COMPUTER SECURITY AND CYBER LAW (CSCL)

LECTURER: ROLISHA STHAPIT/ DIKSHYA SINGH

UNIT 2

Unit 2: Cryptography and Cryptographic Algorithms LH 4

- Cryptography, Data Encryption Standard, Symmetric key Cryptography(Block and stream ciphers), Asymmetric key Cryptography, Public key Cryptography (RSA), Message Digest 5, Hash Function, Message Authentication Code (MAC).

Some Basic Terminologies

- **Cipher -**The algorithm for transforming plaintext to ciphertext
- **Key** –The information used in cipher known only to sender/receiver is called key.
- **Encipher (encrypt)** –converting plaintext to ciphertext
- **Decipher (decrypt)** – recovering ciphertext from plaintext
- **Encryption:** The process of converting from plaintext to ciphertext is called encryption or enciphering which is one of the best way to prevent unauthorized access.
- **Cryptographic Algorithm:** It is a set of rules based on some mathematical function used for encryption.
- **Decryption**: The process of restoring the plaintext from the cipher is called decryption.
- **Cryptography** – The study of encryption principles/methods is called cryptography
- **Cryptanalysis (codebreaking)** – The study of methods of deciphering ciphertext without knowing key
- **Cryptology** –The field of both cryptography and cryptanalysis

Cryptography

- The word cryptography comes from two Greek words meaning "secret writing" and is the art and science of concealing(hiding) meaning.
- Cryptanalysis is the breaking of codes.
- The basic component of cryptography is a cryptosystem.
- A good cryptosystem protects against all types of attacks The goal of cryptography is to keep enciphered information secret.

Definition 9–1. A *cryptosystem* is a 5-tuple (E, D, M, K, C), where M is the set of *plaintexts*, K the set of *keys*, C is the set of *ciphertexts*, $E: M \times K \rightarrow C$ is the set of *enciphering functions*, and $D: C \times K \rightarrow M$ is the set of *deciphering functions*.

The cryptographic systems are characterized along the three independent dimensions:

- The type of operations used for transforming plaintext to ciphertext
- - substitution / transposition / product
- The number of keys used
- - single-key or private / two-key or public key
- The way in which plaintext is processed
- - block / stream cipher

Cryptanalysis Technique

- **Cryptanalytic attack:** Analysis of the nature or characteristic of the algorithm to attempt to deduce specific plaintext or the key being used.
- **Brute-force attack:** In brute-force approach the attacker tries every possible key on a piece of ciphertext until an intelligent translation into plaintext is obtained.

Types of Cryptanalytic Attacks

Based on the information known to the cryptanalyst, the cryptanalytic attacks are classified as follows:

- **In cipher text only attack**, encryption algorithm and cipher text are known to the cryptanalyst.
- **In known plaintext attack**, information known includes: encryption algorithm, cipher text, and one or more plaintext-cipher text pairs formed with the secret key.
- **In chosen plaintext attack,** information known includes: encryption algorithm, cipher text, and chosen plaintext and its corresponding cipher text generated with the secret key.
- **In chosen cipher text attack,** information known includes: encryption algorithm, cipher text, and chosen cipher text and its corresponding decrypted plaintext with the secret key.
- **In chosen text attack,** information known in both chosen plaintext attack and chosen cipher text attack is available to the cryptanalyst.

How secure is your system?

- **Unconditionally Secure:** An encryption algorithm is unconditionally secure if ciphertext generated by the algorithm does not contain enough information to determine uniquely the corresponding plaintext.

  - No matter how much computer power or time is available, the cipher cannot be broken since the ciphertext provides insufficient information to uniquely determine the corresponding plaintext

- **Computational Secure:** An encryption scheme is computationally secured if:

  - The cost of breaking the cipher exceeds the value of the encrypted information

  - The time required to break the cipher exceeds the useful lifetime of the information. Given limited computing resources (eg time needed for calculations is greater than age of universe), the cipher cannot be broken.

Classical Cryptosystems

- Classical cryptosystems (also called single-key or symmetric cryptosystems) are cryptosystems that use the same key for encipherment and decipherment. There are two basic types of classical ciphers: ***transposition ciphers and substitution ciphers.***

Transposition Ciphers:

- A transposition cipher rearranges the characters in the plaintext to form the ciphertext. The letters are not changed.
- Example: **Rail-Fence Cipher**
- **Rail-Fence Cipher:**
- The Rail Fence Cipher is a form of transposition cipher that derives its name from the way in which it is encoded.
- In the rail fence cipher, the plaintext is written downwards and diagonally on successive "rails" of an imaginary fence, then moving up when we reach the bottom rail.
- When we reach the top rail, the message is written downwards again until the whole plaintext is written out. The message is then read off in rows.

- For example, using 3 "rails" and a message of 'WE ARE DISCOVERED FLEE AT ONCE', the cipherer writes out:

  W . . . E . . . C . . . R . . . L . . . T . . . E

  . E . R . D . S . O . E . E . F . E . A . O . C .

  . . A . . . I . . . V . . . D . . . E . . . N . .

Then reads off:

WECRL TEERD SOEEF EAOCA IVDEN

Similarly, if we have 3 "rails" and a message of THIS IS THE PLAINTEXT, the cipherer writes out (we are not showing diagonal move here just write in down rail a step ahead):

```
T     S     T       P   I     E
.H    I     H     L   N     X
..I       S     E     A   T       T
```

The ciphertext is T S T P I E H I H L N X I S E A T

The problem with Rail Fence Cipher is that the rail fence cipher is not very strong; the number of practical keys is small enough that a cryptanalyst can try them all by hand. To decrypt we get the number of letters to be skipped. For this if the number of rail is n key is so in our e.g. n = 3 and key is 18/3 = 6 i.e. skip 6 letters from the letter you are reading every time to get plaintext (remember to go circular that is if count ends continue from the starting letter leaving the read letter). See below:

T	S	T	P	I	E	H	I	H	L	N	X	I	S	E	A	T	T
1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6

We have selected letter with index 1 THI

Substitution Cipher:

- In substitution ciphers the letters are systematically replaced by other letters or symbols. **Eg. Caesar Cipher, Vigenere Cipher.**

Caesar Cipher:

- It is the simple shift mono-alphabetic classical cipher where each letter is replaced by a letter 3 position (actual Caesar cipher) ahead using the circular alphabetic ordering i.e. letter after Z is A.
- So when we encode HELLO WORLD, the cipher text becomes KHOORZRUOG
- Here we number each English alphabet starting from 0 (A) to 25 (Z).

- The encryption can also be represented using modular arithmetic by first transforming the letters into numbers, according to the scheme, A = 0, B = 1, ..., Z = 25.
- Encryption of a letter c by a shift k can be described mathematically as,

$$c = E_k(m) = (m + k) \mod 26$$

- Decryption is performed similarly,

$$m = D_k(c) = (c + 26 - k) \mod 26$$

Where k = 3

Similarly, consider some examples of Caesar cipher;

- Plaintext: meet me after the toga party
- Ciphertext: PHHW PH DIWHU WKH WRJD SDUWB

- Plaintext: the quick brown fox jumps over the lazy dog
- Ciphertext: WKH TXLFN EURZQ IRA MXPSV RYHU WKH ODCB GRJ

For Example we need to encrypt the original message “SECURITY”. Given the key function is C= (p + x) mod 26 where x=19.

Now, C=(S+x)mod26=(18+19)mod26=37mod26=11 i.e L

C=(E+x)mod26=(4+19)mod26=23mod26=23 i.e X

C=(C+x)mod26=(2+19)mod26=21mod26= 21i.e V

C=(U+x)mod26=(20+19)mod26=39mod26=13 i.e N

C=(R+x)mod26=(17+19)mod26=36mod26=10 i.eK

C=(I+x)mod26=(8+19)mod26=27mod26=1 i.e B

C=(T+x)mod26=(19+19)mod26=38mod26=12 i.e M

C=(Y+x)mod26=(24+19)mod26=43mod26=16 i.e Q

Therefore “SECURITY” is encrypted as “LYVNKBMQ”

- **Attacking the Cipher**

Caesar Cipher is quite easily broken even with cipher text only. One can attack the cipher text using exhaustive search by trying all possible keys until you find the right one. Exhaustive search is best suited if the key space is small and we have only 26 possible keys in Caesar cipher. Another approach of attacking the cipher is statistical analysis where we compare the ciphertext to 1-gram model of English.

Vigenere Cipher: Substitution Cipher (Polyalphateic)

- It is like Caesar cipher, but uses a phrase for e.g. for the message THE BOY HAS THE BALL and the key VIG, encipher using Caesar cipher for each letter:
- key VIGVIGVIGVIGVIGV
- plain THEBOYHASTHEBALL
- cipher OPKWWECIYOPKWIRG
- Here, generally, we repeatedly write key above the plaintext and use the Caesar cipher for each letter in the plaintext where key for each letter being processed is taken from the repeated key letter just above it.

This process is simplified by using the table as below called Tableau

Key

Plain text

	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
A	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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

One-Time Pad

- It is a variant of a Vigenère cipher with a random key at least as long as the message.
- Since it has very high key length it is provably unbreakable.
- Each new message requires a new key of the same length as the new message.
- For ciphertext DXQR plaintext DOIT (key AJIY) and plaintext DONT (key AJDY) are equally likely and any other 4 letters.

Stream vs. Block Cipher:

- **Stream Cipher:** A symmetric encryption algorithm in which ciphertext output is produced bit-by-bit or byte-by-byte from stream of plaintext input. Like, character by character conversion.
- **Block cipher:** A symmetric encryption algorithm in which a block of plaintext (typically 64bits or 128bits) is transformed as a whole into a ciphertext block of the same length.

Stream Cipher:

Let E be an encipherment algorithm, and let Ek(b) be the encipherment of message b with key k. Let a message m = b1b2 …, where each bi is of a fixed length, and let k = k1k2 …. Then a stream cipher is a cipher for which Ek(m) = Ek1(b1)Ek2(b2) …. If the key stream k of a stream cipher repeats itself, it is a periodic cipher and the length of its period is one cycle of k1k2 … Vigenère cipher has bi = 1 character, k = k1k2 …where ki = 1 character and each bi is enciphered using ki mod length(k).

Block Cipher:

Let E be an encipherment algorithm, and let Ek(b) be the encipherment of message b with key k. Let a message m = b1b2 …, where each bi is of a fixed length. Then a block cipher is a cipher for which Ek(m) = Ek(b1)Ek(b2) …. Data Encryption Standard has bi = 64 bits, k = 56 bits and each bi enciphered separately using k

Advantages of Stream Ciphers

- Speed of transformation: Because each symbol is encrypted without regard for any other plaintext symbols, each symbol can be encrypted as soon as it is read. Thus, the time to encrypt a symbol depends only on the encryption algorithm itself, not on the time it takes to receive more plaintext.

- Low error propagation: Because each symbol is separately encoded, an error in the encryption process affects only that character.

Disadvantages of Stream Ciphers

- Low diffusion: Each symbol is separately enciphered. Therefore, all the information of that symbol is contained in one symbol of the ciphertext.

- Susceptibility to malicious insertions and modifications: Because each symbol is separately enciphered, an active interceptor who has broken the code can splice together pieces of previous messages and transmit a spurious new message that may look authentic.

Advantages of Block Ciphers

- High diffusion: Information from the plain-text is diffused into several ciphertext symbols. One ciphertext block may depend on several plaintext letters.

- Immunity to insertion of symbols: Because blocks of symbols are enciphered, it is impossible to insert a single symbol into one block. The length of the block would then be incorrect, and the decipherment would quickly reveal the insertion.

Disadvantages of Block Ciphers

- Slowness of encryption: The person or machine using a block cipher must wait until an entire block of plaintext symbols has been received before starting the encryption process.

- Error propagation: An error will affect the transformation of all other characters in the same block.

Encryption Type

Two fundamental approaches are used

- **Symmetric (Private- Key or Conventional) Encryption**

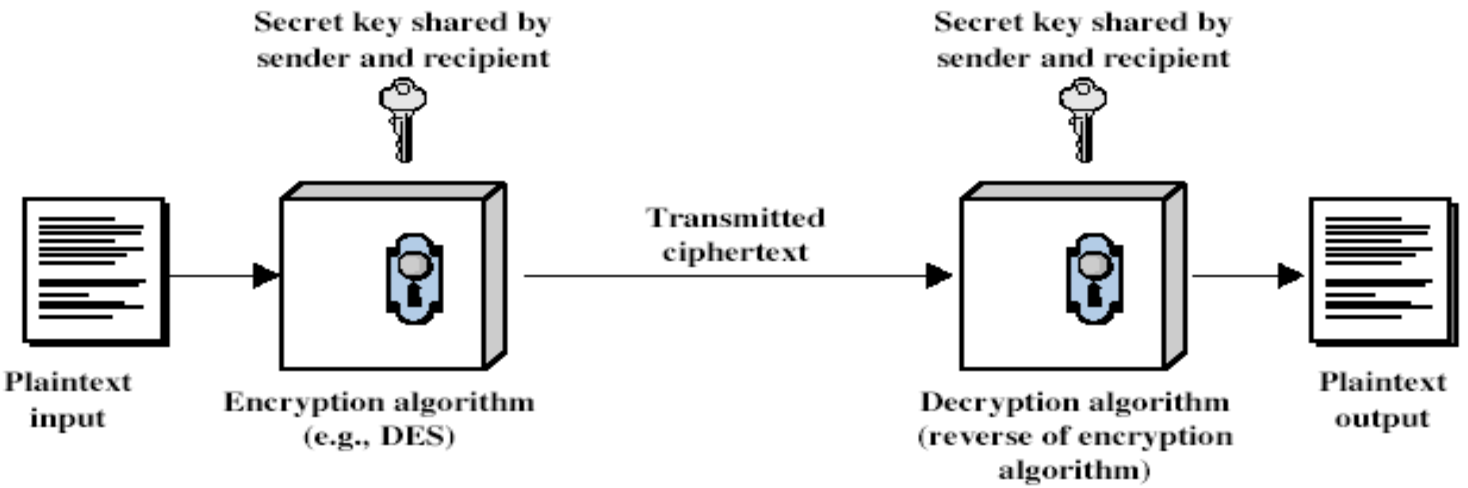
  E.g. Data Encryption Standard (DES), and Advanced Encryption Standard(AES)

- **Asymmetric (Public-key) Encryption**

  E.g. RSA, Deffie Hellman, etc..

Symmetric Encryption

- Symmetric encryption is a form of cryptosystem in which encryption and description are perform using the same key. It is also called conventional encryption or single key or private key encryption.
- In symmetric encryption, the sender and recipient share a common key
- The symmetric encryption transform plaintext into ciphertext using a secret key and an encryption algorithm



A Symmetric cipher scheme has five ingredients:

- Plaintext: This is an original intelligible message or data that is fed into the algorithm as input.
- Encryption Algorithm: The encryption algorithm performs various substitutions and transformation on plaintext
- Secret Key: The secret key is also input to the encryption algorithm
- Ciphertext: It is the coded message obtained after encryption as an output. It depends upon plaintext & key.
- Decryption Algorithm: It is reverse to encryption algorithm. It takes ciphertext and secret key and produces original plaintext.

Data Encryption Standard

- The Data Encryption Standard (DES) was designed to encipher sensitive but non classified data.
- It is bit oriented, unlike the other ciphers we have seen.
- It uses both transposition and substitution and for that reason is sometimes referred to as a product cipher. Its input, output, and key are each 64 bits long.
- The sets of 64 bits are referred to as blocks.
- The cipher consists of 16 rounds, or iterations. Each round uses a separate key of 48 bits.
- These round keys are generated from the key block by dropping the parity bits (reducing the effective key size to 56 bits), permuting the bits, and extracting 48 bits.
- A different set of 48 bits is extracted for each of the 16 rounds. If the order in which the round keys is used is reversed, the input is deciphered.

- The rounds are executed sequentially, the input of one round being the output of the previous round.
- The right half of the input, and the round key, are run through a function f that produces 32 bits of output; that output is then xor'ed into the left half, and the resulting left and right halves are swapped

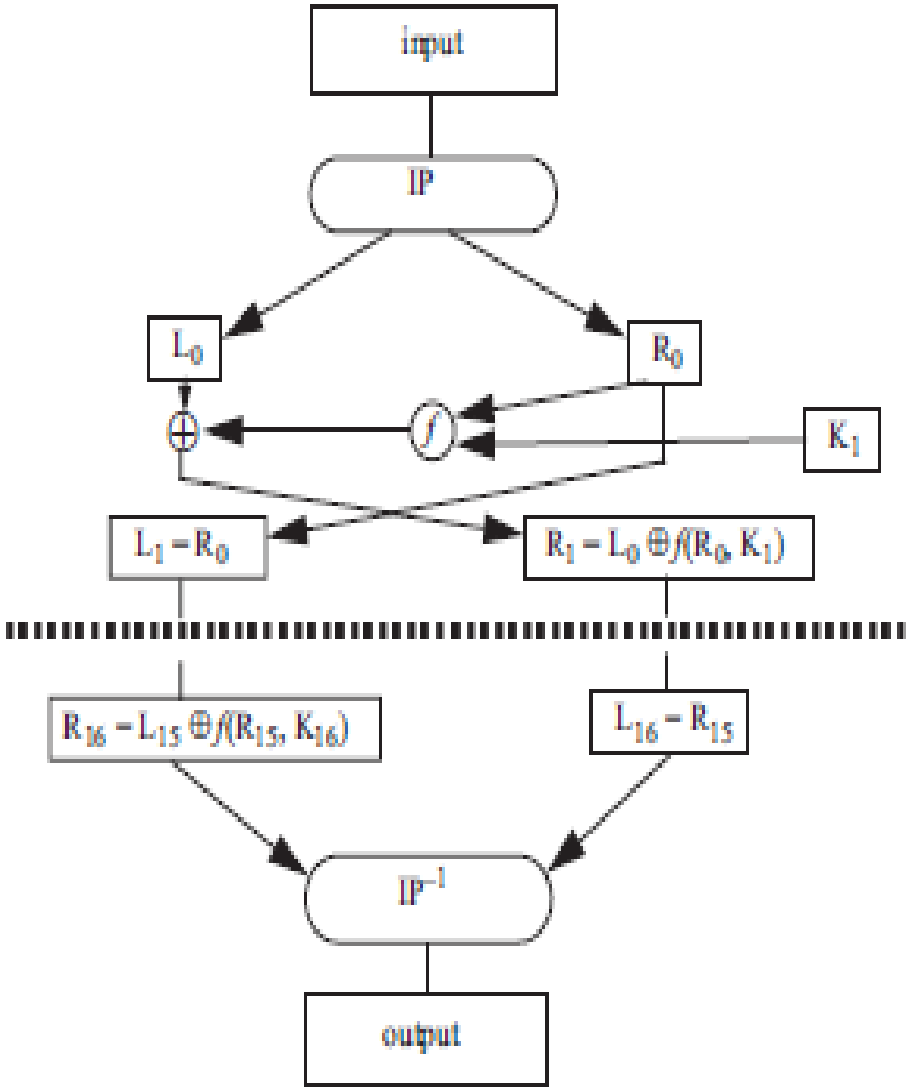
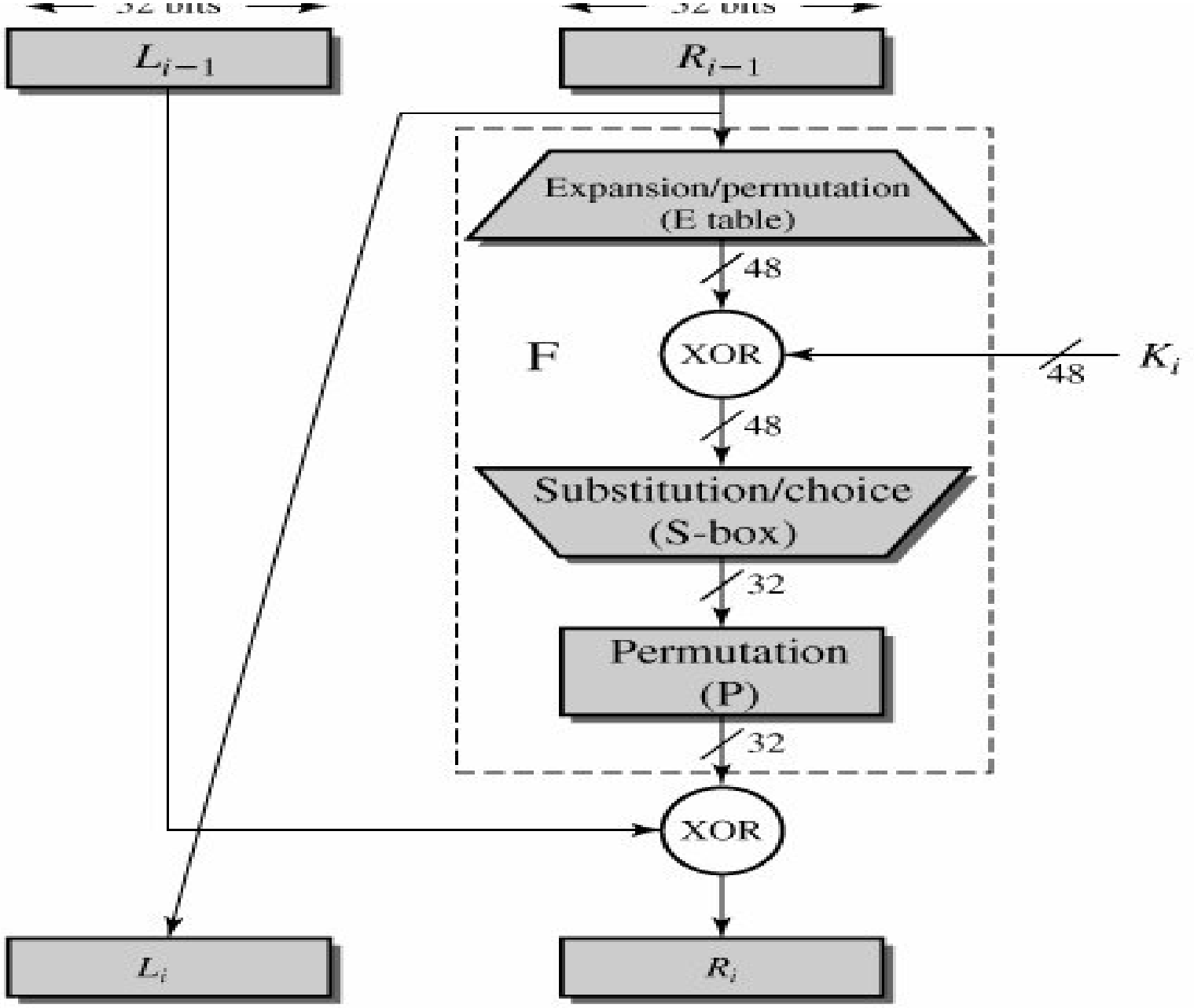
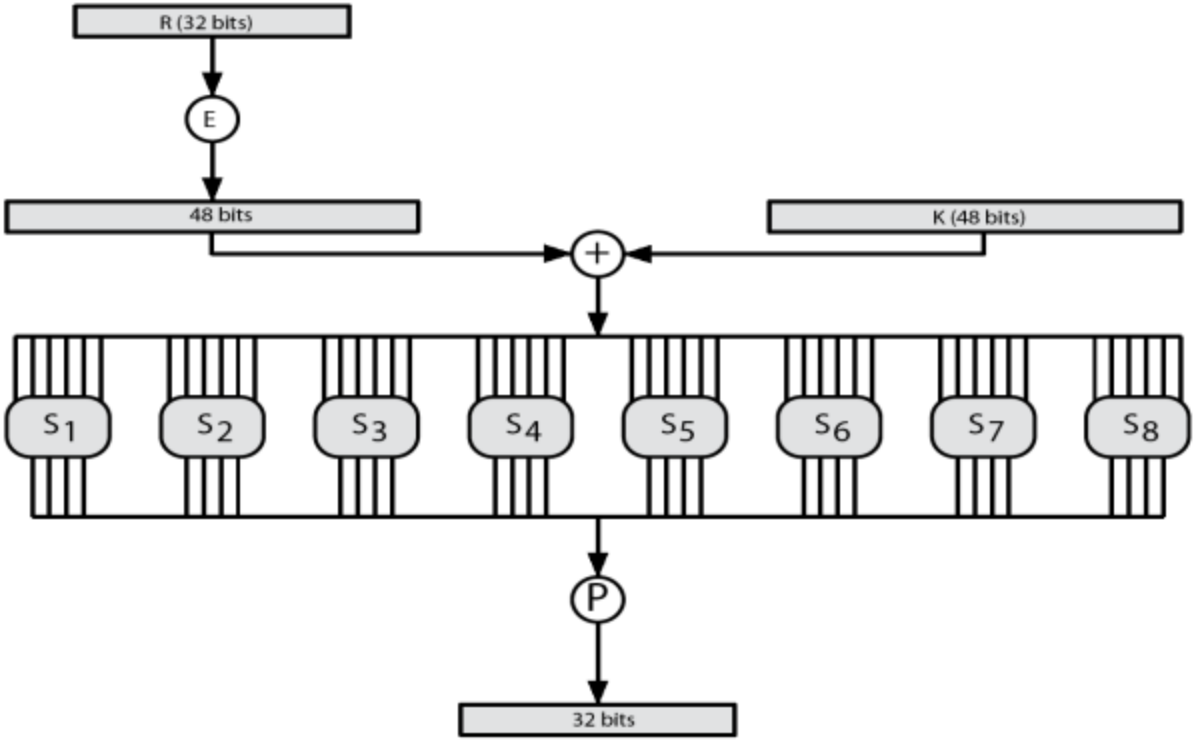


Figure 8–6 DES message encipherment and decipherment.

Details of Single Round

- The function f provides the strength of the DES. The right half of the input (32 bits) is expanded to 48 bits, and this is XOR'ed with the round key.
- The resulting 48 bits are split into eight sets of six bits each, and each set is put through a substitution table called the S-box.
- Each S-box produces four bits of output. They are catenated into a single 32-bit quantity, which is permuted.
- The resulting 32 bits constitute the output of the f function
- The substitution consists of a set of eight S-boxes, each of which accepts 6 bits as input and produces 4 bits as output.


32 bits
32 bits
L_{i-1}
R_{i-1}
Expansion/permutation
(E table)
48
F
XOR
48
K_i
48
Substitution/choice
(S-box)
32
Permutation
(P)
32
XOR
L_i
R_i



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

$$R_i = L_{i-1} \oplus \mathrm{F}(R_{i-1}, K_i)$$

(a) 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

(b) Inverse Initial 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

(c) Expansion Permutation (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

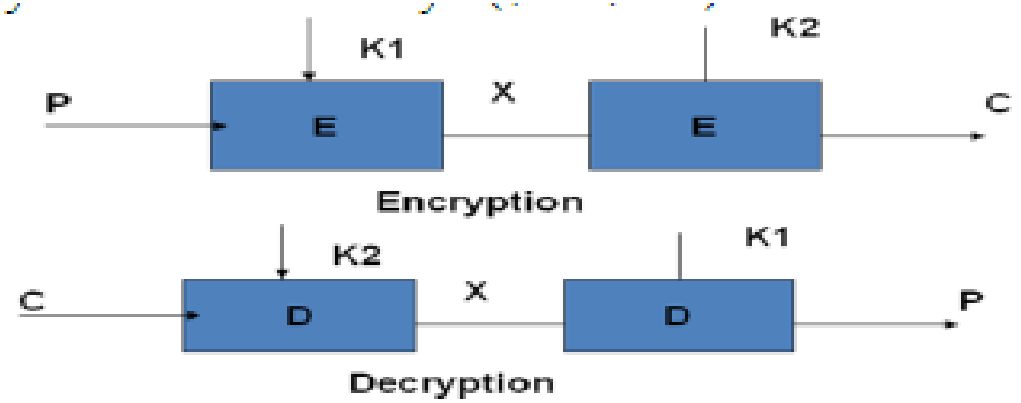
(d) Permutation Function (P)

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

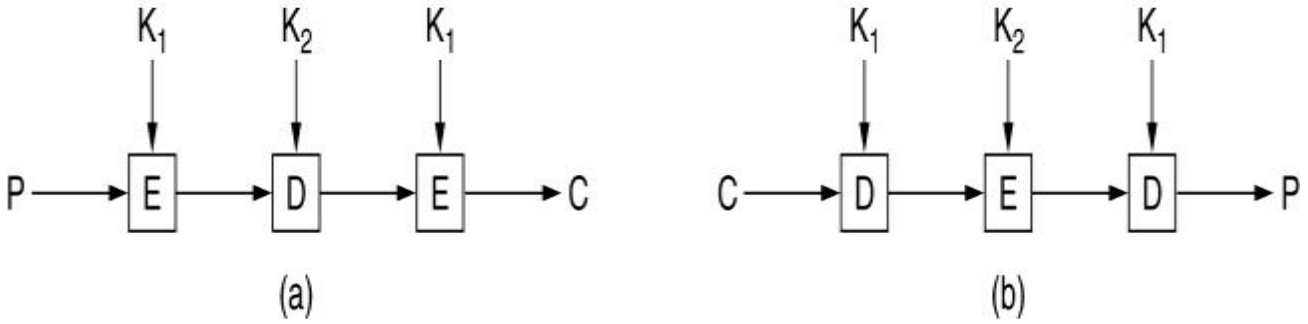
- Double DES: The simplest form of multiple encryption has two encryption stages and two keys is called Double-DES. The 2-DES algorithm encrypts on each block.

$$C = EK2(EK1(P))$$

$$P = DK2(DK1(C))$$



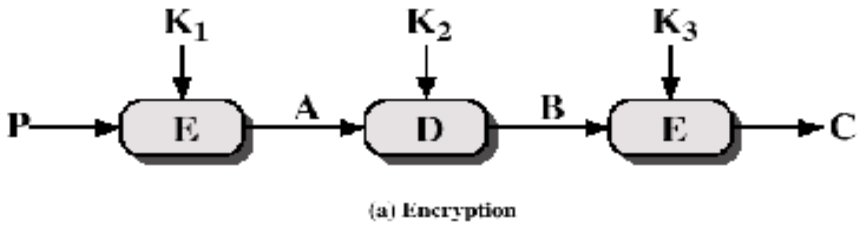
- Triple-DES Variants: 3-DES with 2-Keys, 3-DES with 3-Keys
- Triple-DES with 2-Keys: It is a popular alternative to single-DES, but suffers from being 3 times slower to run. The uses of encryption & decryption stages are equivalent. But the chosen structure allows for compatibility with single-DES implementations.
- It uses two- keys with encrypt-decrypt-encrypt sequence. Encryption Decryption
- C = EK1(DK2(EK1(P))) P = DK1(EK2(DK1(C)))



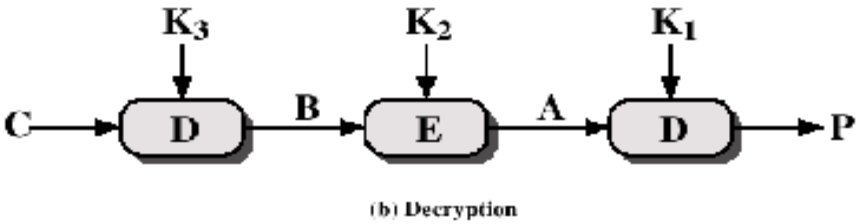
- Triple-DES with 3-Keys:

The use of Triple-DES with Three-Keys to avoid even these possible attacks on 2-Key 3-DES.

- Use three keys and three executions of the DES algorithm
- C = EK3[DK2[EK1[P]]] P = DK3[EK2[DK1[C]]]



(a) Encryption



(b) Decryption

Asymmetric Public Key Cryptography

- Asymmetric encryption is a form of cryptosystem in which encryption and decryption are performed using the different keys: one a public-key and one a private key. It is also known as public-key encryption
- Asymmetric Encryption can be used for confidentiality, authentication, or both.
- The most widely used public-key encryption is RSA.
- It is asymmetric because those who encrypt messages or verify signatures cannot decrypt messages or create signatures.

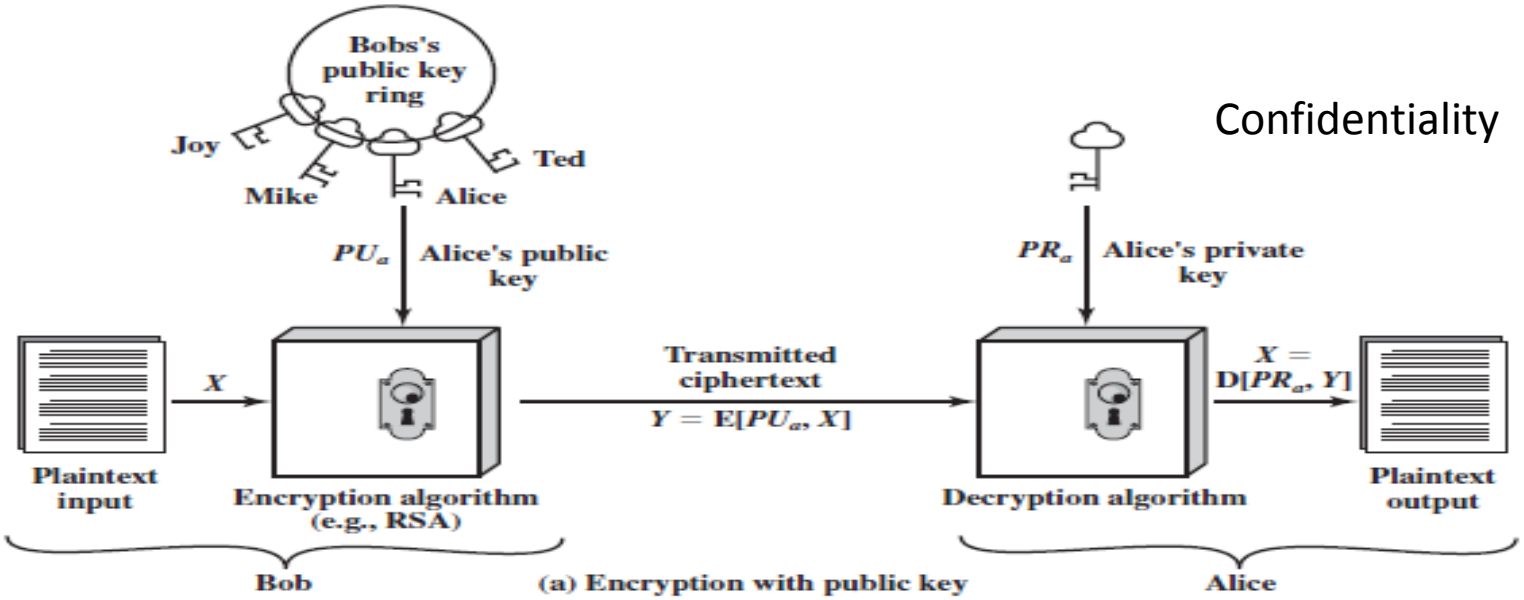
Public key cryptosystem must meet the following three conditions.

1. It must be computationally easy to encipher or decipher a message given the appropriate key.

2. It must be computationally infeasible to derive the private key from the public key.

3. It must be computationally infeasible to determine the private key from a chosen plaintext attack.

Confidentiality



Authenticity

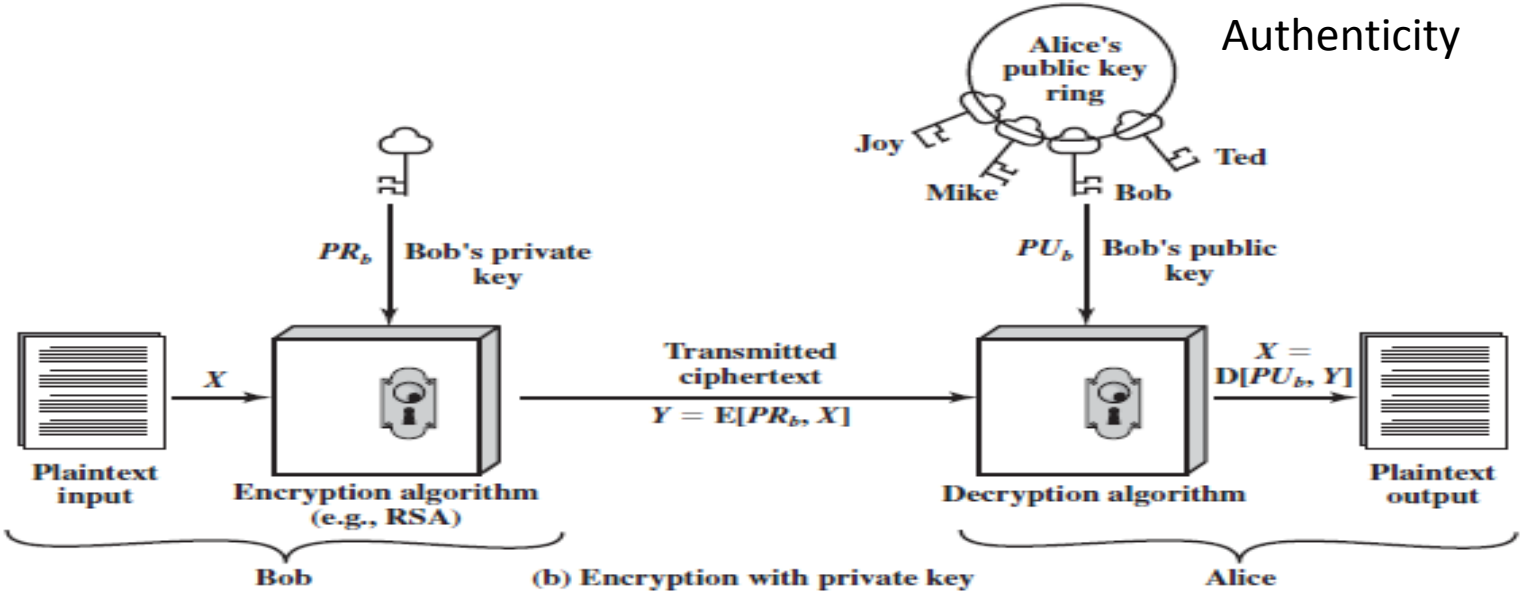


Figure 3.9 Public-Key Cryptography

- A public-key encryption scheme has six ingredients
- Plaintext: This is the readable message or data that is fed into the algorithm as input.
- Encryption algorithm : The encryption algorithm performs various transformations on the plaintext.
- Public and private key: This is a pair of keys that have been selected so that if one is used for encryption, the other is used for decryption. The exact transformations performed by the encryption algorithm depend on the public or private key that is provided as input.
- Ciphertext: This is the scrambled message produced as output. It depends on the plaintext and the key. For a given message, two different keys will produce two different ciphertexts.
- Decryption algorithm: This algorithm accepts the ciphertext and the matching key and produces the original plaintext

PUBLIC-KEY CRYPTOGRAPHY ALGORITHMS

RSA (Rivest, Shamir & Adleman)

- RSA the most widely used general public key encryption algorithm in MIT in 1978.
- It is based on exponentiation in a finite field over integers modulo a prime, using large integers. Its security is due to the cost of factoring large numbers which is also known as integer factorization problem IFP.
- RSA algorithm has three steps: Key generation, Encryption & Decryption
- Operation: RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key.

RSA Key Generation:

1. Choose two distinct large random prime numbers p and q
2. Compute n = pq, n is used as the modulus for both the public and private keys
3. Compute the totient: $\varphi(n) = (p - 1)(q - 1)$.
4. Choose an integer e such that $1 < e < \varphi(n)$, and e and $\varphi(n)$ share no factors other than 1 i.e. e and $\varphi(n)$ are relatively prime)
5. e is released as the public key exponent
6. Compute d to satisfy the congruence relation $ed \equiv 1 \bmod \varphi(n)$ ($ed \bmod \varphi(n) = 1$) ; i.e. $de = 1 + k\varphi(n)$ for some integer k.
7. d is kept as the private key exponent

Encrypting Messages

- Alice transmits her public key (n, e) to Bob and keeps the private key secret. Bob send message M to Alice by turning M into a number m < n by using a reversible protocol called a padding scheme. He then computes the ciphertext c as: $c = m^e \bmod n$. Bob then transmits c to Alice.

Decrypting Messages

- Alice can recover m from c by using her private key exponent d by the following computation: $m = c^d \bmod n$. Given m, she can recover the original message M.

- **RSA Example 1:**

1. Select primes: p=17 & q=11
2. Compute $n = pq$ =17×11=187
3. Compute ø(n)=(p–1)(q-1)=16×10=160
4. Select e *:* gcd(e,160)=1; choose e=7
5. Determine d*: de*=1 mod 160 and d < 160 Value is d=23 since 23×7=161= 10×160+1
6. Publish public key KU={7,187}
7. Keep secret private key KR={23,187}
8. Given message M = 88 (88<187)
9. Encryption: C = 88^7 mod 187 = 11
10. Decryption: M = 11^{23} mod 187 = 88

One More Example:

Consider primes p=11, q=3. Now, compute $n = pq = 11.3 = 33$ and totient $\varphi(n) = (p-1)(q-1) = 10.2 = 20$.

Choose e=3; Check gcd(e, φ(n)) = gcd(3, 20) = 1 (i.e. 3 and 20 have no common factors except 1),

Compute d such that $ed \equiv 1 \pmod{\varphi(n)}$
i.e. find a value for d such that φ(n) divides (ed-1)
i.e. find d such that 20 divides 3d-1.
Simple testing (d = 1, 2, ...) gives d = 7
Check: ed-1 = 3.7 - 1 = 20, which is divisible by φ(n)

The notation '$a \equiv b \pmod{n}$' means a and b have the same remainder when divided by n, or, equivalently,

Public key = (n, e) = (33, 3)
Private key = (n, d) = (33, 7).

This is actually the smallest possible value for the modulus n for which the RSA algorithm works. Now say we want to encrypt the message m = 7,
$c = m^e \bmod n = 7^3 \bmod 33 = 343 \bmod 33 = 13$.
Hence the ciphertext c = 13. To check decryption we compute $m' = c^d \bmod n = 13^7 \bmod 33 = 7$.

Example: Consider, p = 61 and q = 53 now, compute n = pq = 61×53 = 3233

Compute the totient $\phi(n) = (p - 1)(q - 1) = (61-1)(53-1) = 3120$

Choose e > 1 relatively prime to 3120; e = 17

Compute d such that $ed \equiv 1 \bmod \phi(n)$ e.g., by computing the modular multiplicative inverse of e modulo $\phi(n)$: d = 2753 since 17×753 = 46801 = 1 + 15×3120.

The public key is (n = 3233, e = 17).

For a padded message m the encryption function is:

$c = m^e$ **mod** n = m^{17} **mod** 3233.

The private key is (n = 3233, d = 2753). The decryption function is:

$m = c^d$ **mod** n = c^{2753} **mod** 3233.

- For example, to encrypt m = 123, we calculate
- $c = 123^{17}$ **mod** 3233 = 855 to decrypt c = 855, we calculate m = 855^{2753} **mod** 3233 = 123

Notes

- In cryptography, **confusion and diffusion** are two properties of the operation of a secure cipher identified by Claude Shannon.
- Confusion means that each binary digit (bit) of the ciphertext should depend on several parts of the key.
- Diffusion means that if we change a single bit of the plaintext, then half of the bits in the ciphertext should change, and similarly, if we change one bit of the ciphertext, then approximately one half of the plaintext bits should change. Since a bit can have only two states, when they are all re-evaluated and changed from one seemingly random position to another, half of the bits will have changed state.

Message Authentication Code (MAC)

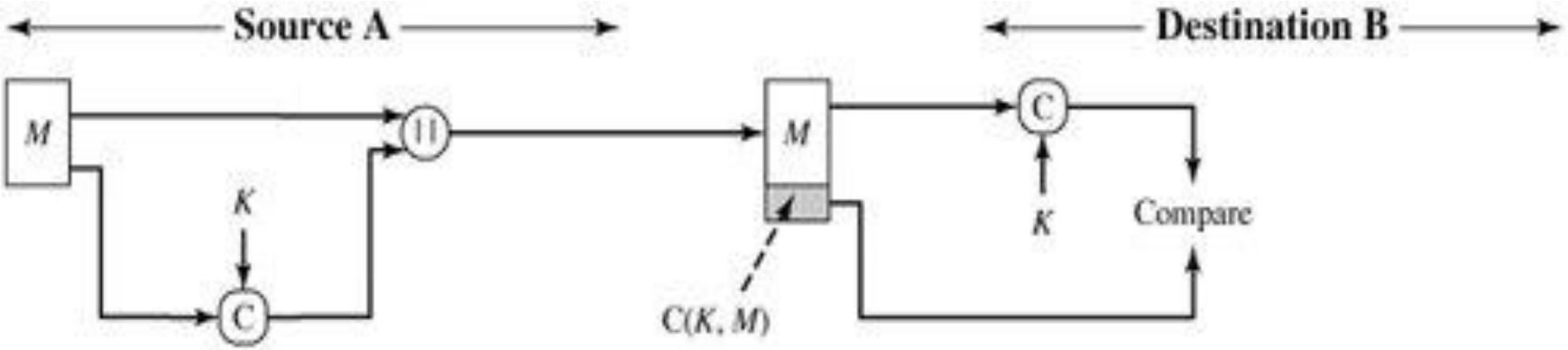
- An alternative authentication technique involves the use of a secret key to generate a small fixed-size block of data, known as a cryptographic checksum or MAC that is appended to the message.
- This technique assumes that two communicating parties, say A and B, share a common secret key K.
- When A has a message to send to B, it calculates the MAC as a function of the message and the key: MAC = C(K,M), where

  M = input message

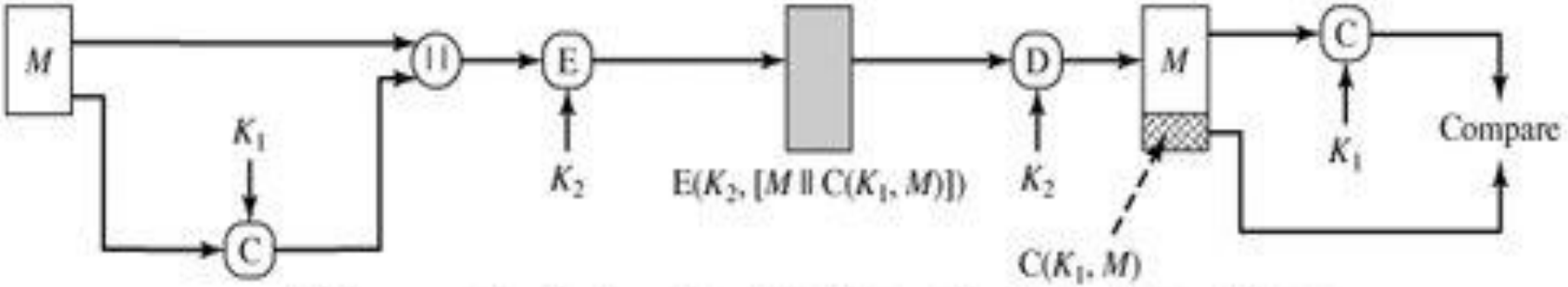
  C = MAC function

  K = shared secret key

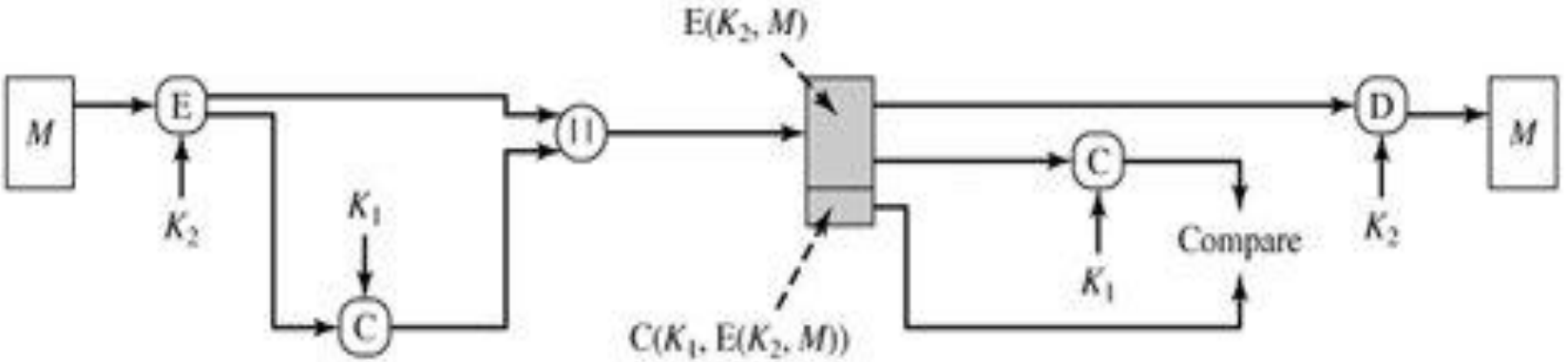
  MAC = message authentication code



(a) Message authentication



(b) Message authentication and confidentiality; authentication tied to plaintext



(c) Message authentication and confidentiality; authentication tied to ciphertext

- The message plus MAC are transmitted to the intended recipient.
- The recipient performs the same calculation on the received message, using the same secret key, to generate a new MAC. The received MAC is compared to the calculated MAC (Figure a).
- If we assume that only the receiver and the sender know the identity of the secret key, and if the received MAC matches the calculated MAC, then

1. The receiver is assured that the message has not been altered. If an attacker alters the message but does not alter the MAC, then the receiver's calculation of the MAC will differ from the received MAC. Because the attacker is assumed not to know the secret key, the attacker cannot alter the MAC to correspond to the alterations in the message.

2. The receiver is assured that the message is from the alleged sender. Because no one else knows the secret key, no one else could prepare a message with a proper MAC.

3. If the message includes a sequence number (such as is used with HDLC, X.25, and TCP), then the receiver can be assured of the proper sequence because an attacker cannot successfully alter the sequence number.

- The process depicted in Figure a provides authentication but not confidentiality, because the message as a whole is transmitted in the clear.
- Confidentiality can be provided by performing message encryption either after (Figure b) or before (Figure c) the MAC algorithm. In both these cases, two separate keys are needed, each of which is shared by the sender and the receiver.
- In the first case, the MAC is calculated with the message as input and is then concatenated to the message. The entire block is then encrypted.
- In the second case, the message is encrypted first. Then the MAC is calculated using the resulting ciphertext and is concatenated to the ciphertext to form the transmitted block.
- Typically, it is preferable to tie the authentication directly to the plaintext, so the method of Figure b is used.

Hash Function

- A hash value h is generated by a function H of the form

$$h = H(M)$$

where M is a variable-length message and H(M) is the fixed-length hash value.

The hash value is appended to the message at the source at a time when the message is assumed or known to be correct.

The receiver authenticates that message by recomputing the hash value.

Requirements for a Hash Function

- The purpose of a hash function is to produce a "fingerprint" of a file, message, or other block of data. To be useful for message authentication, a hash function H must have the following properties:

1. H can be applied to a block of data of any size.

2. H produces a fixed-length output.

3. H(x) is relatively easy to compute for any given x, making both hardware and software implementations practical.

4. For any given value h, it is computationally infeasible to find x such that H(x) = h. This is sometimes referred to in the literature as the one-way property.

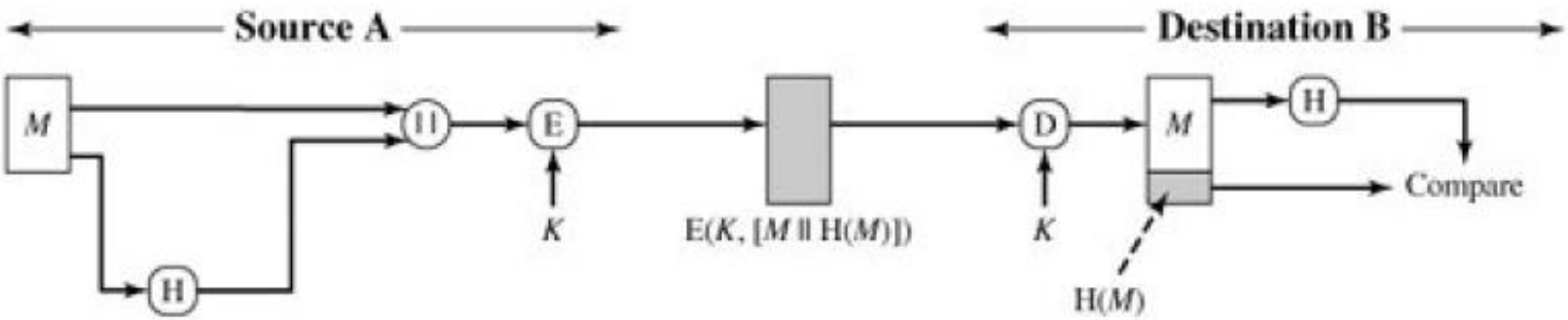
5. For any given block x, it is computationally infeasible to find y x such that H(y) = H(x). This is sometimes referred to as **weak collision resistance.**

6. It is computationally infeasible to find any pair (x, y) such that H(x) = H(y). This is sometimes referred to as **strong collision resistance.**

Cryptographic Hash Function

- A cryptographic hash function is a transformation that takes an input and returns a fixed-size string, which is called the hash value or message digest.
- The hash value is a concise representation of the longer message or document from which it was computed.
- The message digest is a sort of "digital fingerprint" of the larger document.
- Cryptographic hash functions are used to do message integrity checks and digital signatures in various information security applications, such as authentication and message integrity.

- Hash functions are an important type of cryptographic algorithms and are widely used in cryptography such as digital signature, data authentication, e-cash and many other applications.
- Hash functions are at work in the millions of transactions that take place on the internet every day.
- The purpose of the use of hash functions in many cryptographic protocols is to ensure their security as well as improve their efficiency.
- The most widely used hash functions are dedicated hash functions such as MD5 and SHA-1.



(a)

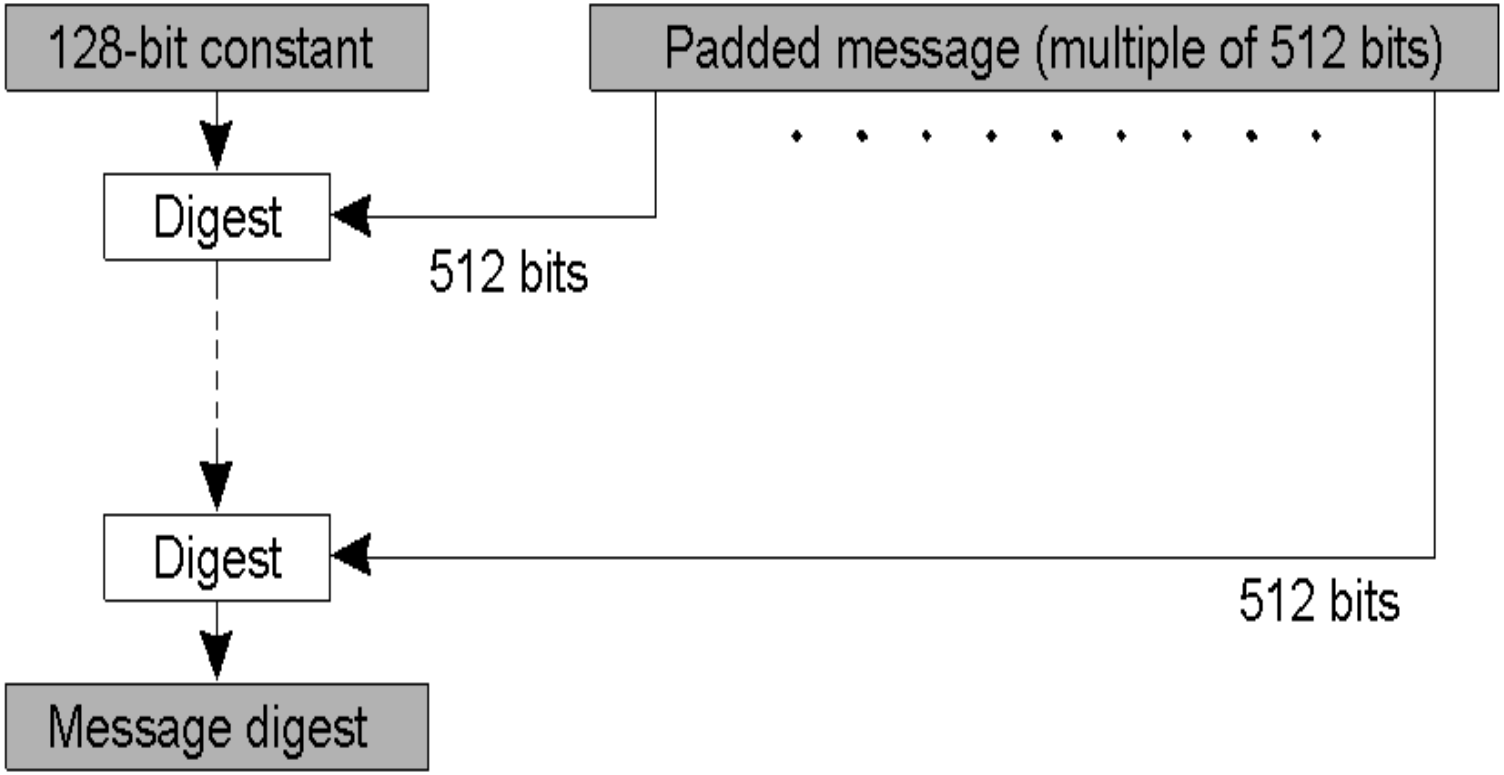
Applications

- **Message Integrity Verification:** Determining whether any changes have been made to a message (or a file), for example, can be accomplished by comparing message digests calculated before, and after, transmission (or any other event).
- **Password Verification:** Passwords are usually not stored in clear text, for obvious reasons, but instead in digest form. To authenticate a user, the password presented by the user is hashed and compared with the stored hash. This is sometimes referred to as one-way encryption.
- **Digital Signatures**: while generating digital signatures, the message digest is created and it is encrypted with the private key so that the signing process becomes faster.

Message Digest 5 (MD5)

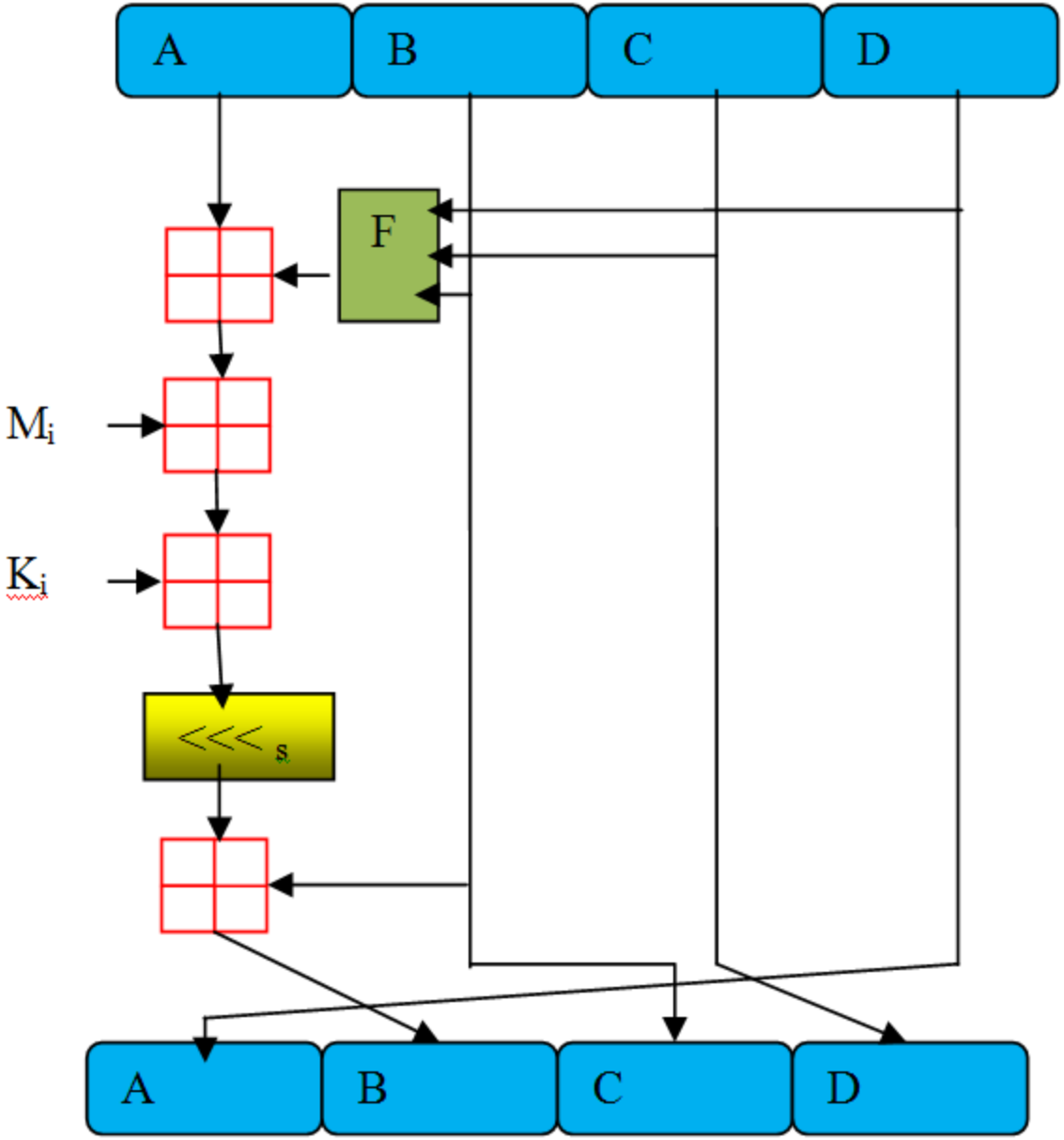
- MD5 algorithm was developed by Professor Ronald L. Rivest in 1991.
- According to RFC 1321, “MD5 message-digest algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input.
- The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA.”

MD5 Algorithm Structure



Original Message	1000……..000.	Original length in bits 64 bits

The original message is padded by adding 1 followed by required number of 0s so that the length of the message is 64 bits less than multiple of 512. The remaining 64 bits is used for providing length of the original message i.e. unpadded message.



One MD5 operation. MD5 consists of 64 of these operations, grouped in four rounds of 16 operations. *F* is a nonlinear function; one function is used in each round. M_i denotes a 32-bit block of the message input, and K_i denotes a 32-bit constant, different for each operation.

$<<<_s$ denotes a left bit rotation by *s* places ; *s* varies for each operation. Red box with plus sign denotes addition modulo 2^{32}

Implementation Steps

Step1 Append padding bits

The input message is "padded" (extended) so that its length (in bits) equals to 448 mod 512. Padding is always performed, even if the length of the message is already 448 mod 512.

Padding is performed as follows: a single "1" bit is appended to the message, and then "0" bits are appended so that the length in bits of the padded message becomes congruent to 448 mod 512. At least one bit and at most 512 bits are appended.

Implementation Steps

Step2. Append length

A 64-bit representation of the length of the message is appended to the result of step1.

The resulting message (after padding with bits and with b) has a length that is an exact multiple of 512 bits. The input message will have a length that is an exact multiple of 16 (32-bit) words.

Implementation Steps

Step3. Initialize MD buffer

A four-word buffer (A, B, C, D) is used to compute the message digest. Each of A, B, C, D is a 32-bit register. These registers are initialized to the following values in hexadecimal, low-order bytes first):

word A: 01 23 45 67

word B: 89 ab cd ef

word C: fe dc ba 98

word D: 76 54 32 10

Implementation Steps

Step4. Process message in 16-word blocks

Four functions will be defined such that each function takes an input of three 32-bit words and produces a 32-bit word output.

F (X, Y, Z) = XY or not (X) Z

G (X, Y, Z) = XZ or Y not (Z)

H (X, Y, Z) = X xor Y xor Z

I (X, Y, Z) = Y xor (X or not (Z))

Implementation Steps

Round 1.

[abcd k s i] denote the operation a = b + ((a + F (b, c, d) + X [k] + T [i]) <<< s).

+ is modulo addition ie. 2^32

Do the following 16 operations.

[ABCD 0 7 1] [DABC 1 12 2] [CDAB 2 17 3] [BCDA 3 22 4]
[ABCD 4 7 5] [DABC 5 12 6] [CDAB 6 17 7] [BCDA 7 22 8]
[ABCD 8 7 9] [DABC 9 12 10] [CDAB 10 17 11] [BCDA 11 22 12]
[ABCD 12 7 13] [DABC 13 12 14] [CDAB 14 17 15] [BCDA 15 22 16]

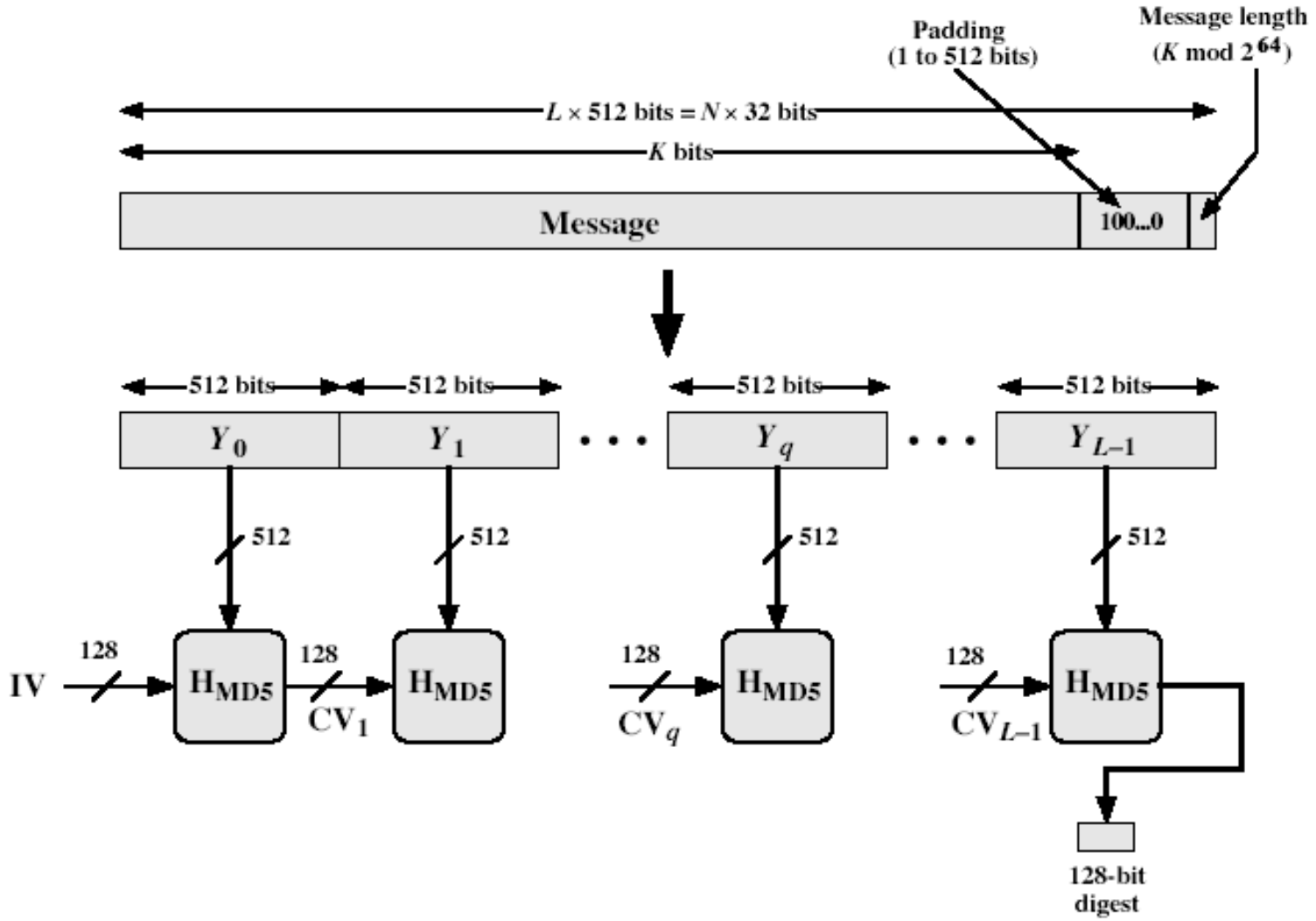
- Comparing to other digest algorithms, MD5 is simple to implement, and provides a "fingerprint" or message digest of a message of arbitrary length.
- It performs very fast on 32-bit machine.
- MD5 is being used heavily from large corporations, such as IBM, Cisco Systems, to individual programmers.
- MD5 is considered one of the most efficient algorithms currently available.

MD5 Overview

1. Pad message so its length is 448 mod 512
2. Append a 64-bit length value to message
3. Initialise 4-word (128-bit) MD buffer (A,B,C,D)
4. Process message in 16-word (512-bit) blocks:

    Using 4 rounds of 16 bit operations on message block & buffer

    Add output to buffer input to form new buffer value
5. Output hash value is the final buffer value


Padding
(1 to 512 bits)
Message length
(K mod 2^{64})
$L \times 512$ bits = $N \times 32$ bits
K bits
Message
100...0
512 bits
512 bits
512 bits
512 bits
Y_0
Y_1
Y_q
Y_{L-1}
512
512
512
512
IV
128
H_{MD5}
128
CV_1
H_{MD5}
128
CV_q
H_{MD5}
128
CV_{L-1}
H_{MD5}
128-bit
digest

Assignment 2

- Write notes on Secure hash Algorithm (SHA1).
- Difference between MD5 and SHA1

End of Chapter 2