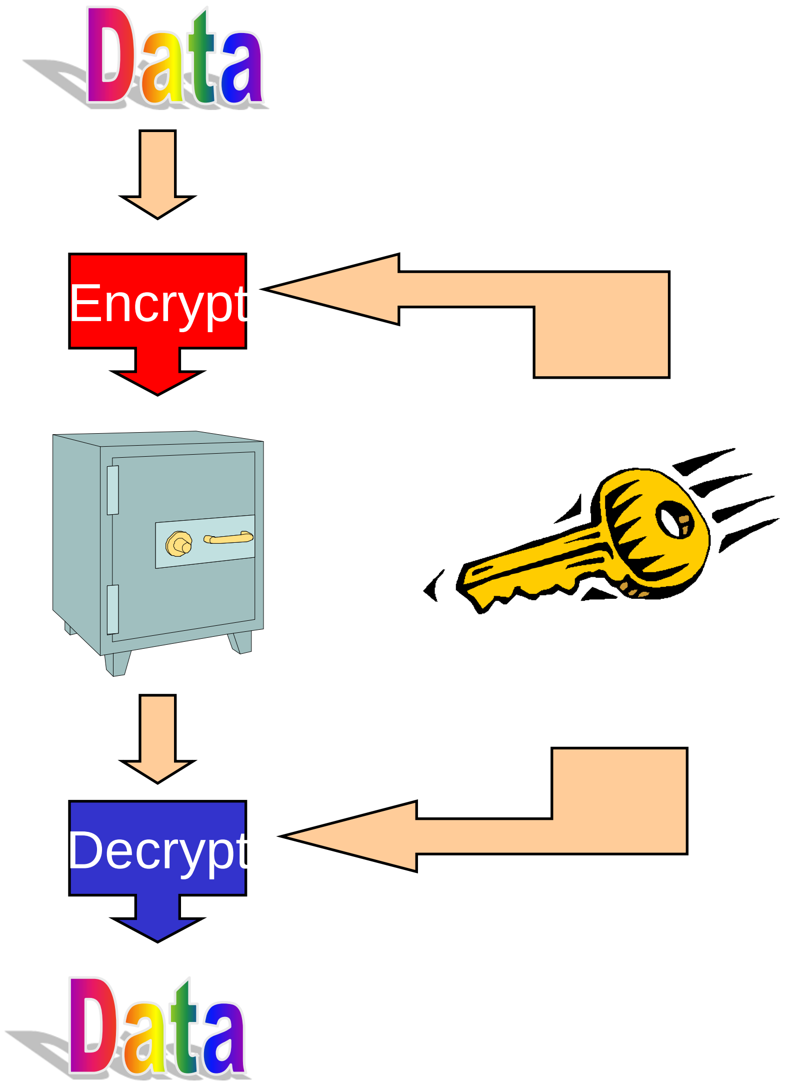
- Decryption is the process of taking encrypted data and a key and feeding it into a function and getting out the original data
  - Encryption and decryption functions are linked



Encryption Techniques

Symmetric Encryption

- Encryption and decryption algorithms that use the same key are called symmetric
  - In this case everyone wanting to read encrypted data must share the same key
- Sender and receive have the same secret key that will encrypt and decrypt plain text.
- Strength of encryption technique depends on key length



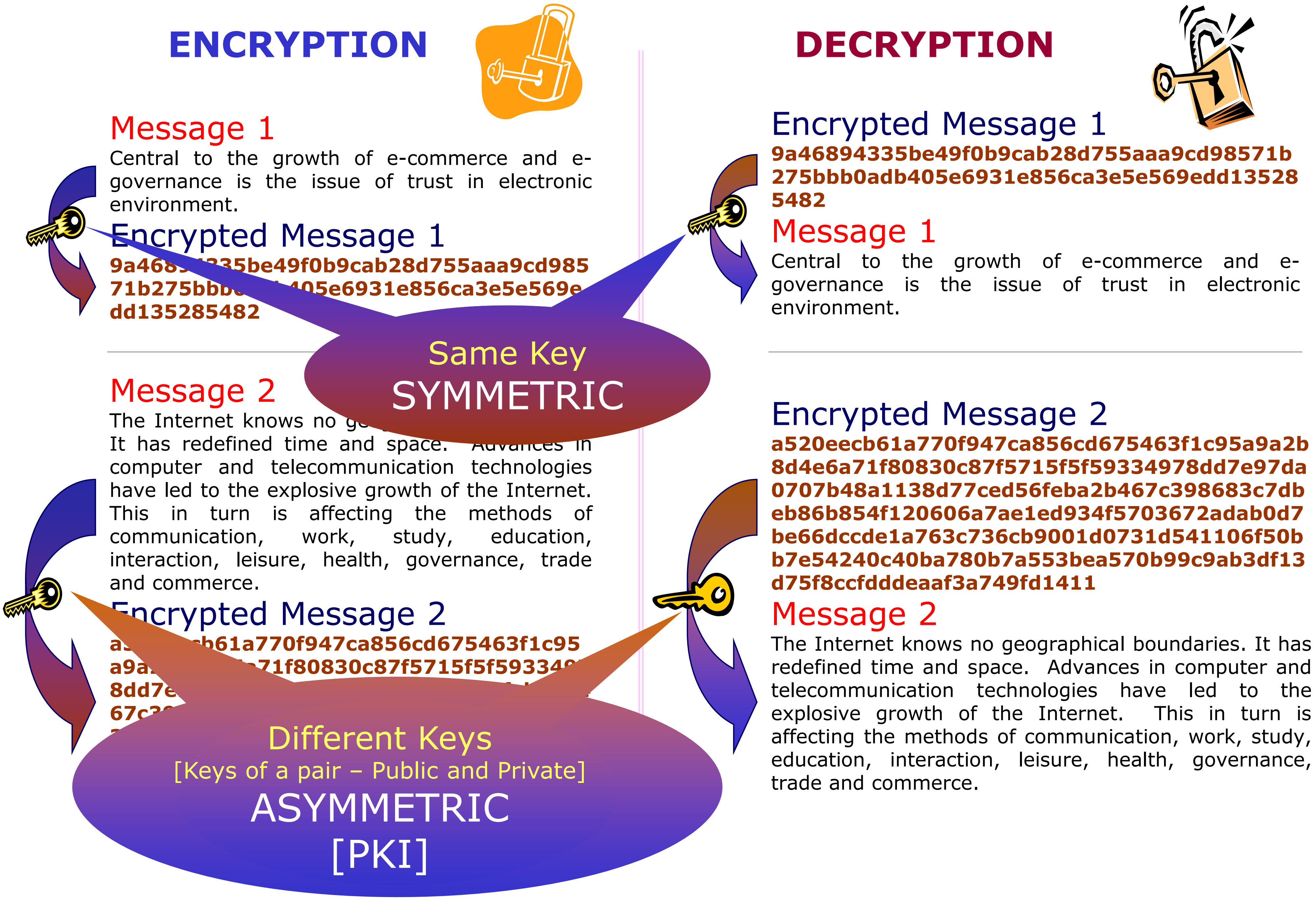
Encryption Techniques…

- Secret Key (Symmetric)
  - Known symmetrical algorithms
    - Data Encryption Standard (DES)
      - 56 bit key
    - Triple DES, DESX, GDES, RDES
      - 168 bit key
    - Advanced Encryption Standard (AES)
      - 128, 192,256 bit key
    - RC2, RC4, RC5
      - variable length up to 2048 bits
    - IDEA - basis of PGP
      - 128 bit key
    - Blowfish, Twofish
      - variable length up to 448 bits

Encryption Techniques…

Asymmetric Encryption

- Encryption and decryption algorithms that use a key pair are called asymmetric
  - Keys are mathematically linked
- Most common algorithm is the RSA (Rivest Shamir Adleman) algorithm with key lengths from 512, 1024, 2048 bits…



ENCRYPTION

Message 1

Central to the growth of e-commerce and e-governance is the issue of trust in electronic environment.

Encrypted Message 1

**9a468 335be49f0b9cab28d755aaa9cd985
71b275bbb 405e6931e856ca3e5e569e
dd135285482**

Message 2

The Internet knows no ge
It has redefined time and space. Advances in computer and telecommunication technologies have led to the explosive growth of the Internet. This in turn is affecting the methods of communication, work, study, education, interaction, leisure, health, governance, trade and commerce.

Encrypted Message 2

**a5 cb61a770f947ca856cd675463f1c95
a9a 71f80830c87f5715f5f59334
8dd7e
67c3**

Same Key

SYMMETRIC

Different Keys

[Keys of a pair – Public and Private]

ASYMMETRIC

[PKI]

DECRYPTION

Encrypted Message 1

**9a46894335be49f0b9cab28d755aaa9cd98571b
275bbb0adb405e6931e856ca3e5e569edd13528
5482**

Message 1

Central to the growth of e-commerce and e-governance is the issue of trust in electronic environment.

Encrypted Message 2

**a520eecb61a770f947ca856cd675463f1c95a9a2b
8d4e6a71f80830c87f5715f5f59334978dd7e97da
0707b48a1138d77ced56feba2b467c398683c7db
eb86b854f120606a7ae1ed934f5703672adab0d7
be66dccde1a763c736cb9001d0731d541106f50b
b7e54240c40ba780b7a553bea570b99c9ab3df13
d75f8ccfdddeaaf3a749fd1411**

Message 2

The Internet knows no geographical boundaries. It has redefined time and space. Advances in computer and telecommunication technologies have led to the explosive growth of the Internet. This in turn is affecting the methods of communication, work, study, education, interaction, leisure, health, governance, trade and commerce.

Encryption Techniques…

- One-Way Function
  - non-reversible “quick” encryption
  - produces a fixed length value called a *hash* or *message digest*
  - used to authenticate contents of a message
  - Common message digest functions
    - MD4 and MD5
      - produces 128 bit hashes
    - SHA
      - produces 160 bit hashes

Cryptographic Services Allow

- Digital Signatures
  - sign messages to validate source and integrity of the contents
- Digital Envelopes (combination of symmetric/asymetric)
  - secure delivery of secret keys
- Message Digests
  - short bit string hash of message
- Digital Certificates
  - used to authenticate: users, web sites, public keys of public/private pair, and information in general
- Secure Channels
  - Encryption can be used to create secure channels over private or public networks

Building Blocks of Encryption Techniques

- Two building blocks of all classical encryption techniques are substitution and transposition.

- Substitution means replacing an element of the plaintext with an element of ciphertext.
  - each element in the plaintext (bit, letter, group of bits or letters) is mapped into another element
- Transposition means rearranging the order of appearance of the elements of the plaintext.

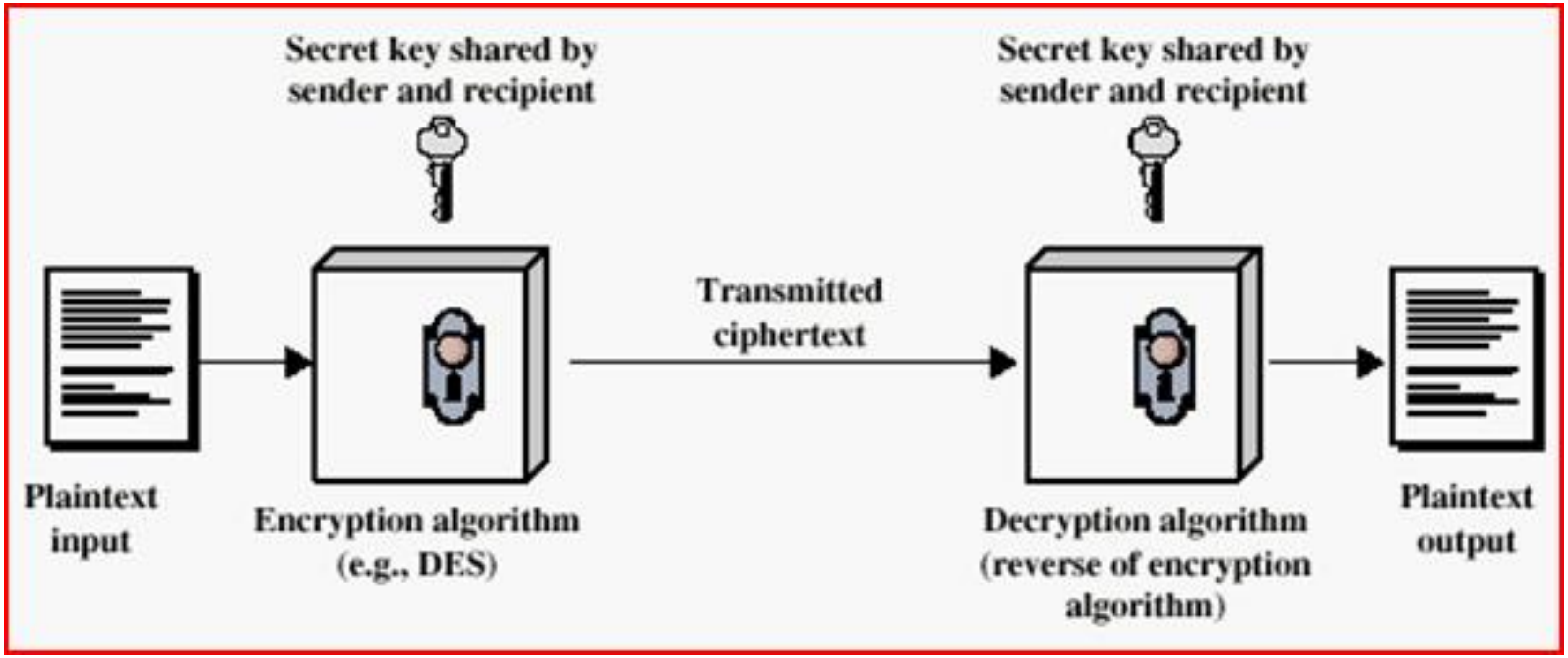
- Transposition is also referred to as permutation.

Cryptography…

- Cryptographic systems can be characterized along these three independent dimensions.
  - type of encryption operations used
    - substitution
    - transposition
    - product
  - number of keys used
    - single-key, secret-key, symmetric or private
    - two-key, asymmetric or public-key
  - way in which plaintext is processed
    - block
    - stream

Cryptography...

- Simplified Encryption Model:


Secret key shared by sender and recipient
Secret key shared by sender and recipient
Transmitted ciphertext
Plaintext input
Encryption algorithm (e.g., DES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Cryptography…

Description

- A sender S wants to transmit message M to a receiver R.
- To protect the message M, the sender first encrypts it into an intelligible message M'.
- After receipt of M', R decrypts the message to obtain M.
- M is called the plaintext
  - What we want to encrypt
- M' is called the ciphertext
  - The encrypted output

Cryptography…

- Mathematical Notation

➢ Given

  - P=Plaintext
  - C=Ciphertext

➢ $C = E_K (P)$ Encryption

➢ $P = D_K (C)$ Decryption

Cryptanalytic Attacks

- Types of attacks
  - An attacker has only the ciphertext and his goal is to find the corresponding plaintext.
  - An attacker has a ciphertext and the corresponding plaintext and his goal is to find the key.

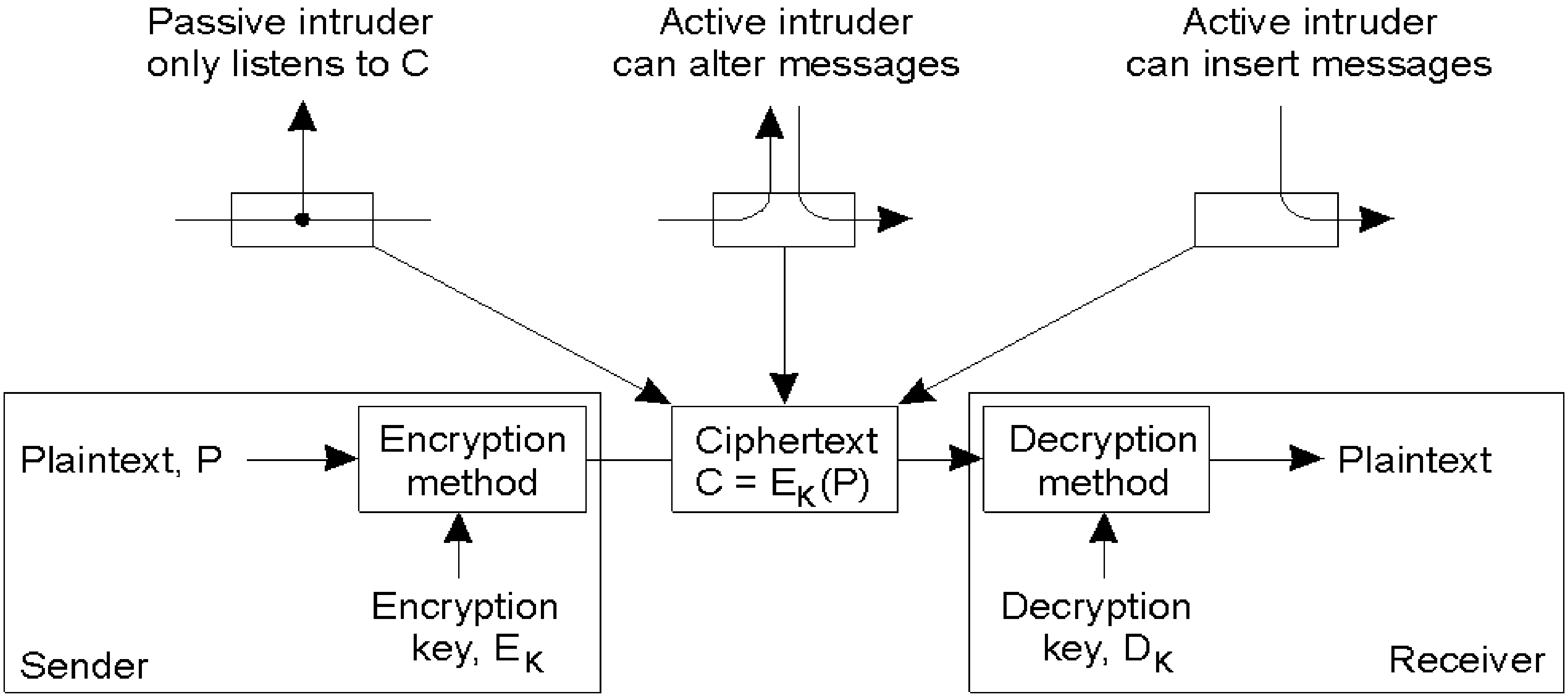
- A good cryptosystem protects against all types of attacks.
- Attackers use both Mathematics and Statistics.

Cryptanalytic Attacks…

- Intruders
  - Eavesdropping (listening/spying the message)
    - An intruder may try to read the message
    - If it is well encrypted, the intruder will not know the content
    - However, just the fact the intruder knows that there is communication may be a threat (Traffic analysis)
  - Modification
    - Modifying a plaintext is easy, but modifying encrypted messages is more difficult
  - Insertion of messages
    - Inserting new message into a ciphertext is difficult

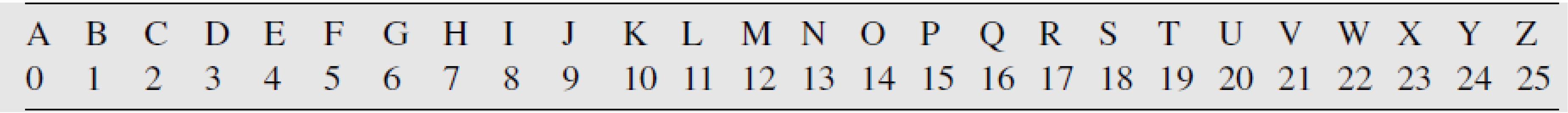
Cryptography and Cryptanalytic Attacks

- Intruders



Cryptography example:
Caesar cipher

- This is the earliest known example of a substitution cipher.
- Each character of a message is replaced by a character three position down in the alphabet.



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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

- Shift of letters:

Plain: ABCDEFGHIJKLMNOPQRSTUVWXYZ

Cipher: DEFGHIJKLMNOPQRSTUVWXYZABC

Example

plaintext: are you ready

ciphertext: duh brx uhdgb

Cryptography example: Caesar cipher

Example: Encipher the message

THIS MESSAGE IS TOP SECRET

- using the ordinary alphabet and a Caesar cipher with a shift of 3.
- When each letter is converted to a number, and we group into blocks of length 5, we get

19 7 8 18 12   4 18 18 0 6   4 8 18 19 14   15 18 4 2 17   4 19

- Here, we group the items in blocks for readability. After applying the enciphering transformation, each number becomes

22 10 11 21 15   7 21 21 3 9   7 11 21 22 17   18 21 7 5 20   7 22

- and the ciphertext message is sent as

WKLVP HVVDI HLVWR SVHFU HW

Cryptography example:
Caesar cipher

- If we represent each letter of the alphabet by an integer that corresponds to its position in the alphabet, the formula for replacing each character 'P' of the plaintext with a character 'C' of the ciphertext can be expressed as

$$C = E(3, P) = (P + 3) \bmod 26$$

- A more general version of this cipher that allows for any degree of shift would be expressed by

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

- The formula for decryption would be

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

- In these formulas, 'k' would be the secret key.
- The symbols 'E' and 'D' represent encryption and decryption.

WEAKNESSES OF THE CAESAR CIPHER

- The Caesar Cipher is a secret key cryptosystem;
  - that is, revealing the enciphering key makes decryption simple.
- In the Caesar cipher, the shift value is the enciphering key.
- Anyone knowing it can immediately decrypt, so it must be protected from unauthorized persons.

WEAKNESSES OF THE CAESAR CIPHER

- Exhaustive Key Search. There is yet another method for breaking the Caesar cipher:
- simply try all the possible keys!
  - After all, there are only 26 viable keys in the ordinary alphabet, and only 255 useful keys in the ASCII alphabet! This kind of attack is called an exhaustive search.
- An exhaustive search is rarely effective against all but the simplest of cryptosystems.
- Seeing that the Caesar cipher is so vulnerable, we attempt to see stronger cryptosystems.

Ciphering with Transposition

- So far we have seen ciphering with substitution.
- We will now talk about a different notion in classical cryptography: permuting the plaintext.

- This is how a pure permutation cipher could work:
  - You write your plaintext message along the rows of a matrix of some size.
  - You generate ciphertext by reading along the columns.
  - The order in which you read the columns is determined by the encryption key.

Ciphering with Transposition…

Key: 4 1 3 6 2 5

Plaintext:

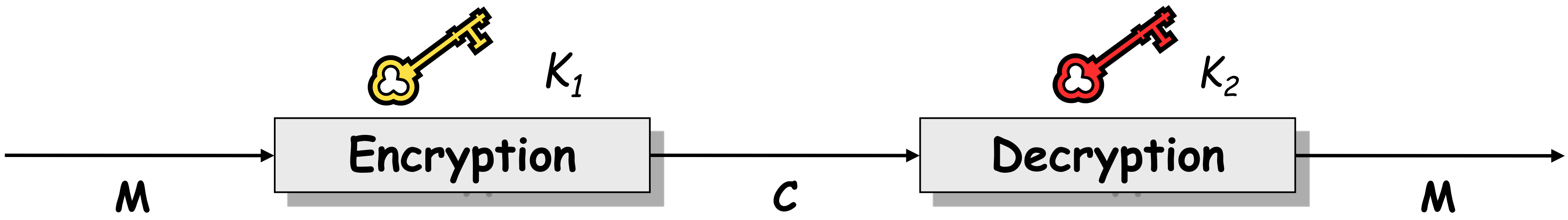
m	e	e	t	m	e
a	t	s	q	u	a
r	e	g	u	a	r
d	e	n	f	o	r
g	o	o	d	d	i
n	n	e	r	o	k

Ciphertext: tqufdrmardgnesgnoeearriketeeonmuaodo

The cipher can be made more secure by performing multiple rounds of such permutations.

Symmetric and Asymmetric ciphering

- Symmetric: the same key is used to encrypt the data
  - Both sides of the communication must have the same key
  - Examples: DES, AES, Blowfish, RC2, RC5, IDEA…
- Asymmetric: different keys are used to encrypt and decrypt the data
  - Example: RSA,DH, Elgamal…



- More formally, using maths:
- Notation
  - Plain text: M
  - Encrypted text: C
  - Encryption with key K_1: $E_{K_1}(M) = C$
  - Decryption with key K_2: $D_{K_2}(C) = M$
- Algorithms
  - **Symmetric**: $\boldsymbol{K_1 = K_2}$
  - **Asymmetric:** $\boldsymbol{K_1 \neq K_2}$

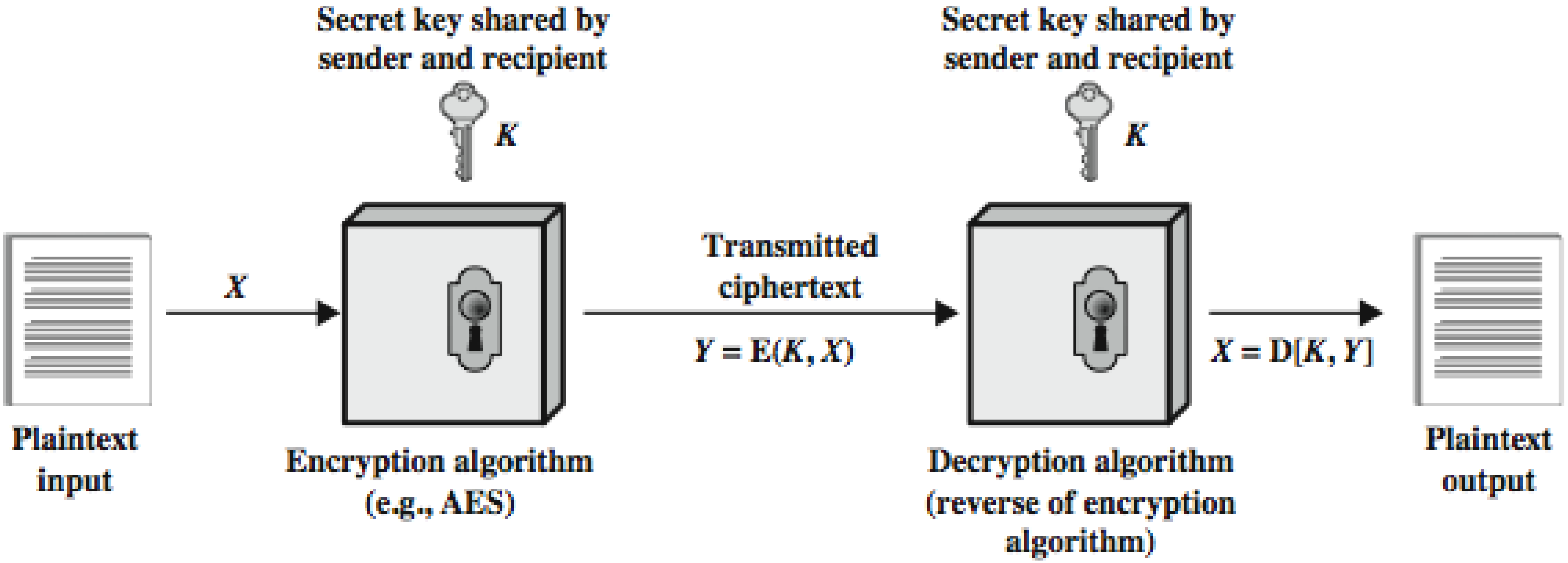
Symmetric and Asymmetric ciphering...

- Symmetric Cryptography
- Also called secret-key/private-key cryptosystem
- The same key is used to encrypt and decrypt a message

$$C = D_K [E_K (P)]$$

- Have been used for centuries in a variety of forms
- The key has to be kept secret
- The key has to be communicated using a secure channel
- They are still in use in combination with public-key cryptosystems due to some of their advantages

Symmetric Cipher Model


Secret key shared by sender and recipient
K
Secret key shared by sender and recipient
K
X
Transmitted ciphertext
Y = E(K, X)
X = D[K, Y]
Plaintext input
Encryption algorithm (e.g., AES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Requirements

- Two requirements for secure use of symmetric encryption:
  - *a strong encryption algorithm*
  - *a secret key known only to sender / receiver*
- mathematically have:

  $C = E(K, P)$ done by sender side

  $P = D(K, C)$ receiver side
- assume encryption algorithm is known
- implies a secure channel to distribute key

Asymmetric ciphering

- Asymmetric Cryptography

- Also called public-key cryptosystem
  - keys for encryption and decryption are different but form a unique pair

    $$C = D_{KD}\ [E_{KE}\ (P)]$$

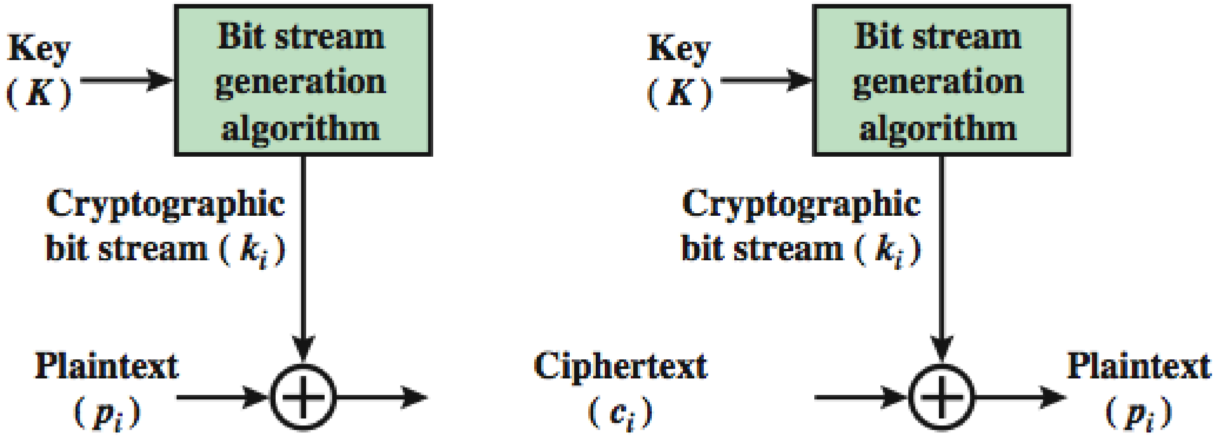
  - Only one of the keys need to be private while the other can be public.

- Invented by Diffie and Hellman in 1976.
- It is a revolutionary concept since it avoids the need of using a secure channel to communicate the key.
- It has made cryptography available for the general public and made many of today's on-line application feasible.

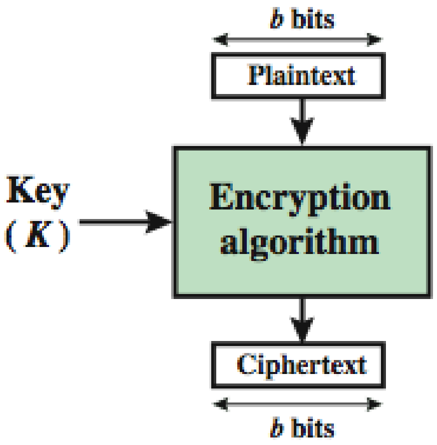
Block vs Stream Ciphers

- Block ciphers process messages into blocks, each of which is then en/decrypted
- like a substitution on very big characters
  - 64-bits or more
- Stream ciphers process messages a bit or byte at a time when en/decrypting
- many current ciphers are block ciphers
- hence are focus of this course

Block vs Stream Ciphers


Key
(K)
Bit stream generation algorithm
Cryptographic bit stream (k_i)
Plaintext
(p_i)
Key
(K)
Bit stream generation algorithm
Cryptographic bit stream (k_i)
Ciphertext
(c_i)
Plaintext
(p_i)

(a) Stream Cipher Using Algorithmic Bit Stream Generator



(b) Block Cipher

Substitution-Permutation Ciphers

- In his 1949 paper, Shannon introduced the idea of substitution-permutation (S-P) networks, which now form the basis of modern block ciphers.

- an S-P network is the modern form of a substitution-transposition product cipher.

- S-P networks are based on the two primitive cryptographic operations we have seen before

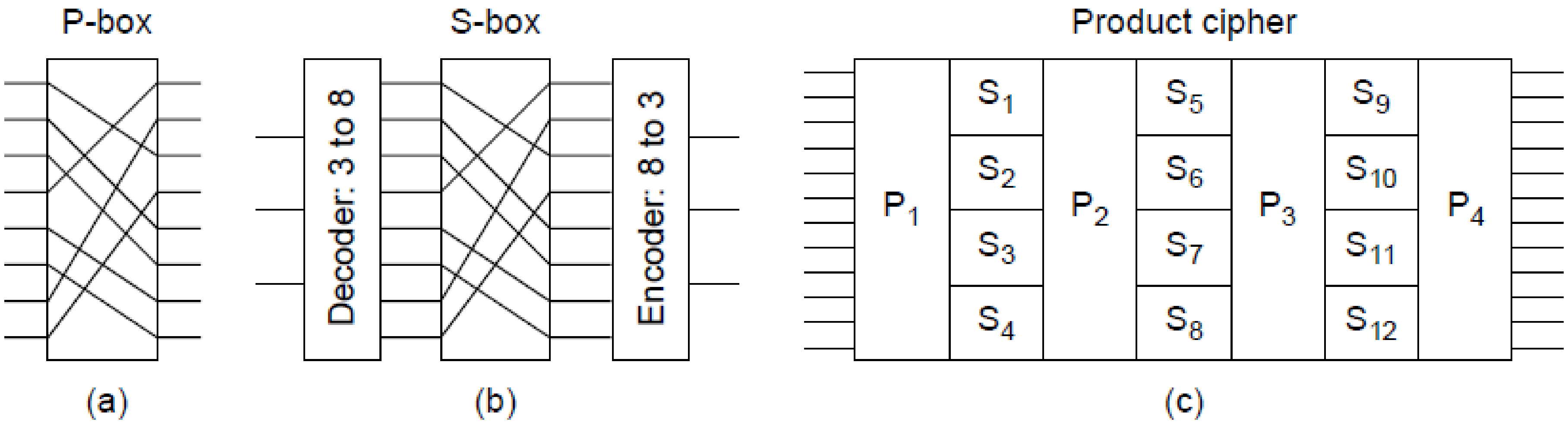
Substitution-Permutation Ciphers…

- Substitution Operation
  - a binary word is replaced by some other binary word
  - *the whole substitution function forms the key*
  - if use n bit words, the key is 2^n ! bits, grows rapidly
  - will call them S-Boxes
- Permutation Operation
  - a binary word has its bits reordered (permuted)
  - *the re-ordering forms the key*
  - if use n bit words, the key is n! bits, which grows more slowly, and hence is less secure than substitution
  - will call these P-Boxes

Substitution-Permutation Ciphers…

- Shannon combined these two primitives
- he called these mixing transformations
- Shannon's mixing transformations are a special form of product ciphers where
- S-Boxes
  - provide **diffusion** of input bits
- P-Boxes
  - provide **confusion** across S-box inputs

Substitution-Permutation Ciphers…



Basic elements of product ciphers.
(a) P-box. (b) S-box. (c) Product.

Cryptographic Algorithms

- Block ciphers (secret/symmetric key, DES)
- Hashes (digital signature)
- Diffie-Hellman key exchange
- RSA (public key encryption and digital signature)
- ElGamal digital signature
- IDEA, RC2, RC5, Blowfish, and many more

Symmetric Block Cipher Algorithms

- DES (Data Encryption Standard)
- 3DES (Triple DES)
- AES (Advanced Encryption Standard)

Data Encryption Standard (DES)

- most widely used block cipher in world
- adopted in 1977 by NBS (now NIST)
  - NBS-National Bureau of Standards
  - NIST - National Institute of Standards and Technology
- encrypts 64-bit data using 56-bit key
- has widespread use
- has been considerable controversy over its security

Symmetric DES…

- The basic process in enciphering a 64-bit data block using the DES consists of:
  - an initial permutation (IP)
  - 16 rounds of a complex key dependent calculation f
  - a final permutation, being the inverse of IP

Symmetric DES…

- DES Utilizes block cipher.
  - During the encryption process, the plaintext is divided into fixed length blocks of 64 bits.
- The key is 56 bits wide. 8-bit out of the total 64-bit block key is used for parity check.
- 56-bit key gives 2^{56} ($\cong 7.2*10^{16}$) possible key variations.
- DES algorithm involves carrying out combinations of substitutions and permutations between the text to be encrypted and the key,
  - while making sure the operations can be performed in both directions (for decryption).
- The combination of substitutions and permutations is called a product cipher.

Symmetric DES…

- DES Encryption starts with an initial permutation (IP) of the 64 input bits.
- These bits are then divided into two 32-bit halves called L and R.
- The encryption then proceeds through 16 rounds, each using the L and R parts, and a roundkey.
- The R and roundkeys are processed in the so called *f*-function, and exclusive-or of the output of the *f*-function with the existing L part to create the new R part.
- The new L part is simply a copy of the incoming R part.

Symmetric DES…

- In the final round, the L and R parts are swapped once more before the final permutation (FP) producing the output block.
- Decryption is identical to encryption, except that the subkeys are used in the opposite order.
- That is, roundkey 16 is used in round 1, roundkey 15 is used in round 2, etc., ending with roundkey 1 being used in round 16.

Symmetric DES...

- The *f*-function mixes the bits of the R portion using the roundkey for the current round.
- First the 32-bit R value is expanded to 48 bits using a permutation E. That value is then exclusive-or'ed with the roundkey.
- The 48 bits are then divided into eight 6-bit chunks, each of which is fed into an S-Box that mixes the bits and produces a 4-bit output.
- Those 4-bit outputs are combined into a 32-bit value, and permuted once again to produce the *f*-function output.

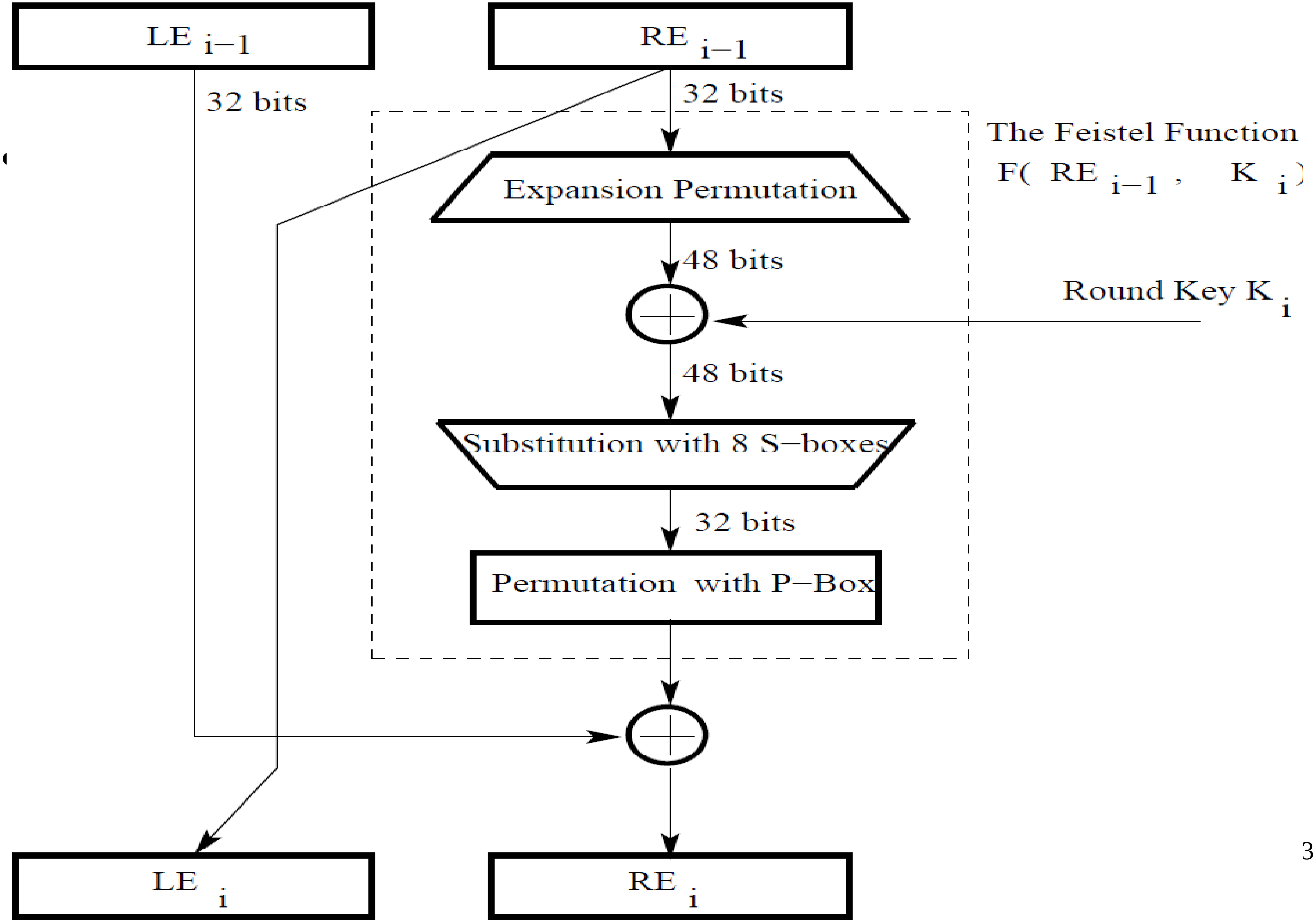
Symmetric DES...

One Round of Processing in DES

- The algorithmic implementation of DES is known as DEA for Data Encryption Algorithm.
- Figure shows a single round of processing in DEA.
  - The dotted rectangle constitutes the F function.
- The 32-bit right half of the 64-bit input data block is expanded by into a 48-bit block.
- This is referred to as the expansion permutation step, or the E-step.

Symmetric DES...

One Round of Processing in DES



Symmetric DES...

One Round of Processing in DEA

- The above-mentioned E-step involves the following:
  - First divide the 32-bit block into eight 4-bit words
  - attach an additional bit on the left to each 4-bit word that is the last bit of the previous 4-bit word
  - attach an additional bit to the right of each 4-bit word that is the beginning bit of the next 4-bit word.
- The 56-bit key is divided into two halves,
  - each half shifted separately, and the combined 56-bit key permuted/contracted to yield a 48-bit round key.
- How this is done will be explained later.

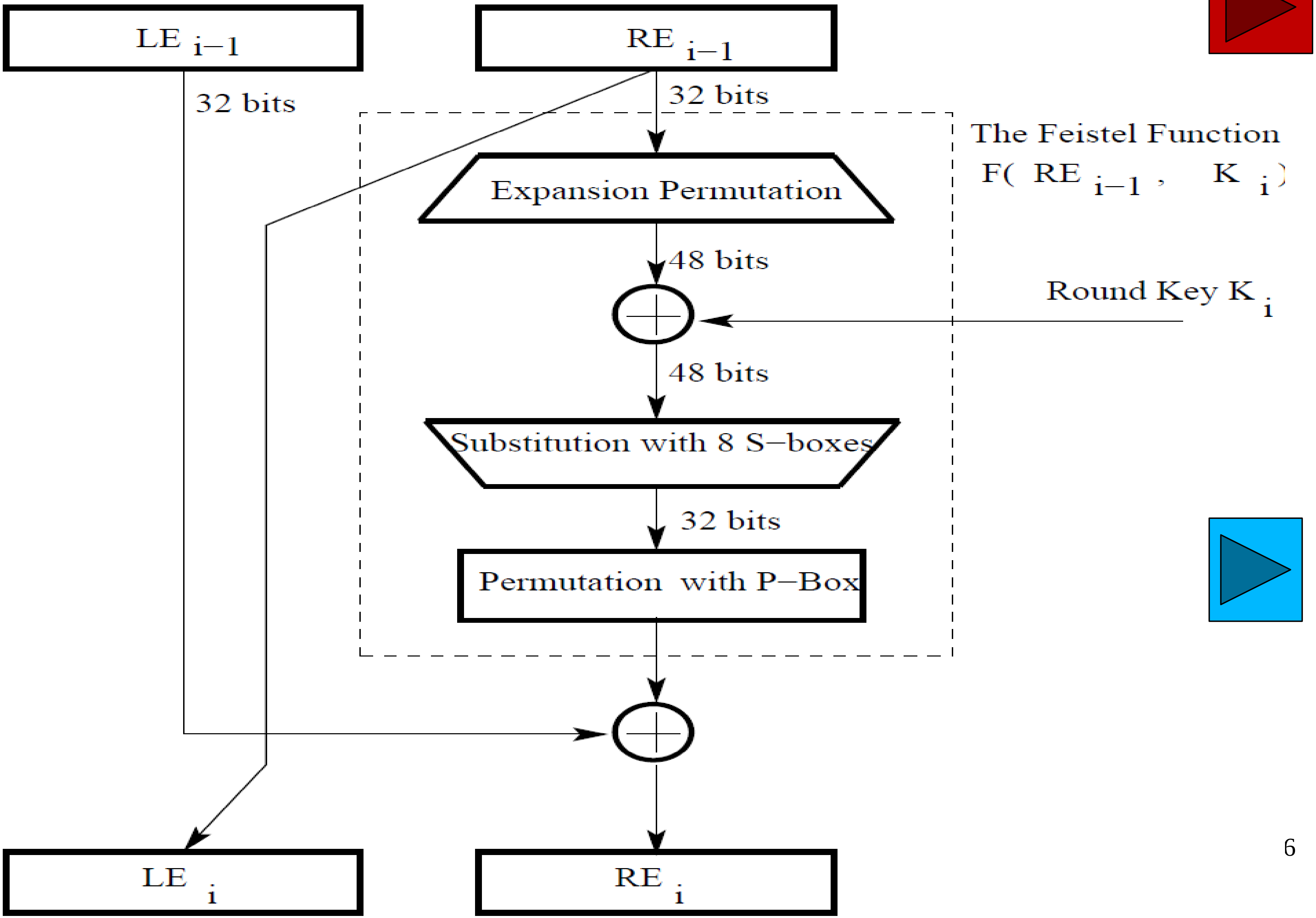
Symmetric DES...

One Round of Processing in DEA

- The 48 bits of the expanded output produced by the E-step are XORed with the round key.
  - This is referred to as key mixing.
- The output produced by the previous step is broken into eight six-bit words.
- Each six-bit word goes through a substitution step.
  - its replacement is a 4-bit word.
- The substitution is carried out with an S-box.
- So after all the substitutions, we again end up with a 32-bit word.

Symmetric DES...

One Round of Processing in DEA



Symmetric DES…

One Round of Processing in DEA

- The 32-bits of the previous step then go through a P-box based permutation.
- What comes out of the P-box is then XORed with the left half of the 64-bit block that we started out with.
- The output of this XORing operation gives us the right half block for the next round.

Symmetric DES...

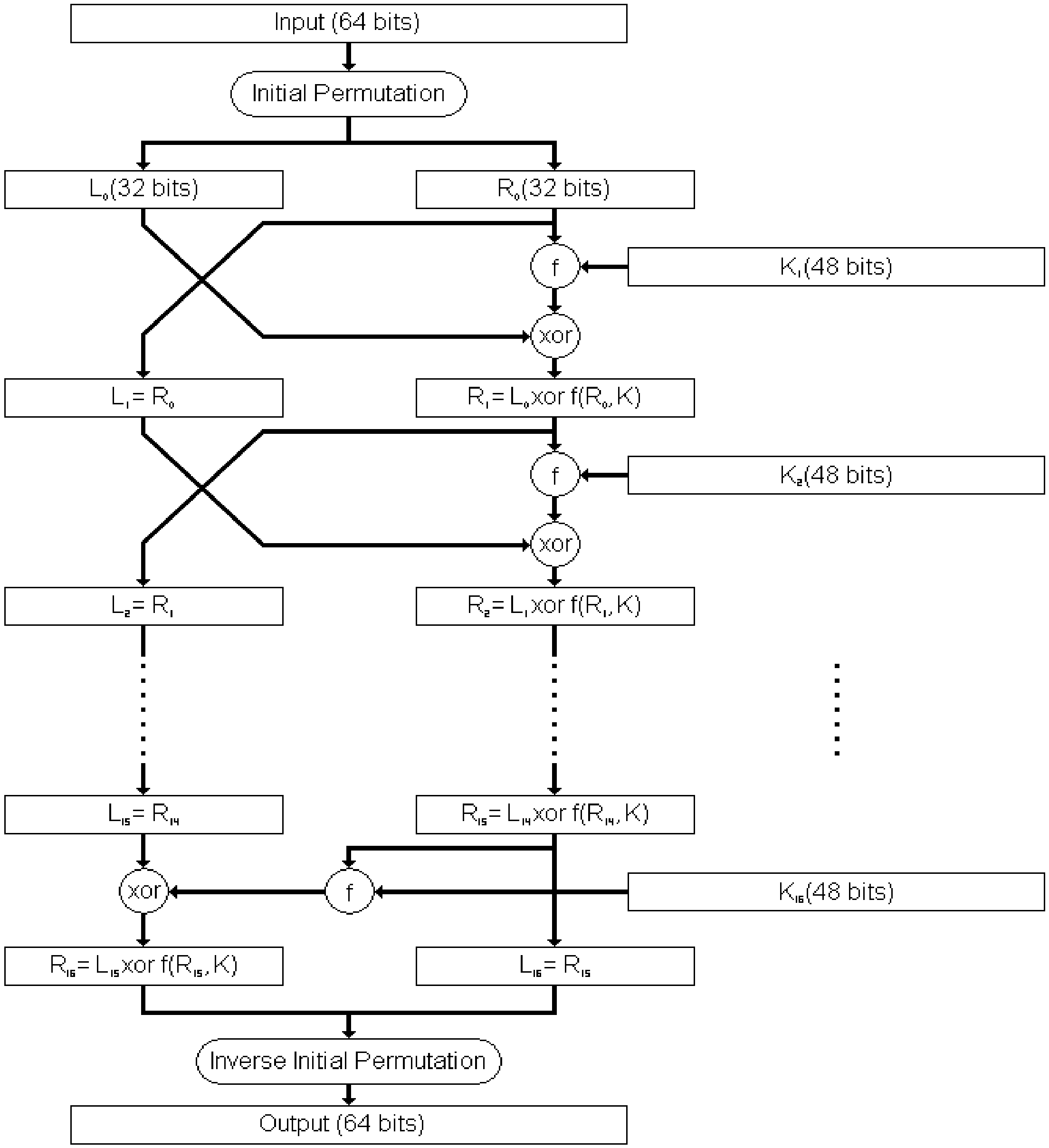
One Round of Processing in DEA

NOTE

- The goal of the substitution step implemented by the S-box is to introduce **diffusion** in the generation of the output from the input.
  - *Diffusion means that each plaintext bit must affect as many ciphertext bits as possible.*
- The strategy used for creating the different round keys from the main key is meant to introduce **confusion** into the encryption process.
  - *Confusion in this context means that the relationship between the encryption key and the ciphertext must be as complex as possible.*
- Diffusion and confusion are the two cornerstones of block cipher design.

Symmetric DES...

- DES algorithm



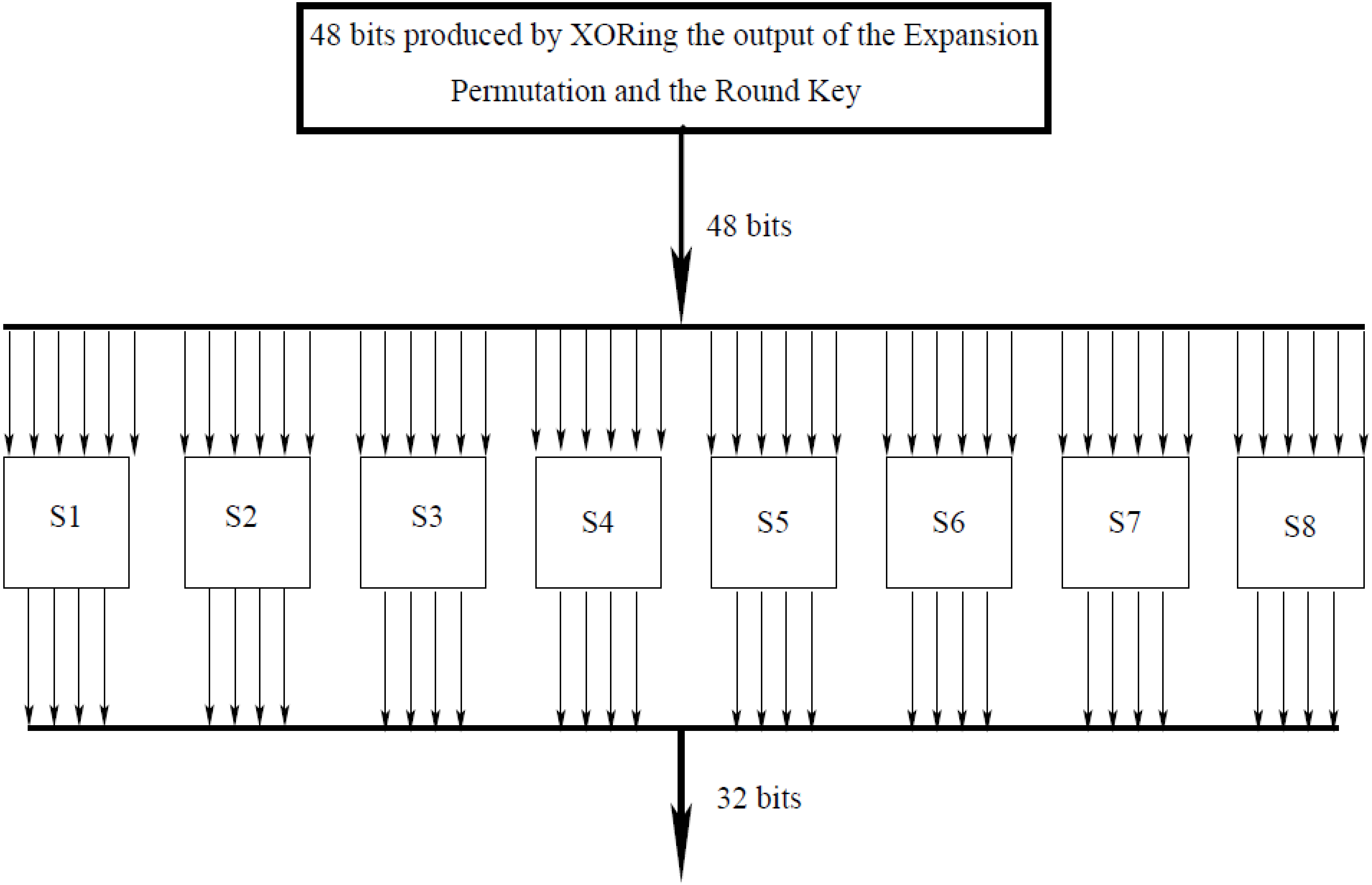
Symmetric DES

The S-Box for the Substitution Step in Each Round

- The 48-bit input word is divided into eight 6-bit words and each 6-bit word fed into a separate S-box.
- Each S-box produces a 4-bit output. Therefore, the 8 S-boxes together generate a 32-bit output.
- The overall substitution step takes the 48-bit input back to a 32-bit output.
- Each of the eight S-boxes consists of a 4×16 table lookup for an output 4-bit word.
  - The first and the last bit of the 6-bit input word are decoded into one of our rows and
  - The middle 4 bits into one of 16 columns for the table lookup.

Symmetric DES

The S-Box for the Substitution Step in Each Round


48 bits produced by XORing the output of the Expansion Permutation and the Round Key
48 bits
S1
S2
S3
S4
S5
S6
S7
S8
32 bits

Symmetric DES

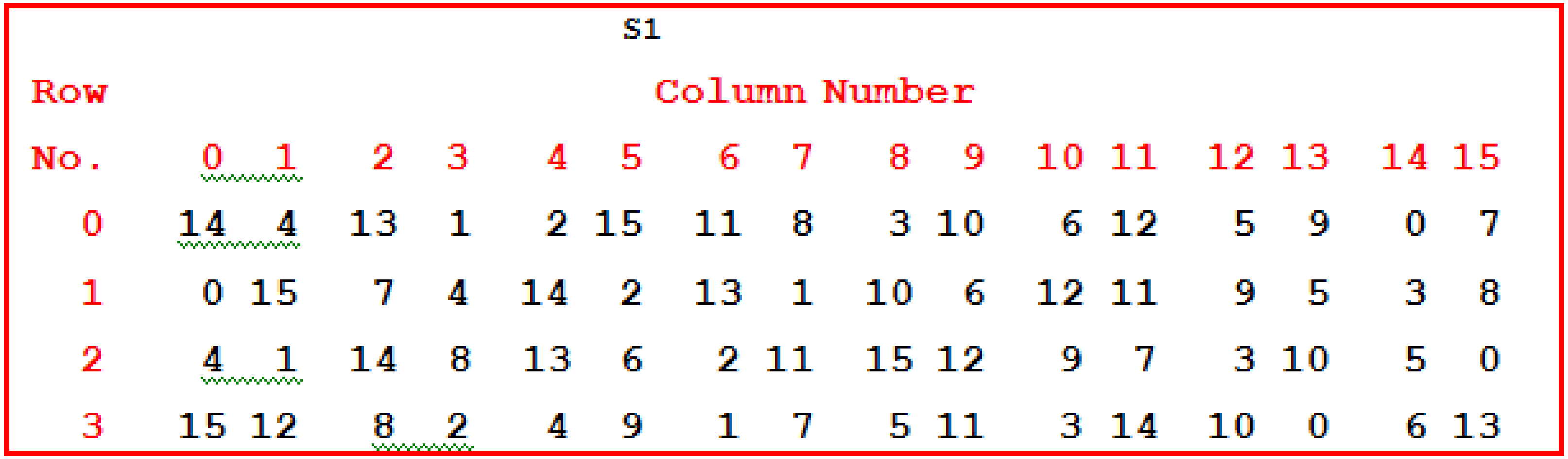
The S-Box for the Substitution Step in Each Round

- The goal of the substitution carried out by an S-box is to enhance diffusion.
- From the E-step described before, the expansion-permutation step (the E-step) expands a 32-bit block into a 48-bit block
  - by attaching a bit at the beginning and a bit at the end of each 4-bit sub-block.
- The two bits needed for these attachments belong to the adjacent blocks.
- Thus, the row lookup for each of the eight S-boxes becomes a function of the input bits for the previous S-box and the next S-box.

Symmetric DES

The S-Box for the Substitution Step in Each Round

The S-Box



S1

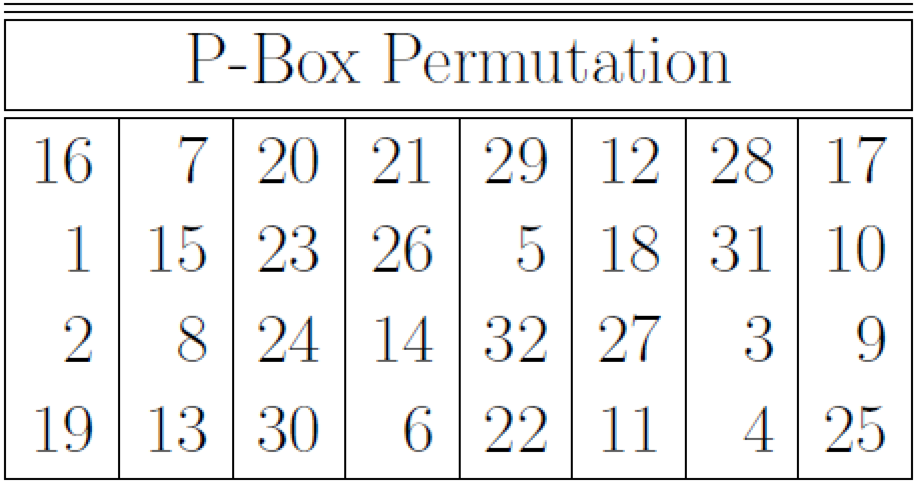
Row No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Column Number															
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

If $\boldsymbol{S_1}$ is the function defined in this table and ***B*** is a block of 6 bits, then $\boldsymbol{S_1(B)}$ is determined as follows: The first and last bits of ***B*** represent in base 2 a number in the decimal range 0 to 3 (or binary 00 to 11). Let that number be ***i***. The middle 4 bits of ***B*** represent in base 2 a number in the decimal range 0 to 15 (binary 0000 to 1111). Let that number be ***j***. Look up in the table the number in the ***i***-th row and ***j***-th column. It is a number in the range 0 to 15 and is uniquely represented by a 4 bit block. That block is the output $\boldsymbol{S_1(B)}$ of $\boldsymbol{S_1}$ for the input ***B***. For example, for input block ***B*** = 011011 the first bit is "0" and the last bit "1" giving 01 as the row. This is row 1. The middle four bits are "1101". This is the binary equivalent of decimal 13, so the column is column number 13. In row 1, column 13 appears 5. This determines the output; 5 is binary 0101, so that the output is 0101. Hence $\boldsymbol{S_1}$(011011) = 0101.

Symmetric DES

The P-Box Permutation in Feistel Function

- The last step in the Feistel function is labeled “Permutation with P-Box”.
- This permutation table simply means that the first output bit will be the *16th* bit of the input, the second output bit the *7th* bit of the input, and so on.



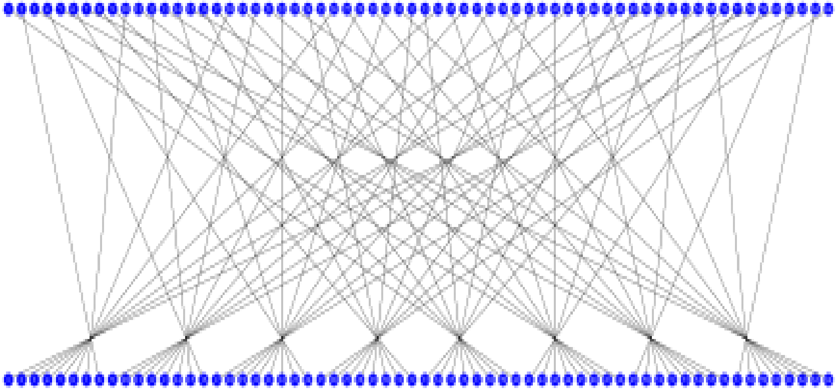
P-Box Permutation							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

- For all of the 32 bits of the output that are obtained from the 32 bits of the input.
- NOTE
  - *bit indexing starts with 1 and not with 0.*

Symmetric DES

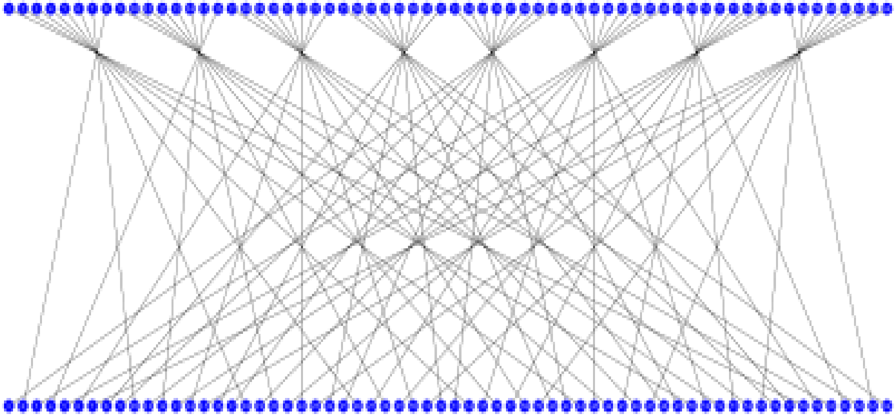
The P-Box Permutation in Feistel Function

Initial Permutation (IP)



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 (FP)



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

"First Bit of the output is taken from the 58th bit of the input, etc..."

Symmetric DES

Round Key Generation

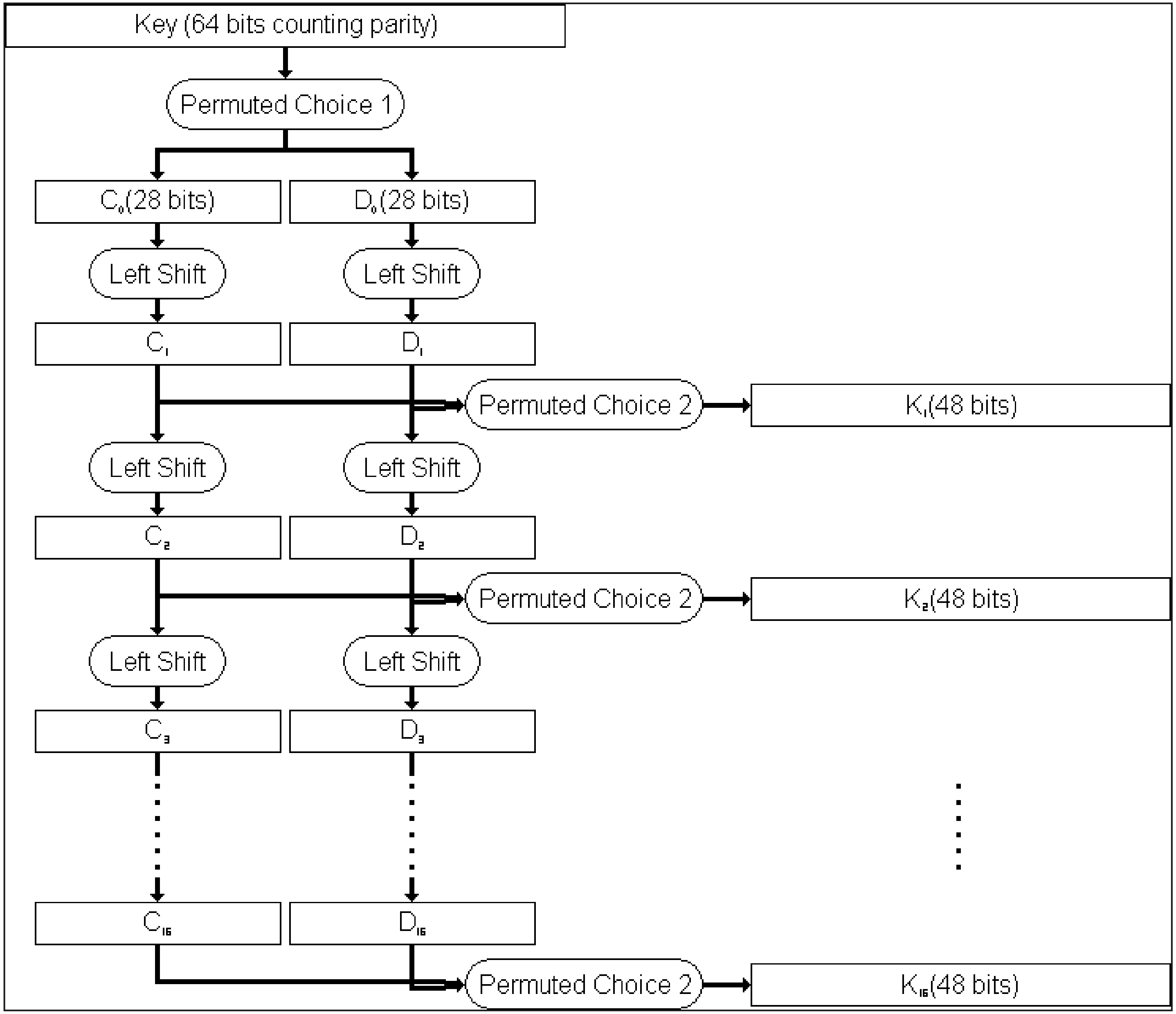
- The initial 64-bit key may be represented as 8 bytes, with the last bit of each byte used as a parity bit.
- The relevant 56 bits are subject to a permutation at the beginning before any round keys are generated. (permutation choice 1)
- At the beginning of each round,
  - we divide the 56 relevant key bits into two 28 bit halves and
  - circularly shift left each half by one or two bits.

Symmetric DES…
Round Key Generation

- To generate the roundkeys, start with the 56-bit key (64 bits if you include the parity bits).
- These are permuted and divided into two halves called C and D.
- For each round, C and D are each shifted left circularly one or two bits.
- The 48-bit roundkey is then selected from the current C and D bits.

Symmetric DES…

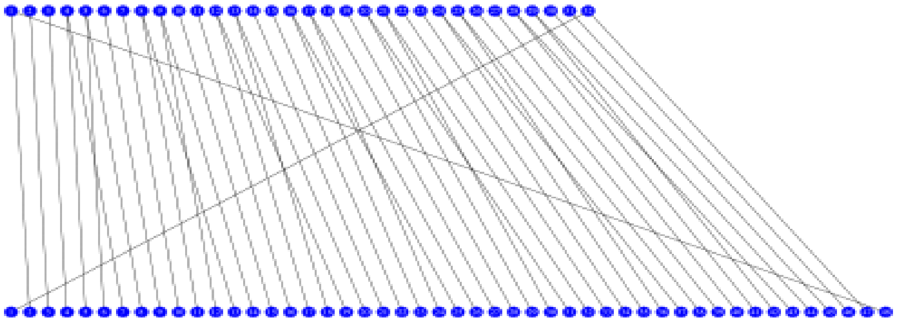
Round Key Generation



Symmetric DES

Permutation in the Feistel Function

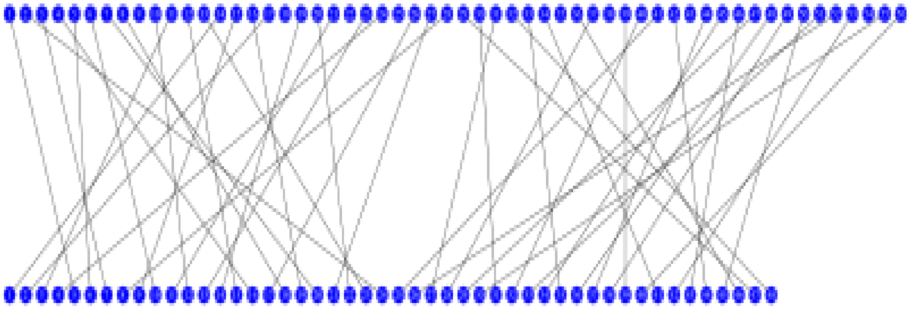
Expansion/Permutation



The 32-bit half-block of data is expanded to 48 bits.

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

Contraction/Permuted Choice (PC-2)

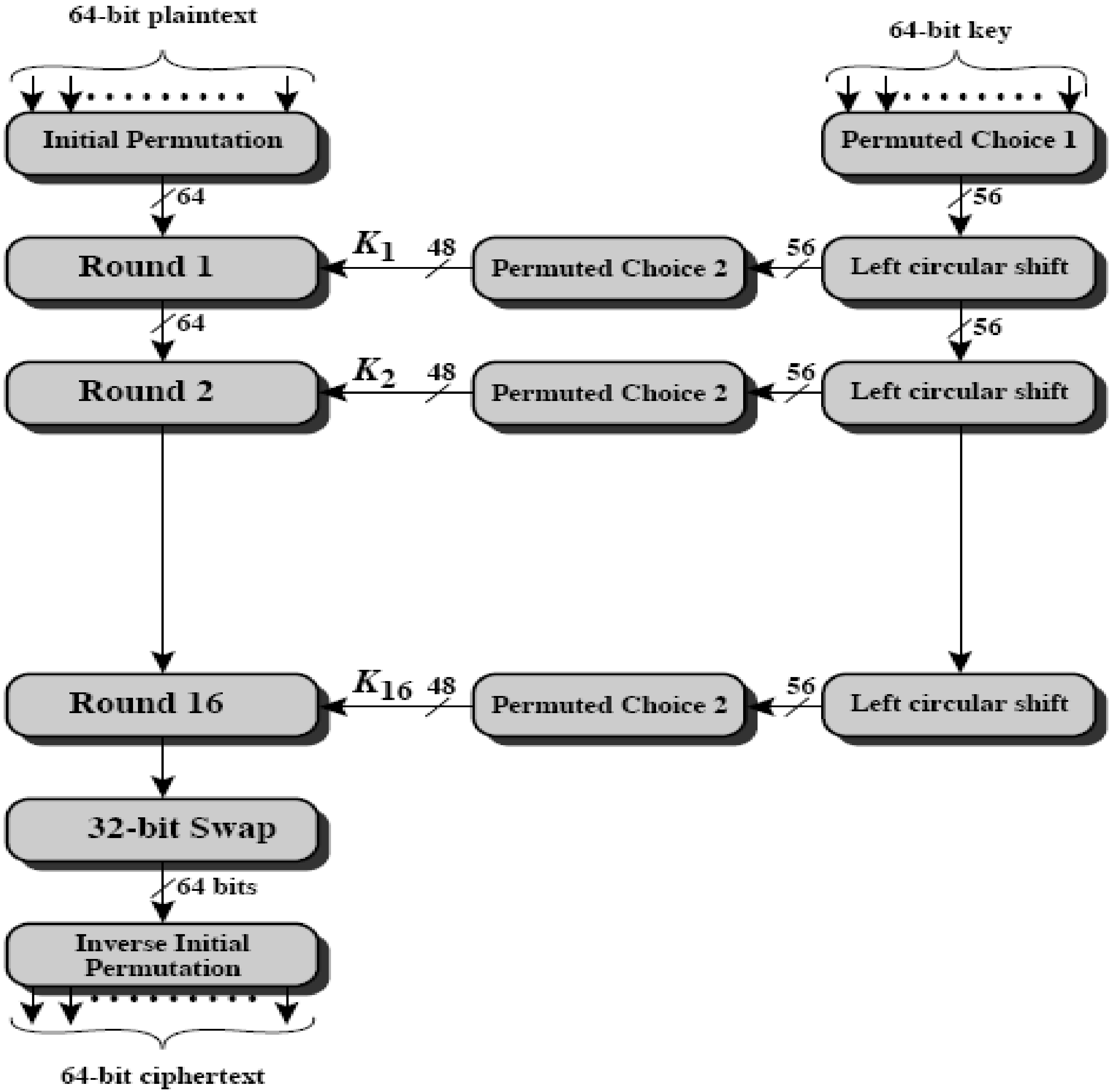


Selects/Extracts the 48-bit subkey for each round from the 56-bit key-schedule state.

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

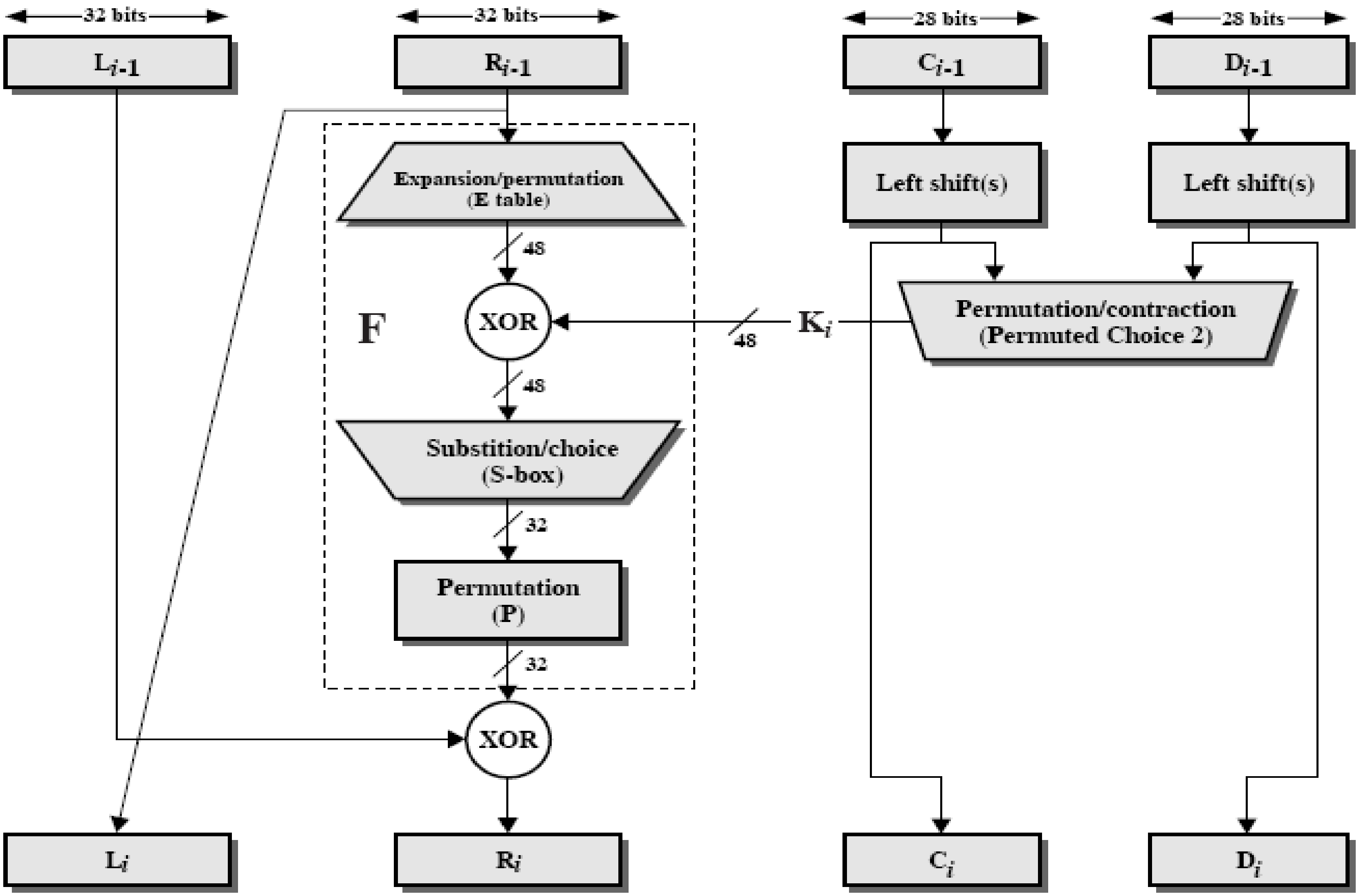
Symmetric DES...

- DES- Algorithm, General representation



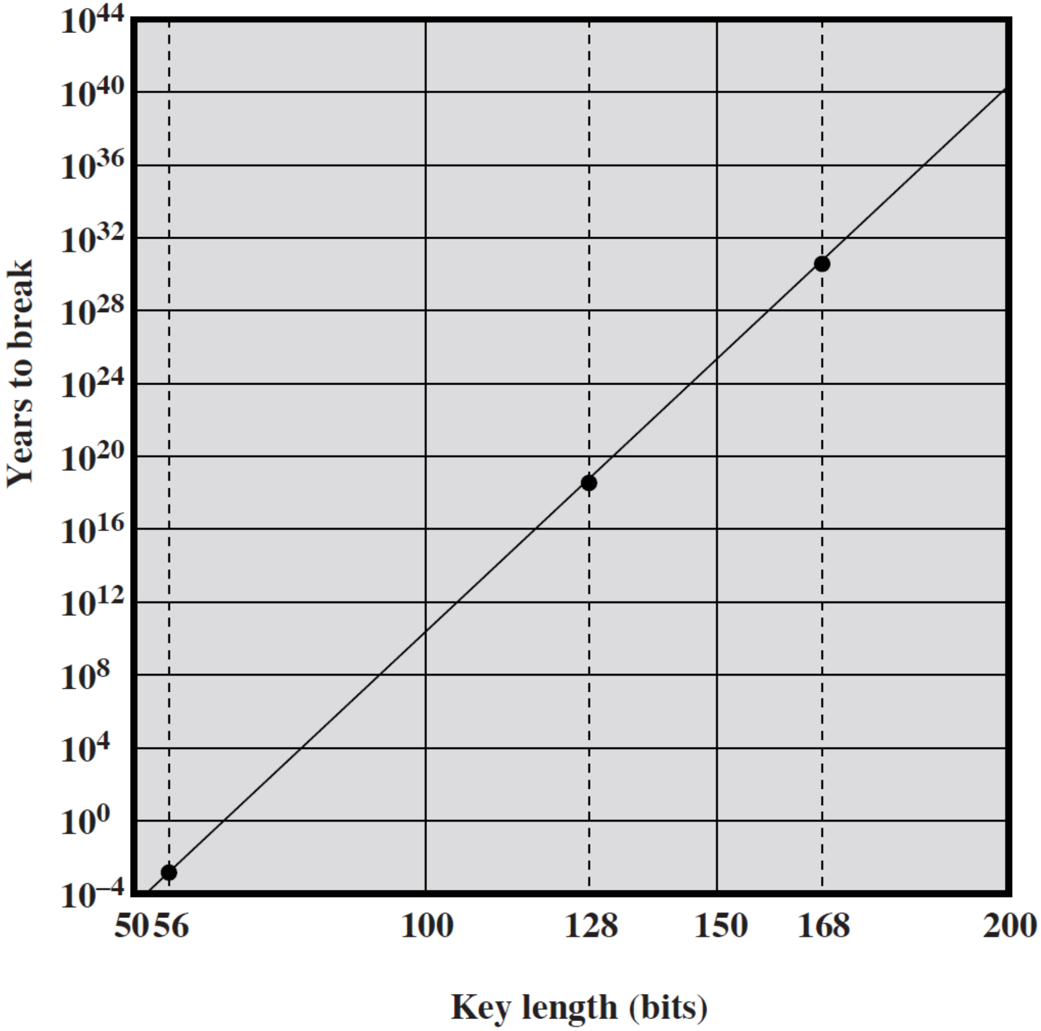
Symmetric DES...

Single round of DES Algorithm



Click for
DES Example

Time to Break a DES Code (assuming 106 decryptions/μs)



Strength of DES – Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
- brute force search looks hard
- recent advances have shown is possible
  - in 1997 on Internet in a few months
  - in 1998 on dedicated h/w (EFF) in a few days
  - in 1999 above combined in 22hrs!
- now considering alternatives to DES

Symmetric DES…

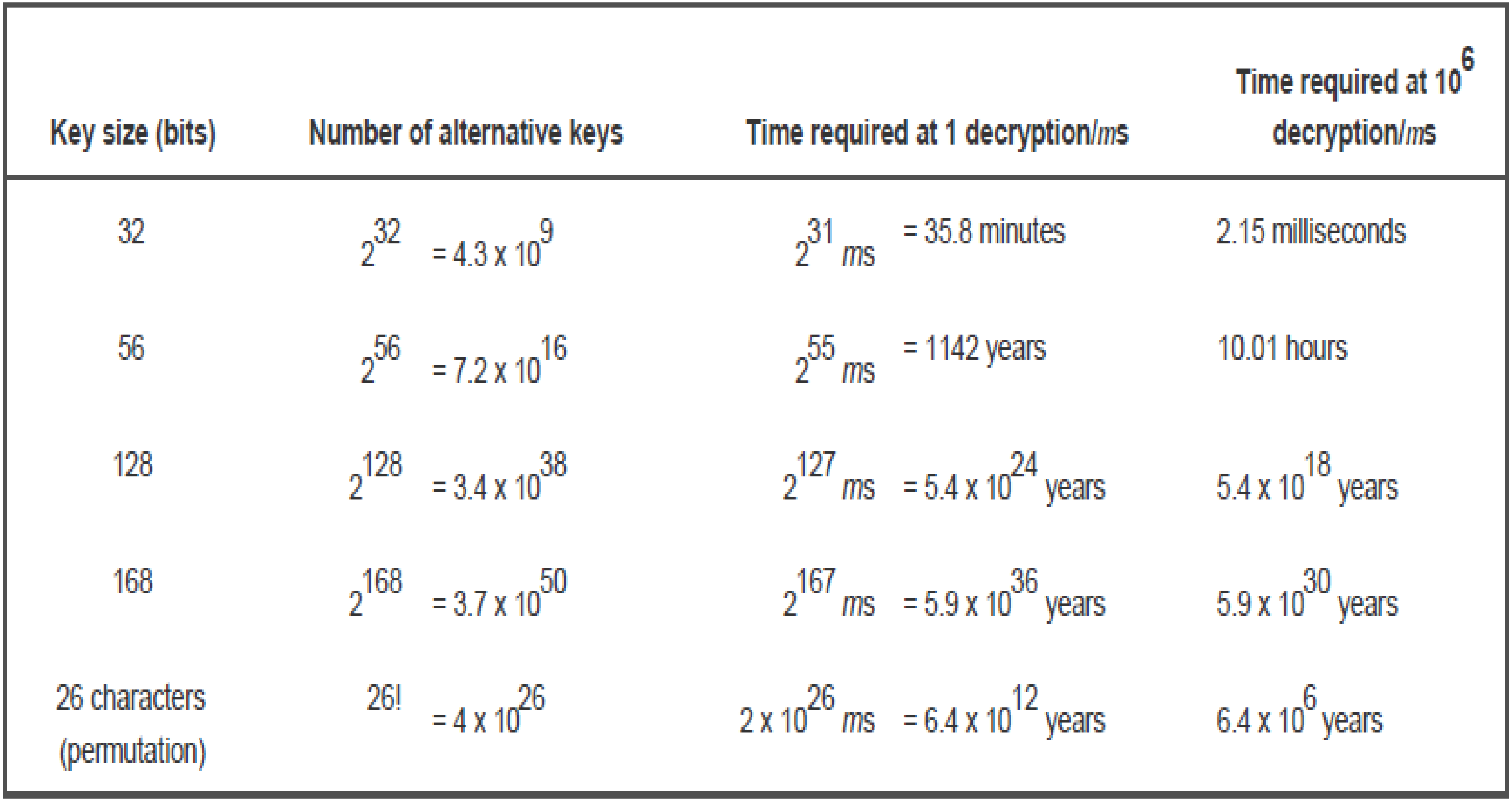
- Cracking: The most basic method of attack for any cipher is brute force - trying every possible key in turn.

- The length of the key determines the number of possible keys, and hence the feasibility of the approach.

- DES is not adequate with this regard due to its key size

- In academia, various proposals for a DES-cracking machine were advanced.
  - ➢ In 1977, Diffie and Hellman proposed a machine, which could find a DES key in a single day.
  - ➢ By 1993, Wiener had proposed a key-search machine costing US $1 million which would find a key within 7 hours.

Average time required for exhaustive key search

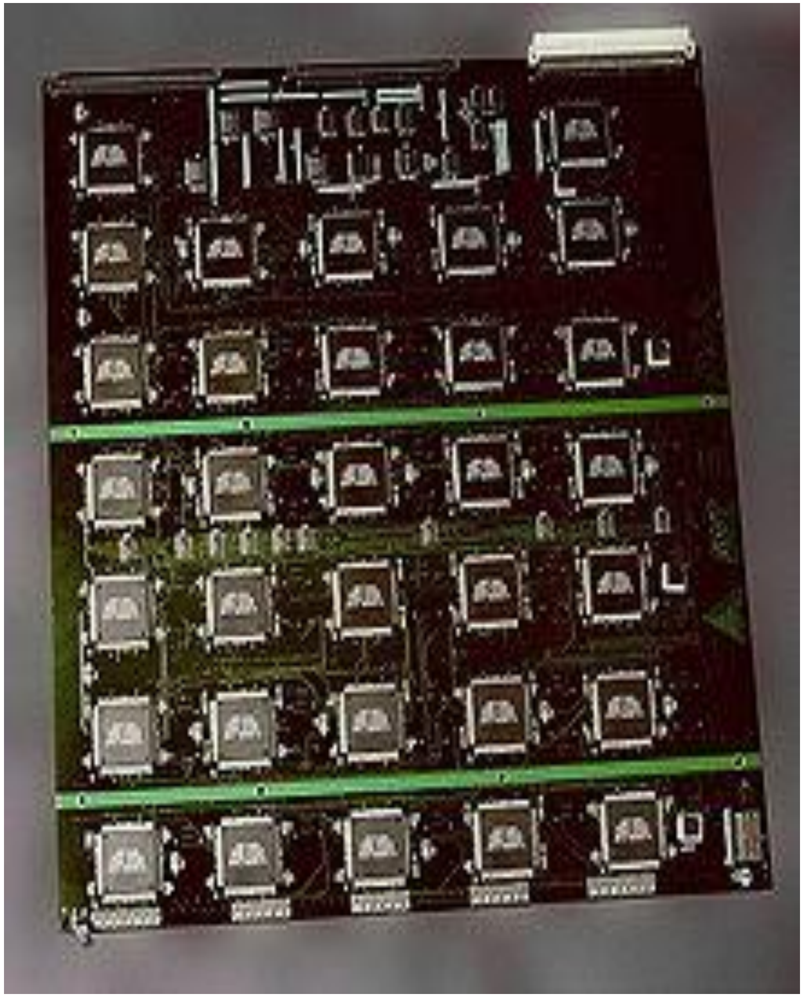


Key size (bits)	Number of alternative keys	Time required at 1 decryption/*ms*	Time required at 10^6 decryption/*ms*
32	$2^{32} = 4.3 \times 10^9$	2^{31} *ms* = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} *ms* = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} *ms* = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} *ms* = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	26! = 4×10^{26}	2×10^{26} *ms* = 6.4×10^{12} years	6.4×10^{6} years

Symmetric DES...

- DES finally and definitively proved insecure in July 1998, when the Electronic Frontier Foundation (EFF) announced that it had broken a DES encryption using a special-purpose "DES cracker" machine that was built for less than $250,000.
- The attack took less than three days.
- The EFF has published a detailed description of the machine, enabling others to build their own cracker [EFF98].

Symmetric DES...



- The EFF's US$250,000 DES cracking machine contained 1,856 custom chips and could brute force a DES key in a matter of days.
- The photo shows a DES Cracker circuit board fitted with several Deep Crack chips.

Multiple Encryption with DES for a More Secure Cipher

- As you already know, the DES cryptographic system was shown not to be very secure about 15 years ago.
- We can obviously use AES cryptography that is designed to be extremely secure, but the world of commerce and finance does not want to give up on DES that quickly
  - because of all the investment that has already been in DES-related software and hardware.
- So that raises questions like: How about a cryptographic system that carries out repeated encryption with DES? Would that be more secure?
- We will now show that whereas double DES may not be that much more secure than regular DES, we can expect triple DES to be very secure.

Double DES

- The simplest form of **multiple encryption** with DES is **double DES** that has two DES-based encryption stages using two different keys.
- Let's say that P represents a 64-bit block of plaintext.
- Let E represent the process of encryption that transforms a plaintext block into a ciphertext block.
- Let's use two 56-bit encryption keys K_1 and K_2 for a double application of DES to the plaintext.
- Let C represent the resulting block of ciphertext. We have

$$C = E(K_2, E(K_1, P))$$

$$P = D(K_1, D(K_2, C))$$

- where D represents the process of decryption.

Triple-DES with Two-Keys

- Triple-DES with two keys is a popular alternative to single-DES,
- but suffers from being 3 times slower to run.
- Although there are no practical attacks, have some indications of attack approaches.
- Hence some are now adopting Triple-DES with three keys for greater security.

Triple-DES with Two-Keys…

- Hence must use 3 encryptions
  - would seem to need 3 distinct keys
- but can use 2 keys with E-D-E sequence
  - $C = E_{K1}[D_{K2}[E_{K1}[P]]]$
  - nb encrypt & decrypt equivalent in security
  - if $K_1=K_2$ then can work with single DES
- standardized in ANSI X9.17 & ISO8732
- no current known practical attacks

Triple-DES with Three-Keys

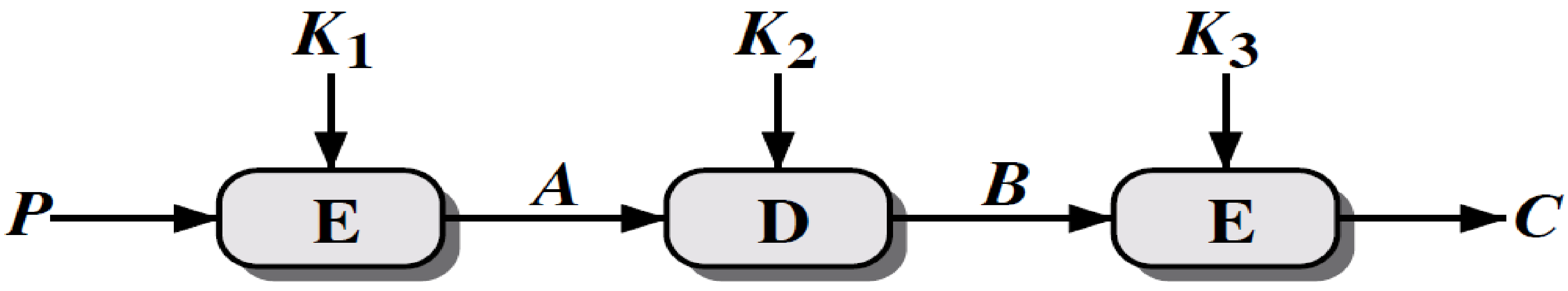
- Although there are no practical attacks on two-key Triple-DES, have some indications
- can use Triple-DES with Three-Keys to avoid even these
  - $C = E_{K3}[D_{K2}[E_{K1}[P]]]$
- has been adopted by some Internet applications, eg PGP, S/MIME
  - PGP- Pretty Good Privacy
  - MIME- Multipurpose Internet Mail Extension
- Three-key 3DES has an effective key length of 168 bits

Triple-DES with Three-Keys…

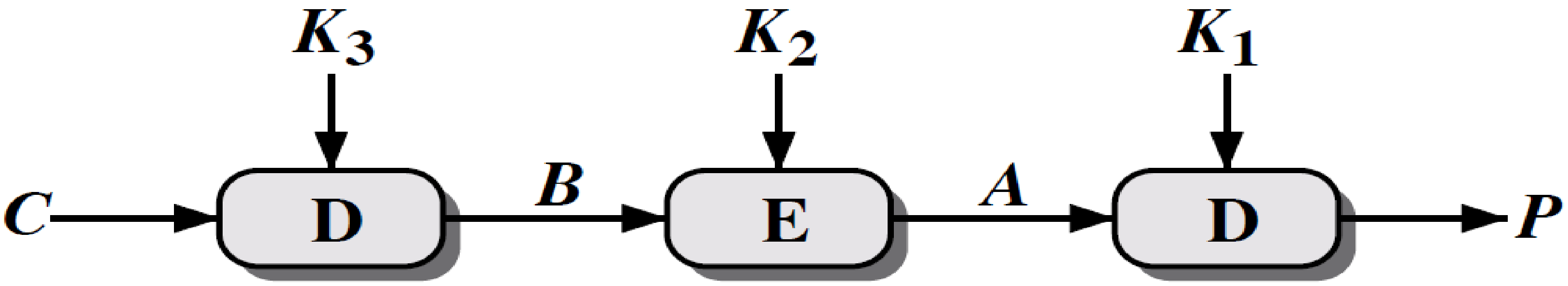
- With triple length key of three 56-bit keys K_1, K_2 & K_3, encryption is:
  - Encrypt with K_1
  - Decrypt with K_2
  - Encrypt with K_3
- Decryption is the reverse process:
  - Decrypt with K_3
  - Encrypt with K_2
  - Decrypt with K_1

- Setting K_3 equal to K_1 in these processes gives us a double length key K_1, K_2.
- Setting K_1, K_2 and K_3 all equal to K has the same effect as using a single-length (56-bit key).
- Thus it is possible for a system using triple-DES to be compatible with a system using single-DES.

Triple DES…



(a) Encryption



(b) Decryption