Network Security

First security issue: Computer Security – protect data stored into computer

Network Security – ***protect data during transmissions & guarantee that data transmissions are authentic***

Security Requirements

Confidentiality – data accessed & read only by authorized parties

Integrity – data modification by authorized parties

Availability – data available to authorized parties

Network Security Problems (what to allow for):

Secrecy

Keeping information private (out of unauthorized parties)

Authentication

Proving one's identity, before revealing info

Non-repudiation

Showing (proving) that a message was sent; use of signatures

Integrity

Showing that a message wasn't modified

Attacks on Network Security

Passive Attacks

Nature of: eavesdropping (monitoring) on transmissions

Goal: to obtain information that is being transmitted;

Two types of passive attacks:

Release of message contents

Outsider learns content of transmission

Traffic analysis

By monitoring frequency and length of messages, even encrypted, nature of communication may be guessed

Difficult to detect, because attacks don't alter data; can be prevented, rather than detected; use of *encryption*

Active Attacks

Involve some data stream modification, or creation of a false stream

Masquerade

Pretending to be a different entity

Replay

Capture of data unit and retransmission for an unauthorized effect

Modification of messages

Some portion of a legitimate message is altered

Denial of service

Prevents or inhibits normal use of communications facilities

Easy to detect: detection may lead to a deterrent effect (helps prevention)

Hard to prevent (requires all time physical protection)

Use of *authentication*

Encryption

Two approaches: **conventional encryption** and **public-key encryption**

Conventional Encryption

Referred also as single-key or symmetric encryption

Terms

Original message: plaintext

Transformed via a: key

Into: ciphertext (scrambled message)

Breaking ciphers: cryptanalysis

Science: cryptology

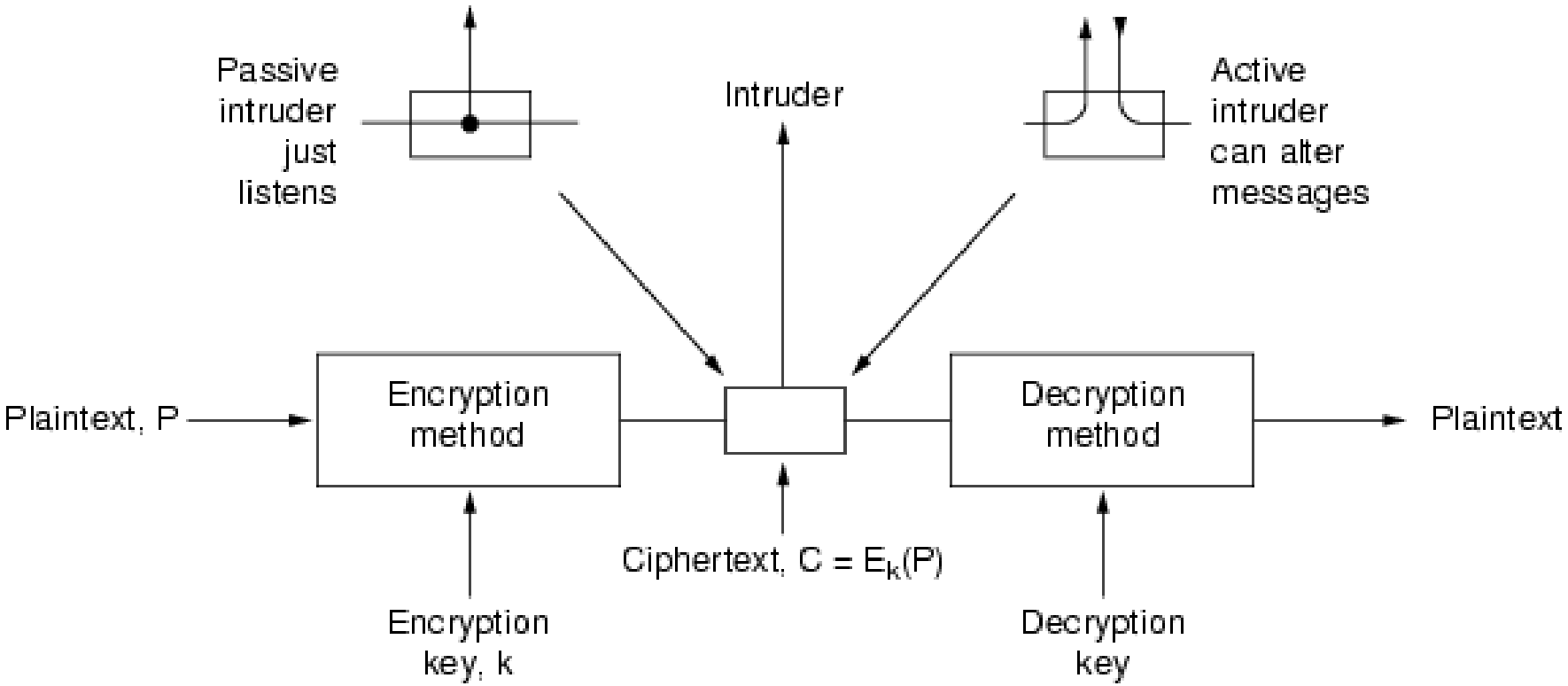
D(E(P)) = P

See next slide

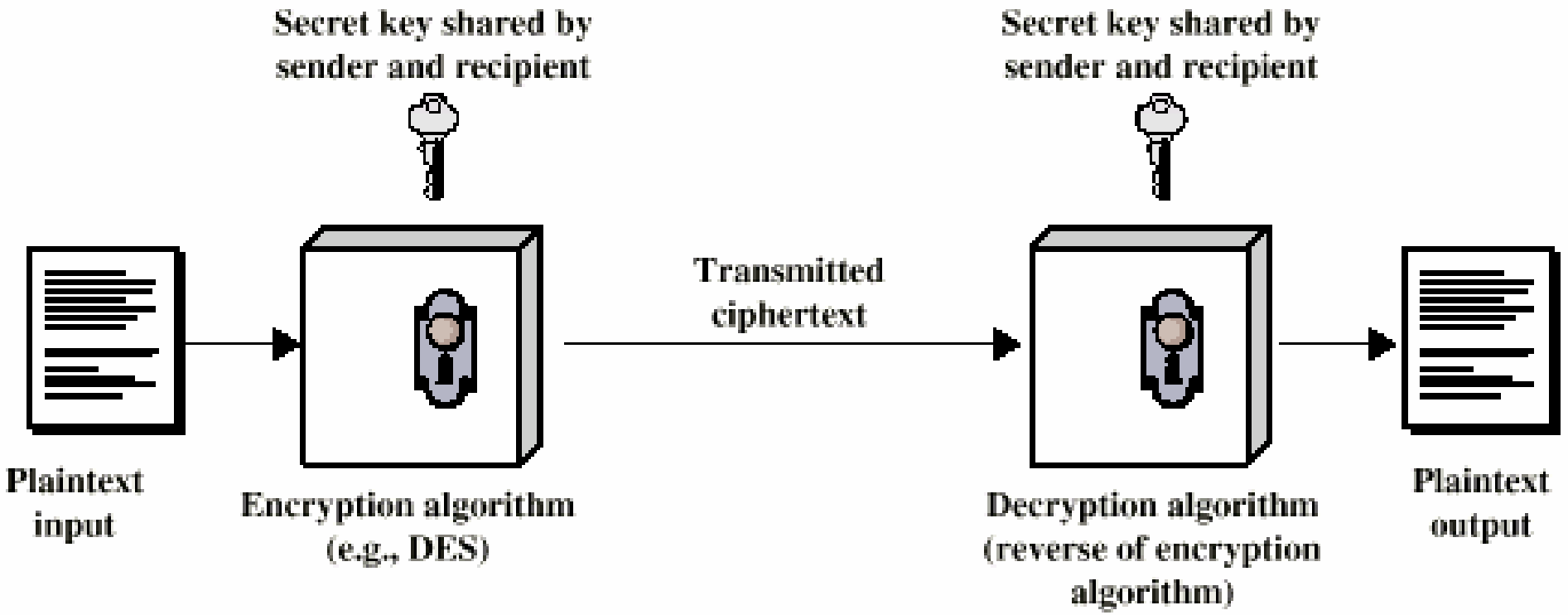
Provides:

Confidentiality

Protection against passive attacks



The encryption model.



Requirements for Security

Strong encryption algorithm

- Even if algorithm known, should not be able to decrypt or work out key
- Even if a number of cipher texts are available together with plain texts of them, opponent unable to decrypt cipher text or discover the key

Sender and receiver must obtain secret key securely

Once key is known, all communication using this key is readable

Attacking Encryption

Crypt analysis

- Relay on nature of encryption algorithm, plus some knowledge of general characteristics of plain text
- Attempt to deduce plain text or key

Brute force attack

- Try every possible key on a piece of cipher-text until plain text is achieved

Cryptographic Algorithms

Private methods

Public methods

- Considered better as they are reviewed
- Key length is the issue: for a 32bit key, a required time for key search would be in the order of milliseconds, but for a 128bit key, at same encryption rate, key search would take tens of thousands years

Encryption Algorithms

Traditional simple ciphers

- Substitution cipher
- Transposition cipher

Block cipher

- Process plain text in fixed block sizes producing block of cipher text of equal size
- Data encryption standard (DES)
- Triple DES (TDES)

Historical Ciphers

Substitution ciphers

- Each letter replaced with another
- Easy to decode with letter frequency

Transposition ciphers

- Change the order of the letters, not disguise them

See next example

M	E	G	A	B	U	C	K
7	4	5	1	2	8	3	6
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Plaintext

pleasetransferonemilliondollarsto
myswissbankaccountsixtwotwo

Ciphertext

AFLLSKSOSELAWAIATOOSSCTCLNMOMANT
ESILYNTWRNNTSOWDPAEDOBUOERIRICXB

MEGABUCK is the key

The columns are numbered after the proximity
of the letter to the start of the alphabet

Secret-Key Cipher Algorithms

Today algorithms use same idea as traditional cryptography, but relies algorithms with long keys.

Transpositions & substitutions ciphers can be implemented with simple (hardware) circuits: P-box & S-box. Together are used for a product cipher, cascading series of boxes.

Also known as **block cipher** algorithms.

Process plain text in fixed block sizes producing block of cipher text of equal size

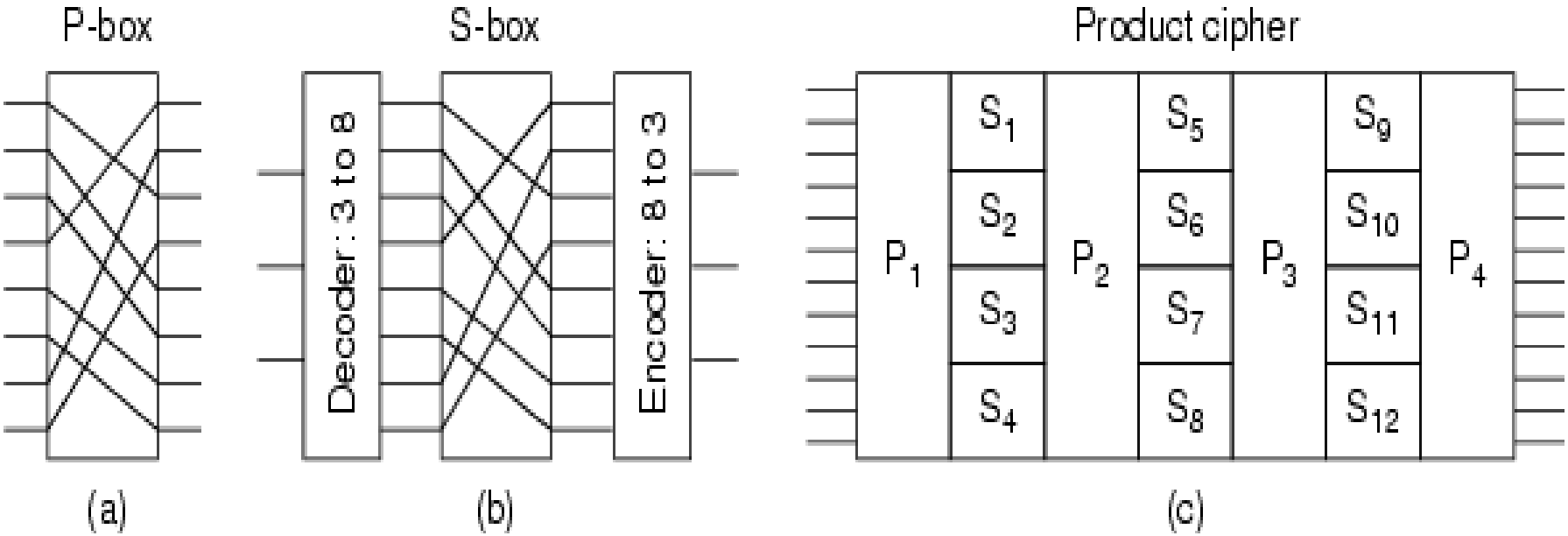
Example:

Data encryption standard (DES)

Triple DES (TDES)

Used Algorithm: DEA (Data Encryption Algorithm)

See next slide



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

Data Encryption Standard (DES)

US standard, from 1977; based on DEA (Data Encryption Algorithm)

64 bit plain text blocks; 56 bit key

Strength of DES

Declared insecure in 1998 by Electronic Frontier Foundation

Use of DES Cracker machine; DES now worthless

Alternatives include TDEA

DES Phases:

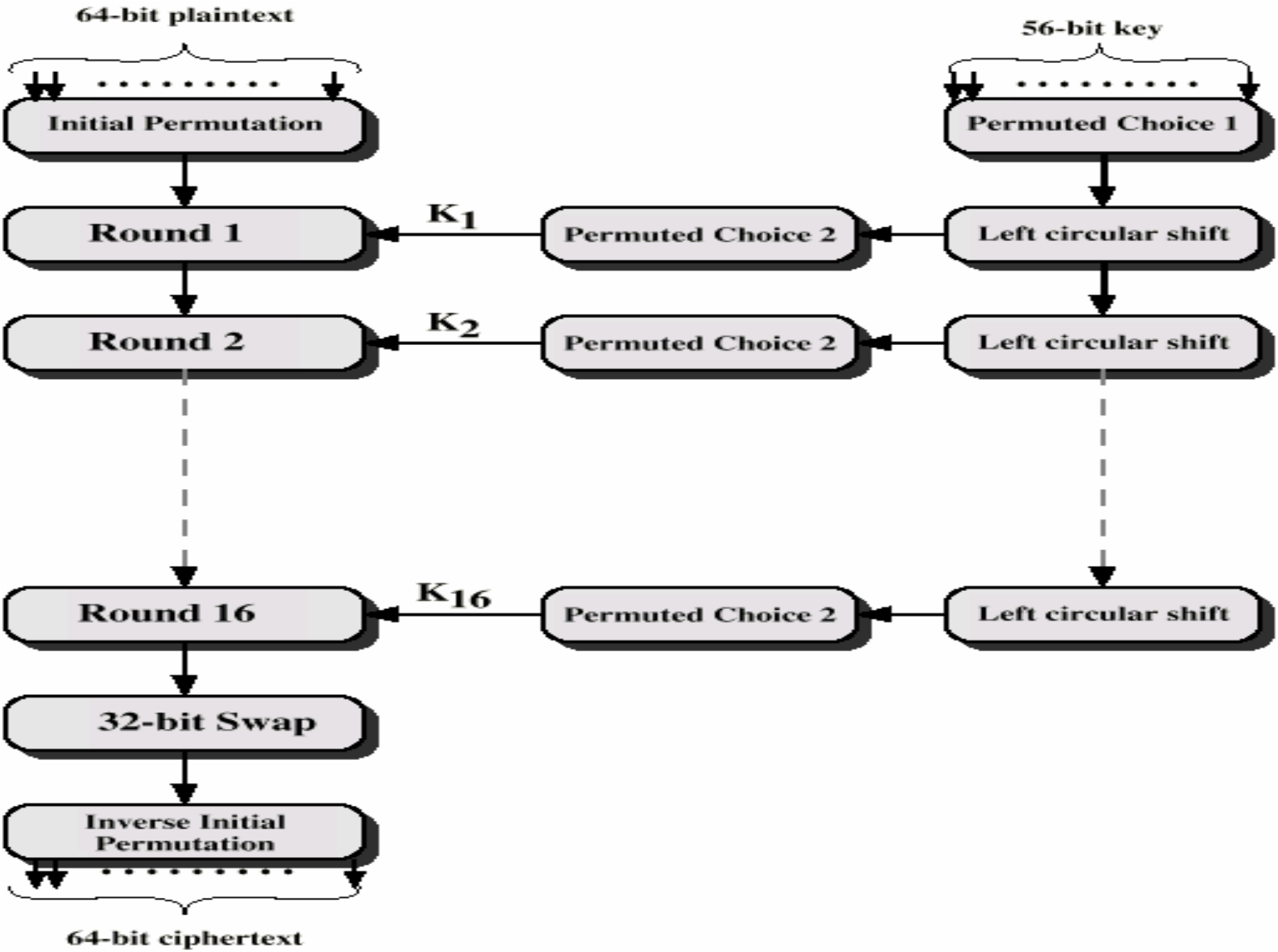
Initial Permutation, key independent

16 iterations of same functions, resulting a 64bit string depending on the input string and some encryption key

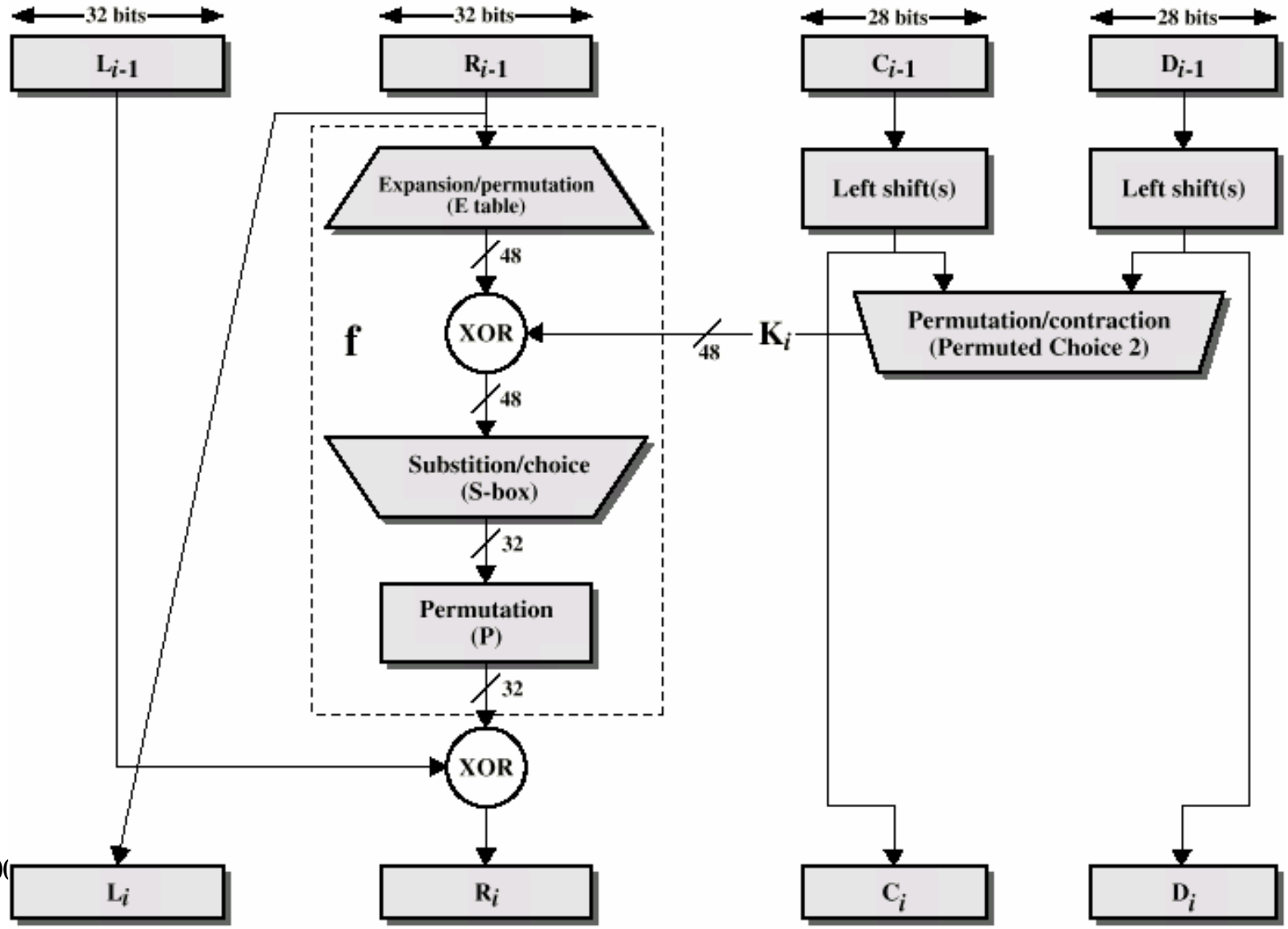
Swap of the left & right string halves

Inverse initial permutation, resulting 64bit cipher text

For each iteration is used a sub-key, obtained from the initial key, using a pair: left circular shift plus a permutation (see right part of next illustration)


64-bit plaintext
56-bit key
Initial Permutation
Permuted Choice 1
Round 1
K1
Permuted Choice 2
Left circular shift
Round 2
K2
Permuted Choice 2
Left circular shift
Round 16
K16
Permuted Choice 2
Left circular shift
32-bit Swap
Inverse Initial Permutation
64-bit ciphertext

Single Round of DEA



DEA Improvements:

Triple DEA

ANSI X9.17 (1985)

Incorporated in DEA standard 1999

Uses 3 keys and 3 executions of DEA algorithm

Effective key length 168 bit

Use 3 keys & 3 DEA executions, allowing for a sequence Encryption/Decryption/Encryption

Key length: 168bit, but if $K_1 = K_3$, key length becomes 112bit (usually enough)

Decryption used for backward compatibility (with one key DEA, also $K_1 = K_2$)

Triple encryption using DES.

Other approach:

IDEA (International DEA)

Swiss ‘idea’

128-bit key, so immune to brute force attacks

Use of 3 operations

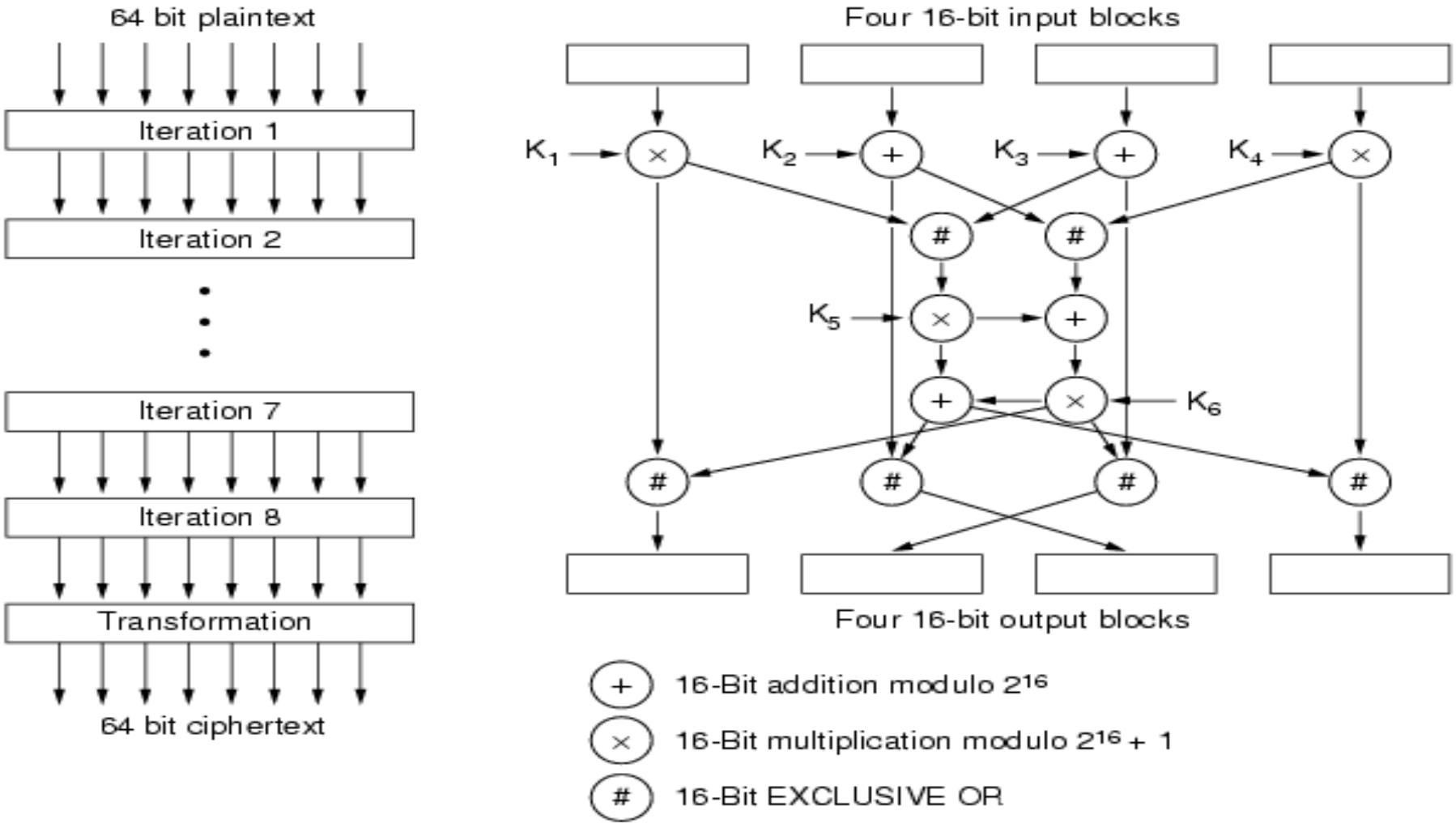
Has 8 iterations

128bit key generates 52 sub-keys of 16bit each

6 sub-keys for each iteration

4 sub-keys for final transformation

See next slide



. (a) IDEA. (b) Detail of one iteration.

Location of Encryption Gear

What to encrypt but also where to locate encrypting gear?

Two alternatives: link and end-to-end encryption

See next slide

Link encryption device

Each communication link equipped at both ends

All traffic secure

High level of security

Requires lots of encryption devices

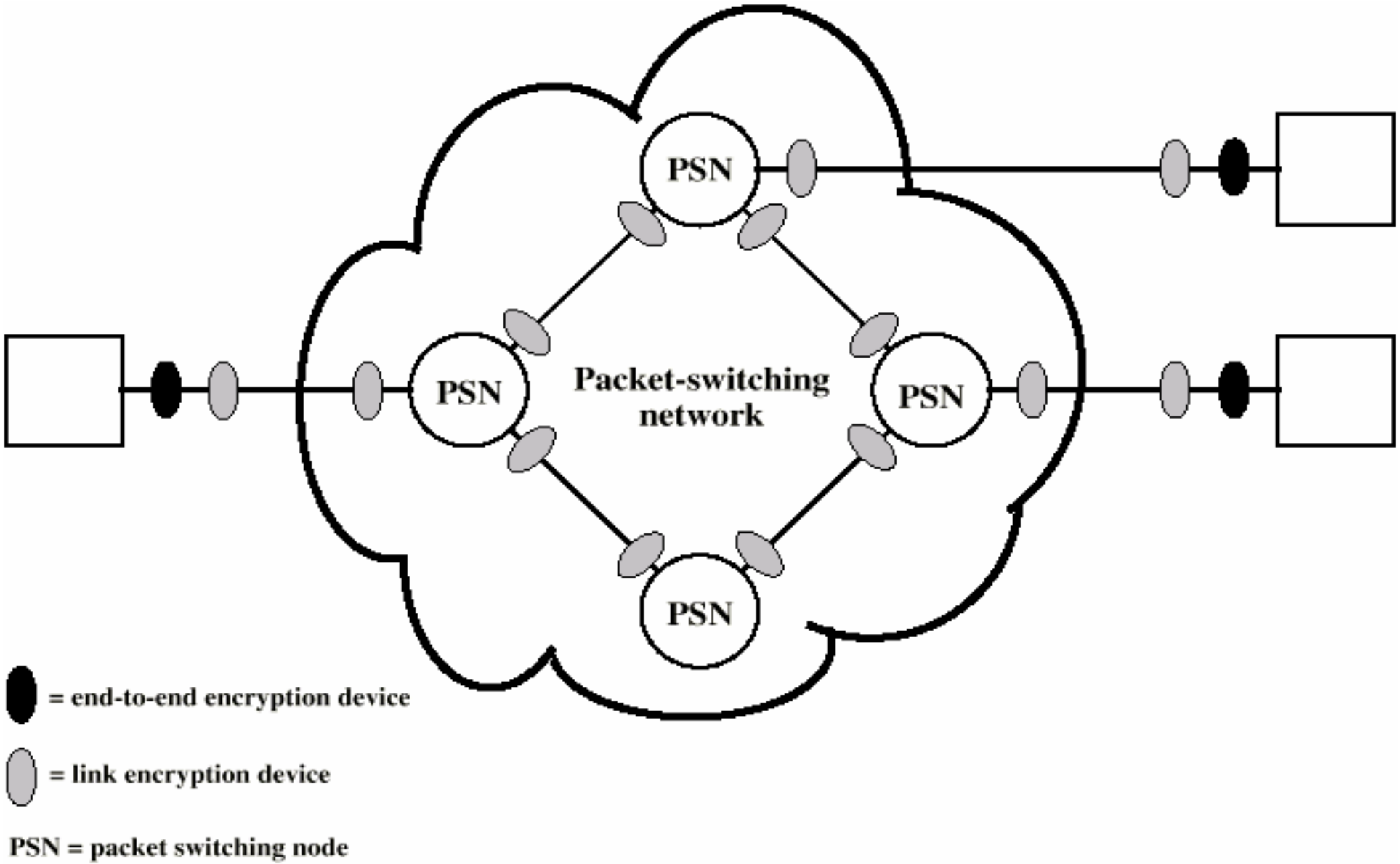
Message must be decrypted at each switch

to read address (virtual circuit number)

Security vulnerable at switches

- Particularly on public network

Encryption across a packet-switching network



End-to-end encryption devices

Encryption done at ends of system

Data in encrypted form crosses network unaltered

Destination shares key with source to decrypt

Host can only encrypt user data

Otherwise switching nodes could not read header or route packet

Traffic pattern not secure

Conclusion?

Need for use of both techniques!

Improvement of traffic security: Traffic Padding

Produce cipher text continuously

If no plain text to encode, send random data

Make traffic analysis impossible (no traffic differences at each end system)

Key Distribution

Another strengthens of the cryptographic system!

A number of approaches:

Key selected by A and delivered to B

Third party selects key and delivers to A and B

Allows for manual key delivery; use for link encryption devices

Use old key to encrypt a new key and transmit new key from A to B

If one key found, all revealed

Both parts have encrypted connections to a third party C, and C uses old key to transmit new key to A and B

End-to-end encryption gear

Such implementations are expensive, requiring more keys and special software (see next figure)

Session Key, used for duration of one logical connection, destroyed at end of session, used for encryption of user data

Permanent key, used for distribution of session keys

Key distribution center (KDC)

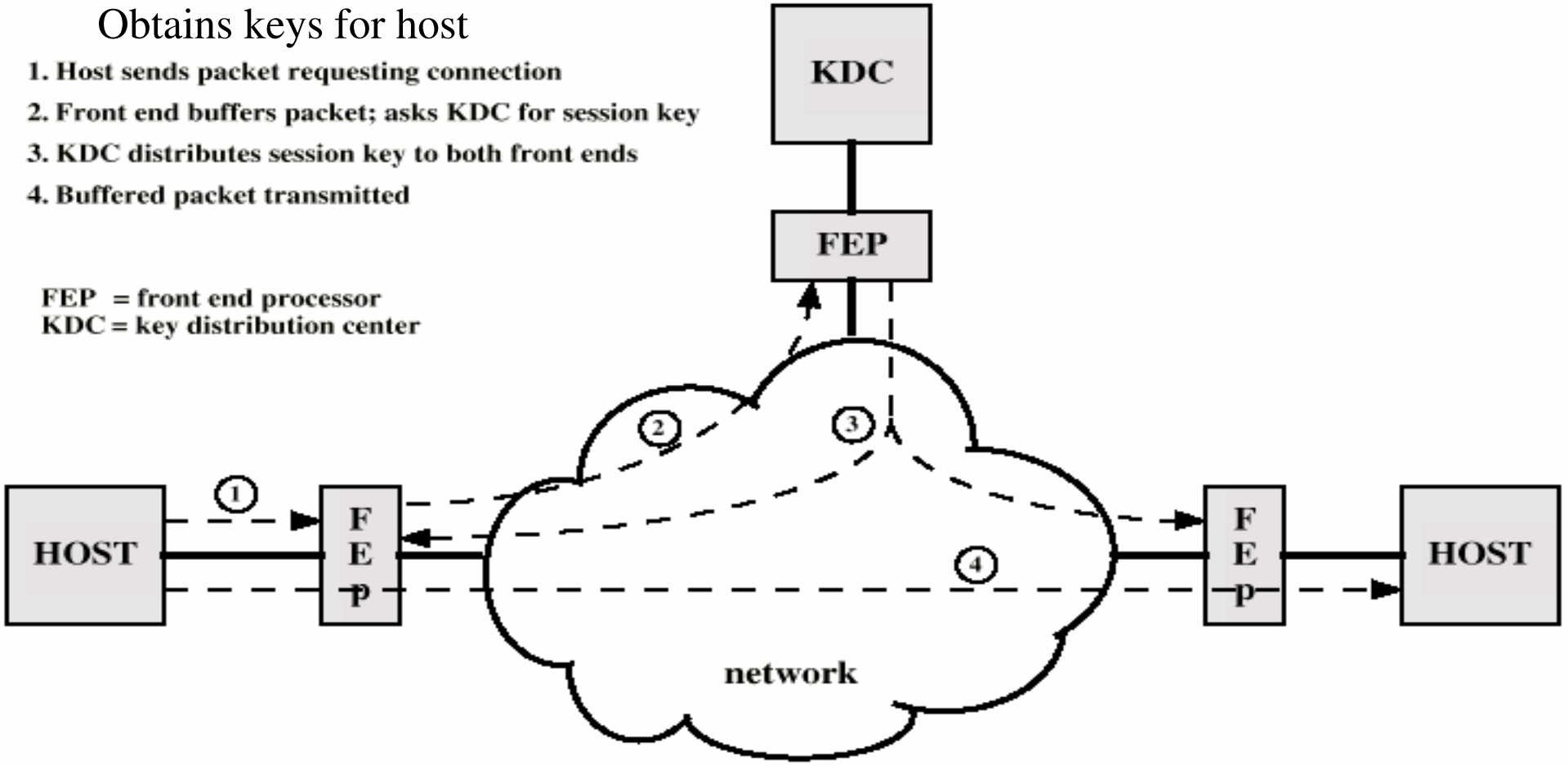
Determines which systems may communicate

Provides one session key for that connection

Front end processor (FEP)

Performs end to end encryption

Obtains keys for host



Message Authentication

Protection against active attacks

- Falsification of data and transactions

Message is **authentic** if it is genuine and comes from the authentic source

Authentication allows receiver to verify that message is authentic

- Message has not altered during communication
- Message is from authentic source
- Message timeline ok (not extra-delay or replays)

Authentication Using Conventional Encryption

Assumes sender and receiver are the only entities that know key

Message includes (for fulfilling above requirements):

- error detection code (if there are alterations)
- sequence number (correct sequencing)
- time stamp (correct timeline)

Authentication Without Message Encryption

Authentication tag generated and appended to each message

Message itself not encrypted (confidentiality not provided)

Useful (sometimes) for:

- Messages broadcast to multiple destinations
  - Have one destination responsible for authentication
- One side heavily loaded
  - Encryption adds to workload
  - Can authenticate (check) messages randomly
- Programs authenticated without encryption can be executed without decoding

Need for both **authentication & encryption** in meeting security requirements!

Authentication Techniques

Message Authentication Code

Generate message authentication code based on shared secret key and the original message

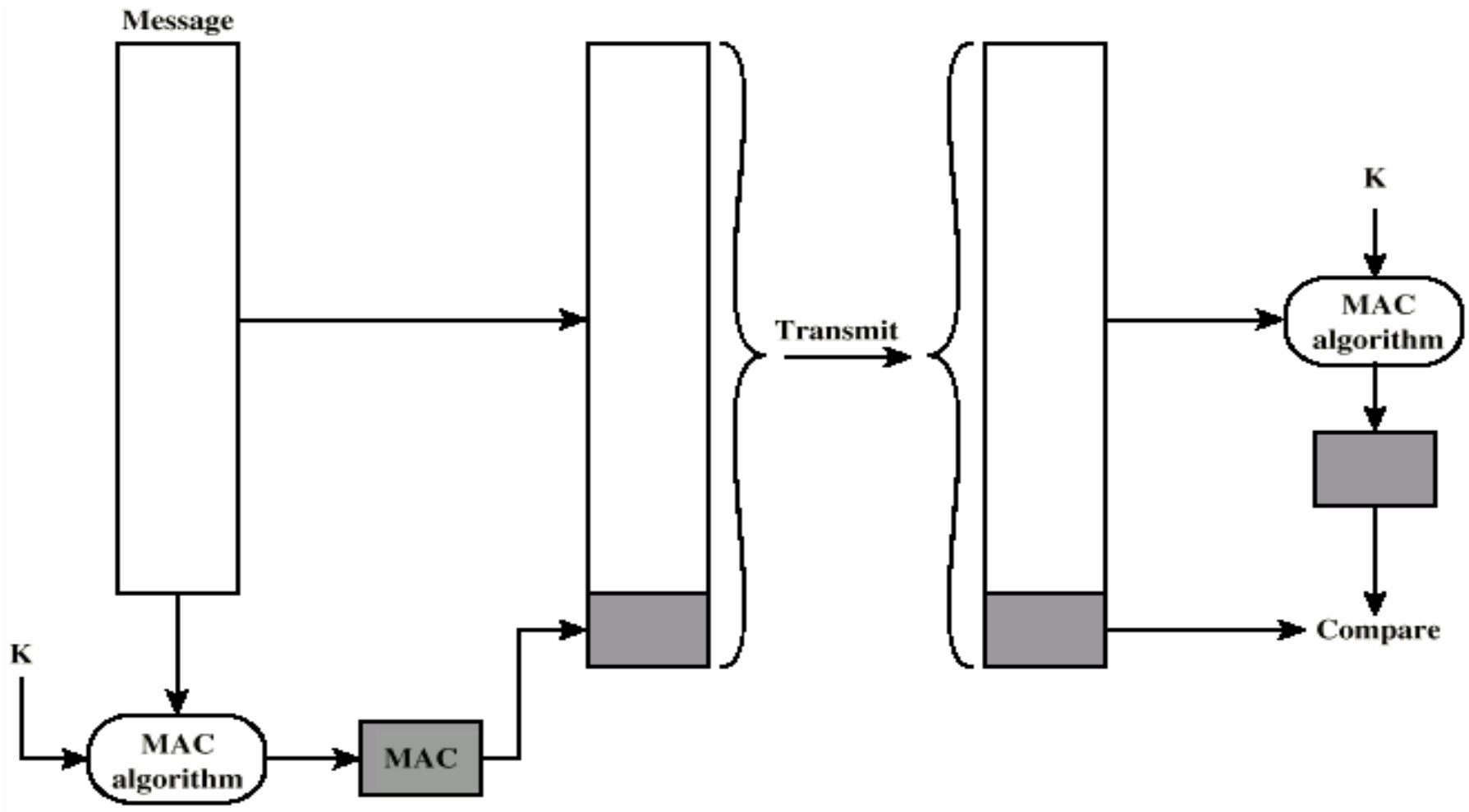
This block of data appended to message (see next slide)

Assumption: Common key shared between the two parts

If only sender and receiver know key, and code matches:

- Receiver assured message has not altered
- Receiver assured message is from alleged sender
- If message has sequence number, receiver assured of proper sequence

Message Authentication Code



One Way Hash Function

Other modern approach for authentication (variation of the previous)

Accepts variable size message and produces fixed size tag (message digest)

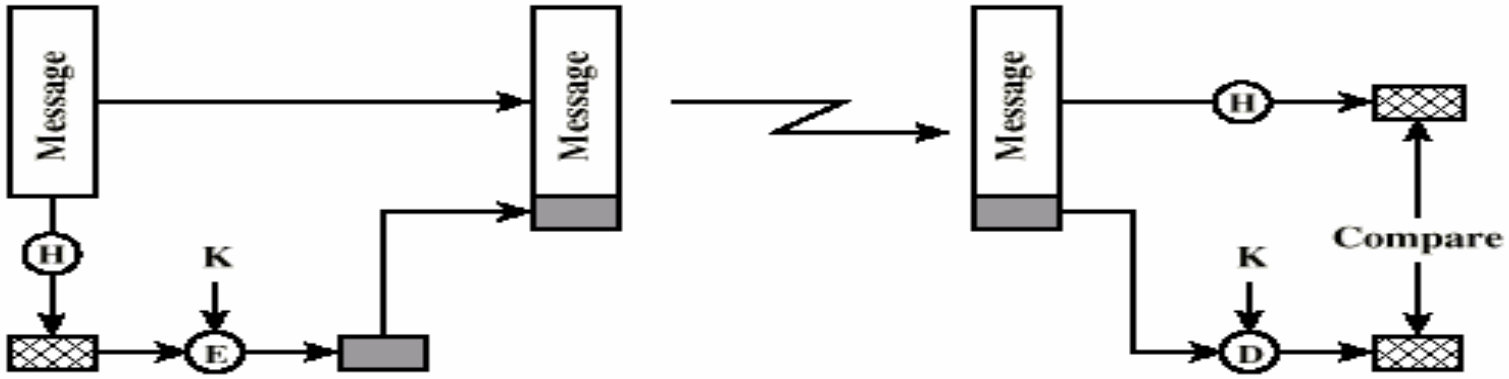
Next slide's figure shows 3 approaches using hash functions

First two use encryption

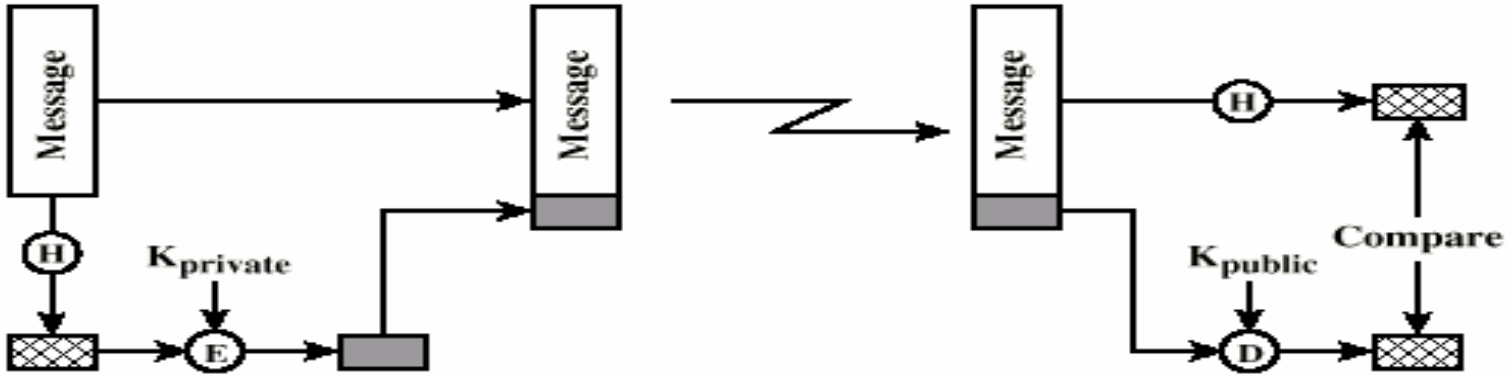
Third use only hash function, operating over the message and a shared secret value

Advantages of authentication without encryption

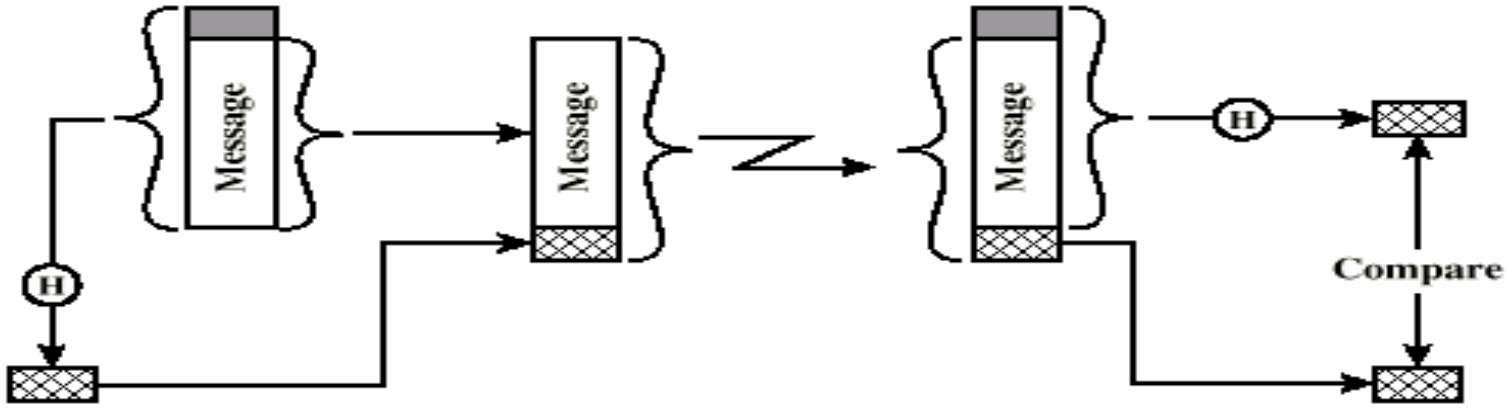
- Encryption is slow
- Encryption hardware expensive
- Encryption hardware optimized to large data
- Algorithms covered by patents
- Algorithms subject to export controls



(a) Using conventional encryption



(b) Using public-key encryption



(c) Using secret value

Secure Hash Functions

Hash function (used in message authentication, digital signatures …) must have following properties:

- Can be applied to any size data block
- Produce fixed length output (e.g. message digest)
- H(x) easy to compute, cheap hardware + software
- Not feasible to reverse; for a given code *h*, not finding *x*, such that *H(x) = h*
- Not feasible to find two message that give the same hash: not H(x)=H(y), for x not y

SHA-1 (Secure Hash Algorithm 1)

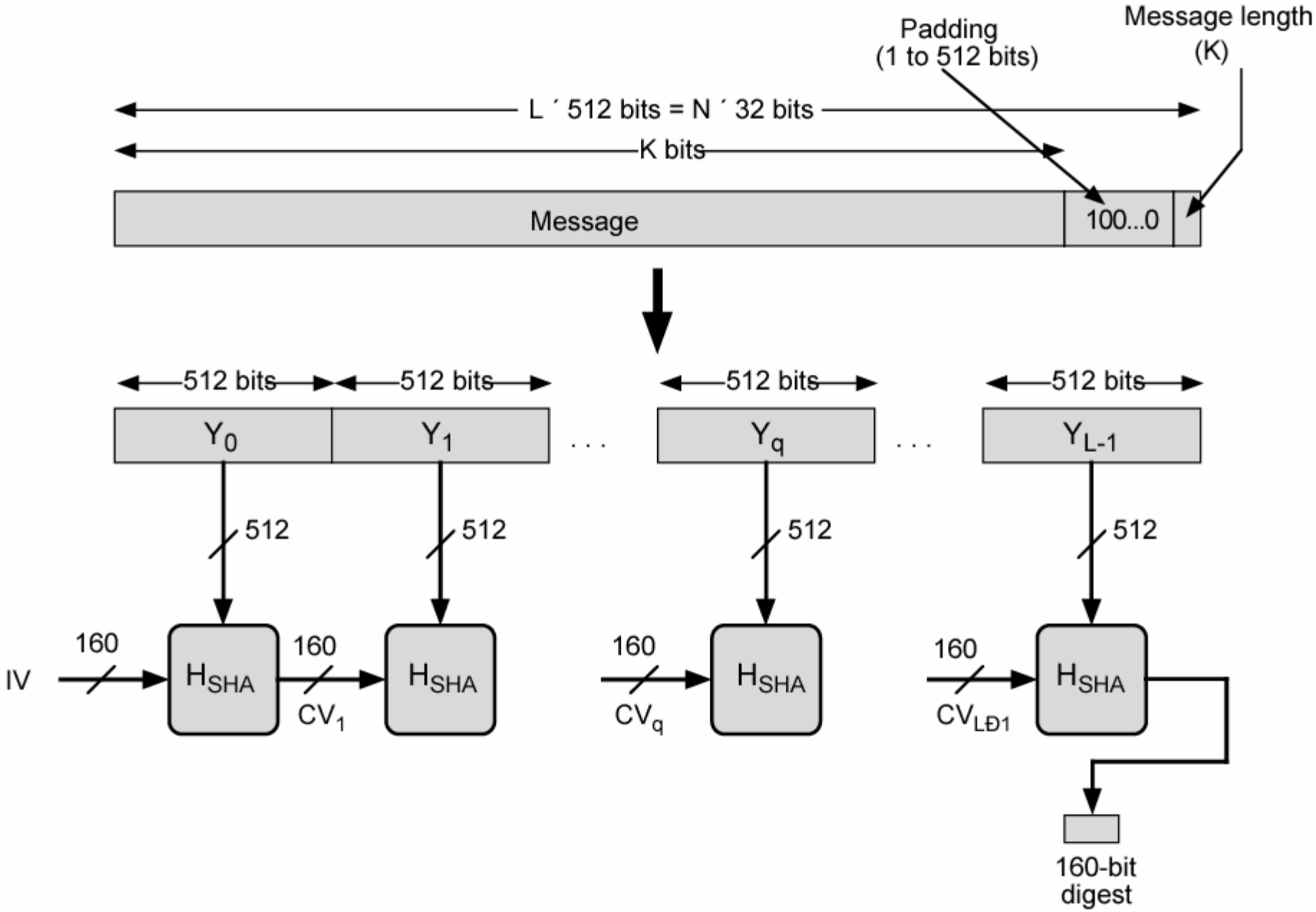
Input message less than 2^{64} bits

- Processed in 512 bit blocks; need for padding bits; compression function with 4 rounds of 20 steps each

Output 160 bit digest (2^{160} operations to find a message with given digest)

Each output bit is function of every input bit: low probability for matching two messages for same digest

Message Digest Generation Using SHA-1



MD5 – Message Digest Algorithm

Produces a 128 bit message digest, based on processing a number of 512 bit blocks

Compression function based on 4 rounds of 16 steps each

More vulnerable than SHA

RIPEMD-160

Derived also from MD4, so similar to MD5 and SHA1

Compression function involves 160 steps