UNIT-I

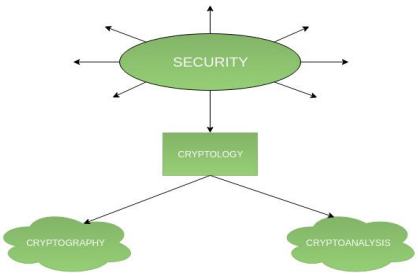
➢ Cryptography

Cryptography is the study of secure communications techniques that allow only the sender and intended recipient of a message to view its contents. The term is derived from the Greek word *kryptos*, which means hidden. It is closely associated to encryption, which is the act of scrambling ordinary text into what's known as ciphertext and then back again upon arrival. In addition, cryptography also covers the obfuscation of information in images using techniques such as microdots or merging. Ancient Egyptians were known to use these methods in complex hieroglyphics, and Roman Emperor Julius Caesar is credited with using one of the first modern ciphers.

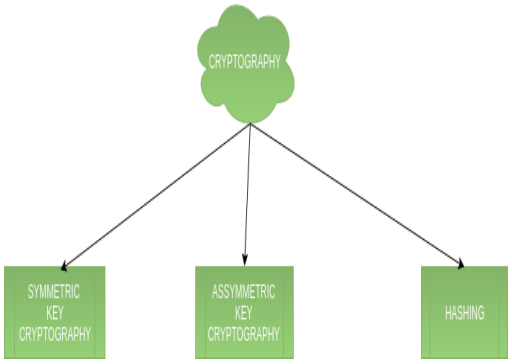
When transmitting electronic data, the most common use of cryptography is to encrypt and decrypt email and other plain-text messages. The simplest method uses the symmetric or "secret key" system. Here, data is encrypted using a secret key, and then both the encoded message and secret key are sent to the recipient for decryption. The problem? If the message is intercepted, a third party has everything they need to decrypt and read the message. To address this issue, cryptologists devised the asymmetric or "public key" system. In this case, every user has two keys: one public and one private. Senders request the public key of their intended recipient, encrypt the message and send it along. When the message arrives, only the recipient's private key will decode it — meaning theft is of no use without the corresponding private key.

➢ Introduction to Crypto-terminologies

Cryptography is an important aspect when we deal with network security. 'Crypto' means secret or hidden. Cryptography is the science of secret writing with the intention of keeping the data secret. Cryptanalysis, on the other hand, is the science or sometimes the art of breaking cryptosystems. These both terms are a subset of what is called as Cryptology.



Classification –
The flowchart depicts that cryptology is only one of the factors involved in securing networks. Cryptology refers to study of codes, which involves both writing (cryptography) and solving (cryptanalysis) them. Below is a classification of the crypto-terminologies and their various types.



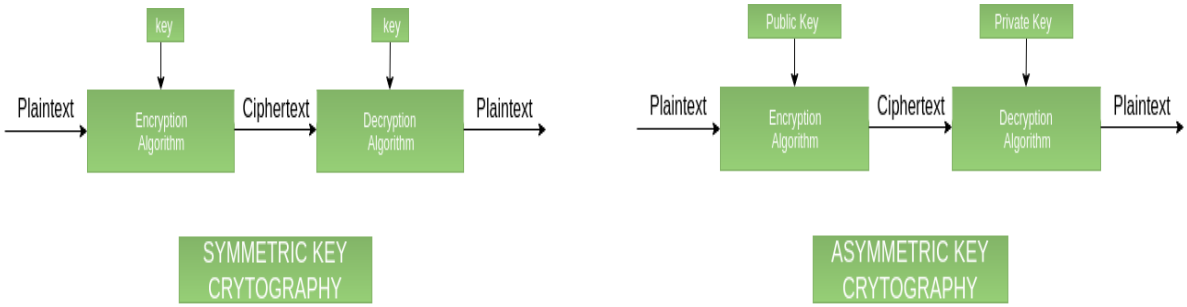
1. Cryptography –
Cryptography is classified into symmetric cryptography, asymmetric cryptography and hashing. Below are the description of these types.

1. Symmetric key cryptography

It involves usage of one secret key along with encryption and decryption algorithms which help in securing the contents of the message. The strength of symmetric key cryptography depends upon the number of key bits. It is relatively faster than asymmetric key cryptography. There arises a key distribution problem as the key has to be transferred from the sender to receiver through a secure channel.

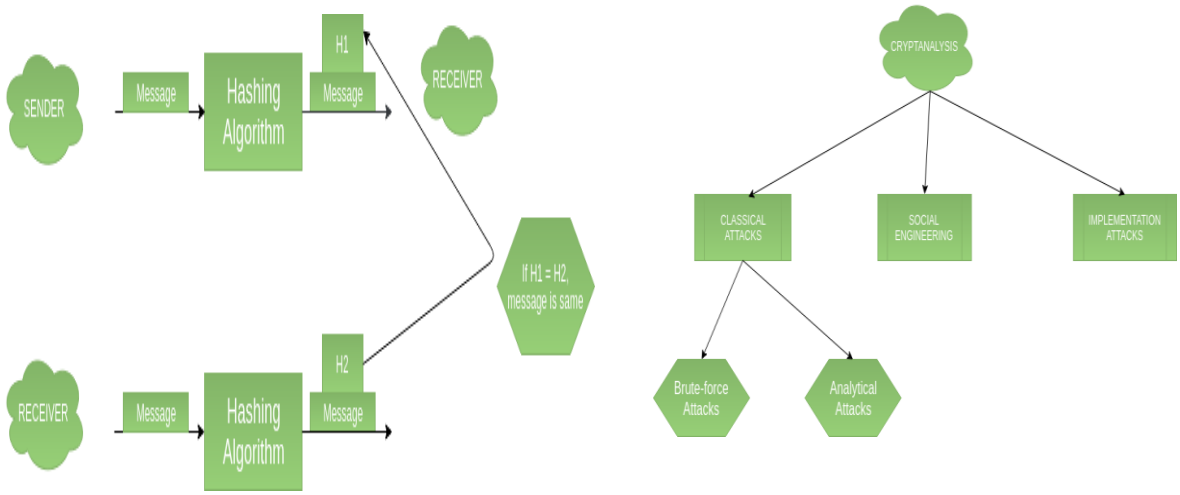
2. Asymmetric key cryptography

It is also known as public key cryptography because it involves usage of a public key along with secret key. It solves the problem of key distribution as both parties uses different keys for encryption/decryption. It is not feasible to use for decrypting bulk messages as it is very slow compared to symmetric key cryptography.



3.Hashing –

It involves taking the plain-text and converting it to a hash value of fixed size by a hash function. This process ensures integrity of the message as the hash value on both, sender\'s and receiver\'s side should match if the message is unaltered.



2. Cryptanalysis –

1. Classical attacks

It can be divided into a)Mathematical analysis and b) Brute-force attacks. Brute-force attacks run the encryption algorithm for all possible cases of the keys until a match is found. Encryption algorithm is treated as a black box. Analytical attacks are those attacks which focus on breaking the cryptosystem by analysing the internal structure of the encryption algorithm.

2. Social Engineering attack

It is something which is dependent on the human factor. Tricking someone to reveal their passwords to the attacker or allowing access to the restricted area comes under this attack. People should be cautious when revealing their passwords to any third party which is not trusted.

3.Implementation attacks

Implementation attacks such as side-channel analysis can be used to obtain a secret key. They are relevant in cases where the attacker can obtain physical access to the cryptosystem.

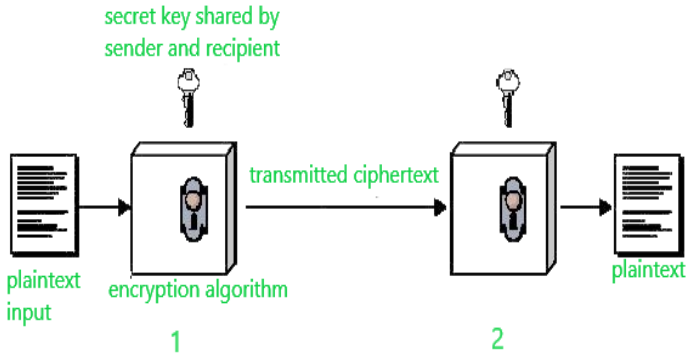
➢Conventional Encryption Model

Conventional encryption is a cryptographic system that uses the same key used by the sender to encrypt the message and by the receiver to decrypt the message. It was the only type of encryption in use prior to the development of public-key encryption.

It is still much preferred of the two types of encryption systems due to its simplicity. It is a relatively fast process since it uses a single key for both encryption and decryption In this encryption model, the sender encrypts plaintext using the receiver's secret key, which can be later used by the receiver to decrypt the ciphertext. Below is a figure that illustrates this concept.

Suppose A wants to send a message to B, that message is called plaintext. Now, to avoid hackers reading plaintext, the plaintext is encrypted using an algorithm and a secret key (at 1). This encrypted plaintext is called ciphertext. Using the same secret key and encryption algorithm run in reverse(at 2), B can get plaintext of A, and thus the message is read and security is maintained.

The idea that uses in this technique is very old and that's why this model is called conventional encryption.



Conventional encryption has mainly 5 ingredients :

Plain text –

It is the original data that is given to the algorithm as an input.

Encryption algorithm –

This encryption algorithm performs various transformations on plain text to convert it into ciphertext.

Secret key –

The secret key is also an input to the algorithm. The encryption algorithm will produce different outputs based on the keys used at that time.

Ciphertext –

It contains encrypted information because it contains a form of original plaintext that is unreadable by a human or computer without proper cipher to decrypt it. It is output from the algorithm.

Decryption algorithm –

This is used to run encryption algorithms in reverse. Ciphertext and Secret key is input here and it produces plain text as output.

Requirements for secure use of conventional encryption:

1.We need a strong encryption algorithm.

2. The sender and Receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure.

Advantages of Conventional Encryption :

1.Simple –

This type of encryption is easy to carry out.

2.Uses fewer computer resources –

Conventional encryption does not require a lot of computer resources when compared to public-key encryption.

3.Fast –

Conventional encryption is much faster than asymmetric key encryption.

Disadvantages of Conventional Encryption Model:

1.Origin and authenticity of the message cannot be guaranteed, since both sender and receiver use the same key, messages cannot be verified to have come from a particular user.

2.It isn't much secured when compared to public-key encryption.

3.If the receiver lost the key, he/she cant decrypt the message and thus making the whole process useless.

4.This scheme does not scale well to a large number of users because both the sender and the receiver have to agree on a secret key before transmission.

➢Steganography

Steganography is a technique of hiding communication by concealing the secret message into a fake message. The term Steganography has a Greek influence which means "covered writing". The main idea behind the Steganography is to prevent suspicion about the existence of the information.

- Pure Steganography does not require the exchange of a cipher such as a stego-key. It assumes that no other party is aware of the communication.
- Secret key Steganography where the secret (stego) key is exchanged prior to communication. This is most susceptible to interception. Secret Key Steganography takes a cover message and embeds the secret message inside of it by using a secret key (stego-key). Only the parties who know the secret key can reverse the process and read the secret message.
- Public key Steganography where a public key and a private key is used for secure Communication. The sender will use the public key during the encoding process and only the private key, which has a direct mathematical relationship with the public key, can decipher the secret message.

Types of Steganography

Image Steganography

- The image Steganography is used to hide a secret message inside an image. The most widely used technique to hide secret bit inside the LSB of the cover image. Because this method uses bits of each pixel in the image, it is necessary to use a lossless compression format, otherwise the hidden information will get lost in the transformations of a lossy compression algorithm.
- When using a 24 bit color image, a bit of each of the red, green and blue color components can be used, so a total of 3 bits can be used for each pixel, in this way we can use more secret bit to hide data in it.

Audio Steganography

- Audio stenography can conceal the secret message in the audio file with the help of its digital representation. It can be achieved easily as a typical 16-bit file has 216 sound levels, and a few levels difference could not be detectable by the human ear.
- The sender embeds secret data of any type using a key in a digital cover file to produce a stego file, in such a way that an observer cannot detect the existence of the hidden message. In many schemes a method of audio Steganography based on modification of least significant bits (LSB) the audio samples in the temporal domain or transform domain have been proposed.

Video Steganography

- Video Steganography brings more possibilities of disguising a large amount of data because it is a combination of image and sound. Therefore, image and audio Steganography techniques can also be employed on the video.
- Video files are generally a collection of images and sounds, so most of the presented techniques on images and - audio can be applied to video files too.
- The great advantage of video is the large amount of data that can be hidden inside and the fact that it is a moving stream of images and sounds.
- The Video Steganography is nothing but a combination of Image Steganography and Audio Steganography.

Text Steganography:

- Steganography can be applied to different types of media including text, audio, image and video etc. However, text Steganography is considered to be the most difficult kind of Steganography due to lack of redundancy in text as compared to image or audio but still has smaller memory occupation and simpler communication.
- The method that could be used for text Steganography is data compression. Data compression encodes information in one representation into another representation. The new representation of data is smaller in size.
- One of the possible schemes to achieve data compression is Huffman coding. Huffman coding assigns smaller length code words to more frequently occurring source symbols and longer length code-words to less frequently occurring source symbols.

➢ Classical Encryption Techniques: A SYMMETRIC CIPHER MODEL:

Symmetric encryption, also referred to as conventional encryption or single-key encryption, was the only type of encryption in use prior to the development of public key encryption in the1970s.

Some basic terminologies used:

- ciphertext - the coded message
- cipher - algorithm for transforming plaintext to ciphertext
- key - info used in cipher known only to sender/receiver
- encipher (encrypt) - converting plaintext to ciphertext
- decipher (decrypt) - recovering ciphertext from plaintext
- cryptography - study of encryption principles/methods
- cryptanalysis (code breaking) - the study of principles/ methods of deciphering ciphertext without knowing key
- cryptology - the field of both cryptography and cryptanalysis

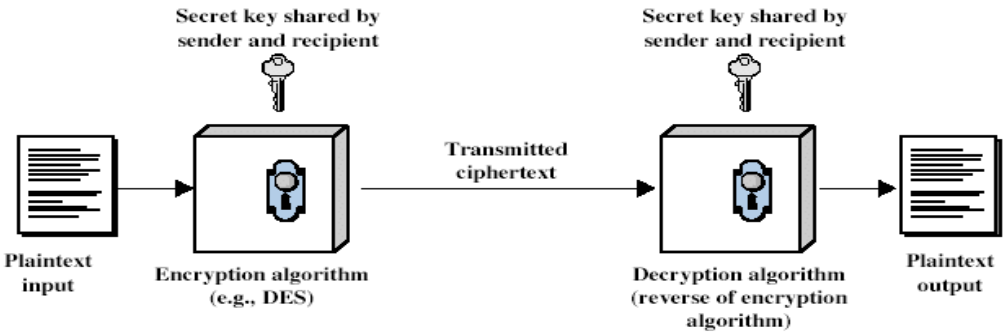


Fig.7 Simplified Model of Symmetric Encryption

A symmetric encryption scheme has five ingredients

A symmetric encryption scheme has five ingredients (Fig.7). Here the original message, referred to as plaintext, is converted into apparently random nonsense, referred to as cipher text. The encryption process consists of an algorithm and a key.

The key is a value independent of the plaintext. Changing the key changes, the output of the algorithm. Once the cipher text is produced, it may be transmitted. Upon reception, the cipher text can be transformed back to the original plaintext by using a decryption algorithm and the same key that was used for encryption.

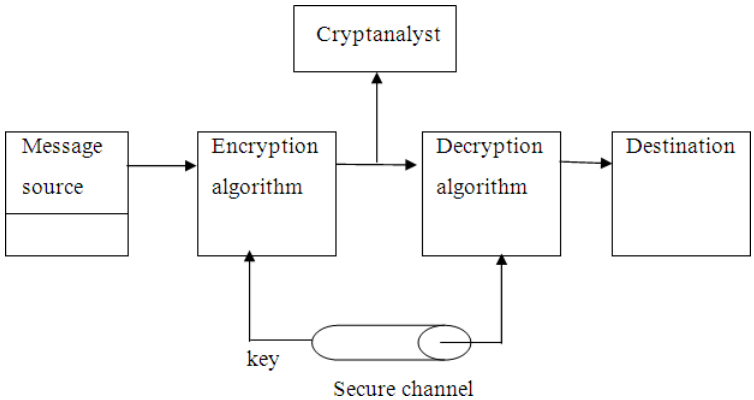
The security depends on several factors. First, the encryption algorithm must be powerful enough that it is impractical to decrypt a message on the basis of cipher text alone. Beyond that, the security depends on the secrecy of the key, not the secrecy of the algorithm.

Two requirements for secure use of symmetric encryption:

- A strong encryption algorithm
- A secret key known only to sender / receiver
- Y = EK(X)
- X = DK(Y)

assume encryption algorithm is known implies a secure channel to distribute key

Fig.8. conventional cryptosystem



A source produces a message in plaintext, X = [X1, X2, …, XM] where M are the number of letters in the message. A key of the form K = [K1, K2, …, KJ] is generated. If the key is generated at the source, then it must be provided to the destination by means of some secure channel. With the message X and the encryption key K as input, the encryption algorithm forms

the cipher text Y = [Y1, Y2, …, YN]. This can be expressed as Y = EK(X)

The intended receiver, in possession of the key, is able to invert the transformation: X = DK(Y) An opponent, observing Y but not having access to K or X, may attempt to recover X or K or both. It is assumed that the opponent knows the encryption and decryption algorithms. If the opponent is interested in only this particular message, then the focus of effort is to recover X by generating a plaintext estimate. Often if 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 anestimate.

Substitution Encryption Techniques:

Substitution encryption technique is one type of classic encryption technique, A substitution technique is one in which the letters of plaintext are replaced by other letters or by numbers or symbols. If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns.

- (i)Caesar cipher (or) shift cipher
- The earliest known use of a substitution cipher and the simplest was by Julius Caesar.
- The Caesar Cipher is a type of shift cipher. Shift Ciphers work by using the modulo operator to encrypt and decrypt messages. The Shift Cipher has a key K, which is an integer from 0 to 25. We will only share this key with people that we want to see our message
- The Caesar cipher involves replacing each letter of the alphabet with the letter standing 3 places further down the alphabet.
- e.g., Plain text: pay more mone Cipher text: SDB PRUH PRQHB
- Note that the alphabet is wrapped around, so that letter following „z" is „a".
- Note that the alphabet is wrapped around, so that the letter following Z isA.
- We can define the transformation by listing all possibilities, as follows: plain: 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 cipher: 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
- Let us assign a numerical equivalent to each letter:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12

n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

For Encrypt each plaintext letter p, substitute the cipher text letter c such that C = E(p) = (p+3) mod 26,

a shift may be any amount, so that general Caesar algorithm isC = E (p) = (p+k) mod 26

where k takes on a value in the range 1 to 25.

The decryption algorithm is simply P = D(C) = (C-k) mod 26 (or) to Encrypt a message M. Convert the letter into the number that matches its order in the alphabet starting from 0, and call this number X, (A=0, B=1, C=2, ..., Y=24, Z=25).

Calcúlate: Y = (X + K) mod 26

Convert the number Y into a letter that matches its order in the alphabet starting from 0.Example:

By using the Shift Cipher with key K=19 for our message. We encrypt the message "KHAN", as follows

ENCRYPTION

	K	H	A	N	
	10	7	0	13	
+	19	19	19	19	
(	29	26	19	32	) mod 26
	3	0	19	6	
	D	A	T	G	

So, after applying the Shift Cipher with key K=19 our message text "KHAN" gaveus cipher text "DATG".

- For every letter in the cipher text C, convert the letter into the number that matches itsorder in the alphabet starting from 0, and call this number Y.
- If it is known that a given ciphertext is a Caesar cipher, then a brute-force cryptanalysis is easily performed: Simply try all the 25 possible keys.

Monoalphabetic Ciphers:

With only 25 possible keys, the Caesar cipher is far from secure. A dramatic increase in the key space can be achieved by allowing an arbitrary substitution. Before proceeding, the term permutation can be defined.

A permutation of a finite set of elements S is an ordered sequence of all the elements of S, with each element appearing exactly once.

For example, if S = {a, b, c}, there are six permutations of S:abc, acb, bac, bca, cab, cba

In general, there are n! permutations of a set of n elements, because the first element can be chosen in one of n ways, the second in n - 1 ways, the third in n – 2 ways, and so on.

plain: 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 Caesar cipher: 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

If, instead, the “cipher” line can be any permutation of the 26 alphabetic characters, then there are 26! or greater than 4 * 1026 possible keys.

This is 10 orders of magnitude greater than the key space for DES and would seem to eliminate brute-force techniques for cryptanalysis. Such an approach is referred to as a mono alphabetic substitution cipher, because a single cipher alphabet (mapping from plain alphabet to cipher alphabet) is used per message.

Monoalphabetic ciphers are easy to break because they reflect the frequency data of the original alphabet.

A countermeasure is to provide multiple substitutes known as homophones, for a single letter. For example, the letter e could be assigned a number of different cipher symbols, such as 16, 74, 35, and 21, with each homophone assigned to a letter in rotation or randomly.

Playfair Cipher:

The best-known multiple-letter encryption cipher is the Playfair, which treats diagrams in the plaintext as single units and translates these units into cipher text diagrams

he Playfair algorithm is based on the use of a 5 * 5 matrix of letters constructed using a keyword. Here is an example, solved by Lord Peter Wimsey in Dorothy Sayers‟s Have HisCarcase

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

In this case, the keyword is monarchy. The matrix is constructed by filling in the letters of the keyword (minus duplicates) from left to right and from top to bottom, and then filling in the remainder of the matrix with the remaining letters in alphabetic order. The letters I and J count as one letter.

Plaintext is encrypted two letters at a time, according to the following rules:

Repeating plaintext letters that are in the same pair are separated with a filler letter, such as x, so that balloon would be treated as ba lx lo on.

Two plaintext letters that fall in the same row of the matrix are each replaced by the letter to the right, with the first element of the row circularly following the last. For example, ar is encrypted as RM.

Two plaintext letters that fall in the same column are each replaced by the letter beneath, with the top element of the column circularly following the last. For example, mu is encrypted as CM.

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. Thus, hs becomes BP and ea becomes IM

The Playfair cipher is a great advance over simple monoalphabetic ciphers. For one thing, whereas there are only 26 letters, there are 26 * 26 = 676 digrams, so that identification of individual digrams is more difficult. Furthermore, the relative frequencies of individual letters exhibit a much greater range than that of digrams, making frequency analysis muchmore difficult.

For these reasons, the Playfair cipher was for a long time considered unbreakable. It was used as the standard field system by the British Army in World War I and still enjoyed considerable use by the U.S. Army and other Allied forces during World War II.

Hill Cipher:

Another interesting multiletter cipher is the Hill cipher, developed by the mathematician Lester Hill in 1929.

The Hill Algorithm

This encryption algorithm takes m successive plaintext letters and substitutes for them m ciphertext letters. The substitution is determined by m linear equations in which each character is assigned a numerical value (a = 0, b = 1,….., z = 25). For m = 3, the system canbe described as

$c_1 = (k_{11}p_1 + k_{21}p_2 + k_{31}p_3) \bmod 26$ $c_2 = (k_{12}p_1 + k_{22}p_2 + k_{32}p_3) \bmod 26$ $c_3 = (k_{13}p_1 + k_{23}p_2 + k_{33}p_3) \bmod 26$

This can be expressed in terms of row vectors and matrices:

$$(c_1 c_2 c_3) = (p_1 p_2 p_3) \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \bmod 26$$

or

$$C = PK \bmod 26$$

Where C and P are row vectors of length 3 representing the plaintext and ciphertext, and K is a 3 x 3 matrix representing the encryption key. Operations are performed mod 26.

Polyalphabetic ciphers

A polyalphabetic cipher is any cipher based on substitution, using multiple substitution alphabets. The Vigenère cipher is probably the best-known example of a polyalphabeticcipher.

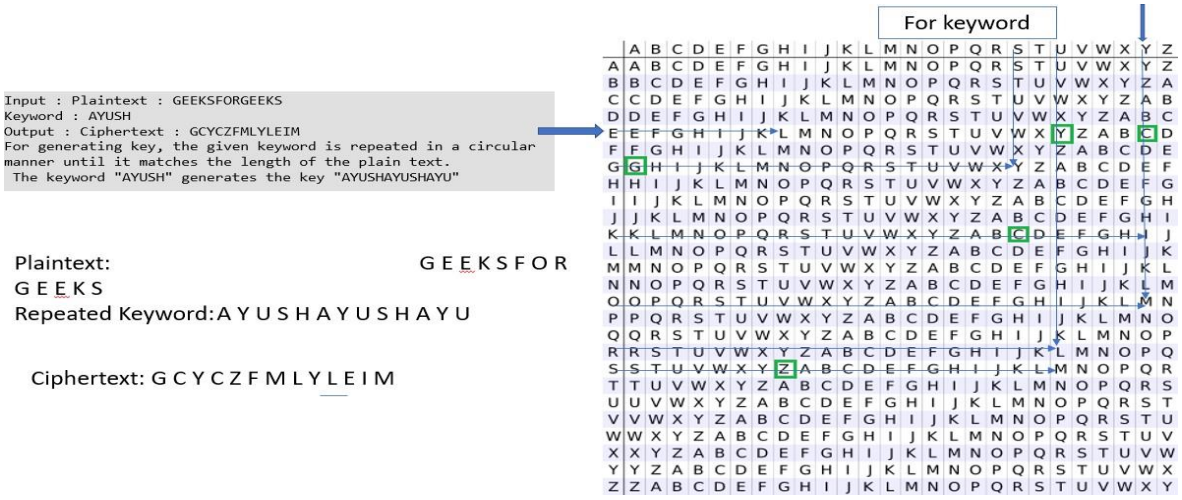
Difference between monoalphabetic cipher and polyalphabetic cipher:

A monoalphabetic cipher is a substitution cipher in which the cipher alphabet is fixed through the encryption process.....A polyalphabetic cipher is a substitution cipher in which the cipher alphabet changes during the encryption process.

Vigenere cipher:

 Vigenere Cipher is a method of encrypting alphabetic text. It uses a simple form of polyalphabetic substitution. A polyalphabetic cipher is any cipher based on substitution, using multiple substitution alphabets .The encryption of the originaltext is done using the Vigenère square or Vigenère table.

- The table consists of the alphabets written out 26 times in different rows, each alphabet shifted cyclically to the left compared to the previous alphabet, corresponding to the 26 possible Caesar Ciphers.
- At different points in the encryption process, the cipher uses a different alphabet from one of the rows.



- The alphabet used at each point depends on a repeating keyword

The Vigenère cipher can be expressed in the following manner. Assume a sequence of plaintext letters P = p_0, p_1, p_2,.....,p_{n-1} and a key consisting of the sequence of letters K = k_0, k_1, k_2,..... , k_{m-1}, where typically m <n. The sequence of ciphertext letters C = C_0, C_1, C_2, ……, C_{n-1} is calculated as follows:

C = C_0, C_1, C_2, …..,C_{n-1} = E(K, P) = E[(k_0, k_1, k_2, ….., k_{m-1}), (p_0, p_1, p_2…., p_{n-1})]

= (p_0 + k_0) mod 26, (p_1 + k_1) mod 26, ……, (p_{m-1} + k_{m-1}) mod 26, (p_m+ k_0) mod 26, (p_{m+1} + k_1) mod 26, ……. , (p_{2m-1} + km-1) mod 26,…..

Thus, the first letter of the key is added to the first letter of the plaintext, mod 26, the second letters are added, and so on through the first m letters of the plaintext. For the next m letters of the plaintext, the key letters are repeated. This process continues until all of the plaintext sequence is encrypted. A general equation of the encryption process is

C_i= (p_i+ k_i mod m) mod 26

A general equation for decryption is

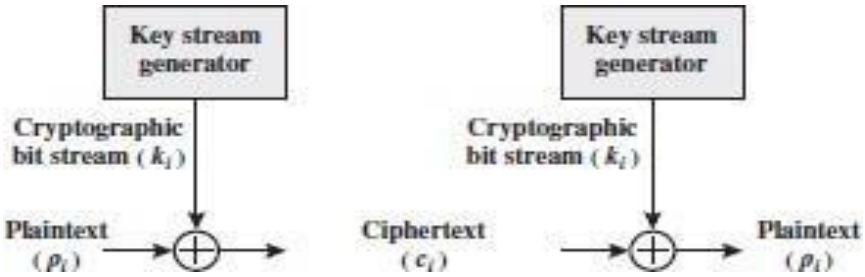
p_i= (C_i– k_i mod m) mod 26

To encrypt a message, a key is needed that is as long as the message. Usually, the key is a repeating keyword. For example, if the keyword is deceptive, the message "we are discovered save yourself" is encrypted as

Key :deceptivedeceptivedeceptive plaintext :
Wearediscoveredsaveyourselfciphertext: ZICVTWQNGRZGVTWAVZHCQYGLMGJ

The strength of this cipher is that there are multiple ciphertext letters for each plaintext letter, one for each unique letter of the keyword. Thus, the letter frequency information is obscured. However, not all knowledge of the plaintext structure is lost.

Vernam Cipher The ultimate defense against such a cryptanalysis is to choose a keyword that is as long as the plaintext and has no statistical relationship to it. Such a system was introduced by an



AT&T engineer named Gilbert Vernam in 1918.

- The system can be expressed as:

$$ci = pi \oplus ki$$

where

pi= ith binary digit of plaintext
ki= ith binary digit of key
ci= ith binary digit of ciphertext
$\oplus$ s= exclusive-or (XOR) operation

- Thus, the ciphertext is generated by performing the bitwise XOR of the plaintext and the key. Because of the properties of the XOR, decryption simply involves the same bitwise operation

One Time Pad Cipher

It is an unbreakable cryptosystem. It represents the message as a sequence of 0s and 1s. this can be accomplished by writing all numbers in binary, for example, or by using ASCII. The key is a random sequence of 0"s and 1"s of same length as the message. Once a key is used, it is discarded and never used again. The system can be expressed as follows:

Ci = Pi Ki

Ci - ith binary digit of cipher text Pi - ith binary digit of plaintext Ki - ith binary digit of key – exclusive OR opearaiton

Thus, the cipher text is generated by performing the bitwise XOR of the plaintext and the key. Decryption uses the same key. Because of the properties of XOR, decryption simply involves the same bitwise operation:

Pi = Ci Ki

e.g., plaintext = 0 0 1 0 1 0 0 1

Key = 1 0 1 0 1 1 0 0

ciphertext = 1 0 0 0 0 1 0 1

Advantage:

- Encryption method is completely unbreakable for a ciphertext only attack.

Disadvantages

- It requires a very long key which is expensive to produce and expensive to transmit.
- Once a key is used, it is dangerous to reuse it for a second message; any knowledge on the first message would give knowledge of the second.

STEGANOGRAPHY:

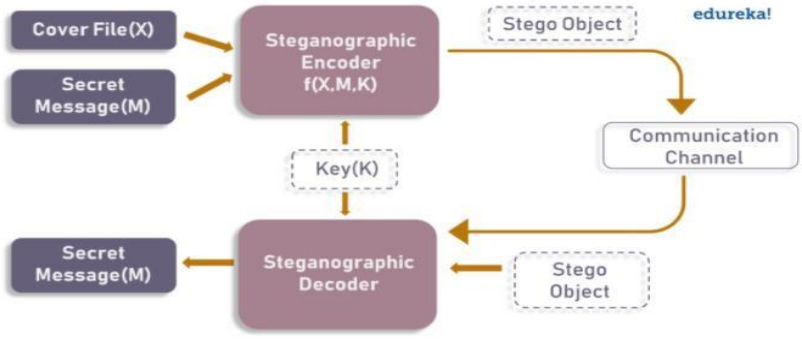
 Steganography is the technique of hiding secret data within an ordinary, non-secret, file or message in order to avoid detection; the secret data is then extracted at its destination.

The use of steganography can be combined with encryption as an extra step for hiding or protecting data.

 It stems from two Greek words, which are steganos, means covered and graphia, means writing

 Examples,

1.Playing an audio track backwards to reveal a secret message
2.Playing a video at a faster frame rate (FPS) to reveal a hidden image
3.Embedding a message in the red, green, or blue channel of an RGB image
4.Hiding information within a file header or metadata
5.Embedding an image or message within a photo through the addition of digital noise



•As the image depicts, both cover file(X) and secret message(M) are fed into steganographic encoder as input.

•Steganographic Encoder function, f(X,M,K) embeds the secret message into a coverfile.

• Resulting Stego Object looks very similar to your cover file, with no visible changes.

• This completes encoding. To retrieve the secret message, Stego Object is fed into Steganographic Decoder.

 Steganography Techniques

 Depending on the nature of the cover object (actual object in which secret data is embedded), steganography can be divided into five types:

1.Text Steganography
2.Image Steganography
3.Video Steganography
4.Audio Steganography
5.Network Steganography

 Text Steganography: Text Steganography is hiding information inside the text files. Various techniques used to hide the data in the text are:

•Format Based Method
•Random and Statistical Generation
•Linguistic Method

 Image Steganography: Hiding the data by taking the cover object as the image is known as image steganography. There are a lot of ways to hide information inside an image. Common approaches include:

•Least Significant Bit Insertion
•Masking and Filtering
•Redundant Pattern Encoding
•Encrypt and Scatter
•Coding and Cosine Transformation

 Audio Steganography: In audio steganography, the secret message is embedded into an audio signal which alters the binary sequence of the corresponding audio file. Different methods of audio steganography include:

•Least Significant Bit Encoding
•Parity Encoding
•Phase Coding
•Spread Spectrum

 Video Steganography: In Video Steganography you can hide kind of data into digital video format. Two main classes of Video Steganography include:

•embedding data in uncompressed raw video and compressing it later
•Embedding data directly into the compressed data stream
•Network Steganography (Protocol Steganography): It is the technique of embedding information within network control protocols used in data transmission such TCP, UDP, ICMP etc. For Example, you can hide information in the header of a TCP/IP packet in some fields that are either optional.

Example:

(i) the sequence of first letters of each word of the overall message spells out the real (hidden) message.

(ii) Subset of the words of the overall message is used to convey the hidden message.

Various other techniques have been used historically, some of them are:

Character marking – selected letters of printed or typewritten text are overwritten in pencil.

The marks are ordinarily not visible unless the paper is held to an angle to bright light. Invisible ink – a number of substances can be used for writing but leave no visible trace until heat or some chemical is applied to the paper.

Pin punctures – small pin punctures on selected letters are ordinarily not visible unless the paper is held in front of the light.

Typewritten correction ribbon – used between the lines typed with a black ribbon, the results of typing with the correction tape are visible only under a strong light.

Drawbacks of steganography

Requires a lot of overhead to hide a relatively few bits of information. Once the system is discovered, it becomes virtually worthless.

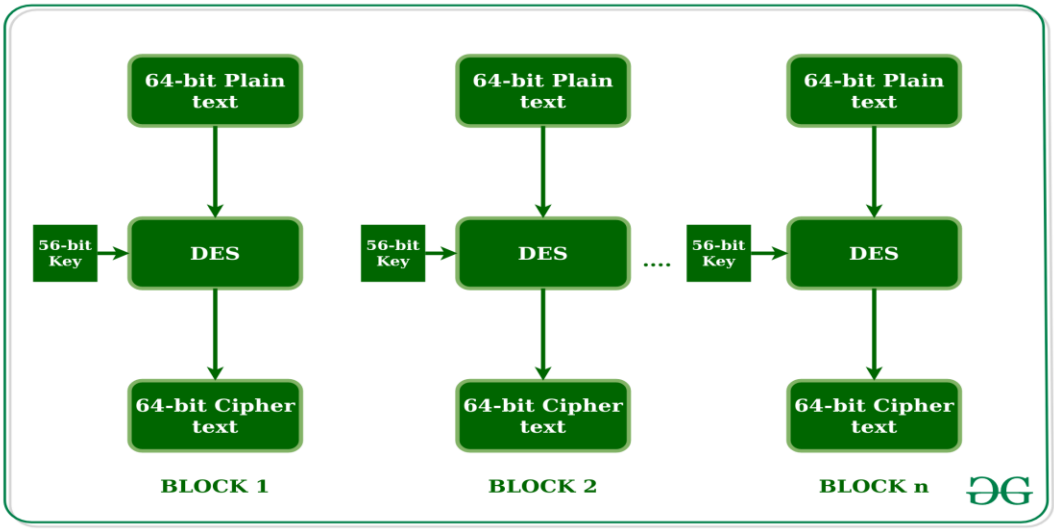
 TRANSPOSITION TECHNIQUES:

All the techniques examined so far involve the substitution of a cipher text symbol for a plaintext symbol. A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters. This technique is referred to as a transposition cipher.

Rail fence is simplest of such cipher, in which the plaintext is written down as a sequence of diagonals and then read off as a sequence of rows.
Plaintext = meet at the school house
To encipher this message with a rail fence of depth 2, we write the message asfollows:
m e a t e c o l o se t t h s H o h u e
The encrypted message is MEATECOLOSETTHSHOHUE
Row Transposition Ciphers-A more complex scheme is to write the message in a rectangle, row by row, and read the message off, column by column, but permute the order of the columns. The order of columns then becomes the key of the algorithm.
e.g., plaintext = meet at the school houseKey = 4 3 1 2 5 6 7
PT = m e e t a t th e s c h o o
l h o u s e
CT = ESOTCUEEHMHLAHSTOETO
A pure transposition cipher is easily recognized because it has the same letter frequencies as the original plaintext. The transposition cipher can be made significantly more secure by performing more than one stage of transposition. The result is more complex permutation that is not easily reconstructed

➢Data encryption standard (DES)

Data encryption standard (DES) has been found vulnerable against very powerful attacks and therefore, the popularity of DES has been found slightly on the decline. DES is a block cipher and encrypts data in blocks of size of 64 bits each, which means 64 bits of plain text goes as the input to DES, which produces 64 bits of ciphertext. The same algorithm and key are used for encryption and decryption, with minor differences. The key length is 56 bits. The basic idea is shown in the figure.



We have mentioned that DES uses a 56-bit key. Actually, the initial key consists of 64 bits. However, before the DES process even starts, every 8th bit of the key is discarded to produce a 56-bit key. That is bit positions 8, 16, 24, 32, 40, 48, 56, and 64 are discarded.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64

Figure - discording of every 8th bit of original key

Thus, the discarding of every 8th bit of the key produces a 56-bit key from the original 64-bit key. DES is based on the two fundamental attributes of cryptography: substitution (also called confusion) and transposition (also called diffusion). DES consists of 16 steps, each of which is called a round. Each round performs the steps of substitution and transposition. Let us now discuss the broad-level steps in DES.

1.In the first step, the 64-bit plain text block is handed over to an initial Permutation (IP) function.

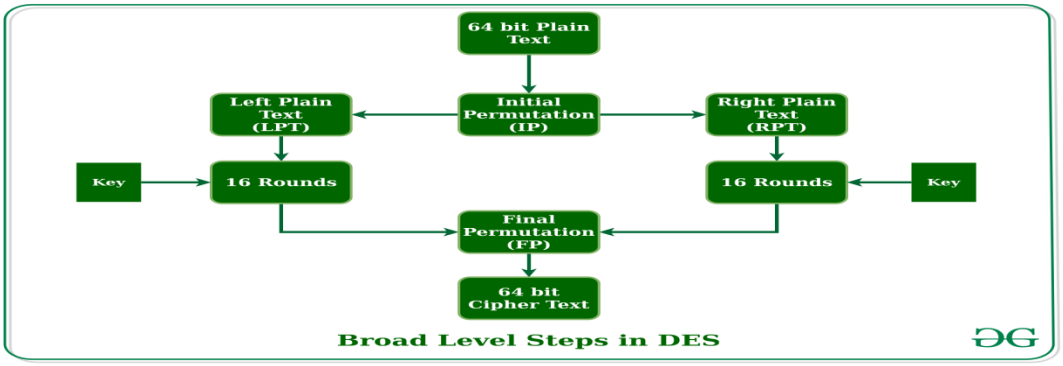
2. The initial permutation is performed on plain text.

3. Next, the initial permutation (IP) produces two halves of the permuted block; says Left Plain Text (LPT) and Right Plain Text (RPT).

4. Now each LPT and RPT go through 16 rounds of the encryption process.

5. In the end, LPT and RPT are rejoined and a Final Permutation (FP) is performed on the combined block

6. The result of this process produces 64-bit ciphertext.



Initial Permutation (IP)

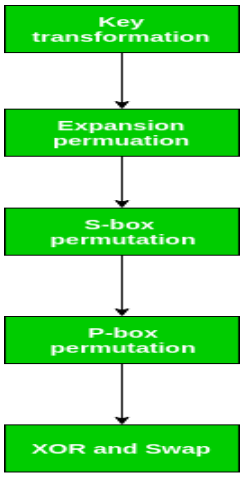
As we have noted, the initial permutation (IP) happens only once and it happens before the first round. It suggests how the transposition in IP should proceed, as shown in the figure. For example, it says that the IP replaces the first bit of the original plain text block with the 58th bit of the original plain text, the second bit with the 50th bit of the original plain text block, and so on

This is nothing but jugglery of bit positions of the original plain text block. the same rule applies to all the other bit positions shown in the figure.

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	33	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Figure - Initial permutation table

As we have noted after IP is done, the resulting 64-bit permuted text block is divided into two half blocks. Each half-block consists of 32 bits, and each of the 16 rounds, in turn, consists of the broad level steps outlined in the figure.



Step-1: Key transformation

We have noted initial 64-bit key is transformed into a 56-bit key by discarding every 8th bit of the initial key. Thus, for each a 56-bit key is available. From this 56-bit key, a different 48-bit Sub Key is generated during each round using a process called key transformation. For this, the 56-bit key is divided into two halves, each of 28 bits. These halves are circularly shifted left by one or two positions, depending on the round
For example, if the round numbers 1, 2, 9, or 16 the shift is done by only position for other rounds, the circular shift is done by two positions. The number of key bits shifted per round is shown in the figure.

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
#key bits shifted	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Figure - number of key bits shifted per round

After an appropriate shift, 48 of the 56 bits are selected. for selecting 48 of the 56 bits the table is shown in the figure given below. For instance, after the shift, bit number 14 moves on the first position, bit number 17 moves on the second position, and so on. If we observe the table carefully, we will realize that it contains only 48-bit positions. Bit number 18 is discarded (we will not find it in the table), like 7 others, to reduce a 56-bit key to a 48-bit key. Since the key transformation process involves permutation as well as a selection of a 48-bit subset of the original 56-bit key it is called Compression Permutation.

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Figure - compression permutation

Because of this compression permutation technique, a different subset of key bits is used in each round. That makes DES not easy to crack.

Step-2: Expansion Permutation

Recall that after initial permutation, we had two 32-bit plain text areas called Left Plain Text(LPT) and Right Plain Text(RPT). During the expansion permutation, the RPT is expanded from 32 bits to 48 bits. Bits are permuted as well hence called expansion permutation. This happens as the 32-bit RPT is divided into 8 blocks, with each block consisting of 4 bits. Then, each 4-bit block of the

previous step is then expanded to a corresponding 6-bit block, i.e., per 4-bit block, 2 more bits are added.

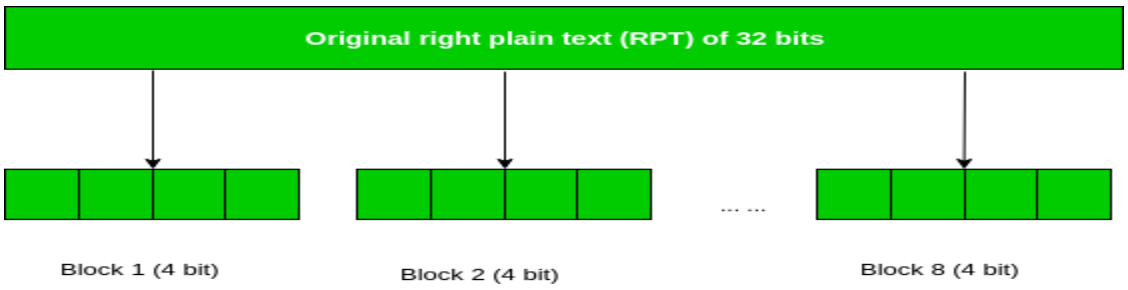


Figure - division of 32 bit RPT into 8 bit blocks

This process results in expansion as well as a permutation of the input bit while creating output. The key transformation process compresses the 56-bit key to 48 bits. Then the expansion permutation process expands the 32-bit RPT to 48-bits. Now the 48-bit key is XOR with 48-bit RPT and the resulting output is given to the next step, which is the S-Box substitution.

Block Cipher and Public Key Cryptography

Block Cipher:

A block cipher takes a block of plaintext bits and generates a block of ciphertext bits, generally of same size. The size of block is fixed in the given scheme. The choice of block size does not directly affect to the strength of encryption scheme. The strength of cipher depends up on the key length.

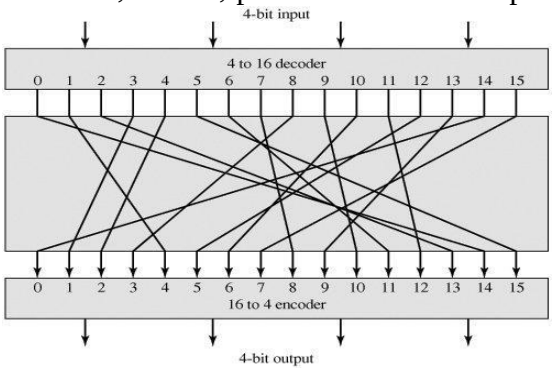
Block Size:

Block ciphers use a block of bits as the unit of encryption and decryption. To encrypt a 64-bit block, one has to take each of the 2^{64} input values and map it to one of the 2^{64} output values. The mapping should be one-to-one. Encryption and decryption operations of a block cipher are shown in Fig.

Some operations, such as permutation and substitution, are performed on the block of bits based on a key (a secret number) to produce another block of bits.

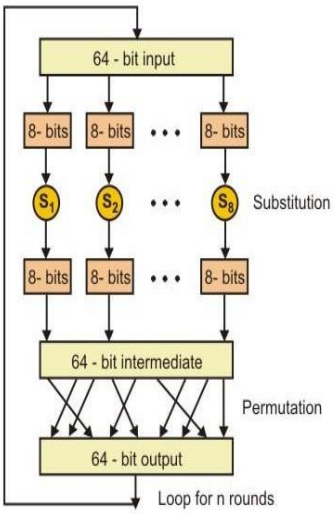
Permutation: The permutation is performed by a permutation box at the bit-level, which keeps the number of 0s and 1s same at the input and output. Although it can be implemented either by hardware or software, the hardware implementation is faster.

Substitution: Fig. shows the substitution is implemented with the help of three building blocks – a decoder, one p-box and an encoder. For an n-bit input, the decoder produces an 2n bit output having only one 1, which is applied to the P-box. The P-box permutes the output of the decoder and it is applied to the encoder. The encoder, in turn, produces an n-bit output. For example, if the input to the decoder is 011, the



output of the decoder is 00001000. Let the permuted output is 01000000, the output of the encoder is 011.

Most symmetric block ciphers are based on a Feistel Cipher Structure needed since must be able to decrypt ciphertext to recover messages efficiently. block ciphers look like an extremely It devid input into 8-Bit pieces Substitute each 8-bit based on functions derived from the key. Permute the bitsbased on the key



Requires table of 264 entries for a 64-bit block

- Instead create from smaller building blocks
- using idea of a product cipher in 1949 Claude Shannon introduced idea of substitution- permutation (S-P) networks called modern substitution-transposition product cipher these form the basis of modern block ciphers
- S-P networks are based on the two primitive cryptographic operations we have seen before:
- substitution (S-box)
- permutation (P-box)
- provide confusion and diffusion of message
- diffusion – dissipates statistical structure of plaintext over bulk of ciphertext
- confusion – makes relationship between ciphertext and key as complex as possible

Block Cipher Schemes:

There is a vast number of block ciphers schemes that are in use. Many of them are publicly known. Most popular and prominent block ciphers are listed below.

Digital Encryption Standard (DES) − The popular block cipher of the 1990s. It is now considered as a 'broken' block cipher, due primarily to its small key size.

Triple DES − It is a variant scheme based on repeated DES applications. It is still a respected block ciphers but inefficient compared to the new faster block ciphers available.

Advanced Encryption Standard (AES) − It is a relatively new block cipher based on the encryption algorithm

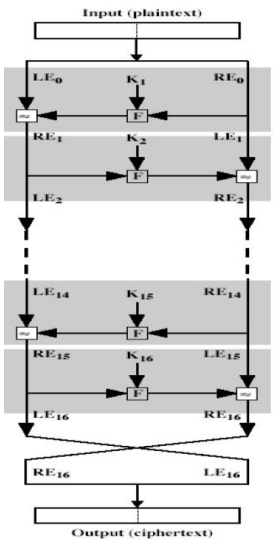
Feistel cipher structure:

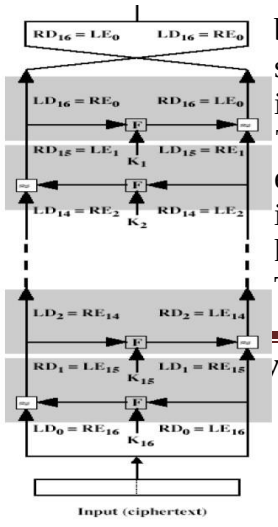
The plaintext block is divided into two halves L0 and R0. The two halves of the data pass through „n" rounds of processing and then combine to produce the ciphertext block.

Each round „i" has inputs Li-1 and Ri-1, derived from the previous round, as well as the subkey Ki, derived from the overall key K. in general, the subkeys Ki are different from K and from each other.

All rounds have the same structure. 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 XOR of the output of that function and the left half of the data.





The round function has the same general structure for each round but is parameterized by the round subkey ki. Following this substitution, a permutation is performed that consists of the interchange ofthe two halves of the data.

The process of decryption is essentially the same as the encryption process. The rule is as follows: use the cipher text as input to the algorithm, but use the subkey ki in reverse order. i.e., kn in the first round, kn-1 in second round and so on.

The exact realization of a Feistel network depends on the choice

of the following parameters and design features:

- Block size - Increasing size improves security, but slows cipher
- Key size - Increasing size improves security, makes exhaustive key searching harder, but may slow cipher
- Number of rounds - Increasing number improves security, but slows cipher
- Subkey generation - Greater complexity can make analysis harder, but slows cipher
- Round function - Greater complexity can make analysis harder, but slows cipher
- Fast software en/decryption & ease of analysis - are more recent concerns for practical use and testing.

The exact realization of a Feistel network depends on the choice of the following parameters and design features:

- Block size: Larger block sizes mean greater security (all other things being equal) but reduced 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.
- 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 to be inadequate, and 128 bits has become a common size.
- 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.
- Subkey generation algorithm: Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.
- Round function F: Again, greater complexity generally means greater resistance to cryptanalysis. There are two other considerations in the design of a Feistel cipher:
- Fast software encryption/decryption: In many cases, encryption is embedded in applications or utility functions in such a way as to preclude a hardware implementation.

Accordingly, the speed of execution of the algorithm becomes a concern.

- 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.

Block Cipher Schemes

There is a vast number of block ciphers schemes that are in use. Many of them are publicly known. Most popular and prominent block ciphers are listed below.

Digital Encryption Standard (DES) − The popular block cipher of the 1990s. It is now considered as a 'broken' block cipher, due primarily to its small key size.

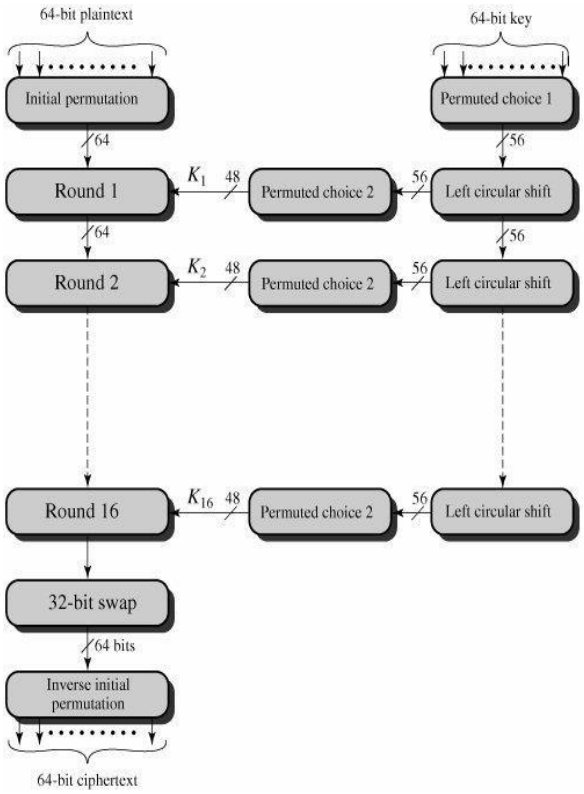
Triple DES − It is a variant scheme based on repeated DES applications. It is still a respected block ciphers but inefficient compared to the new faster block ciphers available.

Advanced Encryption Standard (AES) − It is a relatively new block cipher based on the encryption algorithm

Data Encryption

In May 1973, and again in Aug 1974 the NBS (now NIST) called for possible encryption algorithms for use in unclassified government applications response was mostly disappointing, however, IBM submitted their Lucifer design following a period of redesign and comment it became the Data Encryption Standard (DES) it was adopted as a (US) federal standard in Nov 76, published by NBS as a hardware only scheme in Jan 77 and by

ANSI for both hardware and software standards in ANSI X3.92-
1981 (also X3.106-1983 modes of use) subsequently it has been widely adopted and is now published in many standards around the world cf Australian Standard AS2805.5-1985 one of the largest users of the DES is the banking industry, particularly with EFT, and EFTPOS it is for this use that the DES has primarily been standardized, with ANSI having twice reconfirmed its recommended use for 5 year periods - a further extension is not expected however although the standard is public, the design criteria used are classified and have yet to be released there has been considerable controversy over the design, particularly in the choice of a 56-bit key, recent analysis has shown despite this that the choice was appropriate, and that DES is well designed, rapid advances in computing speed though have rendered the 56-bit key susceptible to exhaustive key search, as predicted by Diffie & Hellman.



The overall scheme for DES encryption is illustrated in Fig. As with any encryption scheme, there are two inputs to the encryption function: the plaintext to be encrypted and the key. In this case, the plaintext must be 64 bits in length and the key is 56 bits in length.

Looking at the left-hand side of the figure, we can see that the processing of the plaintext proceeds in three phases. First, the 64-bit plaintext passes through an initial permutation (IP) that rearranges the bits to produce the permuted input. This is followed by a phase consisting of 16 rounds of the same function, which involves both permutation and substitution functions. The output of the last (sixteenth) round consists of 64 bits that are a function of the input plaintext and the key. The left and right halves of the output are swapped to produce the preoutput.

Finally, the preoutput is passed through a permutation (IP-1) that is the inverse of the initial permutation function, to produce the 64-bit ciphertext. With the exception of the initial and final permutations, DES has the exact structure of a Feistel cipher.

➢Block Cipher Modes of Operation

Block cipher mode of operation is an algorithm that uses a block cipher to processes the data blocks of fixed size provide information security such as confidentiality or authenticity.the long message is divided into a series of sequential message blocks, and the cipher operates on these blocks one at a time.

A block cipher by itself is only suitable for the secure cryptographic transformation (encryption or decryption) of one fixed-length group of bits called a block.A mode of operation describes how to repeatedly apply a cipher's single-block operation to securely transform amounts of data larger than a block.

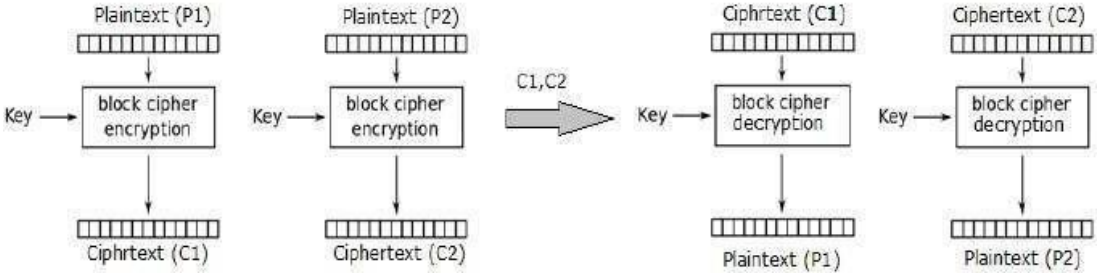
Electronic Code Book (ECB) Mode

This mode is a most straightforward way of processing a series of sequentially listed message blocks.

Operation:

The user takes the first block of plaintext and encrypts it with the key to produce the first block of ciphertext. He then takes the second block of plaintext and follows the same process with same key and so on so forth.

The ECB mode is deterministic, that is, if plaintext block P1, P2,…, Pm are encrypted twice under the same key, the output ciphertext blocks will be the same.
In fact, for a given key technically we can create a codebook of cipher texts for all possible plaintext blocks. Encryption would then entail only looking up for required plaintext and select the corresponding ciphertext. Thus, the operation is analogous to the assignment of code words in a codebook, and hence gets an official name − Electronic Codebook mode of operation (ECB). It is illustrated as follows



Mode	Description	sTypical Application
Electronic Codebook (ECB)	Each block of plaintext bits is encoded independently using the same key.	r Secure transmission of single values (e.g., an encryption key)
Cipher Block Chaining (CBC)	The input to the encryption algo- rithm is the XOR of the next block of plaintext and the preceding block of ciphertext.	r General-purpose block-oriented transmission r Authentication
Cipher Feedback (CFB)	Input is processed *s* bits at a time. Preceding ciphertext is used as input to the encryption algorithm to produce pseudorandom output, which is XORed with plaintext to produce next unit of ciphertext.	r General-purpose stream-oriented transmission r Authentication
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption algorithm is the preceding encryption output, and full blocks are used.	r Stream-oriented transmission over noisy channel (e.g., satellite communication)
Counter (CTR)	Each block of plaintext is XORed with an encrypted counter. The counter is incremented for each subsequent block.	r General-purpose block-oriented transmission r Useful for high-speed requirements

Analysis of ECB Mode:

In reality, any application data usually have partial information which can be guessed. For example, the range of salary can be guessed. A ciphertext from ECB can allow an attacker to guess the plaintext by trial-and-error if the plaintext message is within predictable.

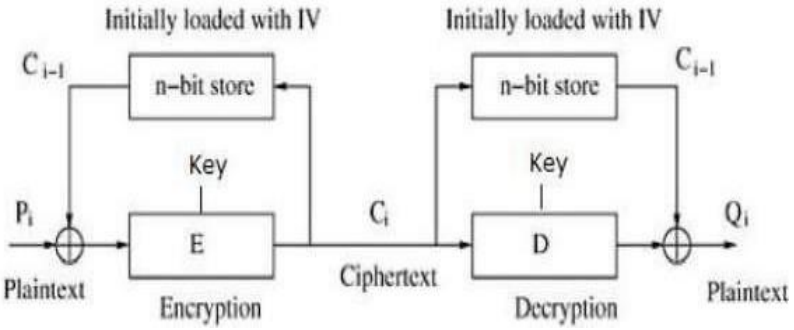
For example, if a ciphertext from the ECB mode is known to encrypt a salary figure, then a small number of trials will allow an attacker to recover the figure. In general, we do not wish to use a deterministic cipher, and hence the ECB mode should not be used in most applications.

Cipher Block Chaining (CBC) Mode

CBC mode of operation provides message dependence for generating ciphertext and makes the system non-deterministic.

Operation:

The operation of CBC mode is depicted in the following illustration. The steps are as follows. Load the n-bit Initialization Vector (IV) in the top register. XOR the n-bit plaintext block with data value in top register. Encrypt the result of XOR operation with underlying block cipher with key K. Feed ciphertext block into top register and continue the operation till all plaintext blocks are processed.



For decryption, IV data is XORed with first ciphertext block decrypted. The first ciphertext block is also fed into to register replacing IV for decrypting next ciphertext block.

Analysis of CBC Mode

In CBC mode, the current plaintext block is added to the previous ciphertext block, and then the result is encrypted with the key. Decryption is thus the reverse process, which involves decrypting the current ciphertext and then adding the previous ciphertext block to the result.

Advantage of CBC over ECB is that changing IV results in different ciphertext for identical message. On the drawback side, the error in transmission gets propagated to few further blocks during decryption due to chaining effect.

It is worth mentioning that CBC mode forms the basis for a well-known data origin authentication mechanism. Thus, it has an advantage for those applications that require both symmetric encryption and data origin authentication.

Cipher Feedback (CFB) Mode

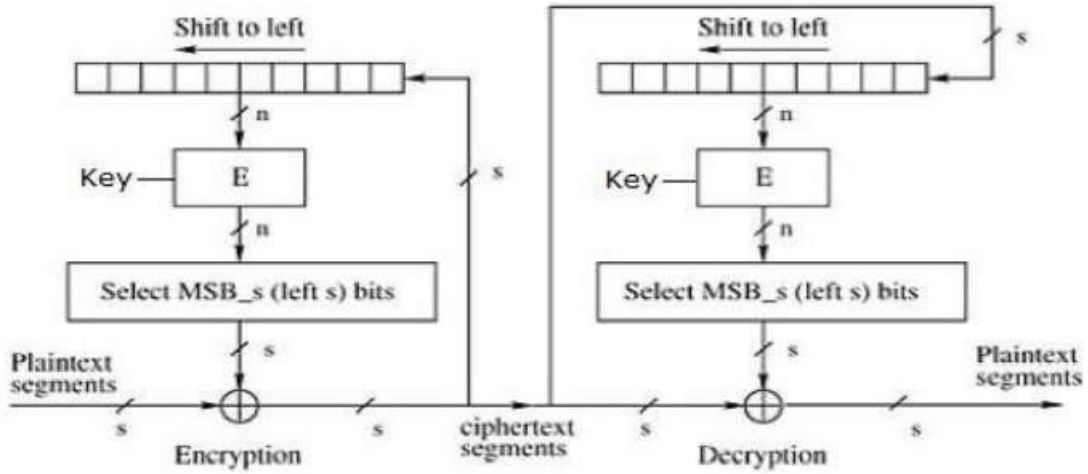
In this mode, each ciphertext block gets 'fed back' into the encryption process in order to encrypt the next plaintext block.

Operation

The operation of CFB mode is depicted in the following illustration. For example, in the present system, a message block has a size 's' bit where $1 < s < n$. The CFB mode requires an initialization vector (IV) as the initial random n-bit input block. The IV need not be secret.

Steps of operation are, Load the IV in the top register. Encrypt the data value in top register with underlying block cipher with key K. Take only 's' number of most significant bits (left bits) of output of encryption process and XOR them with 's' bit plaintext message block to generate ciphertext block. Feed ciphertext block into top register by shifting already present data to the left and continue the operation till all plaintext blocks are processed. Essentially, the previous ciphertext block is encrypted with the key, and then the result is XORed to the current plaintext block. Similar steps are followed for decryption. Pre-decided IV is initially

loaded at the start of decryption.



Analysis of CFB Mode

CFB mode differs significantly from ECB mode, the ciphertext corresponding to a given plaintext block depends not just on that plaintext block and the key, but also on the previous ciphertext block. In other words, the ciphertext block is dependent of message.

CFB has a very strange feature. In this mode, user decrypts the ciphertext using only the encryption process of the block cipher. The decryption algorithm of the underlying block cipher is never used.

Apparently, CFB mode is converting a block cipher into a type of stream cipher. The encryption algorithm is used as a key-stream generator to produce key-stream that is placed in the bottom register. This key stream is then XORed with the plaintext as in case of stream cipher.

By converting a block cipher into a stream cipher, CFB mode provides some of the advantageous properties of a stream cipher while retaining the advantageous properties of a block cipher. On the flip side, the error of transmission gets propagated due to changing of blocks.

Output Feedback (OFB) Mode

It involves feeding the successive output blocks from the underlying block cipher back to it. These feedback blocks provide string of bits to feed the encryption algorithm which act as the key- stream generator as in case of CFB mode. The key stream generated is XOR-ed with the plaintext sblocks. The OFB mode requires an IV as the initial random n-bit input block. The IV need not be secret. The operation is depicted in the following illustration

Counter (CTR) Mode:

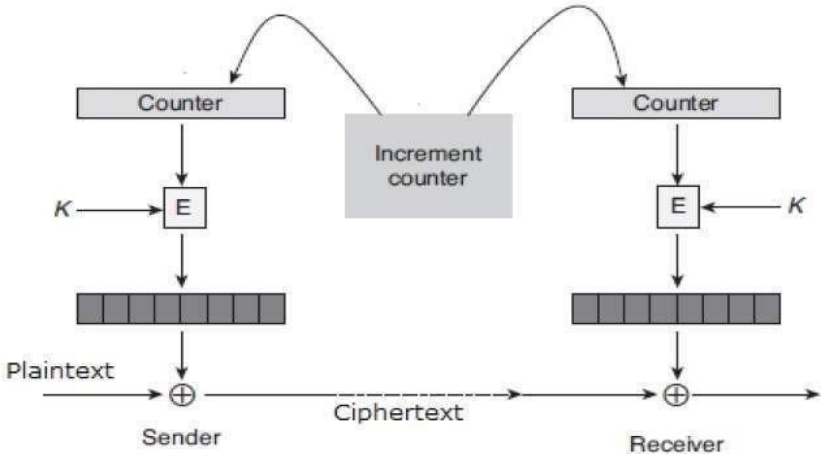
It can be considered as a counter-based version of CFB mode without the feedback. In this mode, both the sender and receiver need to access to a reliable counter, which computes a new shared value each time a ciphertext block is exchanged. This shared counter is not necessarily a secret value, but challenge is that both sides must keep the counter synchronized.

Operation

Both encryption and decryption in CTR mode are depicted in the following illustration. Steps in operation are, Load the initial counter value in the top register is the same for both the sender and the receiver. It plays the same role as the IV in CFB (and CBC) mode.

Encrypt the contents of the counter with the key and place the result in the bottom register. Take the first plaintext block P1 and XOR this to the contents of the bottom register. The result of this is C1. Send C1 to

the receiver and update the counter. The counter update replaces the ciphertext feedback in CFB mode. Continue in this manner until the last plaintext block has been encrypted. The decryption is the reverse process. The ciphertext block is XORed with the output of encrypted contents of counter value. After decryption of each ciphertext block counter is updated as in case of encryption.



Analysis of Counter Mode

It does not have message dependency and hence a ciphertext block does not depend on the previous plaintext blocks. Like CFB mode, CTR mode does not involve the decryption process of the block cipher. This is because the CTR mode is really using the block cipher to generate a key-stream, which is encrypted using the XOR function. In other words, CTR mode also converts a block cipher to a streamcipher.

The serious disadvantage of CTR mode is that it requires a synchronous counter at sender and receiver. Loss of synchronization leads to incorrect recovery of plaintext. However, CTR mode has almost all advantages of CFB mode. In addition, it does not propagate error of transmission at all.