RGPVNOTES.IN

Program : **B.E**

Subject Name: **Information Security**

Subject Code: **IT-8001**

Semester: **8th**

Unit I: Basic of Cryptography

Cryptography is the art and science of making a cryptosystem that is capable of providing information security.

- Cryptography deals with the actual securing of digital data. It refers to the design of mechanisms based on mathematical algorithms that offer vital information security services.
- A cryptosystem is an implementation of cryptographic techniques and their accompanying infrastructure to provide information security services. A cryptosystem is also referred to as a **cipher system**.
- A simple model of a cryptosystem that provides confidentiality to the information being transmitted, this basic model is depicted in figure 1.1

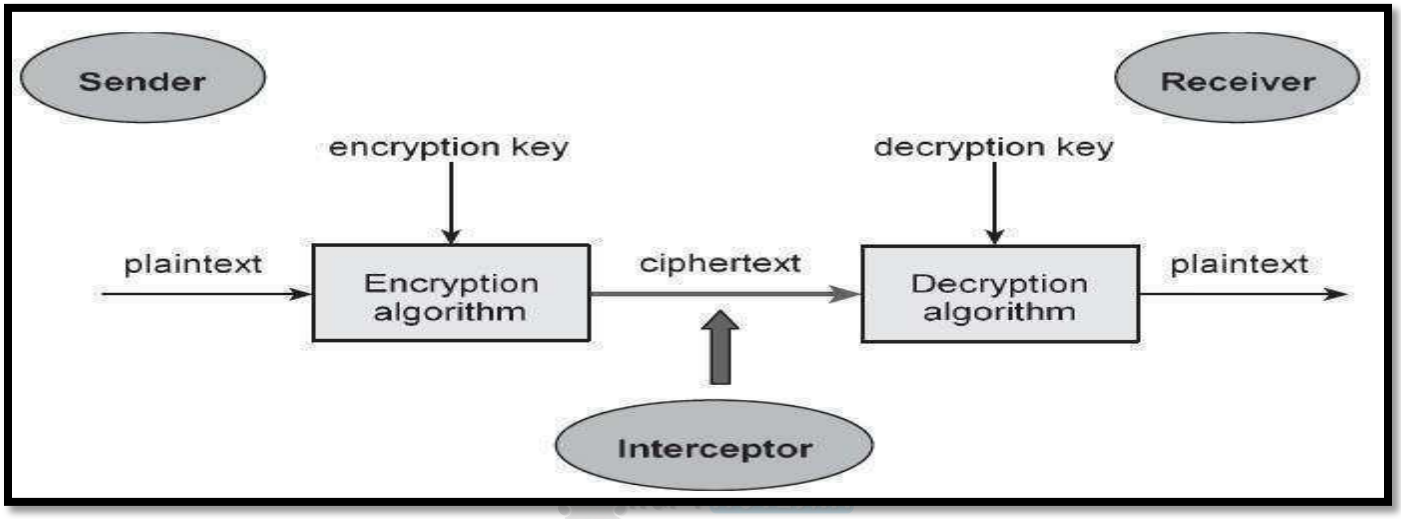

Fig 1.1: Cryptography Process

It shows a sender who wants to transfer some sensitive data to a receiver in such a way that any party intercepting or eavesdropping on the communication channel cannot extract the data.

Components of a Cryptosystem

The various components of a basic cryptosystem are as follows −

- **Plaintext.** It is the data to be protected during transmission.
- **Encryption Algorithm.** It is a mathematical process that produces a ciphertext for any given plaintext and encryption key. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.
- **Ciphertext.** It is the scrambled version of the plaintext produced by the encryption algorithm using a specific encryption key. The ciphertext is not guarded; It flows on a public channel. It can be intercepted or compromised by anyone who has access to the communication channel.
- **Decryption Algorithm,** It is a mathematical process, that produces a unique plaintext for any given ciphertext and decryption key. It is a cryptographic algorithm that takes a ciphertext and a decryption key as input, and outputs a plaintext. The decryption algorithm essentially reverses the encryption algorithm and is thus closely related to it.
- **Encryption Key.** It is a value that is known to the sender. The sender inputs the encryption key into the encryption algorithm along with the plaintext in order to compute the ciphertext.
- **Decryption Key.** It is a value that is known to the receiver. The decryption key is related to the encryption key, but is not always identical to it. The receiver inputs the decryption key into the decryption algorithm along with the ciphertext to compute the plaintext.

Types of Cryptosystems

Fundamentally, there are two types of cryptosystems based on the manner in which encryption-decryption is

carried out in the system −

- Symmetric Key Encryption
- Asymmetric Key Encryption

The main difference between these cryptosystems is the relationship between the encryption and the decryption key. Logically, in any cryptosystem, both the keys are closely associated. It is practically impossible to decrypt the ciphertext with the key that is unrelated to the encryption key.

Secret key cryptography/Symmetric Key Encryption

The encryption process where **the same keys are used for encrypting and decrypting** the information is known as Symmetric Key Encryption. The study of symmetric cryptosystems is referred to as **symmetric cryptography**. Symmetric cryptosystems are also sometimes referred to as **secret key cryptosystems**.

A few well-known examples of symmetric key encryption methods are − Digital Encryption Standard (DES), Triple-DES (3DES), IDEA, and BLOWFISH.

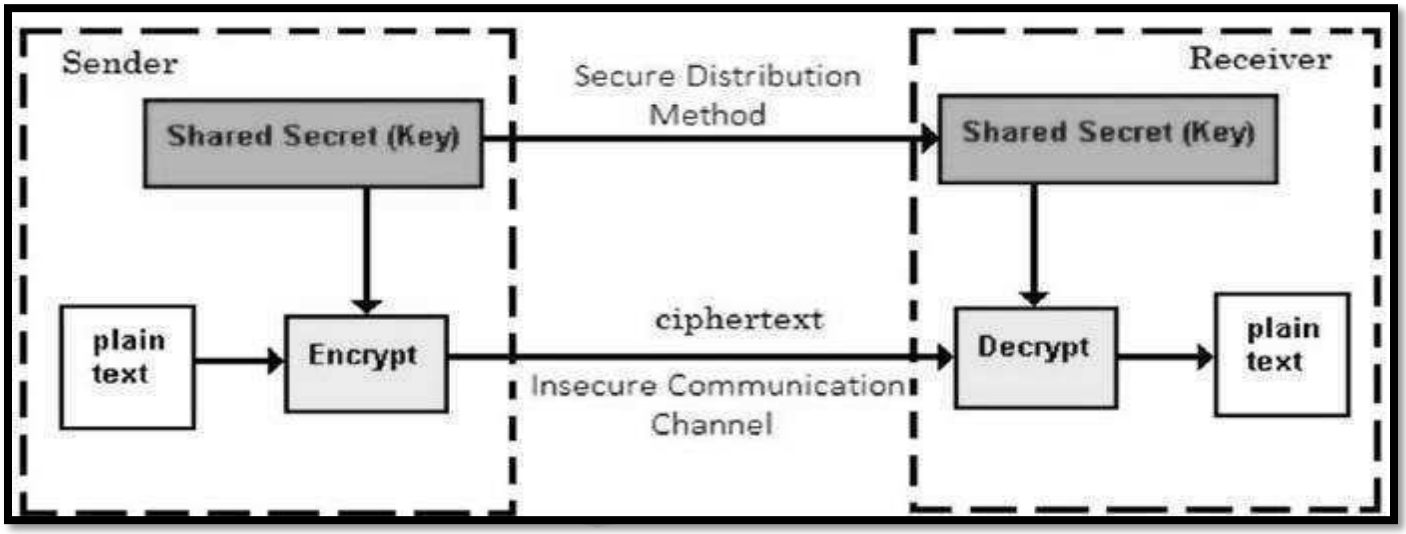

Fig 1.2: Symmetric key encryption

The salient features of cryptosystem based on symmetric key encryption are −

- Persons using symmetric key encryption must share a common key before exchange of information.
- Keys are recommended to be changed regularly to prevent any attack on the system.
- A robust mechanism needs to exist to exchange the key between the communicating parties. As keys are required to be changed regularly, this mechanism becomes expensive and cumbersome.
- In a group of **n** people, to enable two-party communication between any two persons, the number of keys required for a group is **n × (n – 1)/2**.

Asymmetric Key Encryption

The encryption process where **different keys are used for encrypting and decrypting the information** is known as Asymmetric Key Encryption. Though the keys are different, they are mathematically related and hence, retrieving the plaintext by decrypting ciphertext is feasible. The process is depicted in the following figure 1.3. The salient features of this encryption scheme are as follows −

- Every user in this system needs to have a pair of different keys, **private key** and **public key**. These keys are mathematically related − when one key is used for encryption, and the other can decrypt the ciphertext back to the original plaintext.
- It requires to put the public key in public repository and the private key as a well-guarded secret. Hence, this scheme of encryption is also called **Public Key Encryption**.
- Though public and private keys of the user are related, it is computationally not feasible to find one from another. This is the strength of this scheme.
- When Host1 needs to send data to Host2, he obtains the public key of Host2 from the repository, encrypts the data, and transmits.

- Host2 uses his private key to extract the plaintext.
- Length of Keys (number of bits) in this encryption is large, and hence, the process of encryption-decryption is slower than symmetric key encryption.
- The processing power of the computer system required to run asymmetric algorithm is higher.

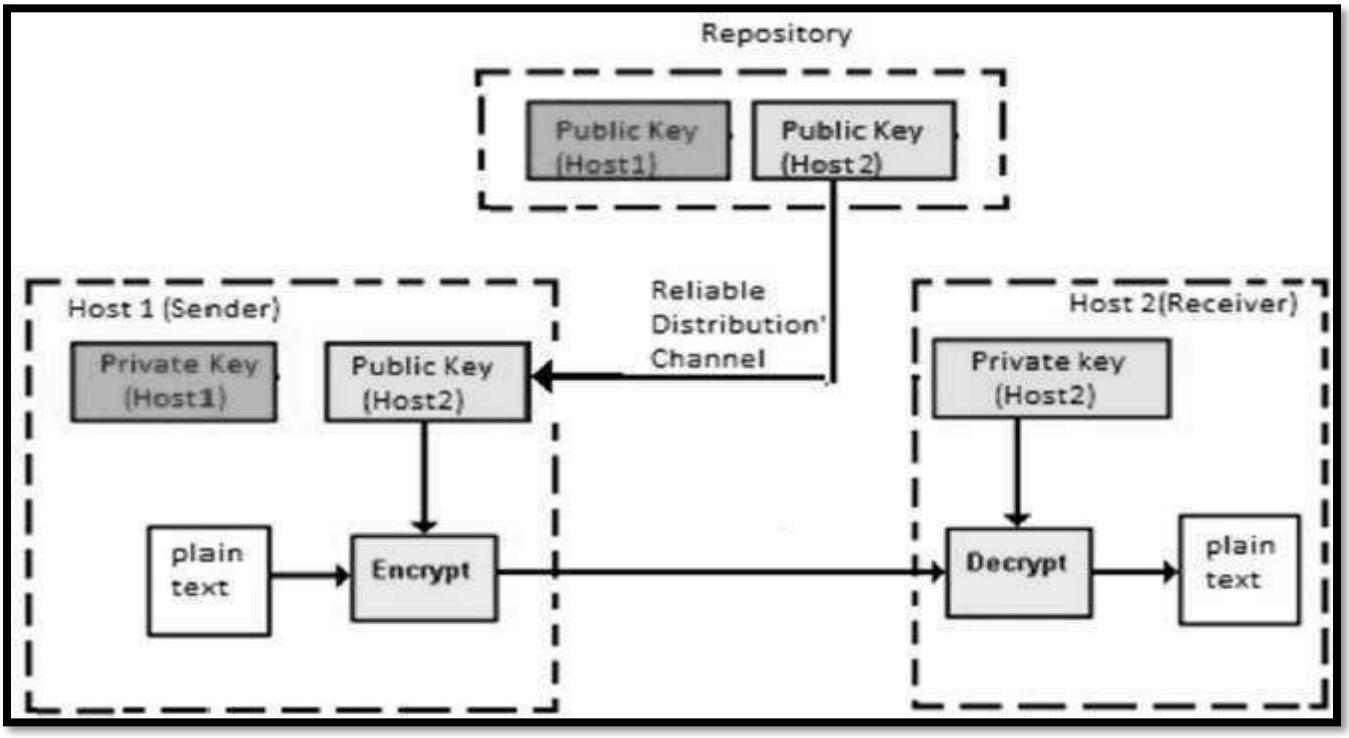

Fig 1.3: Asymmetric Key Encryption

Types of attack

Attacks are typically categorised based on the action performed by the attacker. An attack, thus, can be **passive** or **active**.

Passive Attacks

- The primary goal of a passive attack is to obtain **unauthorised access to the information**. For example, actions such as intercepting and eavesdropping on the communication channel can be regarded as a passive attack.
- These actions are passive in nature, as they neither affect information nor disrupt the communication channel. A passive attack is often seen as stealing information. The only difference in stealing physical goods and stealing information is that theft of data still leaves the owner in possession of that data.

Active Attacks

An active attack involves changing the information in some way by conducting some process on the information. For example,

- Alteration of authentication data such as originator name.
- Unauthorized deletion of data.
- Denial of access to information for legitimate users (denial of service).

Substitution ciphers

Caesar Cipher

It is a mono-alphabetic cipher wherein each letter of the plaintext is substituted by another letter to form the ciphertext. It is the simplest form of substitution cipher scheme. This cryptosystem is generally referred to as the **Shift Cipher**. The concept is to replace each alphabet by another alphabet which is 'shifted' by some fixed number between 0 and 25. For this type of scheme, both sender and receiver agree on a 'secret shift number' for shifting the alphabet. This number which is between 0 and 25 becomes the key of encryption. The name

'Caesar Cipher' is occasionally used to describe the Shift Cipher when the 'shift of three' is used.

The process of Shift Cipher

- To encrypt a plaintext letter, the sender positions the sliding ruler underneath the first set of plaintext letters and slides it to LEFT by the number of positions of the secret shift.
- The plaintext letter is then encrypted to the ciphertext letter on the sliding ruler underneath. The result of this process is depicted in the following illustration for an agreed shift of three positions. In this case, the plaintext 'tutorial' is encrypted to the ciphertext 'WXWRULDO'. Here is the ciphertext alphabet for a Shift of 3 –

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

Fig 1.4: Shift Cipher (plaintext to ciphertext)

- On receiving the ciphertext, the receiver who also knows the secret shift positions his sliding ruler underneath the ciphertext alphabet and slides it to RIGHT by the agreed shift number, 3 in this case.
- He then replaces the ciphertext letter by the plaintext letter on the sliding ruler underneath. Hence the ciphertext 'WXWRULDO' is decrypted to 'tutorial'. To decrypt a message encoded with a Shift of 3, generate the plaintext alphabet using a shift of '-3' as shown below. –

Ciphertext Alphabet	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
Plainrtext Alphabet	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

Fig 1.5: Shift Cipher (ciphertext to plaintext)

Simple Substitution Cipher: The process of Simple Substitution Cipher

- Write the alphabets A, B, C,..., Z in the natural order.
- The sender and the receiver decide on a randomly selected permutation of the letters of the alphabet.
- Underneath the natural order alphabets, write out the chosen permutation of the letters of the alphabet. For encryption, the sender replaces each plaintext letters by substituting the permutation letter that is directly beneath it in the table. This process is shown in the following illustration. In this example, the chosen permutation is K, D, G, ..., O. The plaintext 'point' is encrypted to 'MJBXZ'.

Here is a jumbled Ciphertext alphabet, where the order of the ciphertext letters is a key.

Plaintext Alphabet	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext Alphabet	K	D	G	F	N	S	L	V	B	W	A	H	E	X	J	M	Q	C	P	Z	R	T	Y	I	U	O

Fig 1.6: Substitution Cipher

- On receiving the ciphertext, the receiver, who also knows the randomly chosen permutation, replaces each ciphertext letter on the bottom row with the corresponding plaintext letter in the top row. The ciphertext 'MJBXZ' is decrypted to 'point'.

Mono-alphabetic and Poly-alphabetic Cipher

Mono-alphabetic cipher is a substitution cipher in which for a given key, the cipher alphabet for each plain alphabet is fixed throughout the encryption process. For example, if 'A' is encrypted as 'D', for any number of occurrence in that plaintext, 'A' will always get encrypted to 'D'.

Poly-alphabetic Cipher is a substitution cipher in which the cipher alphabet for the plain alphabet may be different at different places during the encryption process. The next two examples, **playfair and Vigenere Cipher are polyalphabetic ciphers**.

Playfair Cipher

In this scheme, pairs of letters are encrypted, instead of single letters as in the case of simple substitution cipher. In Playfair cipher, initially, a key table is created. The key table is a 5×5 grid of alphabets that acts as the

key for encrypting the plaintext. Each of the 25 alphabets must be unique, and one letter of the alphabet (usually J) is omitted from the table as we need only 25 alphabets instead of 26. If the plaintext contains J, then it is replaced by I. The sender, and the receiver decides on a particular key, say 'tutorials.' In a key table, the first characters (going left to right) in the table is the phrase, excluding the duplicate letters. The rest of the table will be filled with the remaining letters of the alphabet, in the natural order. The key table works out to be –

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

Fig 1.7: Playfair Cipher

The process of Playfair Cipher

- First, a plaintext message is split into pairs of two letters (digraphs). If there is an odd number of letters, a Z is added to the last letter. Let us say we want to encrypt the message "hide money." It will be written as –
  HI DE MO NE YZ
- The rules of encryption are –
  - If both the letters are in the same column, take the letter below each one (going back to the top if at the bottom)

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

'H' and 'I' are in the same column, hence take letter below them to replace. HI → QC

- If both letters are in the same row, take the letter to the right of each one (going back to the left if at the farthest right)

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

'D' and 'E' are in the same row, hence take a letter to the right of them to replace. DE → EF

- If neither of the preceding two rules is true, form a rectangle with the two letters and take the letters on the horizontal opposite corner of the rectangle.

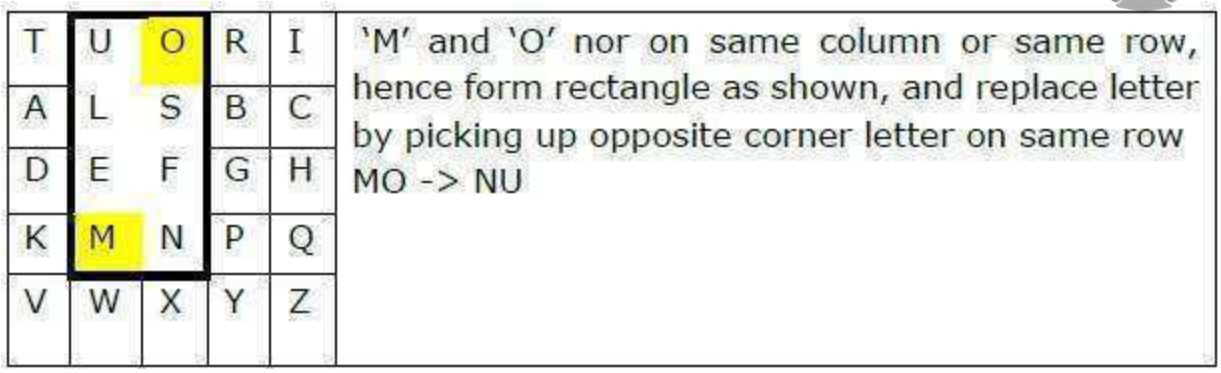

Fig 1.8: an example of Playfair Cipher

Using these rules, the result of the encryption of 'hide money' with the key of 'tutorials' would be –
QC EF NU MF ZV

Decrypting the Playfair cipher is as simple as doing the same process in reverse. The receiver has the same key and can create the same key table, and then decrypt any messages made using that key.

Vigenere Cipher

This scheme of cipher uses a text string (say, a word) as a key, which is then used for doing a number of shifts on the plaintext.

For example, let's assume the key is 'point'. Each alphabet of the key is converted to its respective numeric value: In this case,

p → 16, o → 15, i → 9, n → 14, and t → 20.

Thus, the key is 16 15 9 14 20.

The process of Vigenere Cipher

- The sender and the receiver decide on a key. Say 'point' is the key. Numeric representation of this key is '16 15 9 14 20'.
- The sender wants to encrypt the message, say 'attack from the south-east.' He will arrange plaintext and numeric key as follows –

a	t	t	a	c	k	f	r	o	m	s	o	u	t	h	e	a	s	t
16	15	9	14	20	16	15	9	14	20	16	15	9	14	20	16	15	9	14

Fig 1.9(a): Vigenere Cipher

- He now shifts each plaintext alphabet by the number written below it to create ciphertext as shown below. –

a	t	t	a	c	k	f	r	o	m	s	o	u	t	h	e	a	s	t
16	15	9	14	20	16	15	9	14	20	16	15	9	14	20	16	15	9	14
Q	I	C	O	W	A	U	A	C	G	I	D	D	H	B	U	P	B	H

Fig 1.9(b): Vigenere Cipher

- Here, each plaintext character has been shifted by a different amount – and the key determines that amount. The key must be less than or equal to the size of the message.
- For decryption, the receiver uses the same key and shifts received ciphertext in reverse order to obtain the plaintext.

Q	I	C	O	W	A	U	A	C	G	I	D	D	H	B	U	P	B	H
16	15	9	14	20	16	15	9	14	20	16	15	9	14	20	16	15	9	14
a	t	t	a	c	k	f	r	o	m	s	o	u	t	h	e	a	s	t

Fig 1.9(c): Vigenere Cipher

One-Time Pad

The circumstances are −

- The length of the keyword is the same as the length of the plaintext.
- The keyword is a randomly generated string of alphabets.
- The keyword is used only once.

One-time Pad

Let us say, and we encrypt the name “point” with a one-time pad. It is a five letter text. To break the ciphertext by brute force, you need to try all possibilities of keys and conduct computation for (26 x 26 x 26 x 26 x 26) = 26^5 = 11881376 times. That’s for a message with five alphabets. Thus, for a longer message, the computation grows exponentially with every additional alphabet. This makes it computationally impossible to break the ciphertext by brute force.

Transposition Cipher

It is another type of cipher where the order of the alphabets in the plaintext is rearranged to create the ciphertext. The actual plaintext alphabets are not replaced. An example is a ‘simple columnar transposition’ cipher where the plaintext is written horizontally with a certain alphabet width. Then the ciphertext is read vertically as shown. For example, the plaintext is **“golden statue is in an eleventh cave”** and the secret random key chosen is **“five.**" We arrange this text horizontally in the table with a number of columns equal to key value. The resulting text is shown below.

g	o	l	d	e
n	s	t	a	t
u	e	i	s	i
n	e	l	e	v
e	n	t	h	c
a	v	e		

Fig 1.10: Transposition Cipher

- The ciphertext is obtained by reading column vertically downward from first to the last column. The ciphertext is ‘gnuneaoseenvltiltedasehetivc’.
- To decrypt, the receiver prepares a similar table. The number of columns is equal to the key number.
- The number of rows is obtained by dividing the number of total ciphertext alphabets by key value and rounding of the quotient to the next integer value.
- The receiver then writes the received ciphertext vertically down and from left to right column. To obtain the text, he reads horizontally left to right and from top to bottom row.

Block Ciphers

In this scheme, the plain binary text is processed in blocks (groups) of bits at a time; i.e., a block of plaintext bits is selected, a series of operations are performed on this block to generate a block of ciphertext bits. The number of bits in a block is fixed. For example, the schemes DES and AES have block sizes of 64 and 128, respectively.

Stream Ciphers

In this scheme, the plaintext is processed one bit at a time, i.e., one bit of plaintext is taken, and a series of operations are performed on it to generate one bit of ciphertext. Technically, stream ciphers are block ciphers with a block size of one bit.

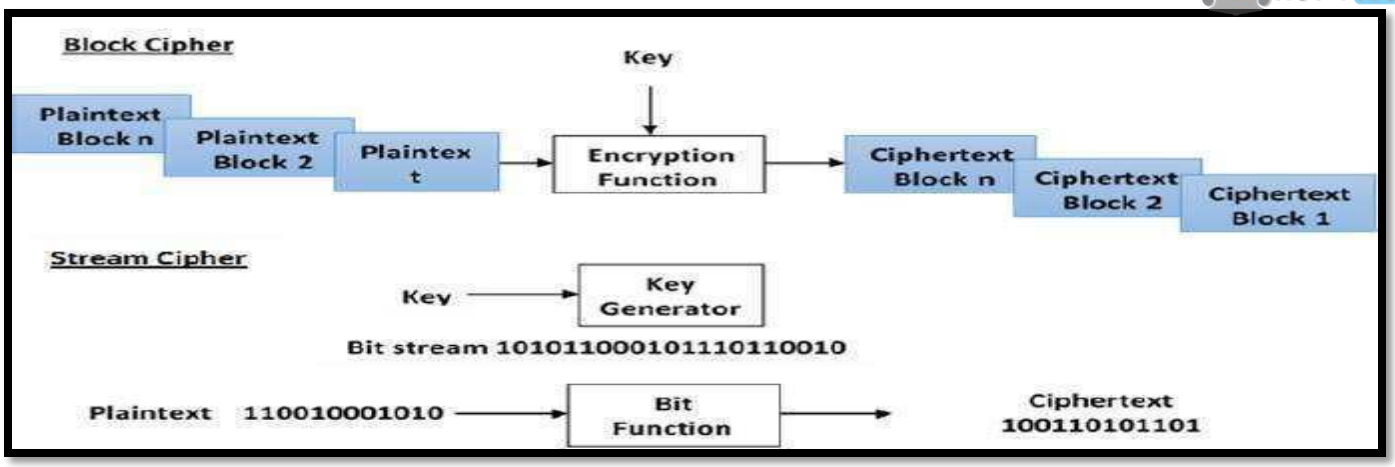

Fig 1.11: Stream Cipher

The basic scheme of a block cipher is depicted as follows −

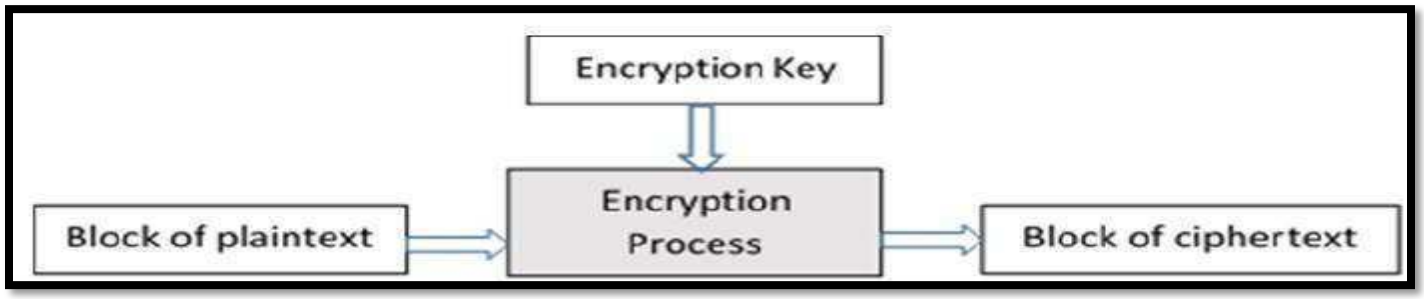

Fig 1.12: Block cipher

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

Block Size

Though any size of the block is acceptable, following aspects are borne in mind while selecting a size of a block.

- **Multiples of 8 bit** − A preferred block size is a multiple of 8 as it is easy for implementation as most computer processor handles data in multiple of 8 bits.

Padding in Block Cipher

Block ciphers process blocks of fixed sizes (say 64 bits). The length of plaintexts is mostly not a multiple of the block size. For example, a 150-bit plaintext provides two blocks of 64 bits each with the third block of balance 22 bits. The last block of bits needs to be padded up with redundant information so that the length of the final block equal to block size of the scheme. In our example, the remaining 22 bits need to have an additional 42 redundant bits added to provide a complete block. The process of adding bits to the last block is referred to as **padding**.

Block Cipher Schemes

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

- **Digital Encryption Standard (DES)** − The famous 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 cipher 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 **Rijndael** that won the AES design competition.
- **IDEA** − It is a sufficiently strong block cipher with a block size of 64 and a key size of 128 bits. Many applications use IDEA encryption, including early versions of Pretty Good Privacy (PGP) protocol. The use of IDEA scheme has a limited adoption due to patent issues.

- **Twofish** – , This scheme of a block cipher, uses a block size of 128 bits and a key of variable length. It was one of the AES finalists. It is based on the earlier block cipher Blowfish with a block size of 64 bits.
- **Serpent** – A block cipher with a block size of 128 bits and key lengths of 128, 192, or 256 bits, which was also an AES competition finalist. It is slower but has more secure design than other block ciphers.

The Data Encryption Standard (DES)

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST). DES is an implementation of a Feistel Cipher. It uses a 16 round Feistel structure. The block size is 64-bit. Though the key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only). General Structure of DES is depicted in figure 1.7

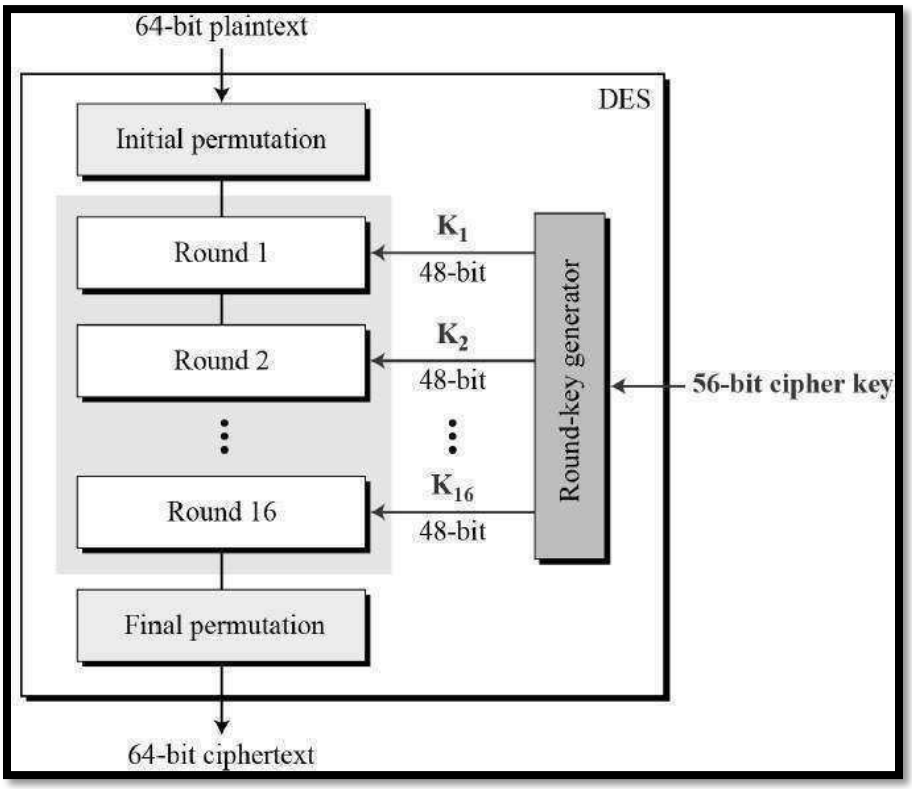

Fig: 1.13: DES Algorithm

Since DES is based on the Feistel Cipher, all that is required to specify DES is –

- Round function
- Key schedule
- Any additional processing – Initial and final permutation

Initial and Final Permutation

The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES. The initial and final permutations are shown as follows –

Round Function

The heart of this cipher is the DES function, f. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.

- **Expansion Permutation Box** – Since right input is 32-bit and the round key is a 48-bit, we first need to expand right input to 48 bits. The graphically depicted permutation logic is generally described as a table in DES specification.
- **XOR (Whitener).** – After the expansion permutation, DES does XOR operation on the expanded right section and the round key. The round key is used only in this operation.
- **Substitution Boxes.** – The S-boxes carry out the real mixing (confusion). DES uses 8 S-boxes, each with a

6-bit input and a 4-bit output.

- There are a total of eight S-box tables. The output of all eight s-boxes is then combined into32-bit section.
- **Straight Permutation** – , The 32-bit output of S-boxes, is then subjected to the straight permutation .

Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key. The process of key generation is depicted in the following illustration. –

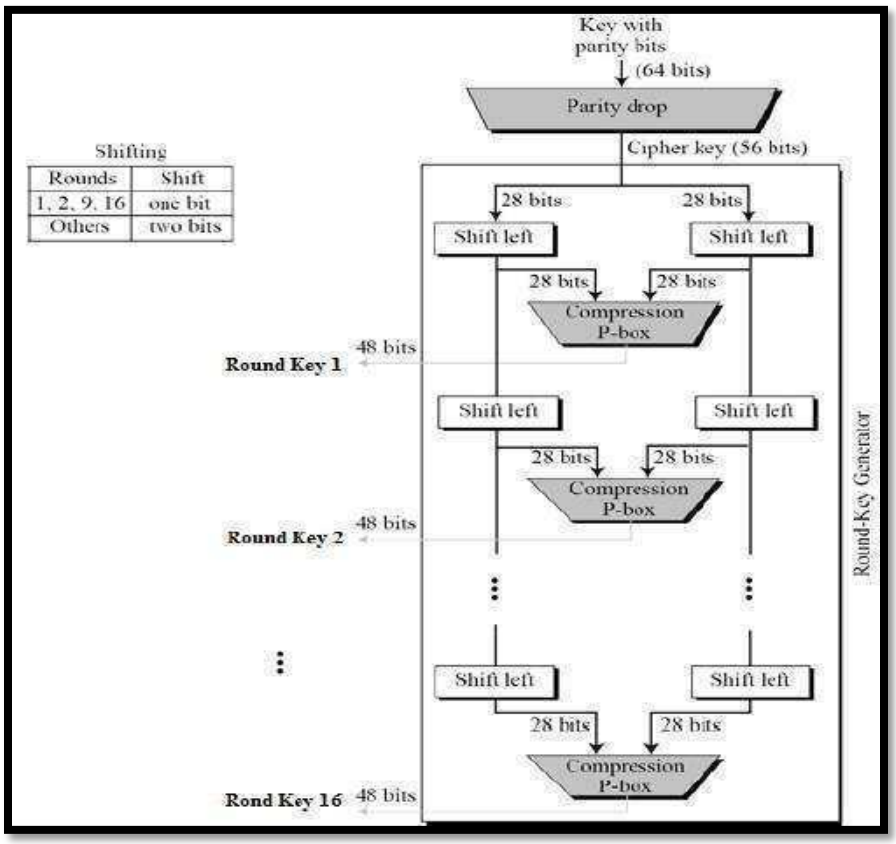

Fig: 1.14: Key Generation Algorithm

Block Cipher Modes of Operation:

A block cipher processes the data blocks of fixed size. Usually, the size of a message is larger than the block size. Hence, the long message is divided into a series of sequential message blocks, and the cipher operates on these blocks one at a time.

Electronic Code Book (ECB) Mode

This mode is the 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 the 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 ciphertexts 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 codewords in a codebook, and hence gets an official name – Electronic Codebook mode of operation (ECB). It is illustrated as follows. –

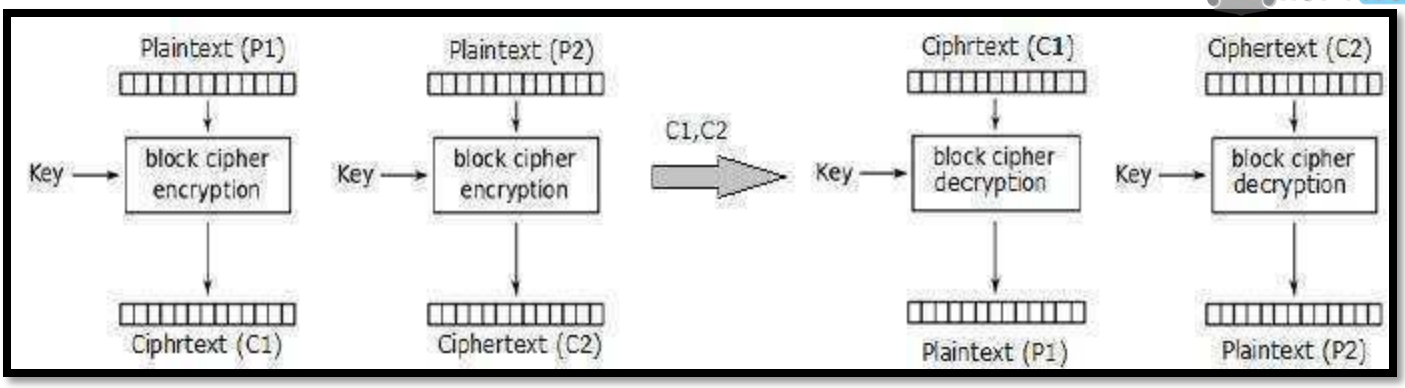

Fig: 1.15: ECB Mode

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 a data value in the top register.
- Encrypt the result of XOR operation with underlying block cipher with key K.
- Feed ciphertext block into the 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 to replace IV for decrypting next ciphertext block.

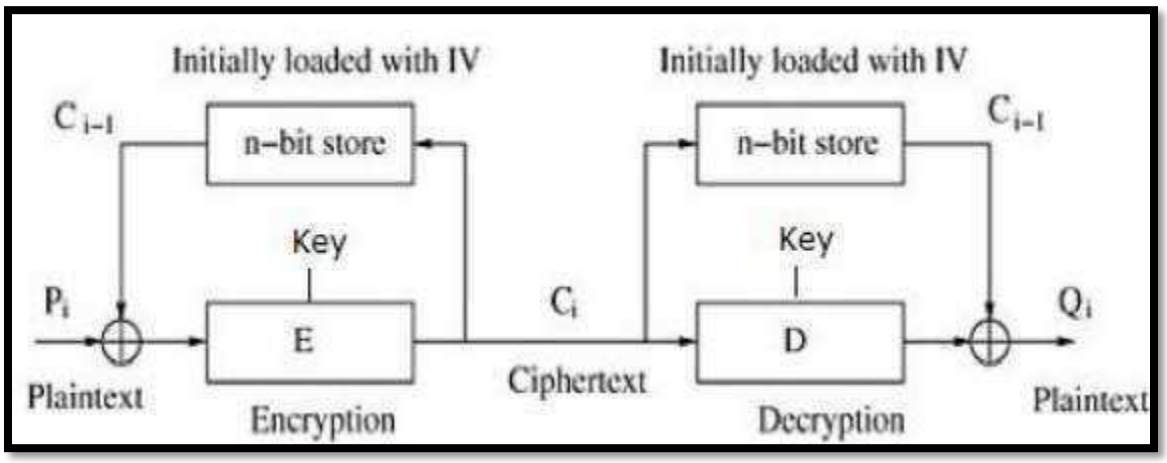

Fig: 1.16: CBC Mode

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.

- The advantage of CBC over ECB is that changing IV results in different ciphertext for an 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 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' bits where 1 < s < n. The CFB mode requires an initialisation 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 the output of the encryption process and XOR them with 's' bit plaintext message block to generate a ciphertext block.
- Feed ciphertext block into a 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.

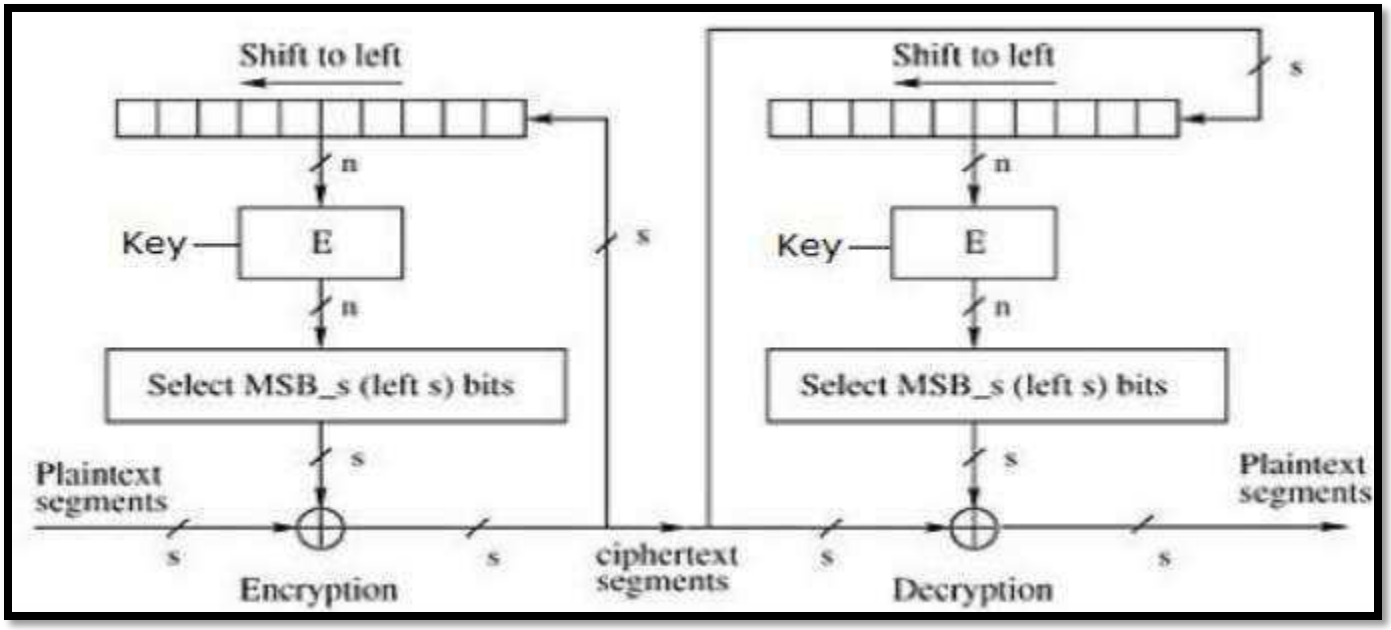

Fig: 1.17: CFB Mode

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 on the message.
- CFB has an extraordinary feature. In this mode, the user decrypts the ciphertext using only the encryption process of the block cipher. The decryption algorithm of the underlying block cipher is never used.

- 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 a 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 beneficial features of a block cipher.

Output Feedback (OFB) Mode

- It involves feeding the successive output blocks from the underlying block cipher back to it. These feedback blocks provide a string of bits to feed the encryption algorithm which acts as the key-stream generator as in case of CFB mode.
- The keystream generated is XOR-ed with the plaintext blocks. 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. −

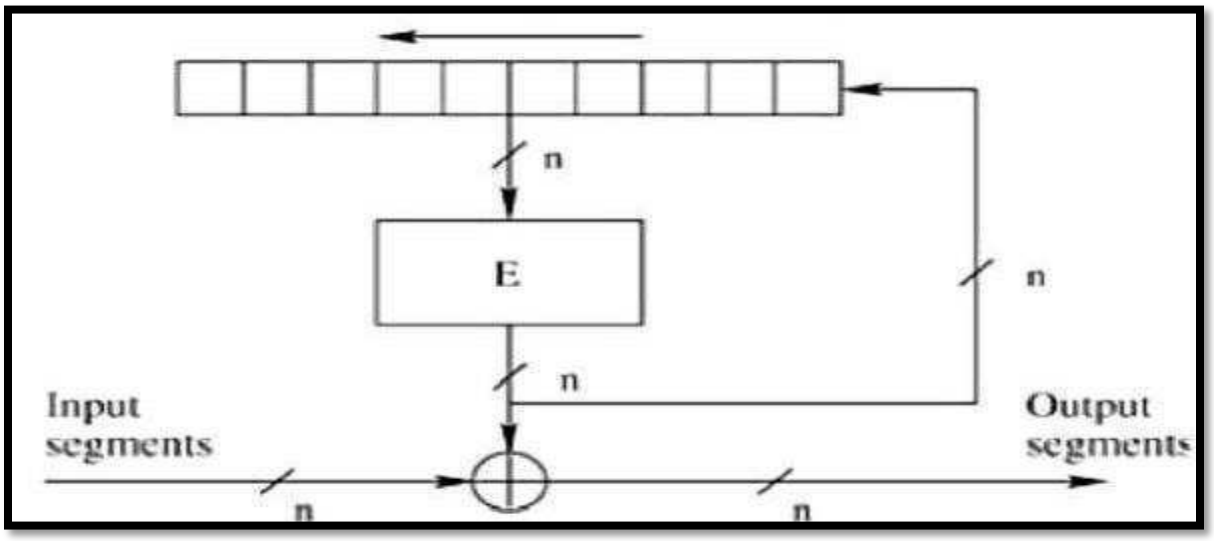

Fig: 1.18: OFB Mode

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 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 the 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 process 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, the block counter is updated as in case of encryption.

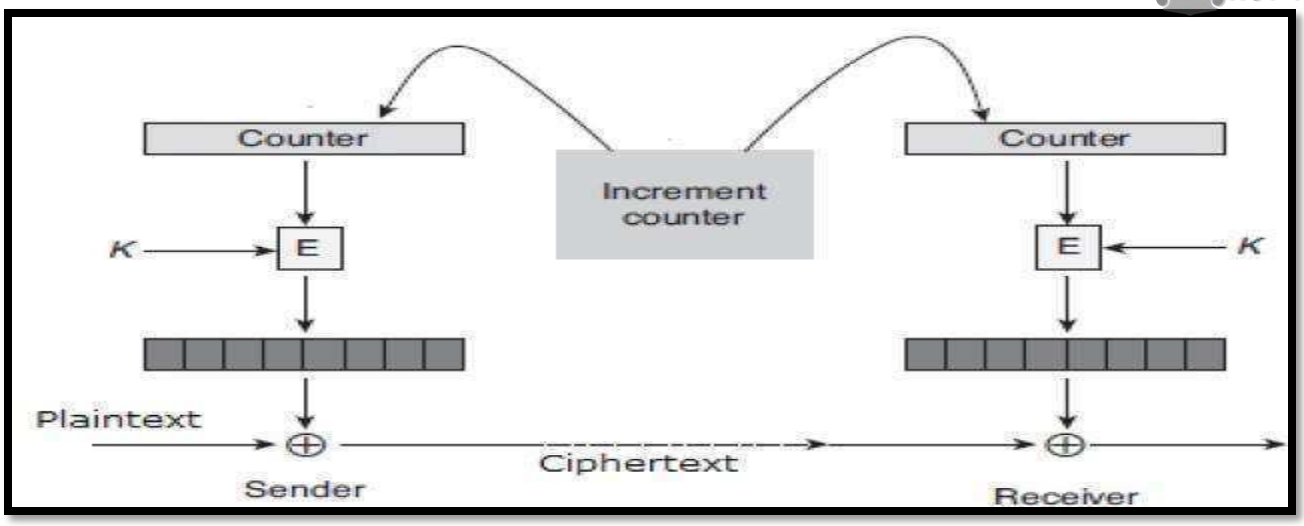

Fig: 1.19: Counter Mode

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 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 stream cipher.
- The severe disadvantage of CTR mode is that it requires asynchronous counter at sender and receiver. Loss of synchronisation leads to incorrect recovery of plaintext.

Cryptanalysis:

Cryptanalysis is the study of ciphertext, ciphers and cryptosystems with the aim of understanding how they work and finding and improving techniques for defeating or weakening them.

The objective of cryptanalysis: the goal of cryptanalysis is to find weaknesses in or otherwise defeat cryptographic algorithms, cryptanalysts' research results are used by cryptographers to improve and strengthen or replace flawed algorithms. Both cryptanalysis, which focuses on deciphering encrypted data, and cryptography, which focuses on creating and enhancing encryption ciphers and other algorithms, are aspects of cryptology, the mathematical study of codes, ciphers and related algorithms.

Cryptanalysis techniques and attacks: There are many different types of cryptanalysis attacks and procedures, which vary depending on how much information the analyst has about the ciphertext being analysed. Some cryptanalytic methods include:

- In a **ciphertext-only attack**, the attacker only has access to one or more encrypted messages but knows nothing about the plaintext data, the encryption algorithm being used or any data about the cryptographic key being used.
- In a **known plaintext attack**, the analyst may have access to some or all of the plaintext of the ciphertext; the analyst's goal, in this case, is to discover the key used to encrypt the message and decrypt the message. Once the key is discovered, an attacker can decrypt all messages that had been encrypted using that key.
- In a **chosen plaintext attack**, the analyst either knows the encryption algorithm or has access to the device used to do the encryption. The analyst can encrypt the chosen plaintext with the targeted algorithm to derive information about the key.
- A **differential cryptanalysis attack** is a type of chosen plaintext attack on block ciphers that analyses pairs of plaintexts rather than single plaintexts, so the analyst can determine how the targeted algorithm works when it encounters different types of data.

- **Integral cryptanalysis attacks** are similar to differential cryptanalysis attacks, but instead of pairs of plaintexts, it uses sets of plaintexts in which part of the plaintext is kept constant, but the rest of the plaintext is modified.
- A **dictionary attack** is a technique typically used against password files and exploits the human tendency to use passwords based on natural words or easily guessed sequences of letters or numbers. The dictionary attack works by encrypting all the words in a dictionary and then checking whether the resulting hash matches an encrypted password stored in the SAM file format or another password file.

Other types of cryptanalytic attacks can include techniques for convincing individuals to reveal their passwords or encryption keys, developing Trojan horse programs that steal secret keys from victims' computers and send them back to the cryptanalyst or tricking a victim into using a weakened cryptosystem.

Brute force attack:

- In the world of Cybercrimes, brute force attack is an activity which involves successive repetitive attempts of trying various password combinations to break into any website. This attempt is carried out vigorously by the hackers who also make use of bots they have installed maliciously in other computers to boost the computing power required to run such type of attacks.
- A Brute Force Attack is the simplest method to gain access to a site or server (or anything that is password protected). It tries various combinations of usernames and passwords again and again until it gets in. This repetitive action is like an army attacking a fort.

Security Goal:

Confidentiality, integrity, and availability, also known as the **CIA triad**, is a model designed to guide policies for information security within an organisation. The model is also sometimes referred to as the AIC triad (availability, integrity, and confidentiality) to avoid confusion with the Central Intelligence Agency.

Confidentiality:

- Confidentiality is roughly equivalent to privacy. Measures were undertaken to ensure confidentiality is designed to prevent sensitive information from reaching the wrong people while making sure that the right people can get it: Access must be restricted to those authorised to view the data
- An excellent example of methods used to ensure confidentiality is an account number or routing number when banking online.

Integrity:

- Integrity involves maintaining the consistency, accuracy, and trustworthiness of data over its entire lifecycle.
- Data must not be changed in transit, and steps must be taken to ensure that data cannot be altered by unauthorised people (for example, in a breach of confidentiality).

Availability:

- Availability is a guarantee of reliable access to the information by authorized people
- Availability is best ensured by rigorously maintaining all hardware, performing hardware repairs immediately when needed and maintaining a correctly functioning operating system environment that is free of software conflicts.
- It's also important to keep current with all necessary system upgrades.
- To prevent data loss from such occurrences, a backup copy may be stored in a geographically-isolated location, perhaps even in a fireproof, waterproof safe.
- Extra security equipment or software such as firewalls and proxy servers can guard against downtime and unreachable data due to malicious actions such as denial-of-service (DoS) attacks and network intrusions.

RGPVNOTES.IN

We hope you find these notes useful.

You can get previous year question papers at https://qp.rgpvnotes.in .

If you have any queries or you want to submit your study notes please write us at rgpvnotes.in@gmail.com

LIKE & FOLLOW US ON FACEBOOK
facebook.com/rgpvnotes.in