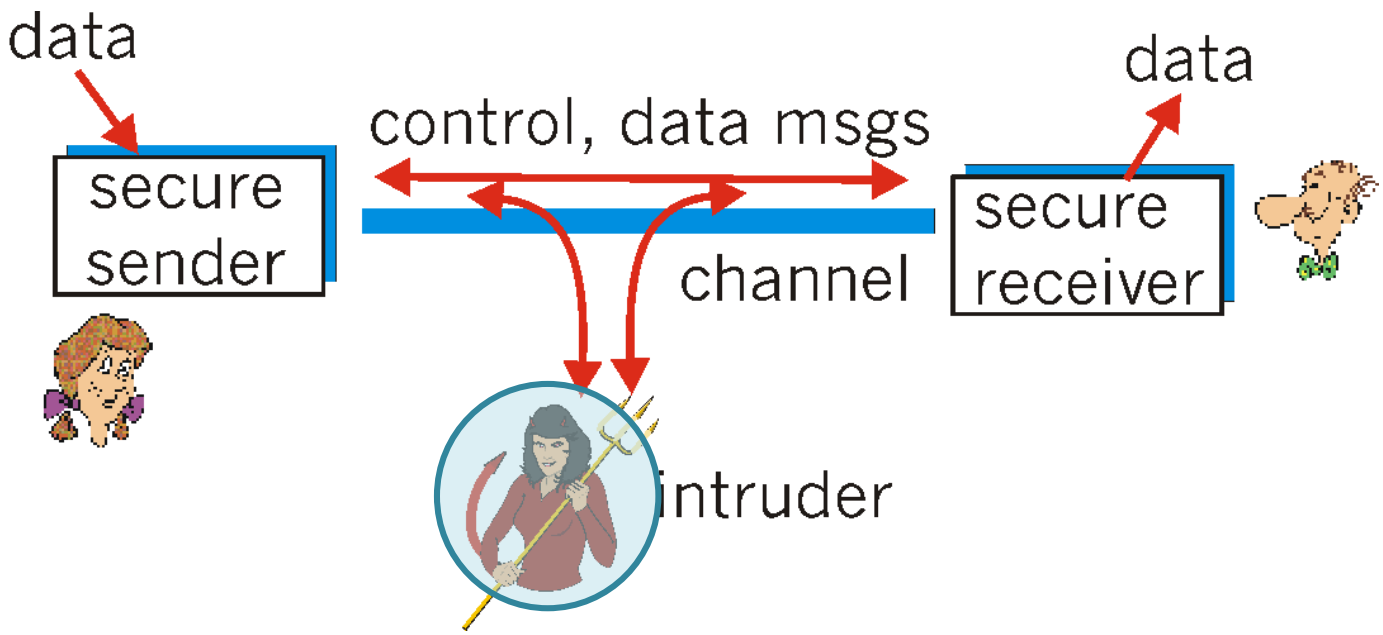
CS321: Computer Networks

Security in Computer Networks, Cryptography

Dr. Manas Khatua
Assistant Professor
Dept. of CSE
IIT Jodhpur
E-mail: manaskhatua@iitj.ac.in

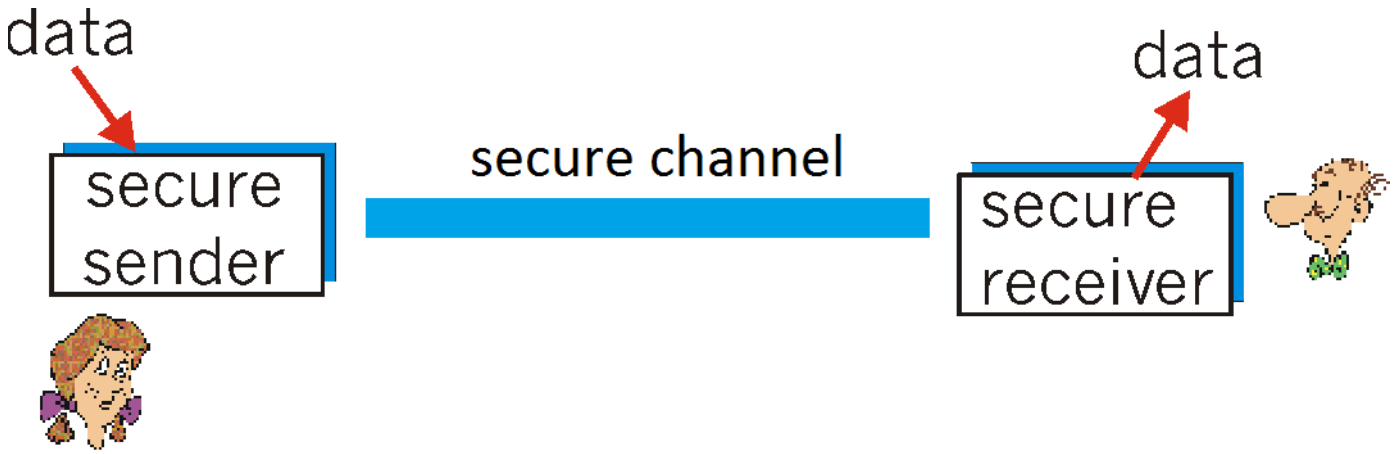
Introduction

- information is an asset that has a value like any other asset
- information needs to be secured from attacks
- Information passes through network.
- So, network needs to be secured as well.



Security Goals

- Confidentiality
- Message integrity
- End point authentication
- Operational security



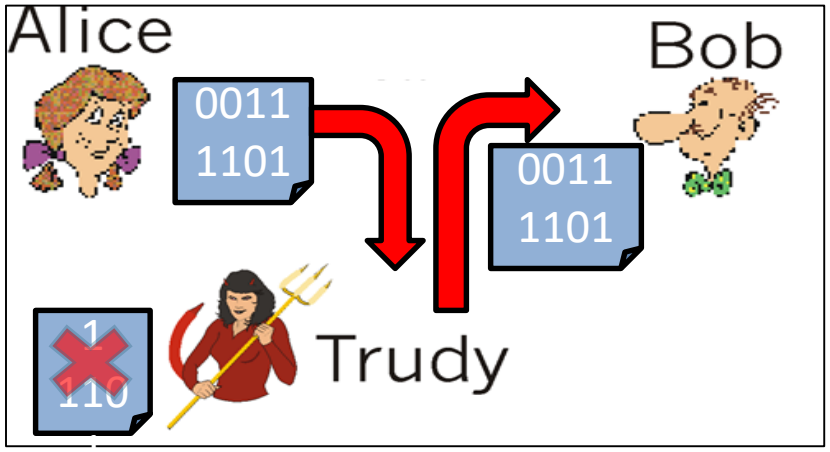
Cont…

Confidentiality | Message integrity | End point authentication | Operational security

- Information needs to be hidden from unauthorized access

- Only the sender and intended receiver should be able to understand the contents of the transmitted message

- How?
  - Using Encryption



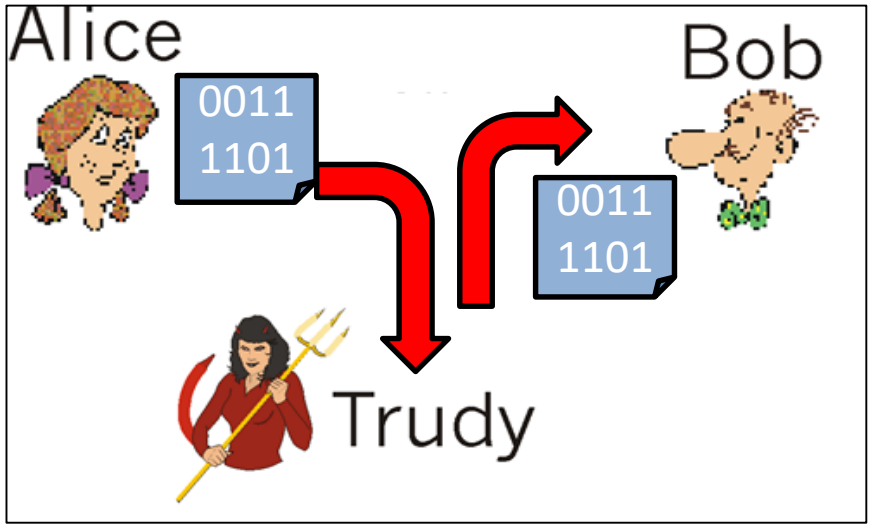
Cont…

Confidentiality | **Message integrity** | End point authentication | Operational security

- Message is protected from unauthorized change

- Alice and Bob want to ensure that the content of their communication is not altered, either maliciously or by accident, in transit.

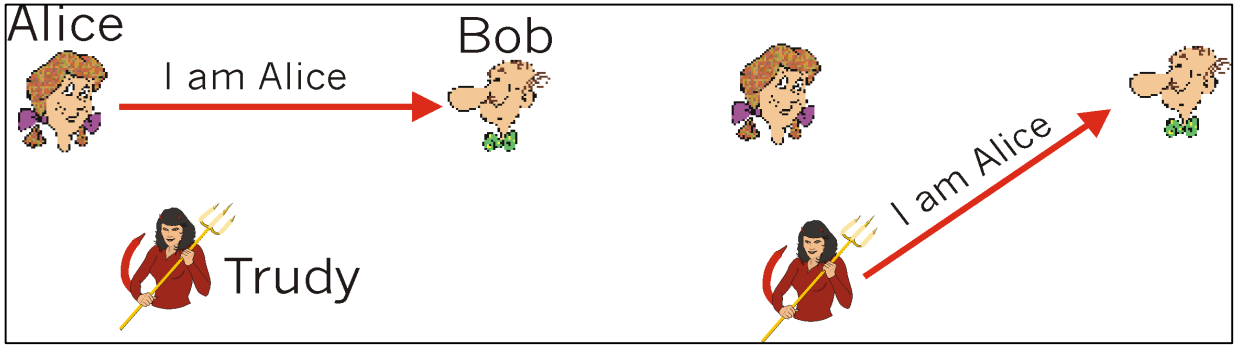
- How?
  - Using message digest and digital signature techniques



Cont…

Confidentiality | Message integrity | **End point authentication** | Operational security

- Transaction is protected from unauthorized access



- Both the sender and receiver should be able to confirm the identity of the other party involved in the communication
- How?
  - Using user authentication mechanism

Cont…

Confidentiality

Message integrity

End point authentication

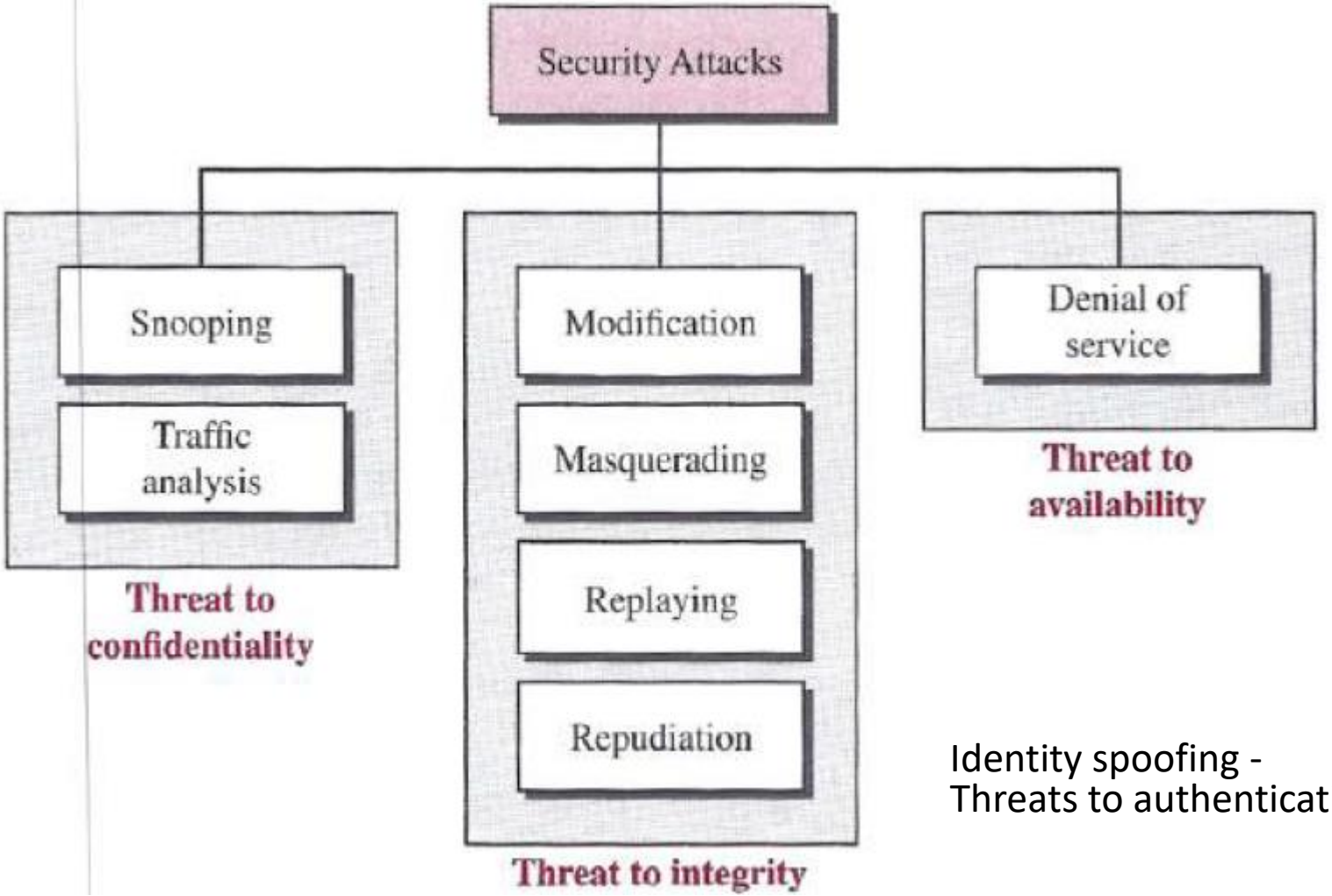
Operational security

Firewall

Viruses from Public network

- Application needs to be properly operational
- How?
  - Using Firewall and IDS (Intrusion Detection System)

Security Attacks



Identity spoofing -
Threats to authentication

Cont…

- *Snooping :* unauthorized access to or interception of data.
- *Traffic Analysis :* obtain some other types of information by monitoring online traffic.

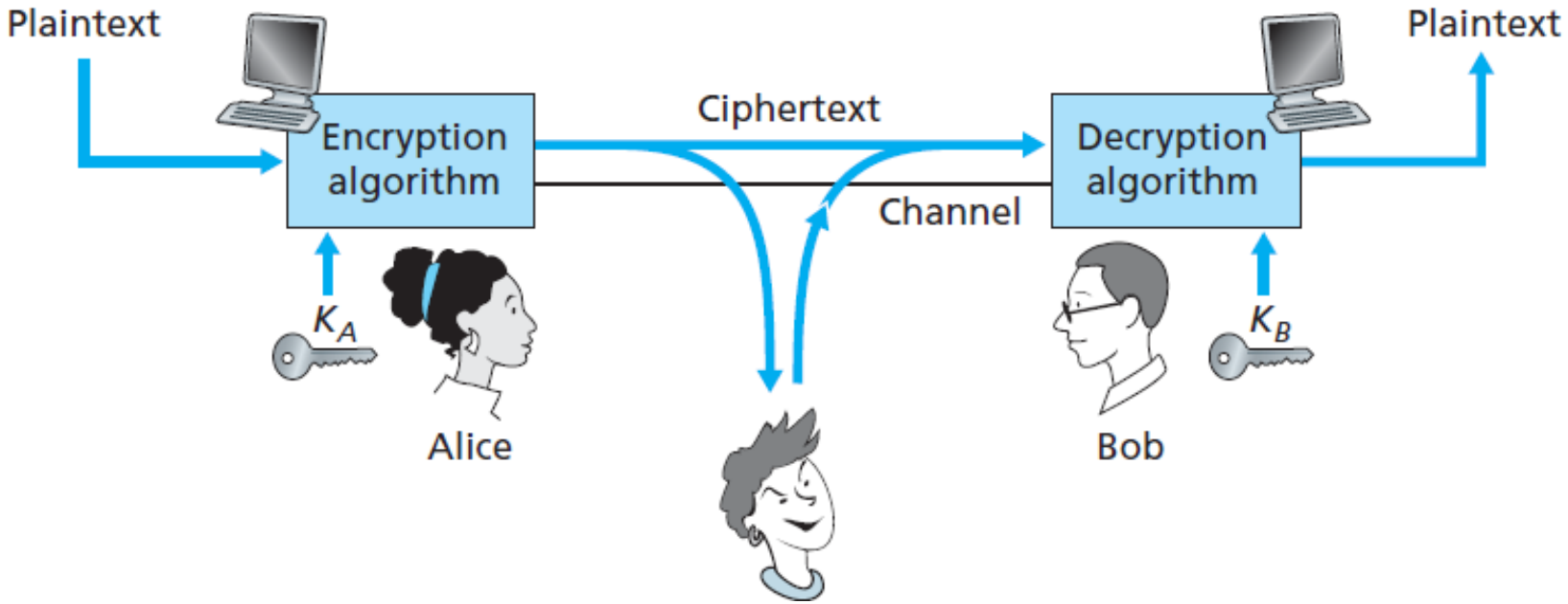
- *Modification :* modifies the information to make it beneficial to the attacker
- *Masquerading* or *spoofing* : the attacker impersonates somebody else.

- *Replaying :* the attacker obtains a copy of a message sent by a user and later tries to replay it.
- *Repudiation :* The sender of the message might later deny that she has sent the message; the receiver of the message might later deny that he has received the message.

- *Denial of Service :* It may slow down or totally interrupt the service of a system.

Principles of Cryptography

- It has a long history dating back at least as far as Julius Caesar.
- Cryptographic techniques allow a sender to disguise data so that an intruder can gain no information from the intercepted data.



Cont…

- To create the ciphertext from the plaintext, Alice uses an encryption algorithm and a key
- To create the plaintext from ciphertext, Bob uses a decryption algorithm and a key
- Based on type of key
  - symmetric key systems : Alice's and Bob's keys are identical and are secret
  - public key systems : a pair of keys is used. One of the keys is known to both Bob and Alice (indeed, it is known to the whole world). The other key is known only by either Bob or Alice (but not both).

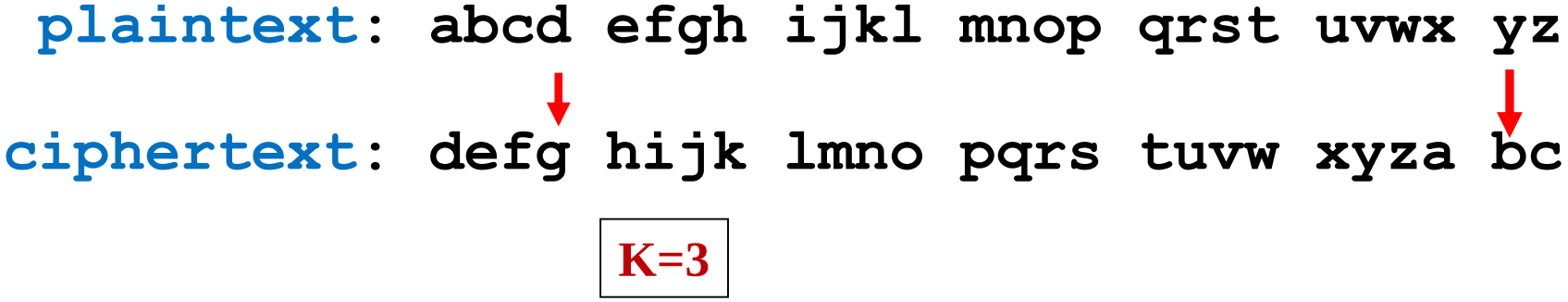
Symmetric Key Cryptography

Symmetric-key Ciphers

- We refer to encryption and decryption algorithms as *ciphers*.
- A *key* is a set of values (numbers) that the cipher operates on.
- It needs secure key exchange mechanism.

Caesar Cipher

- Substitute each letter by a letter *k* index away (allowing wraparound; that is, having the letter z followed by the letter a)



Plaintext: bob, i love you. alice

Ciphertext: ere, l oryh brx. dolfh

Remark: only 25 possible values for *k*, easy to break

Monoalphabetic Cipher

- *Main Idea*:
- Rather than substituting according to a regular pattern (for example, substitution with an offset of k for all letters),

- any letter can be substituted for any other letter, as long as each letter has a unique substitute letter, and vice versa.

Plaintext letter:	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
Ciphertext letter:	m	n	b	v	c	x	z	a	s	d	f	g	h	j	k	l	p	o	i	u	y	t	r	e	w	q

Plaintext: bob, i love you. alice

Ciphertext: nkn, s gktc wky. mgsbc

Remark: 26! (on the order of 10^{26}) possible pairings

Cont…

- A brute-force approach of trying all 10^{26} possible pairings would require far too much work to be a feasible way of breaking the encryption algorithm and decoding the message.

- But, using statistical analysis of the plaintext language, it becomes relatively easy to break this code !
  - the letters *e* and *t* are the most frequently occurring letters in typical English text
  - particular two- and three-letter occurrences of letters appear quite often together (for example, “in,” “it,” “the,” “ion,” “ing,” and so forth)

- By guessing few words related to contextual information
  - For example, if Trudy the intruder is Bob’s wife and suspects Bob of having an affair with Alice, then she might suspect that the names “bob” and “alice” appear in the text.

Polyalphablic Ciphers

- *Main Idea*:

It uses multiple monoalphabetic ciphers, with a specific monoalphabetic cipher to encode a letter in a specific position in the plaintext message.

Plaintext letter:	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
$C_1(k = 5)$:	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
$C_2(k = 19)$:	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

- We might choose to use these two Caesar ciphers, C1 and C2, in the repeating pattern C1, C2, C2, C1, C2.

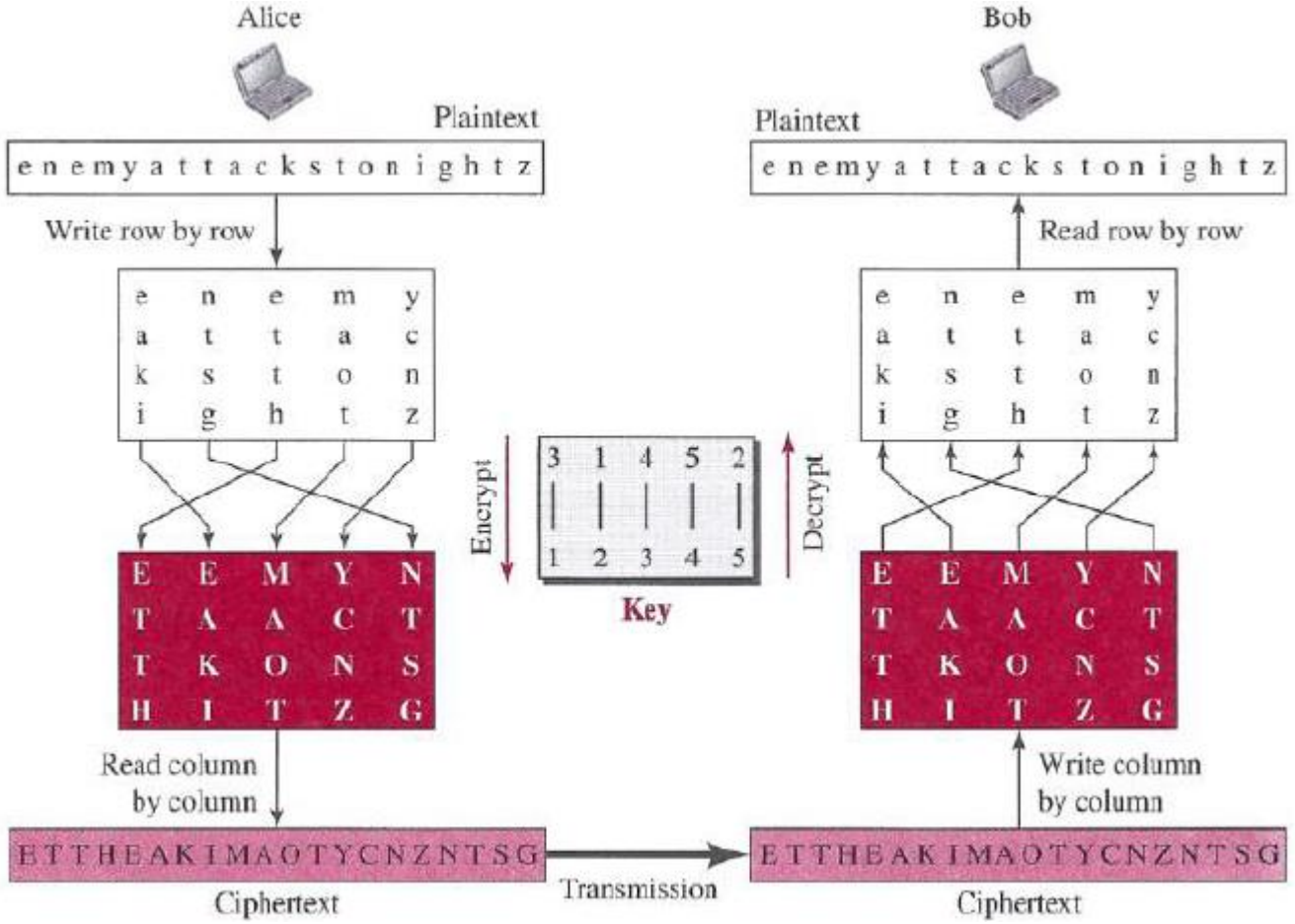
Plaintext: bob, i love you.

Ciphertext: ghu, n etox dhz.

Transposition Ciphers

- *Main Idea*:

It does not substitute one symbol for another;
instead it changes the location of the symbols.



Ciphers in Modern times

- Two broad classes of symmetric-key encryption:
  - stream ciphers
    - where plaintext digits are combined with a pseudorandom cipher digit stream (keystream)
    - used in Wireless LANs
    - E.g., RC4 (Rivest Cipher 4)
  - block ciphers
    - operates on large blocks of digits with a fixed, unvarying transformation
    - used in many secure Internet protocols, including
      - PGP (for secure e-mail),
      - SSL (for securing TCP connections), and
      - IPsec (for securing the network-layer transport).
    - E.g., DES (Data Encryption Standard),
      AES (Advanced Encryption Standard)

Block Cipher

- the message to be encrypted is processed in blocks of *k* bits
- To encode a block, the cipher uses a one-to-one mapping to map the *k*-bit block of cleartext to a *k*-bit block of ciphertext

input	output	input	output
000	110	100	011
001	111	101	010
010	101	110	000
011	100	111	001

Table 8.1 ♦ A specific 3-bit block cipher

- the block cipher maps 3-bit inputs (*cleartext*) to 3-bit outputs (*ciphertext*).

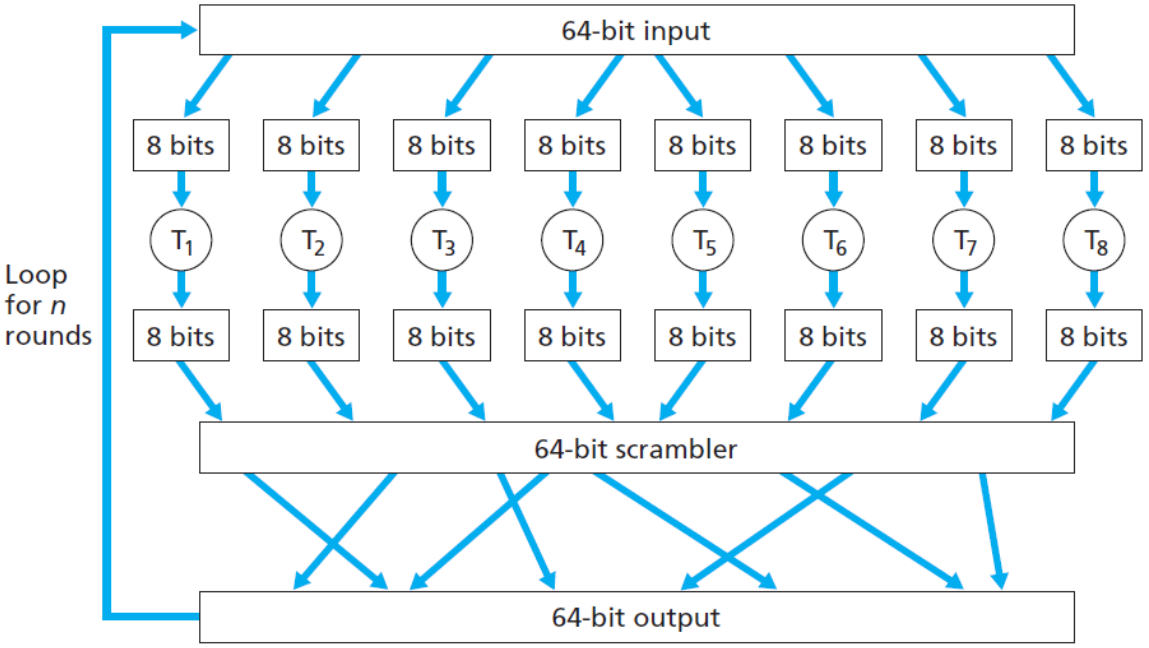
- How many possible mappings are there?
- Answer: permutation of all the possible inputs = $(2^3)!$ = 40,320 different ways
- The brute-force attack for this cipher can decrypt the ciphtertext using a desktop PC!

Block Cipher

- To thwart brute-force attacks, block ciphers typically use much larger blocks, consisting of k = 64 bits or even larger

- Table based mappings are unfortunately difficult to implement.
- Every station would need to maintain a table with 2^k input values, which is an infeasible task.

- Solution: typically use functions that simulate randomly permuted tables

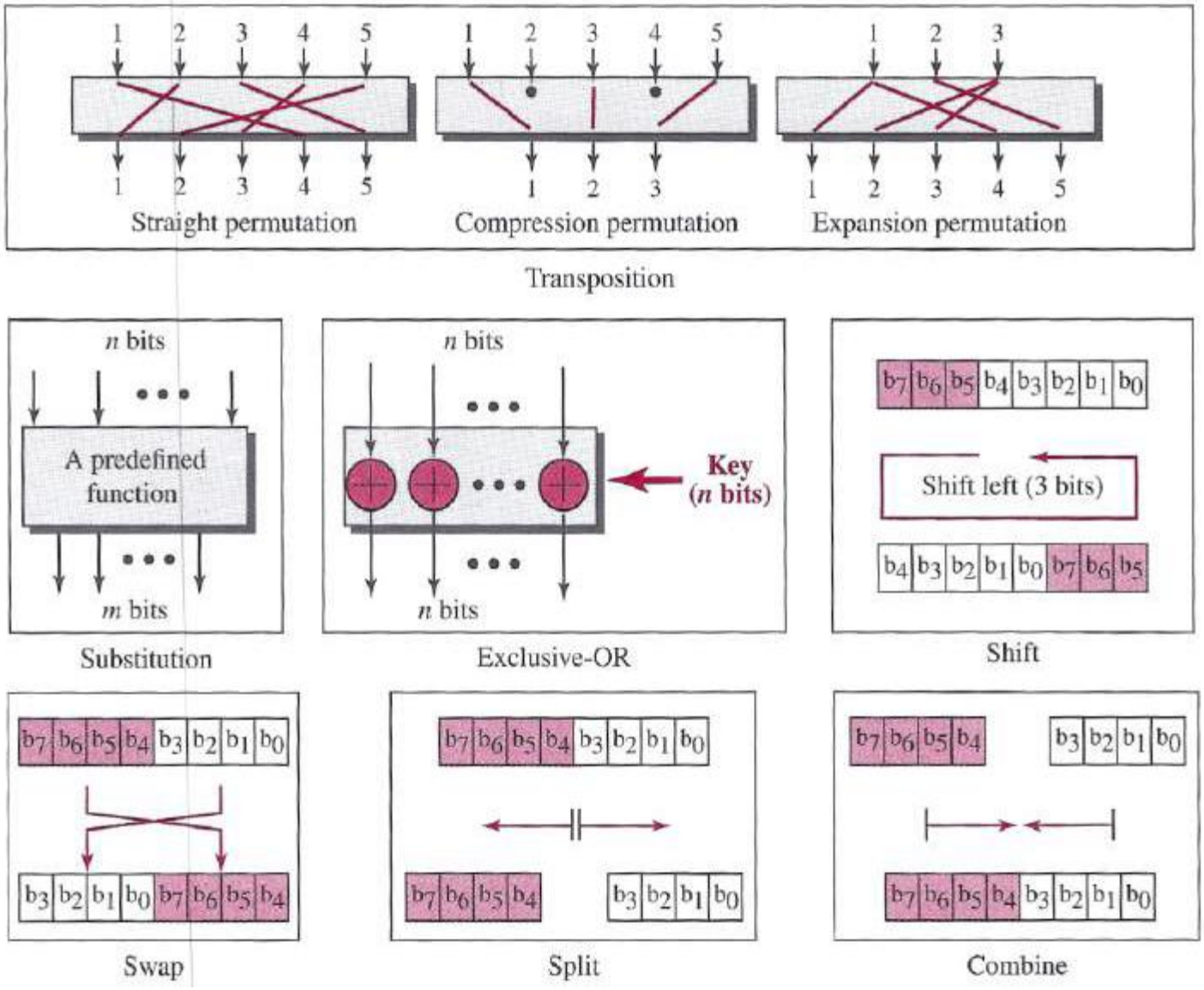


- Each 8-bit chunk is processed by an 8-bit to 8-bit table, which is of manageable size. (T_1--T_8)

- After n such cycles, the function provides a 64-bit block of ciphertext.

- The **key** for this block cipher algorithm would be the 8 permutation tables

Cont…


1 2 3 4 5
1 2 3 4 5
Straight permutation
1 2 3 4 5
1 2 3
Compression permutation
1 2 3
1 2 3 4 5
Expansion permutation
Transposition
n bits
A predefined function
m bits
Substitution
n bits
Key (n bits)
n bits
Exclusive-OR
b7 b6 b5 b4 b3 b2 b1 b0
Shift left (3 bits)
b4 b3 b2 b1 b0 b7 b6 b5
Shift
b7 b6 b5 b4 b3 b2 b1 b0
b3 b2 b1 b0 b7 b6 b5 b4
Swap
b7 b6 b5 b4 b3 b2 b1 b0
b7 b6 b5 b4
b3 b2 b1 b0
Split
b7 b6 b5 b4
b3 b2 b1 b0
b7 b6 b5 b4 b3 b2 b1 b0
Combine

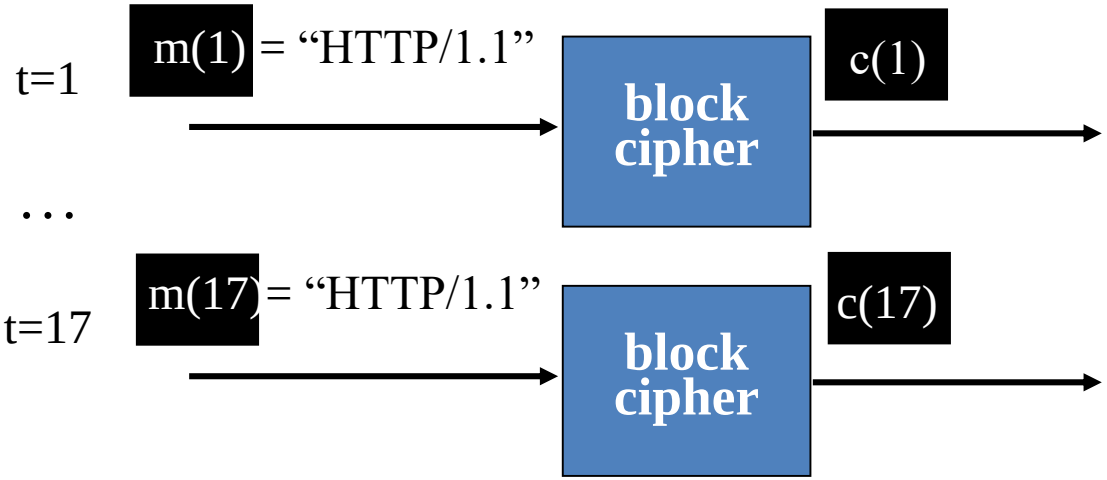
Cont…

- Popular block ciphers:
  - DES (Data Encryption Standard),
  - AES (Advanced Encryption Standard),

- DES uses 64-bit blocks with a 56-bit key
- AES uses 128-bit blocks and can operate with keys that are 128, 192, and 256 bits long.

- NIST estimates that a machine that could crack 56-bit DES in one second (that is, try all 2^{56} keys in one second) would take approximately 149 trillion years to crack a 128-bit AES key.

Cipher-Block Chaining (CBC)

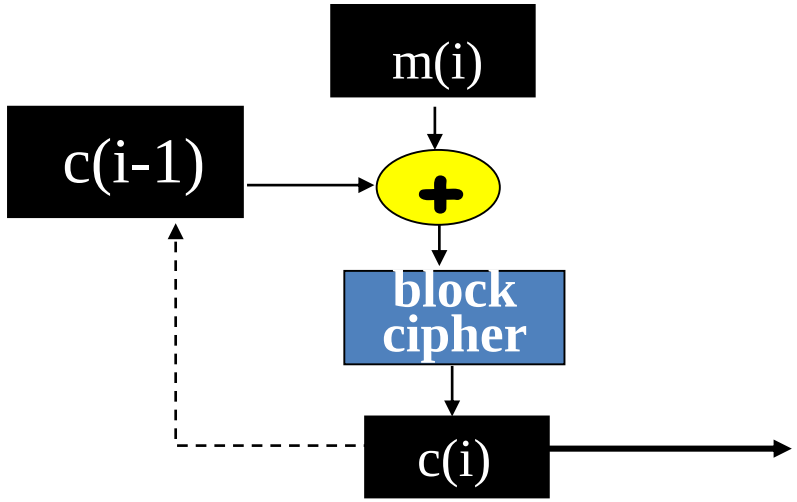


- *Drawback in block cipher:*

if input block is repeated, this coding will produce same cipher text

- Solution: Cipher Block Chaining
  - XOR the i^{th} input block, m(i), with previous block of cipher text, c(i-1)

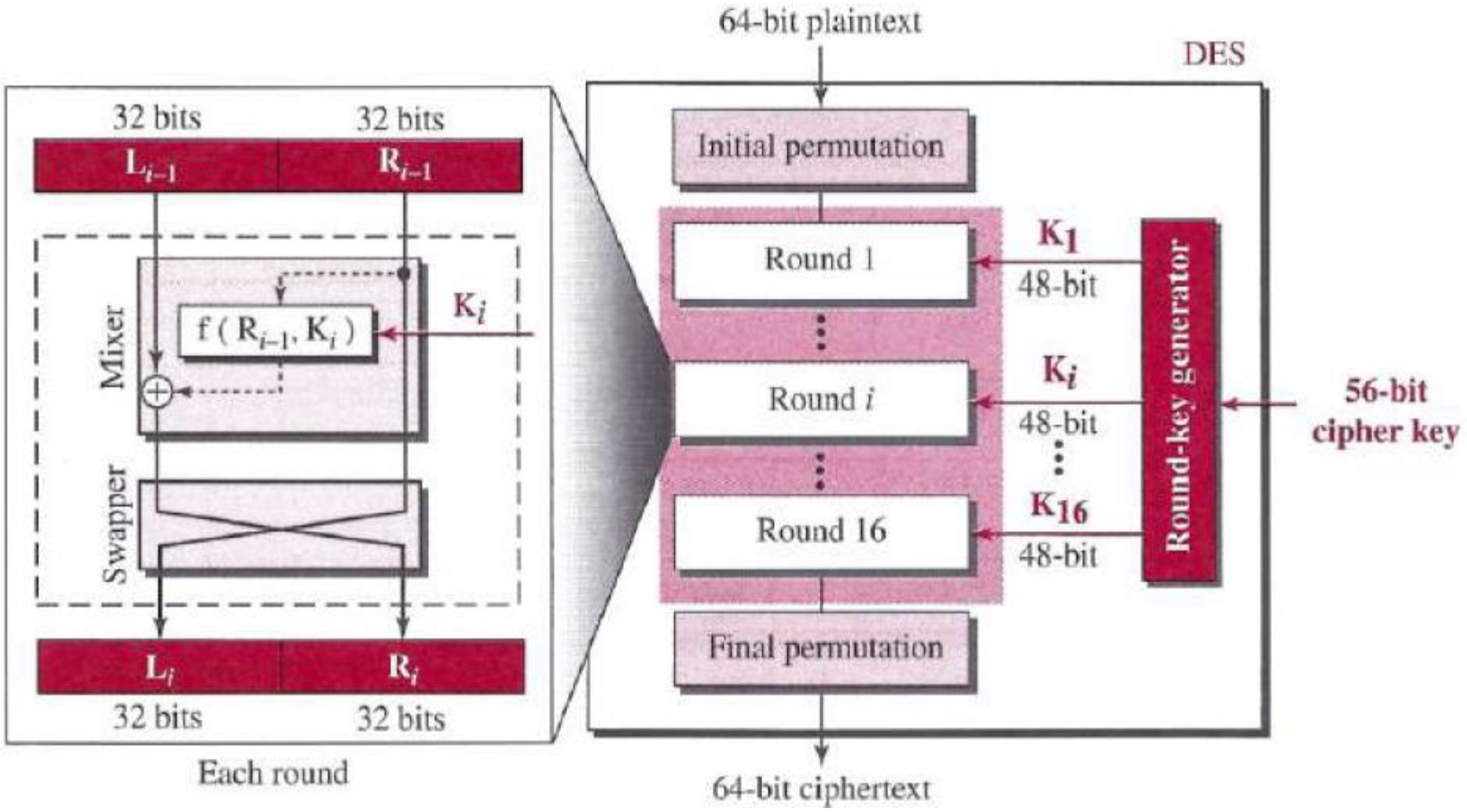
- Initially, the sender generates a random *k*-bit string for c(0), called the Initialization Vector (IV).
- The sender sends the IV to the receiver *in cleartext*.



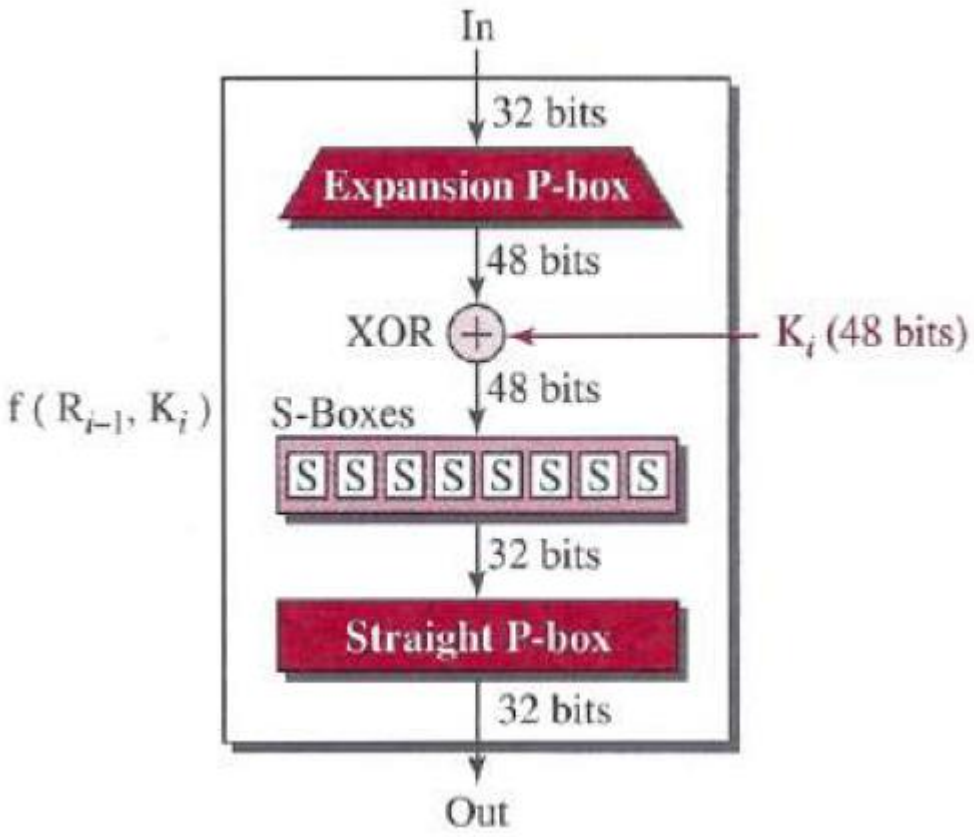
CBC

- Consequences of CBC:
  - receiver will still be able to recover the original message
  - even if two cleartext blocks are identical, the corresponding ciphtertexts (almost always) will be different
  - although the sender sends the IV in the clear, an intruder will still not be able to decrypt the ciphertext blocks, since the intruder does not know the secret key, S
  - the sender only sends one overhead block (the IV), thereby negligibly increasing the bandwidth usage for long messages (consisting of hundreds of blocks)

Data Encryption Standard (DES)



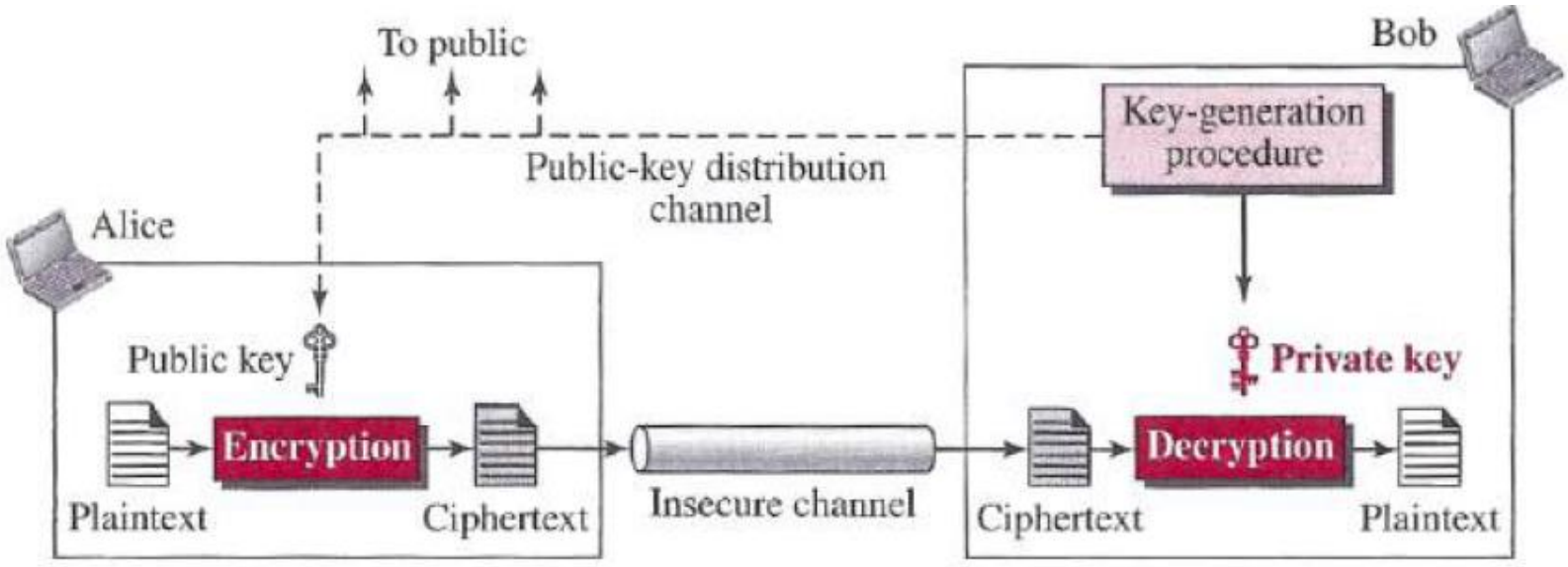
Cont…



- DES takes a 64-bit plaintext and creates a 64-bit ciphertext;
- The same 56-bit cipher key is used for both encryption and decryption
- DES uses 16 rounds
- The heart of DES is the DES function.
- The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.

Public Key Cryptography

Public-Key Cryptography

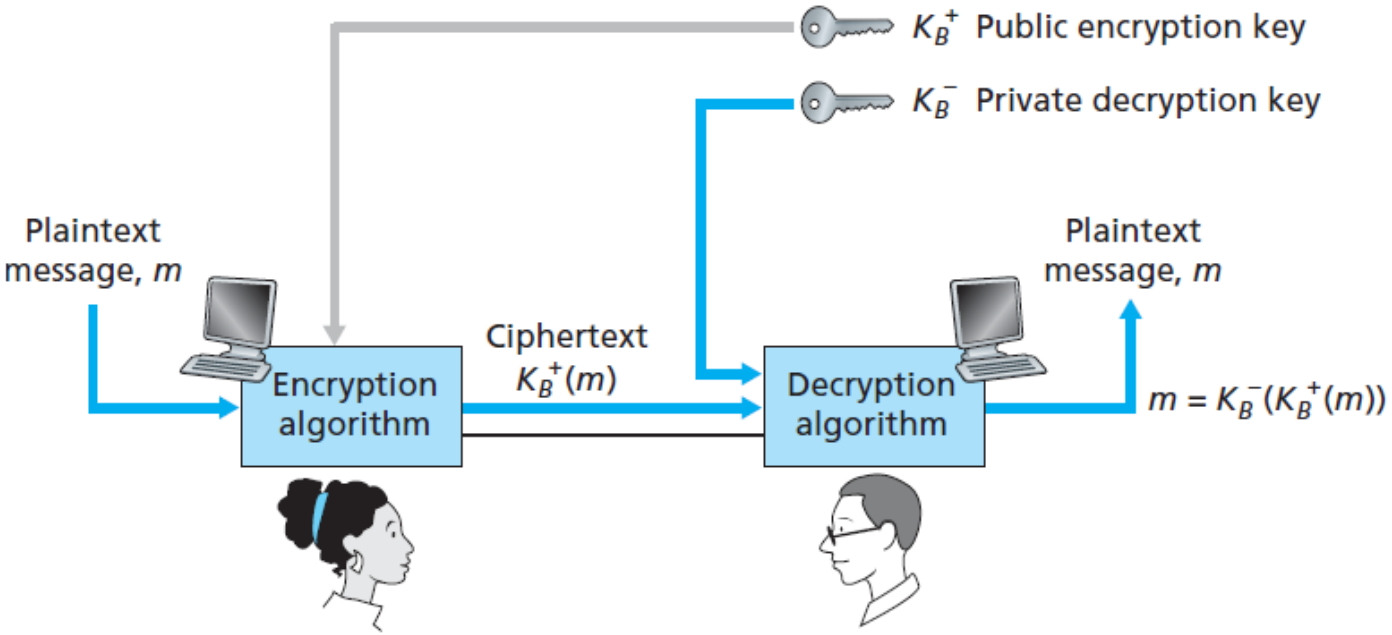


- Symmetric-key cryptography is based on sharing secrecy;
- asymmetric-key cryptography is based on personal secrecy.

- In symmetric-key cryptography, symbols are permuted or substituted;
- in asymmetric-key cryptography, numbers are manipulated.

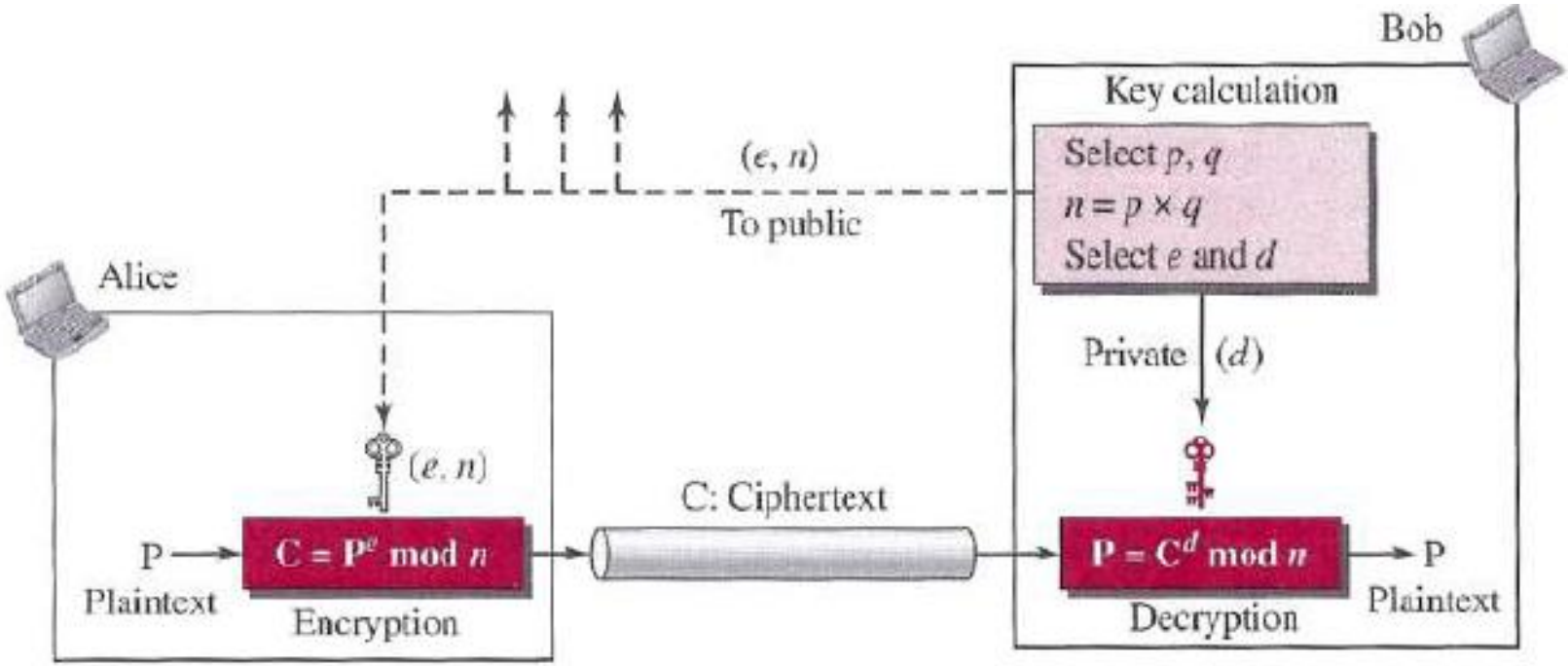
- Example: RSA Algorithm, Diffie-Hellman Algorithm

Cont…



- Two concerns
  - Intruder knows Bob's public key which Alice used. Intruder can intercept the ciphertext transmitted from Alice
    - Soln: strong cipher & long key
  - Since Bob's encryption key is public, anyone can send an encrypted message to Bob, including Alice or someone claiming to be Alice.
    - Soln: Digital Signature

RSA Algorithm



- RSA (Rivest, Shamir, and Adleman)
- RSA uses two exponents, *e* and *d,* where *e* is public and *d* is private
- Suppose P is the plaintext and C is the ciphertext.
  $C = P^e \bmod n$; $P = C^d \bmod n$; n is a large number

Cont…

- How can we get those *e*, *d*, *n*?

- Procedure:

  choose two large numbers, *r* and *q,* and calculates

  $n = r \times q$ and $z = (r - 1) \times (q - 1)$

  Then, selects *e* and *d* such that $(e \times d) \bmod z = 1$.

- Example:

  let Bob choose 7 and 11 as *r* and *q.*

  So, $n = 7 \times 11 = 77$; $z = 6 \times 10 = 60$

  If, he chooses e=13, then, d=37.

  Let, Alice wants to send the plaintext 5 to Bob.

  So, $C = 5^{13} \bmod 77 = 26$

  $P = 26^{37} \bmod 77 = 5$

Cont…

- Modular arithmetic:

  $[(a \bmod n) + (b \bmod n)] \bmod n = (a + b) \bmod n$

  $[(a \bmod n) - (b \bmod n)] \bmod n = (a - b) \bmod n$

  $[(a \bmod n) \bullet (b \bmod n)] \bmod n = (a \bullet b) \bmod n$

- *Pros and Cons:*
  - useful for short messages
  - it is very slow if the message is long

  - is used in digital signature
  - is also used for end point authentication

Session Keys

- Problem in RSA:
  - The exponentiation required by RSA is a time-consuming process
  - DES is at least 100 times faster in software and between 1,000 and 10,000 times faster in hardware than RSA
- RSA is often used in practice in combination with symmetric key cryptography
- How?
  - First Alice chooses a key that will be used to encode the data itself; this key is referred to as a **session key**, and is denoted by *K*.
  - Alice encrypts the session key using Bob's public key and send the encrypted key to Bob.
  - Bob receives the RSA-encrypted session key, c, and decrypts it to obtain the session key, K.
  - Bob now knows the session key that Alice will use for her encrypted data transfer.

Thanks!