4	Find the solutions to each of the following linear equations a)8x≡7(mod19) b). 3x≡4(mod5) c) 8x≡6(mod5) d). 2x≡7(mod23)	[10]
5	a. Using the Euclidean Algorithm find the greatest common divisor of 24140 and 16762. b. Write a short note on transposition cipher and one time pad, explain them with examples.	[5] [5]
6	With the help of neat diagram, explain the AES encryption and decryption process	[10]
7	a. Develop a set of additive and multiplicative tables for modulo-9. b. Find multiplicative inverse of 550in(mod1759).	[5] [5]
8	Illustrate the following with necessary diagrams: (i) Feistel encryption and decryption process. (ii) Single DES encryption.	[10]

CELEBRATING 25 YEARS
CMRIT
CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC

INTERNAL ASSESSMENT TEST – I

Sub:	CRYPTOGRAPHY							Code:	18EC744
Date:	31/ 10 / 2023	Duration:	90 mins	Max Marks:	50	Sem:	VII	Branch:	ECE

Answer any 5 full questions

		Marks	CO	RBT
1	Draw the model of symmetric cryptosystem and explain in detail.	[10]	CO1	L1
2	Encrypt the message "EXAMPOSTPONED" using, play fair cipher with the keyword "BANGLORE"and decrypt the cipher text "QBZKREDJEBCX" to recover the original message. Give the rules forencryption and decryption.	[10]	CO1	L2
3	Encrypt the plaintext "DONOTPAYMONEY"using Hill cipher with key K= [17 17 5 21 18 21 2 2 19]. Show your calculations in obtaining cipher text. (Use A =0, B=1 …Z=25).	[10]	CO1	L2

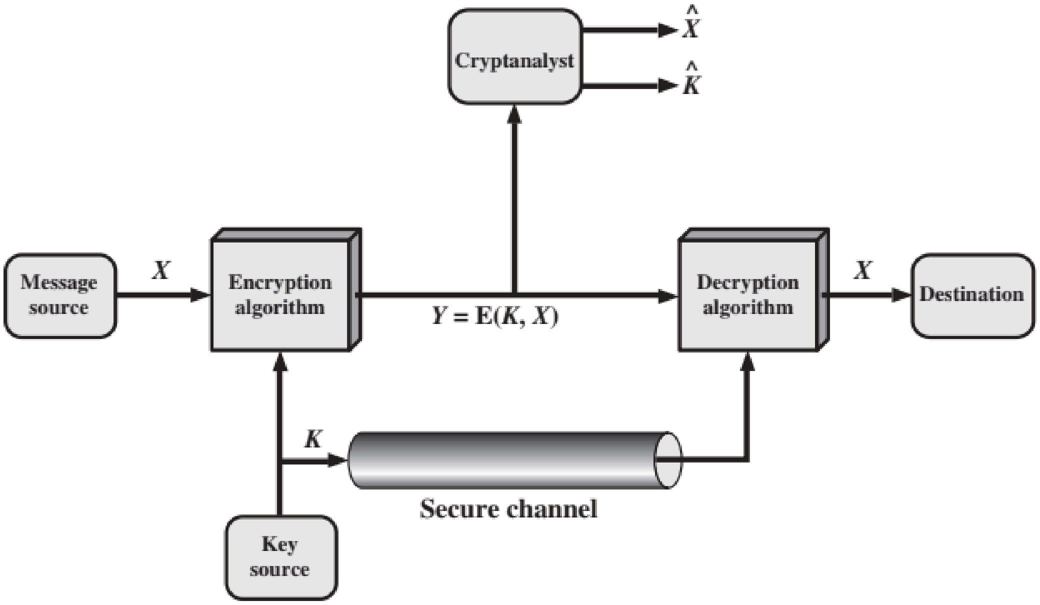
CCI
HOD

CCI
HOD

4	Find the solutions to each of the following linear equations a)8x≡7(mod19) b). 3x≡4(mod5) c) 8x≡6(mod5) d). 2x≡7(mod23)	[10]
5	a. UsingtheEuclideanAlgorithmfindthegreatestcommondivisorof24140and 16762. b. Writeashortnoteontranspositioncipherandonetimepadandexplainwithexamples.	[5] [5]
6	a. Write a note on finite field of form GF(P) b. Prove that if a×c mod n ≡b×c mod n then (a mod n) ≡b (mod n).	[5] [5]
7	a. Develop a set of additive and multiplicative tables for modulo-9. b. Find multiplicative inverse of 550in(mod1759).	[10]
8	Illustrate the following with necessary diagrams: (i) Feistel encryption and decryption process. (ii) Single DES encryption.	[10]

IAT-1 Solutions

1. Draw the model of the symmetric cryptosystem and explain in detail.



Model of Symmetric Cryptosystem

A symmetric encryption scheme has five ingredients:

• Plaintext: This is the original intelligible message or data that is fed into the algorithm as input.

Encryption algorithm: The encryption algorithm performs various substitutions and transformations on the plaintext.

• Secret key: The secret key is also input to the encryption algorithm. The key is a value independent of the plaintext and of the algorithm. The algorithm will produce a different output depending on the specific key being used at the time. The exact substitutions and transformations performed by the algorithm depend on the key.

• Ciphertext: This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different ciphertexts. The ciphertext is an apparently random stream of data and, as it stands, is unintelligible.

• Decryption algorithm: This is essentially the encryption algorithm run in

reverse. It takes the ciphertext and the secret key and produces the original plaintext.
There are two requirements for secure use of conventional encryption:

1. We need a strong encryption algorithm. At a minimum, we would like the algorithm to be such that an opponent who knows the algorithm and has access to one or more ciphertexts would be unable to decipher the ciphertext or figure out the key. This requirement is usually stated in a stronger form: The opponent should be unable to decrypt ciphertext or discover the key even

if he or she is in possession of a number of ciphertexts together with the plaintext that produced each ciphertext.

2. Sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure. If someone can discover the key and knows the algorithm, all communication using this key is readable.

We assume that it is impractical to decrypt a message on the basis of the ciphertext plus knowledge of the encryption/decryption algorithm. In other words, we do not need to keep the algorithm secret; we need to keep only the key secret. This feature of symmetric encryption is what makes it feasible for widespread use. The fact that the algorithm need not be kept secret means that manufacturers can and have developed low-cost chip implementations of data encryption algorithms. These chips are widely available and incorporated into a number of products. With the use of symmetric encryption, the principal security problem is maintaining the secrecy of the key.

Let us take a closer look at the essential elements of a symmetric encryption scheme. A source produces a message in plaintext, $X = [X_1, X_2, \ldots, X_M]$. The M elements of X are letters in some finite alphabet. Traditionally, the alphabet usually consisted of the 26 capital letters. Nowadays, the binary alphabet {0, 1} is typically used. For encryption, a key of the form $K = [K_1, K_2, \ldots, K_J]$ is generated. If the key is generated at the message source, then it must also be provided to the destination by means of some secure channel. Alternatively, a third party could generate the key and securely deliver it to both source and destination.

With the message X and the encryption key K as input, the encryption algorithm forms the ciphertext $Y = [Y_1, Y_2, \ldots, Y_N]$. We can write this as

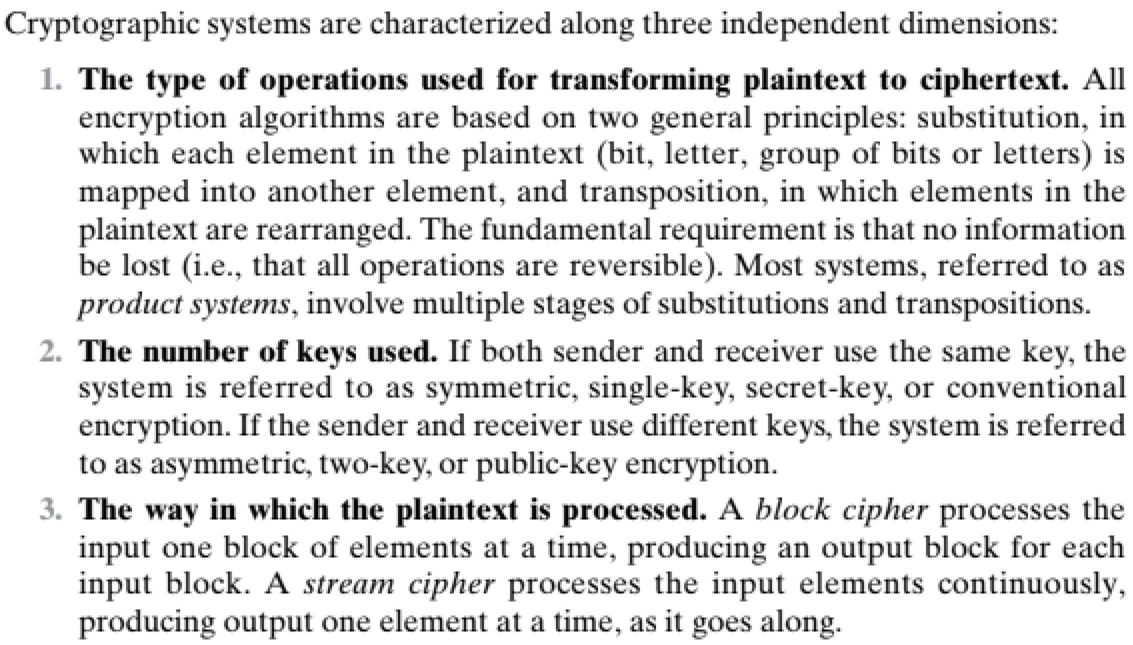
$$Y = \mathrm{E}(K, X)$$

This notation indicates that Y is produced by using encryption algorithm E as a function of the plaintext X, with the specific function determined by the value of the key K.

The intended receiver, in possession of the key, is able to invert the transformation:

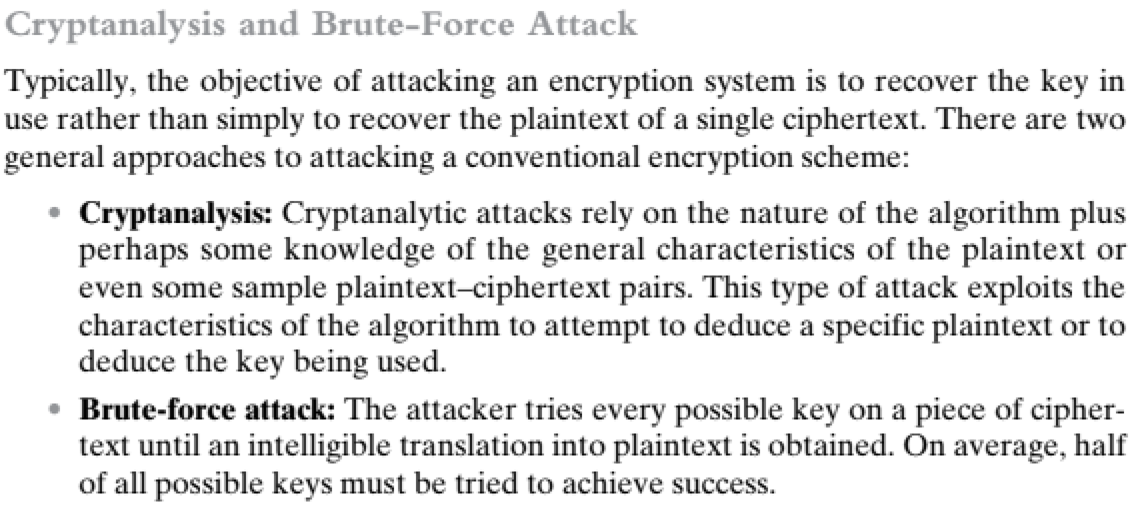
$$X = \mathrm{D}(K, Y)$$

An opponent, observing Y but not having access to K or X, may attempt to recover X or K or both X and K. It is assumed that the opponent knows the encryption (E) and decryption (D) algorithms. If the opponent is interested in only this particular message, then the focus of the effort is to recover X by generating a plaintext estimate $\hat{X}$. Often, however, the opponent is interested in being able to read future messages as well, in which case an attempt is made to recover K by generating an estimate $\hat{K}$.



Cryptographic systems are characterized along three independent dimensions:

1. **The type of operations used for transforming plaintext to ciphertext.** All encryption algorithms are based on two general principles: substitution, in which each element in the plaintext (bit, letter, group of bits or letters) is mapped into another element, and transposition, in which elements in the plaintext are rearranged. The fundamental requirement is that no information be lost (i.e., that all operations are reversible). Most systems, referred to as *product systems*, involve multiple stages of substitutions and transpositions.
2. **The number of keys used.** If both sender and receiver use the same key, the system is referred to as symmetric, single-key, secret-key, or conventional encryption. If the sender and receiver use different keys, the system is referred to as asymmetric, two-key, or public-key encryption.
3. **The way in which the plaintext is processed.** A *block cipher* processes the input one block of elements at a time, producing an output block for each input block. A *stream cipher* processes the input elements continuously, producing output one element at a time, as it goes along.



Cryptanalysis and Brute-Force Attack

Typically, the objective of attacking an encryption system is to recover the key in use rather than simply to recover the plaintext of a single ciphertext. There are two general approaches to attacking a conventional encryption scheme:

- **Cryptanalysis:** Cryptanalytic attacks rely on the nature of the algorithm plus perhaps some knowledge of the general characteristics of the plaintext or even some sample plaintext–ciphertext pairs. This type of attack exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or to deduce the key being used.
- **Brute-force attack:** The attacker tries every possible key on a piece of ciphertext until an intelligible translation into plaintext is obtained. On average, half of all possible keys must be tried to achieve success.

2. **Encrypt the message “EXAMPOSTPONED” using, play fair cipher with the keyword “BANGLORE”and decrypt the ciphertext “QBZKREDJEBCX” to recover the original message. Give the rules for encryption and decryption.**

Marks Distribution: Playfair matrix [2 marks] + Encryption rule [2 marks] + Decryption rule [2 marks] + Encryption [2 marks] + Decryption [2 marks]

B	**A**	**N**	**G**	**L**
O	**R**	**E**	**C**	**D**
F	**H**	**I/J**	**K**	**M**
P	**Q**	**S**	**T**	**U**
V	**W**	**X**	**Y**	**Z**

Plaintext is encrypted two letters at a time.
If a pair is a repeated letter, insert filler like 'X'. Encryption Rule of Play-Fair Cipher:
(1) If both letters fall in the same row, replace each with the letter to its right (circularly).
(2) If both letters fall in the same column, replace each with the letter below it (circularly).
(3) Otherwise, each letter is replaced by the letter in the same row but in the column of the other letter of the pair.
Ciphertext is decrypted two letters at a time. Decryption Rules of Play-Fair Cipher:
(1) Two plaintext letters that fall in the same row of the matrix are each replaced by the letter to the left, with the first element of the row circularly following the last.
(2) Two plaintext letters that fall in the same column are each replaced by the letter above, with the top element of the column circularly following the last.
(3) Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row and the column occupied by the other plaintext letter.

Plain Text: EX, AM, PO, ST, PO, NE, DX
Cipher Text: IN, LH, VF,TU,VF,EI,EZ

Cipher text:QB, ZK, RE, DJ, EB, CX
Plain text: PA,YM,OR, EM,ON,EY

Q3) Encrypt the plaintext “DONOTPAYMONEY”using Hill cipher with key K= [17 17 5, 21 18 21, 2 2 19]**. Show your calculations in obtaining cipher text. (Use A =0, B=1 …Z=25).**

Divide the plain text into blocks of 3 (as here the key is a 3x3 matrix).

DON, OTP, AYM, ONE, YXX
Assign the values:
a=0,b=1,c=2,d=3,e=4,f=5,g=6,h=7,i=8,j=9,k=10,l=11,m=12,n=13,o=14,p= 15, q=16,r=17,s=18,t=19
u=20,v=21,w=22,x=23,y=24,z=25

DON=[3, 14, 13]
OTP=[14, 19, 15]
AYM=[0,24,12]
ONE=[14, 13,4]
YXX=[24,23,23]

Cipher text: Block * Key matrix
[3, 14, 13] * K=[7 17 10]----- hrk
[14, 19, 15]*K=[17 12 0]----[rma]
[0,24,12]*K=[8 14 4]----[ioe]
[14, 13,4]*K=[25 12 3]----[zmd]
[24,23,23]*K=[1 10 0]---[bka]

4) Find the solutions to each of the following linear equations
a)8x≡7(mod19)
x=[inv(8) mod 19* 7 mod 19]= (12 * 7) mod 19 = 8

b). 3x≡4(mod5)
x=inv(3) mod 5 * 4 mod5 = (2 *4)mod 5=3

c) 8x≡6(mod5) — x=2
d). 2x≡7(mod23) x=15

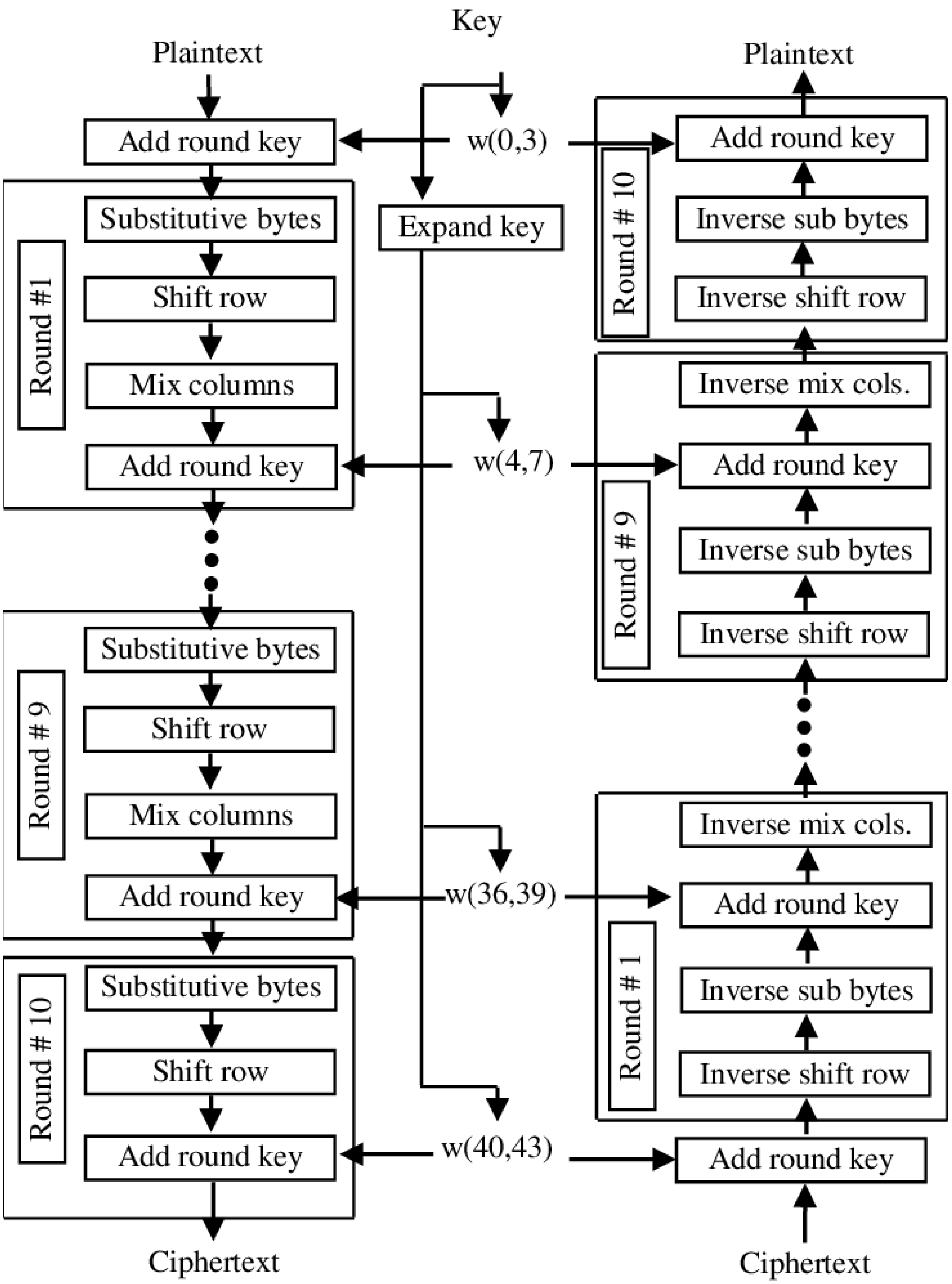
5a) Using the Euclidean Algorithm find the greatest common divisor of 24140 and 16762.

Dividend	**Divisor**	**Remainder**	**Quotient**
24140	**16762**	**7378**	**1**
16762	**7378**	**2006**	**2**
7378	**2006**	**1360**	**3**
2006	**1360**	**646**	**1**
1360	**646**	**68**	
646	**68**	**34 (GCD)**	
68	**34**	**0**	

We know, gcd(a, b) = gcd(b, a mod b) gcd(24140,16762) =gcd(16762,7378) gcd(7378,2006) =gcd(2006,1360) gcd(1360,646) =gcd(646,68) gcd(68,34) = 34 gcd(24140,16762) = 34.

5b) Write a short note on transposition cipher and one time pad, explain them with examples.

6) With the help of neat diagram, explain the AES encryption and decryption process


Key
Plaintext
Add round key
Round #1
Substitutive bytes
Shift row
Mix columns
Add round key
Round # 9
Substitutive bytes
Shift row
Mix columns
Add round key
Round # 10
Substitutive bytes
Shift row
Add round key
Ciphertext
Expand key
w(0,3)
w(4,7)
w(36,39)
w(40,43)
Plaintext
Round # 10
Add round key
Inverse sub bytes
Inverse shift row
Round # 9
Inverse mix cols.
Add round key
Inverse sub bytes
Inverse shift row
Round # 1
Inverse mix cols.
Add round key
Inverse sub bytes
Inverse shift row
Add round key
Ciphertext

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). It is found at least six time faster than triple DES.

A replacement for DES was needed as its key size was too small. With increasing computing power, it was considered vulnerable against exhaustive key search attack. Triple DES was designed to overcome this drawback but it was found slow.

The features of AES are as follows −

Symmetric key symmetric block cipher
128-bit data, 128/192/256-bit keys
Stronger and faster than Triple-DES
Provide full specification and design details

Operation of AES

AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix −

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

Encryption Process

Here, we restrict to description of a typical round of AES encryption. Each round comprise of four sub-processes.

Byte Substitution (SubBytes)

The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

Shiftrows

Each of the four rows of the matrix is shifted to the left. Any entries that ‘fall off’ are re-inserted on the right side of row. Shift is carried out as follows −

First row is not shifted.
Second row is shifted one (byte) position to the left.
Third row is shifted two positions to the left.
Fourth row is shifted three positions to the left.
The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

MixColumns

Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

Addroundkey

The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

Decryption Process

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order −

Add round key
Mix columns
Shift rows
Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms needs to be separately implemented, although they are very closely related.

AES Analysis

In present day cryptography, AES is widely adopted and supported in both hardware and software. Till date, no practical cryptanalytic attacks against AES has been discovered.

Additionally, AES has built-in flexibility of key length, which allows a degree of ‘future-proofing’ against progress in the ability to perform exhaustive key searches.

However, just as for DES, the AES security is assured only if it is correctly implemented and good key management is employed.

7a. Develop a set of additive and multiplicative tables for modulo-9.

Additive Table for Modulo 9

+	0	1	2	3	4	5	6	7	8
0	0	1	2	3	4	5	6	7	8
1	1	2	3	4	5	6	7	8	0
2	2	3	4	5	6	7	8	0	1
3	3	4	5	6	7	8	0	1	2
4	4	5	6	7	8	0	1	2	3
5	5	6	7	8	0	1	2	3	4
6	6	7	8	0	1	2	3	4	5
7	7	8	0	1	2	3	4	5	6
8	8	0	1	2	3	4	5	6	7

Multiplicative Table for Modulo 9

*	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	1	3	5	7
3	0	3	6	0	3	6	0	2	5

4	**0**	**4**	**8**	**3**	**7**	**0**	**5**	**2**	**5**
5	**0**	**5**	**1**	**6**	**0**	**6**	**2**	**8**	**4**
6	**0**	**6**	**3**	**0**	**5**	**3**	**0**	**6**	**3**
7	**0**	**7**	**5**	**2**	**1**	**8**	**4**	**4**	**2**
8	**0**	**8**	**7**	**5**	**5**	**4**	**3**	**2**	**1**

7b. Find multiplicative inverse of 550 in (mod 1759).

Xi=X(i-2) - Qi X(i-1), Yi=Y(i-2) - Qi Y(i-1)

Dividend	Divisor	Ri	Qi	Xi	Yi
1759	550	-	-	1	0
1759	550	-	-	0	1
1759	550	109	3	1	-3
550	109	5	5	-5	16
109	5	4	21	106	-339
5	4	1	1	-111	355
4	1	0	4		

Inverse of 550 in mod 1759=355

Verify: (550*355) mod 1759=1

8) Illustrate the following with necessary diagrams:
(i) Feistel encryption and decryption process.

(i) FEISTEL CIPHER STRUCTURE:

1. The inputs to the encryption algorithm are a plaintext block of length 2w bits and a key K.

2. The plaintext block is divided into two halves, L0 and R0.

3. The two halves of the data pass through n rounds of processing and then combine to produce the ciphertext block.

4. Each round i has as inputs $Li-1$ *and* $Ri-1$ derived from the previous round, as well as a subkey Ki derived from the overall K. The subkeys Ki are different from K and from each other.

5. 16 rounds are used, although any number of rounds could be implemented. All rounds have the same structure.

6. A substitution is performed on the left half of the data. This is done by applying a round function F to the right half of the data and then taking the exclusive-OR of the output of that function and the left half of the data.

7. The round function F has the same general structure for each round. The round function F is represented as $F(REi ,Ki+1)$

8. Following this substitution, a permutation is performed that consists of the interchange of the two halves of the data.

9. Feistel network depends on the choice of the following parameters and design features:

a) Block size: larger block sizes mean greater security, but it reduces encryption/decryption speed for a given algorithm. The greater security is achieved by greater diffusion. Traditionally, a block size of 64 bits has been considered a reasonable tradeoff and was nearly universal in block cipher design. However, the new AES uses a 128-bit block size.

b) Key size: Larger key size means greater security but may decrease encryption decryption speed. The greater security is achieved by greater resistance to brute-force attacks and greater confusion. Key sizes of 64 bits or less are now widely considered being inadequate and 128 bits has become a common size.

c) Number of rounds: The essence of the Feistel cipher is that a single round offers inadequate security but that multiple rounds offer increasing security. A typical size is 16 rounds.

d) Subkey generation algorithm: Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.

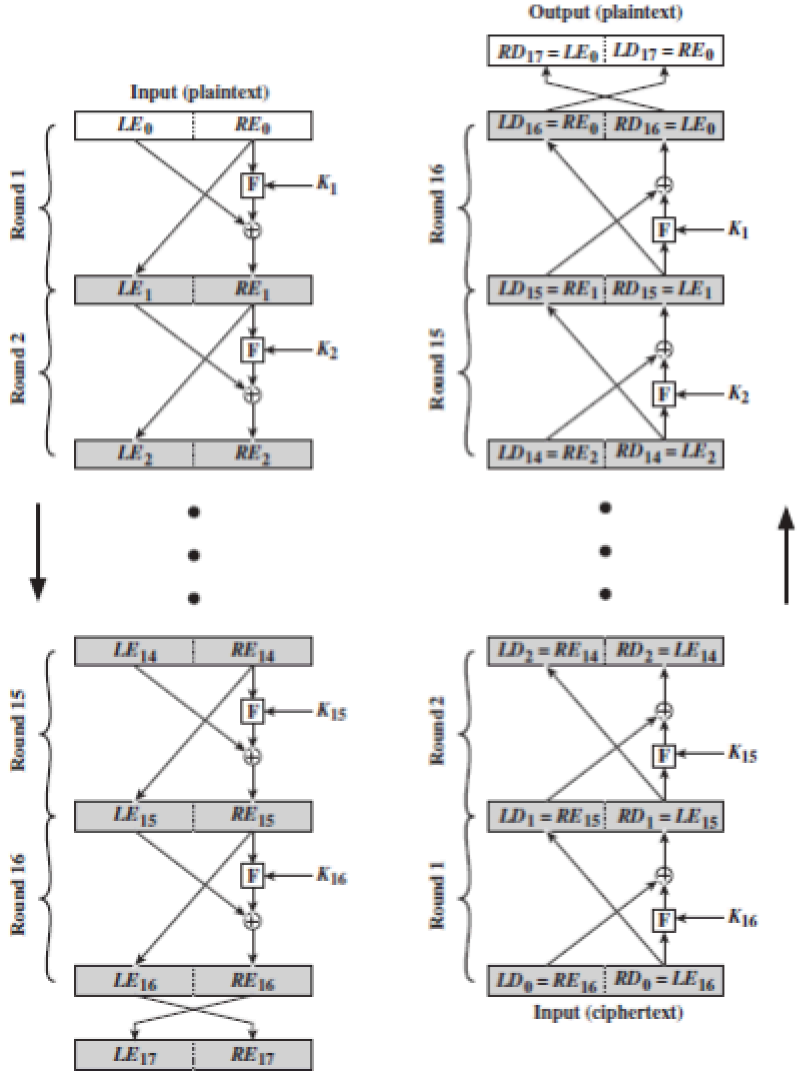
e) Round function F: Again, greater complexity generally means greater resistance to cryptanalysis.

10. There are two other considerations in the design of a Feistel cipher:

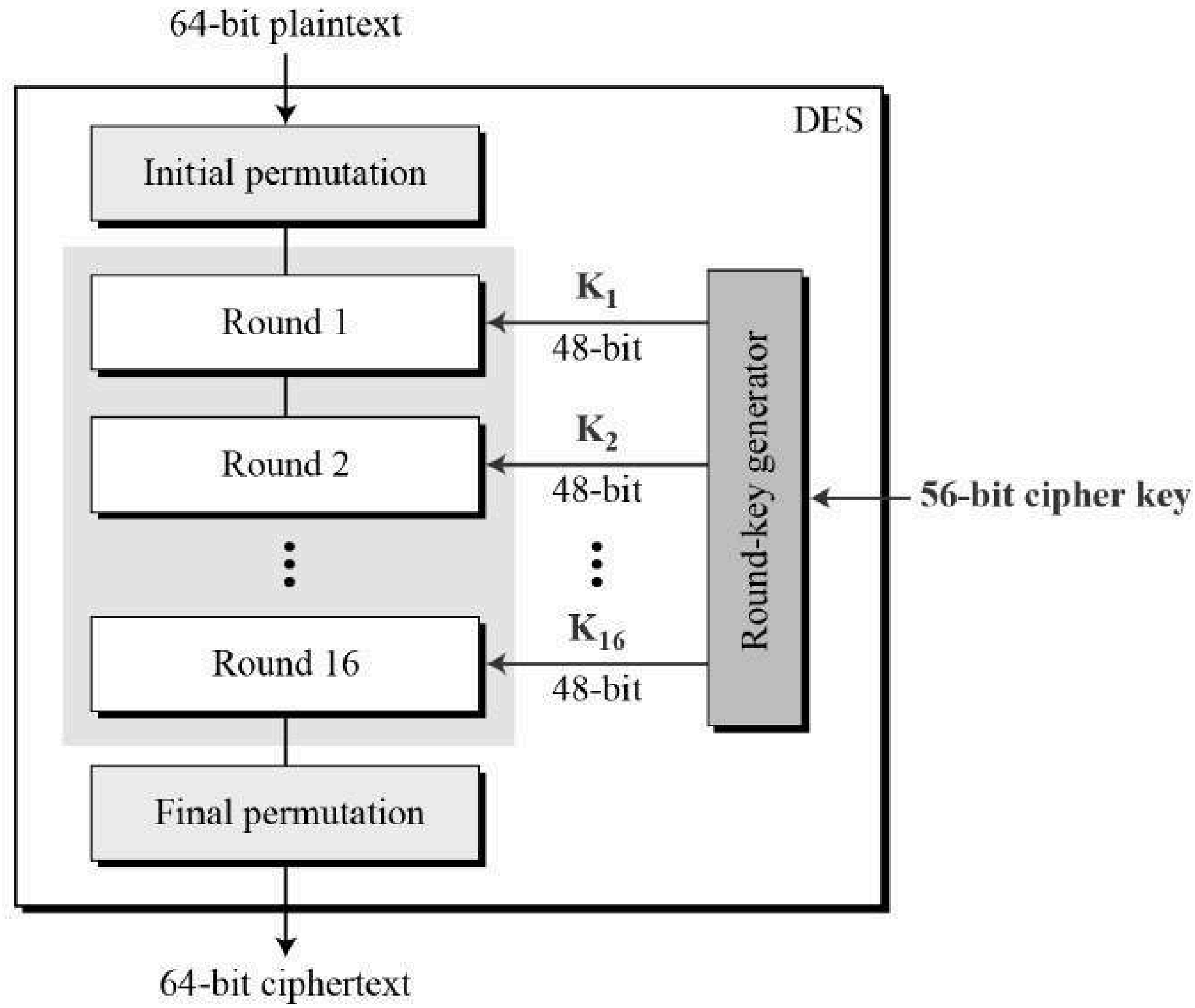
a) Fast software encryption/decryption: Encryption is embedded in applications hence the speed of execution of the algorithm becomes a concern.

b) Ease of analysis: Although we would like to make our algorithm as difficult as possible

To cryptanalyze, there is great benefit in making the algorithm easy to analyze. That is, if the algorithm can be concisely and clearly explained, it is easier to analyze that algorithm for cryptanalytic vulnerabilities and therefore develop a higher level of assurance as to its strength. DES, for example, does not have an easily analyzed functionality.



(ii) Single DES encryption.



64 bit key is used but every 8th bit is the parity bit hence it is taken as 56 bit key. Initially the key is passed through the permutation function. For each 16 round, a sub key Ki is produced by the combination of left circular shift and permutation. The same permutation function is used in each round.
The plain text are processed through these phases
a) Initial Permutation
b) 16 rounds of same function
c) Swap
d) Final Permutation

Initial Permutation and Final Permutation:
The input is 64 bit. These inputs are permuted according to a predefined rule. The permutation table contains a permutation of the number from 1 to 64.

DES Encryption:

a) In DES Encryption, there are two inputs to the encryption function:

i. the plaintext to be encrypted
ii. Key
b) In this case, the plaintext must be 64 bits in length and the key is 56 bits in length.
c) The processing of the plaintext proceeds in three phases.
i. First, the 64-bit plaintext passes through an initial permutation (IP) that rearranges the bits to produce the permuted input.
ii. This is followed by a phase consisting of sixteen rounds of the same function, which involves both permutation and substitution functions.
iii. The left and right halves of the output are swapped to produce the pre output.
iv. Finally, the pre-output is passed through a permutation [IP−1] that is the inverse of the initial permutation function, to produce the 64-bit ciphertext.

d) With the exception of the initial and final permutations, DES has the exact structure of a Feistel cipher.

Key Generation:

a) In DES, 56-bit key is used.
b) Initially, the key is passed through a permutation function.
a) Then, for each of the sixteen rounds, a subkey (Ki) is produced by the combination

of a left circular shift and a permutation.
b) The permutation function is the same for each round, but a different subkey is produced because of the repeated shifts of the key bits.

DES Decryption:

a) As with any Feistel cipher, decryption uses the same algorithm as encryption, except that the application of the subkeys is reversed.
b) Additionally, the initial and final permutations are reversed.