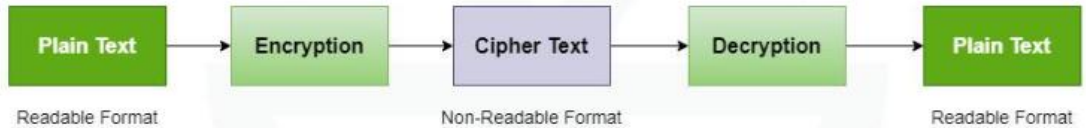
CRYPTOGRAPHY

UNIT - I

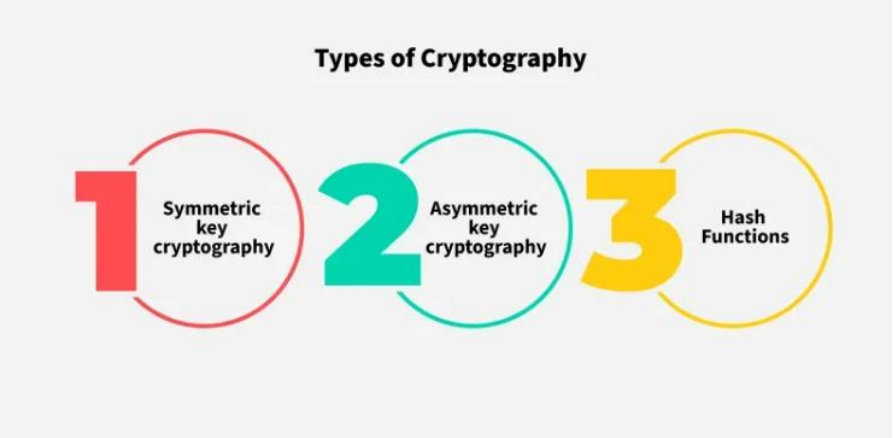
Cryptography is a technique of securing information and communications through the use of codes so that only those persons for whom the information is intended can understand and process it. Thus, preventing unauthorized access to information. The prefix “crypt” means “hidden” and the suffix “graphy” means “writing”. In Cryptography, the techniques that are used to protect information are obtained from mathematical concepts and a set of rule-based calculations known as algorithms to convert messages in ways that make it hard to decode them. These algorithms are used for cryptographic key generation, digital signing, and verification to protect data privacy, web browsing on the internet and to protect confidential transactions such as credit card and debit card transactions.



Features Of Cryptography

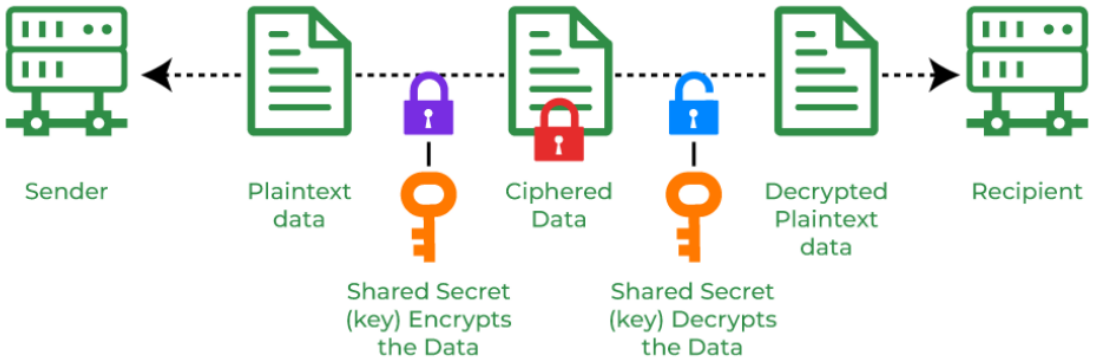
- **Confidentiality:** Information can only be accessed by the person for whom it is intended and no other person except him can access it.
- **Integrity:** Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.
- **Non-repudiation:** The creator/sender of information cannot deny his intention to send information at a later stage.
- **Authentication:** The identities of the sender and receiver are confirmed. As well destination/origin of the information is confirmed.
- **Interoperability:** Cryptography allows for secure communication between different systems and platforms.
- **Adaptability:** Cryptography continuously evolves to stay ahead of security threats and technological advancements.

Types Of Cryptography



1. Symmetric Key Cryptography

It is an encryption system where the sender and receiver of a message use a single common key to encrypt and decrypt messages. Symmetric Key cryptography is faster and simpler but the problem is that the sender and receiver have to somehow exchange keys securely. The most popular symmetric key cryptography systems are Data Encryption Systems (DES) and Advanced Encryption Systems (AES) .



Symmetric Key Cryptography

2. Hash Functions

There is no usage of any key in this algorithm. A hash value with a fixed length is calculated as per the plain text which makes it impossible for the contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

3. Asymmetric Key Cryptography

In Asymmetric Key Cryptography, a pair of keys is used to encrypt and decrypt information. A sender's public key is used for encryption and a receiver's private key is used for decryption. Public keys and Private keys are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone knows his private key. The most popular asymmetric key cryptography algorithm is the RSA algorithm.

Applications of Cryptography

- **Computer passwords:** Cryptography is widely utilized in computer security, particularly when creating and maintaining passwords. When a user logs in, their password is hashed and compared to the hash that was previously stored. Passwords are hashed and encrypted before being stored. In this technique, the passwords are encrypted so that even if a hacker gains access to the password database, they cannot read the passwords.
- **Digital Currencies:** To protect transactions and prevent fraud, digital currencies like Bitcoin also use cryptography. Complex algorithms and cryptographic keys are used to safeguard transactions, making it nearly hard to tamper with or forge the transactions.
- **Secure web browsing:** Online browsing security is provided by the use of cryptography, which shields users from eavesdropping and man-in-the-middle assaults. Public key cryptography is used by the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols to encrypt data sent between the web server and the client, establishing a secure channel for communication.
- **Electronic Signatures:** Electronic signatures serve as the digital equivalent of a handwritten signature and are used to sign documents. Digital signatures are created using cryptography and can be validated using public key cryptography. In many nations, electronic signatures are enforceable by law, and their use is expanding quickly.
- **Authentication:** Cryptography is used for authentication in many different situations, such as when accessing a bank account, logging into a computer, or using a secure network. Cryptographic methods are employed by authentication protocols to confirm the user's identity and confirm that they have the required access rights to the resource.
- **Cryptocurrencies:** Cryptography is heavily used by cryptocurrencies like Bitcoin and Ethereum to protect transactions, thwart fraud, and maintain the network's integrity.

Complex algorithms and cryptographic keys are used to safeguard transactions, making it nearly hard to tamper with or forge the transactions.

- **End-to-end Internet Encryption:** End-to-end encryption is used to protect two-way communications like video conversations, instant messages, and email. Even if the message is encrypted, it assures that only the intended receivers can read the message. End-to-end encryption is widely used in communication apps like WhatsApp and Signal, and it provides a high level of security and privacy for users.

Types of Cryptography Algorithm

- **Advanced Encryption Standard (AES):** AES (Advanced Encryption Standard) is a popular encryption algorithm which uses the same key for encryption and decryption It is a symmetric block cipher algorithm with block size of 128 bits, 192 bits or 256 bits. AES algorithm is widely regarded as the replacement of DES (Data encryption standard) algorithm.
- **Data Encryption Standard (DES):** DES (Data encryption standard) is an older encryption algorithm that is used to convert 64-bit plaintext data into 48-bit encrypted ciphertext. It uses symmetric keys (which means same key for encryption and decryption). It is kind of old by today's standard but can be used as a basic building block for learning newer encryption algorithms.
- **RSA:** RSA is an basic asymmetric cryptographic algorithm which uses two different keys for encryption. The RSA algorithm works on a block cipher concept that converts plain text into cipher text and vice versa.
- **Secure Hash Algorithm (SHA): SHA** is used to generate unique fixed-length digital fingerprints of input data known as hashes**.** SHA variations such as **SHA-2** and **SHA-3** are commonly used to ensure data integrity and authenticity. The tiniest change in input data drastically modifies the hash output, indicating a loss of integrity. Hashing is the process of storing key value pairs with the help of a hash function into a hash table.

Advantages of Cryptography

- Cryptography can be used for access control to ensure that only parties with the proper permissions have access to a resource.
- For secure online communication, it offers secure mechanisms for transmitting private information like passwords, bank account numbers, and other sensitive data over the Internet.
- It helps in the defence against various types of assaults including replay and man-in-the-middle attacks.

- Cryptography can help firms in meeting a variety of legal requirements including data protection and privacy legislation.

Fundamental Network Security Principles

Confidentiality

The degree of confidentiality determines the secrecy of the information. The principle specifies that only the sender and receiver will be able to access the information shared between them. Confidentiality compromises if an unauthorized person is able to access a message.

For Example, let us consider sender A wants to share some confidential information with receiver B and the information gets intercepted by the attacker C. Now the confidential information is in the hands of an intruder C.

Authentication

Authentication is the mechanism to identify the user or system or the entity. It ensures the identity of the person trying to access the information. The authentication is mostly secured by using username and password. The authorized person whose identity is preregistered can prove his/her identity and can access the sensitive information.

Integrity

Integrity gives the assurance that the information received is exact and accurate. If the content of the message is changed after the sender sends it but before reaching the intended receiver, then it is said that the integrity of the message is lost.

- **System Integrity:** System Integrity assures that a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorized manipulation of the system.
- **Data Integrity:** Data Integrity assures that information (both stored and in transmitted packets) and programs are changed only in a specified and authorized manner.

Non-Repudiation

Non-repudiation is a mechanism that prevents the denial of the message content sent through a network. In some cases the sender sends the message and later denies it. But the non-repudiation does not allow the sender to refuse the receiver.

Access Control

The principle of access control is determined by role management and rule management. Role management determines who should access the data while rule management

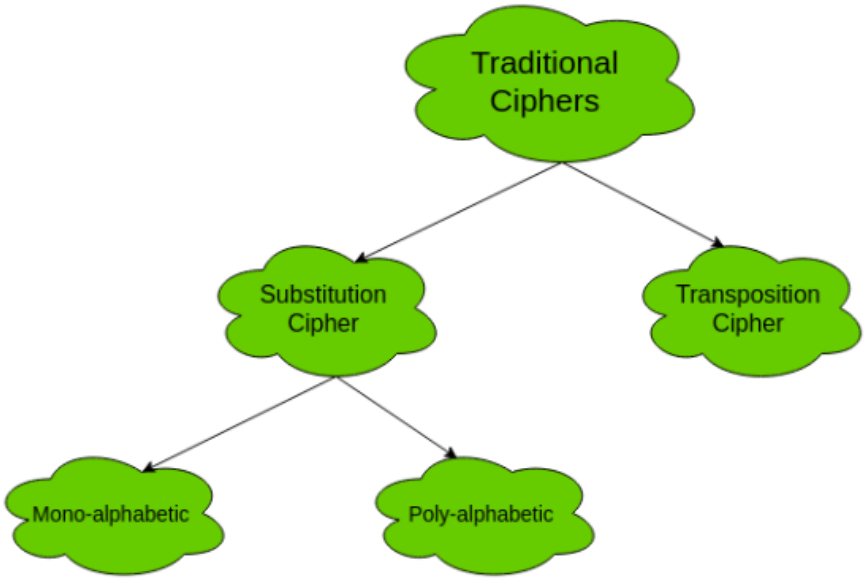
determines up to what extent one can access the data. The information displayed is dependent on the person who is accessing it.

Availability

The principle of availability states that the resources will be available to authorize party at all times. Information will not be useful if it is not available to be accessed. Systems should have sufficient availability of information to satisfy the user request.

Traditional Symmetric Ciphers

The two types of traditional symmetric ciphers are **Substitution Cipher** and **Transposition Cipher**. The following flowchart categories the traditional ciphers:



1. Substitution Cipher:
Substitution Ciphers are further divided into **Mono-alphabetic Cipher** and **Poly-alphabetic Cipher**.

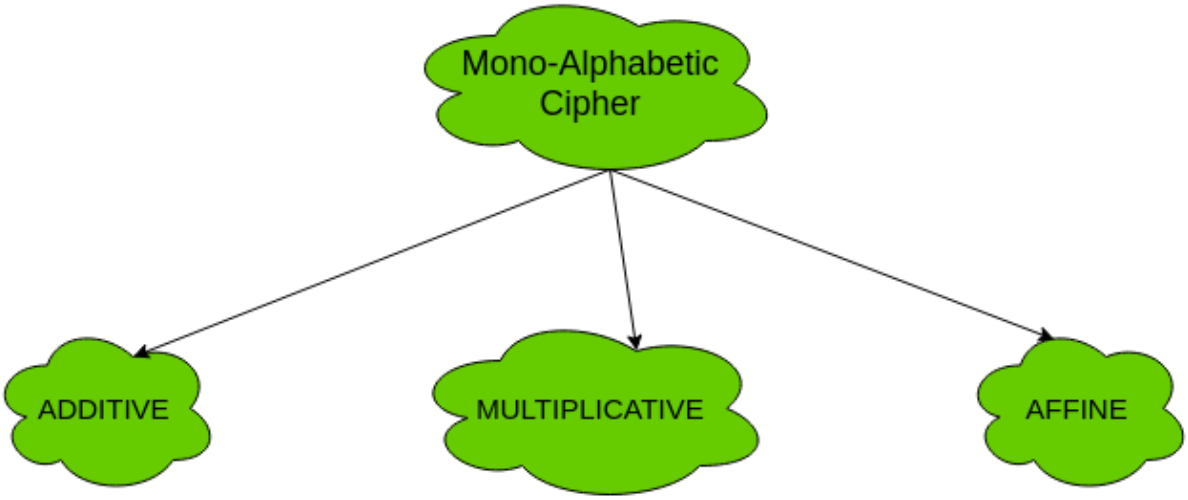
First, let's study about mono-alphabetic cipher.

1. **Mono-alphabetic Cipher –**
   In mono-alphabetic ciphers, each symbol in plain-text (eg; 'o' in 'follow') is mapped to one cipher-text symbol. No matter how many times a symbol occurs in the plain-text, it will correspond to the same cipher-text symbol. For example, if the plain-text is 'follow' and the mapping is :
   - f -> g
   - o -> p

- l -> m
- w -> x

The cipher-text is ‘gpmmpx’.

Types of mono-alphabetic ciphers are:



1. **(a). Additive Cipher (Shift Cipher / Caesar Cipher) –**
   The simplest mono-alphabetic cipher is additive cipher. It is also referred to as ‘Shift Cipher’ or ‘Caesar Cipher’. As the name suggests, ‘addition modulus 2’ operation is performed on the plain-text to obtain a cipher-text.

C = (M + k) mod n
M = (C – k) mod n

where,
C -> cipher-text
M -> message/plain-text
k -> key

The key space is 26. Thus, it is not very secure. It can be broken by brute-force attack.

Caesar Cipher in Cryptography

The **Caesar Cipher** is one of the simplest and oldest methods of encrypting messages, named after Julius Caesar, who reportedly used it to protect his military communications. This technique involves shifting the letters of the alphabet by a fixed number of places. For example, with a shift of three, the letter ‘A’ becomes ‘D’, ‘B’ becomes ‘E’, and so on. Despite its simplicity, the Caesar Cipher formed the groundwork for modern cryptographic

techniques. In this article, we'll explore how the Caesar Cipher works, its significance, and its impact on the development of cryptography with its advantages and disadvantages.

What is Caesar Cipher Technique?

The Caesar cipher is a simple encryption technique that was used by Julius Caesar to send secret messages to his allies. It works by shifting the letters in the plaintext message by a certain number of positions, known as the "shift" or "key". The Caesar Cipher technique is one of the earliest and simplest methods of encryption techniques.

It's simply a type of substitution cipher, i.e., each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet. For example with a shift of 1, A would be replaced by B, B would become C, and so on. The method is apparently named after Julius Caesar, who apparently used it to communicate with his officials.

Cryptography Algorithm For the Caesar Cipher

- Thus to cipher a given text we need an integer value, known as a shift which indicates the number of positions each letter of the text has been moved down.
  The encryption can be represented using modular arithmetic by first transforming the letters into numbers, according to the scheme, A = 0, B = 1,…, Z = 25. Encryption of a letter by a shift n can be described mathematically as.
- For example, if the shift is 3, then the letter A would be replaced by the letter D, B would become E, C would become F, and so on. The alphabet is wrapped around so that after Z, it starts back at A.
- Here is an example of how to use the Caesar cipher to encrypt the message "HELLO" with a shift of 3:

1. Write down the plaintext message: HELLO
2. Choose a shift value. In this case, we will use a shift of 3.
3. Replace each letter in the plaintext message with the letter that is three positions to the right in the alphabet.

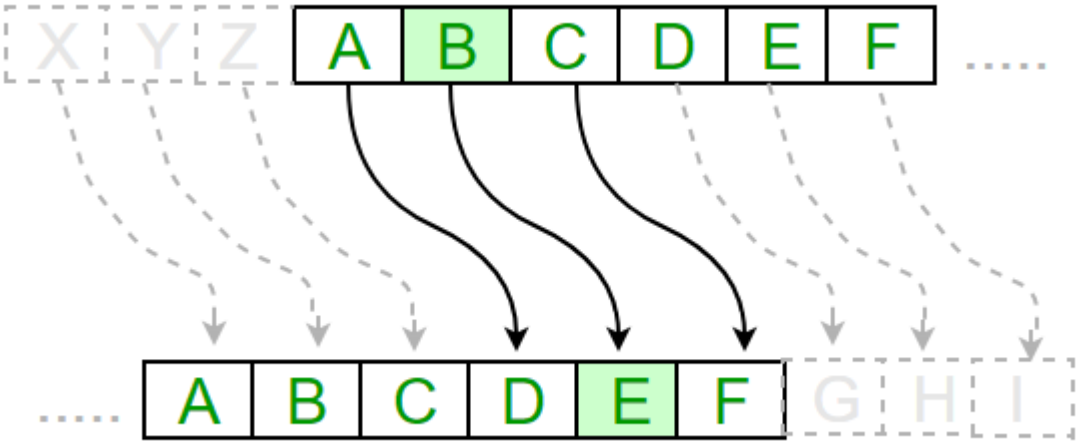
H becomes K (shift 3 from H)
E becomes H (shift 3 from E)
L becomes O (shift 3 from L)
L becomes O (shift 3 from L)
O becomes R (shift 3 from O)

4.The encrypted message is now "KHOOR".

- To decrypt the message, you simply need to shift each letter back by the same number of positions. In this case, you would shift each letter in "KHOOR" back by 3 positions to get the original message, "HELLO".

En(x)=(x+n)mod 26 $En(x)=(x+n) mod\ 26$
(Encryption Phase with shift n)

Dn(x)=(x−n)mod 26 $Dn(x)=(x-n) mod\ 26$
(Decryption Phase with shift n)



Examples :

Text : ABCDEFGHIJKLMNOPQRSTUVWXYZ
Shift: 23
Cipher: XYZABCDEFGHIJKLMNOPQRSTUVW

Text : ATTACKATONCE
Shift: 4
Cipher: EXXEGOEXSRGI

Advantages

- Easy to implement and use thus, making suitable for beginners to learn about encryption.
- Can be physically implemented, such as with a set of rotating disks or a set of cards, known as a scytale, which can be useful in certain situations.
- Requires only a small set of pre-shared information.
- Can be modified easily to create a more secure variant, such as by using a multiple shift values or keywords.

Disadvantages

- It is not secure against modern decryption methods.

- Vulnerable to known-plaintext attacks, where an attacker has access to both the encrypted and unencrypted versions of the same messages.
- The small number of possible keys means that an attacker can easily try all possible keys until the correct one is found, making it vulnerable to a brute force attack.
- It is not suitable for long text encryption as it would be easy to crack.
- It is not suitable for secure communication as it is easily broken.
- Does not provide confidentiality, integrity, and authenticity in a message.

Features of Caesar Cipher

1. **Substitution cipher:** The Caesar cipher is a type of substitution cipher, where each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet.
2. **Fixed key:** The Caesar cipher uses a fixed key, which is the number of positions by which the letters are shifted. This key is known to both the sender and the receiver.
3. **Symmetric encryption:** The Caesar cipher is a symmetric encryption technique, meaning that the same key is used for both encryption and decryption.
4. **Limited keyspace:** The Caesar cipher has a very limited keyspace of only 26 possible keys, as there are only 26 letters in the English alphabet.
5. **Vulnerable to brute force attacks:** The Caesar cipher is vulnerable to brute force attacks, as there are only 26 possible keys to try.
6. **Easy to implement:** The Caesar cipher is very easy to implement and requires only simple arithmetic operations, making it a popular choice for simple encryption tasks.

Rules for the Caesar Cipher

1. Choose a number between 1 and 25. This will be your “shift” value.
2. Write down the letters of the alphabet in order, from A to Z.
3. Shift each letter of the alphabet by the “shift” value. For example, if the shift value is 3, A would become D, B would become E, C would become F, and so on.
4. Encrypt your message by replacing each letter with the corresponding shifted letter. For example, if the shift value is 3, the word “hello” would become “khoor”.
5. To decrypt the message, simply reverse the process by shifting each letter back by the same amount. For example, if the shift value is 3, the encrypted message “khoor” would become “hello”.

Algorithm for Caesar Cipher

Input:

1. Choose a shift value between 1 and 25.
2. Write down the alphabet in order from A to Z.
3. Create a new alphabet by shifting each letter of the original alphabet by the shift value. For example, if the shift value is 3, the new alphabet would be:
4. 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
   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
5. Replace each letter of the message with the corresponding letter from the new alphabet. For example, if the shift value is 3, the word “hello” would become “khoor”.
6. To decrypt the message, shift each letter back by the same amount. For example, if the shift value is 3, the encrypted message “khoor” would become “hello”.

Procedure:

- Traverse the given text one character at a time .
- For each character, transform the given character as per the rule, depending on whether we’re encrypting or decrypting the text.
- Return the new string generated.

One Time Password (OTP) algorithm in Cryptography

Authentication, the process of identifying and validating an individual is the rudimentary step before granting access to any protected service (such as a personal account). Authentication has been built into the cyber security standards and offers to prevent unauthorized access to safeguarded resources. Authentication mechanisms today create a double layer gateway prior to unlocking any protected information. This double layer of security, termed as two factor authentication, creates a pathway that requires validation of credentials (username/email and password) followed by creation and validation of the **One Time Password (OTP)**. The OTP is a numeric code that is randomly and uniquely generated during each authentication event. This adds an additional layer of security, as the password generated is fresh set of digits each time an authentication is attempted and it offers the quality of being unpredictable for the next created session. The two main methods for delivery of the OTP is:

1. **SMS Based:** This is quite straightforward. It is the standard procedure for delivering the OTP via a text message after regular authentication is successful. Here, the OTP is

generated on the server side and delivered to the authenticator via text message. It is the most common method of OTP delivery that is encountered across services.

2. **Application Based:** This method of OTP generation is done on the user side using a specific smartphone application that scans a QR code on the screen. The application is responsible for the unique OTP digits. This reduces wait time for the OTP as well as reduces security risk as compared to the SMS based delivery.

The most common way for the generation of OTP defined by The Initiative For Open Authentication (OATH) is the **Time Based One Time Passwords (TOTP)**, which is a Time Synchronized OTP. In these OTP systems, time is the cardinal factor to generate the unique password. The password generated is created using the current time and it also factors in a secret key. An example of this OTP generation is the Time Based OTP Algorithm (TOTP) described as follows:

1. Backend server generates the secret key
2. The server shares secret key with the service generating the OTP
3. A hash based message authentication code (HMAC) is generated using the obtained secret key and time. This is done using the cryptographic SHA-1 algorithm. Since both the server and the device requesting the OTP, have access to time, which is obviously dynamic, it is taken as a parameter in the algorithm. Here, the Unix timestamp is considered which is independent of time zone i.e. time is calculated in seconds starting from January First 1970. Let us consider "0215a7d8c15b492e21116482b6d34fc4e1a9f6ba" as the generated string from the HMAC-SHA1 algorithm.
4. The code generated is 20 bytes long and is thus truncated to the desired length suitable for the user to enter. Here dynamic truncation is used. For the 20-byte code "0215a7d8c15b492e21116482b6d34fc4e1a9f6ba", each character occupies 4 bits. The entire string is taken as 20 individual one byte string.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
02	15	a7	d8	c1	5b	49	2e	21	11	64	82	b6	d3	4f	c4	e1	a9	f6	ba

We look at the last character, here a. The decimal value of which is taken to determine the offset from which to begin truncation. Starting from the offset value, 10 the next 31 bits are read to obtain the string "6482b6d3". The last thing left to do, is to take our hexadecimal numerical value, and convert it to decimal, which gives 1686288083. All we need now are the last desired length of OTP digits of the obtained decimal string, zero-padded if necessary. This is easily accomplished by taking the decimal string, modulo 10 ^ number of digits required in OTP. We end up with "288083" as our TOTP code.

5. A counter is used to keep track of the time elapsed and generate a new code after a set interval of time
6. OTP generated is delivered to user by the methods described above.

Apart from the time-based method described above, there also exist certain mathematical algorithms for OTP generation for example a one-way function that creates a subsequent OTP from the previously created OTP. The two factor authentication system is an effective strategy that exploits the authentication principles of “something that you know” and “something that you have”.The dynamic nature of the latter principle implemented by the One Time Password Algorithm is crucial to security and offers an effective layer of protection against malicious attackers. The unpredictability of the OTP presents a hindrance in peeling off the layers that this method of cryptography has to offer.

(b). Multiplicative Cipher –
The multiplicative cipher is similar to additive cipher except the fact that the key bit is multiplied to the plain-text symbol during encryption. Likewise, the cipher-text is multiplied by the multiplicative inverse of key for decryption to obtain back the plain-text.

$C = (M * k) \bmod n$
$M = (C * k^{-1}) \bmod n$

where,
k^{-1} -> multiplicative inverse of k (key)

The key space of multiplicative cipher is 12. Thus, it is also not very secure.

(c). Affine Cipher –
The affine cipher is a combination of additive cipher and multiplicative cipher. The key space is 26 * 12 (key space of additive * key space of multiplicative) i.e. 312. It is relatively secure than the above two as the key space is larger.
Here two keys k_1 and k_2 are used.

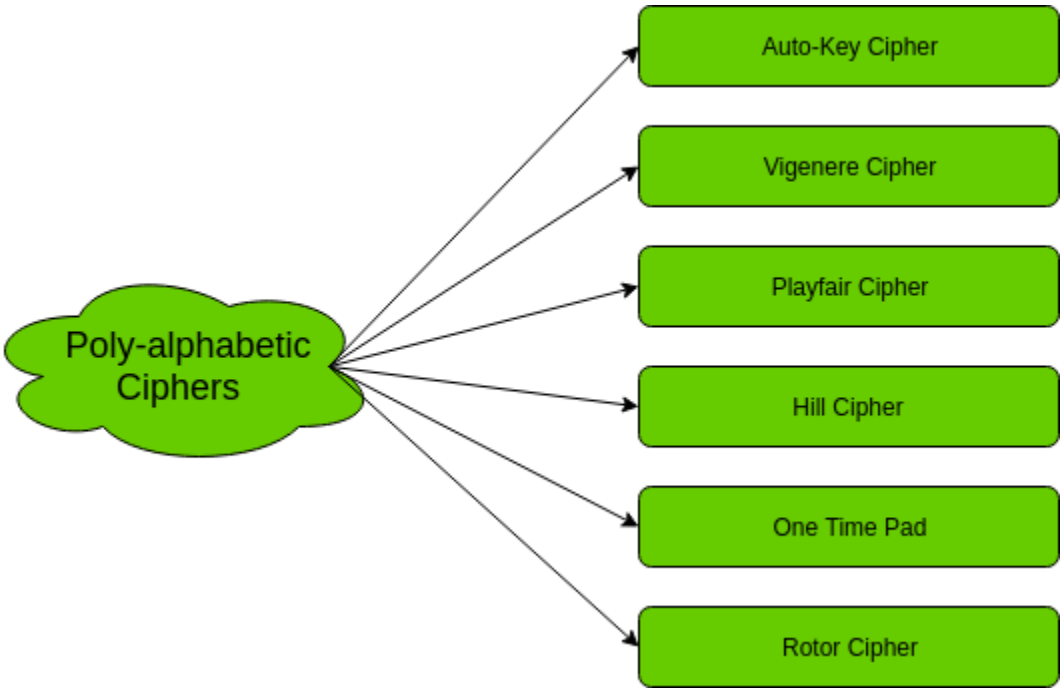
$C = [(M * k_1) + k_2] \bmod n$
$M = [(C – k_2) * k_1^{-1}] \bmod n$

2. **Poly-alphabetic Cipher –**
   In poly-alphabetic ciphers, every symbol in plain-text is mapped to a different cipher-text symbol regardless of its occurrence. Every different occurrence of a symbol has different mapping to a cipher-text. For example, in the plain-text ‘follow’, the mapping is :

f -> q
o -> w
l -> e
l -> r
o -> t
w -> y

Thus, the cipher text is ‘qwerty’.

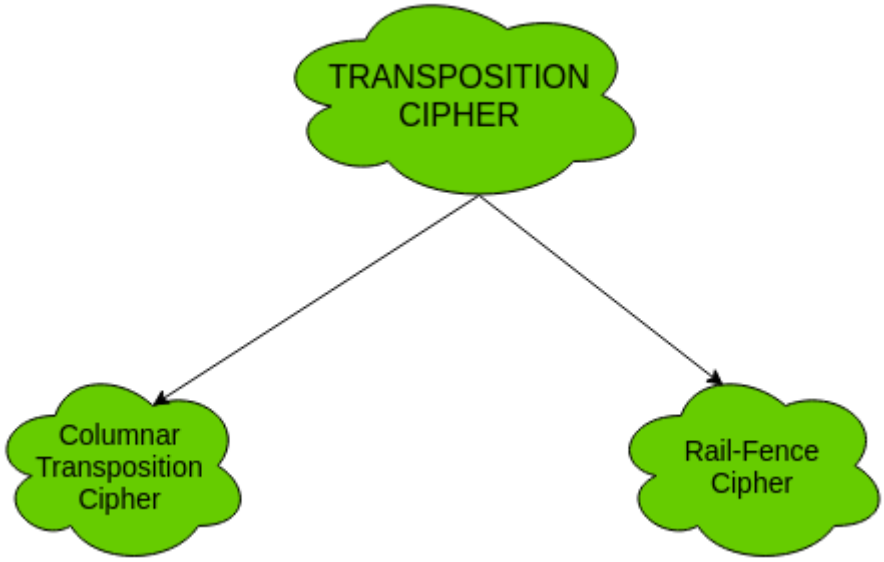
Types of poly-alphabetic ciphers are:



2. Transposition Cipher:

The transposition cipher does not deal with substitution of one symbol with another. It focuses on changing the position of the symbol in the plain-text. A symbol in the first position in plain-text may occur in fifth position in cipher-text.

Two of the transposition ciphers are:



1. **Columnar Transposition Cipher –**
   One type of transposition cipher that represents plaintext in matrix form is called the Columnar Transposition Cipher. Writing the plaintext out in rows and reading the ciphertext out one column at a time is known as columnar transposition. In this tutorial, we have described the columnar transposition cipher's encryption and decryption methods. Probably the most researched transposition cipher is columnar transposition.

h	e	l	l
o	w	o	r
l	d		

How it Works?

The message is structured as a 2-dimensional array. The length of the message determines how many rows and columns there will be. If the message is 30 characters long (including spaces), there is a 50% chance that there will be 15 rows and 2 columns, 10 rows, 3 rows, 5 rows, or 6 rows.

Keep in mind that we have to append a dummy letter at the end of the message if its length exceeds 29.

Encryption

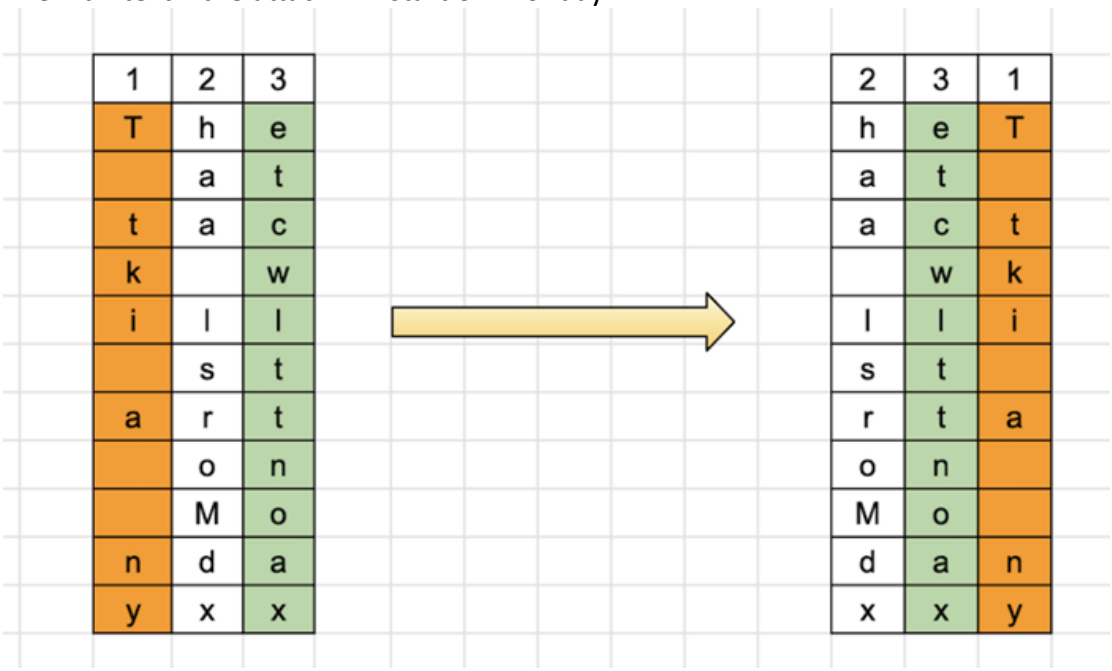
See the encryption process of Columnar transposition cipher below −

- First, the plaintext is written in the rows that are defined in advance, with the key determining the length.
- The order in which the plaintext's columns are transposed can be determined using the key.

- The ciphertext is then created by reading the transposed plaintext column by column.
  Decryption
  So the decryption process of the columnar transposition cipher is as follows −
- Using the same key that was used for encryption, the ciphertext is first transposed by columns.
- The plaintext can be retrieved by reading the transposed ciphertext row by row.

Example of Columnar Transposition

- If the message says, "The attack will start on Monday," so as we can see that it is 28 characters long. However, if we add the dummy letters "x" and "x" at the end, the message will be 30 characters long. We can figure out 30 = 10 X 3 and In the case that (2,3,1) is the key, the columns are arranged as follows −
- The Plaintext: "the attack will start on Monday"



The Ciphertext − "HAA LSROMDXETCWLTTNOAXT TKI A NY" is the Ciphertext, which is calculated from the reading on the table by columns. We rearrange the letters of a keyword, like "TWO," in an alphabetical order to make the key easier to recall. Thus, the array columns will be rearranged using the key (2,3,1).

2. **Rail-Fence Cipher –**
   A basic type of transposition cipher is the rail fence method. It is a kind of cryptographic process where the letters in a message are rearranged to form a new, seemingly unrelated message. The name of the approach comes from the message we write. When a text is created using the rail fence approach, the outcome is a zigzag pattern where each letter is spelled out before going on to the next row.

The message has to be written in the first row of a table in order to be encrypted using the rail fence approach. In addition, the second letter of the message needs to be written in the second row. This procedure must be continued until all of the message's letters have been written. Finally, we read the database row-wise to create the encrypted message.

How Rail Fence Cipher Work?

This section will give a detailed explanation of the encryption and decryption processes used by the rail fence cipher.

Encryption

In order to decrypt a message using the rail fence cipher, we should first choose the number of rails, write the message diagonally in a zigzag pattern using that number, and then combine the letters along each rail from left to right. We will walk through each step with an example below.

Let us start by considering "RAILFENCE" as a plaintext. Let us now assume that there are three rails or fences, which is also known as a key. The zigzag pattern's height will be determined by the key. The message can then be written diagonally, from left to right, in a zigzag pattern −

R				F				E
	A		L		E		C	
		I				N		

In order to create the ciphertext we will merge distinct rows, which in this case is "RFEALECIN."

Decryption

The number of rows and columns in the cipher text needs to be determined before we can start the decryption process. The length of the ciphertext is equal to the number of columns. After that, we need to determine how many rows-which function as the key-were encrypted.

Now that we know how many rows and columns there are, we can build the table and figure out where the letters should go because the rail fence cipher zigzags to encrypt the text diagonally from left to right −

encrypt the text diagonally from left to right −

*				*				*
	*		*		*		*	
		*				*		

The points where letters from the ciphertext are inserted to create the plaintext are indicated by the *(asterisk). Beginning from the top row, which is the first "rail," we fill in the letters going left to right. Up until all of the asterisk spots are filled with letters from the ciphertext, we then carry on with this pattern on the following rail and so on −

R				F				E
	*		*		*		*	
		*				*		

Let us finish the table above −

R				F				E
	A		L		E		C	
		I				N		

Finally, we are able to combine the characters from left to right and top to bottom to get the Plaintext, "RAILFENCE."

Symmetric Cipher Model

Symmetric Encryption is the most basic and old method of encryption. It uses only one key for the process of both the encryption and decryption of data. Thus, it is also known as Single-Key Encryption.

A few basic terms in Cryptography are as follows:

Plain Text: *original message to be communicated between sender and receiver*

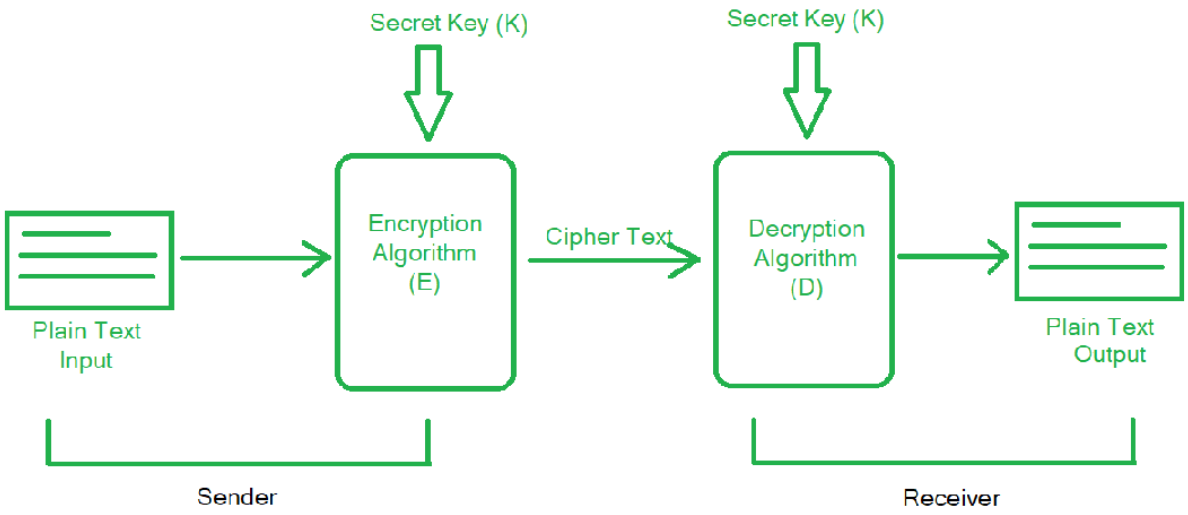
Cipher Text: *encoded format of the original message that cannot be understood by humans*

Encryption (or Enciphering): *the conversion of plain text to cipher text*

Decryption (or Deciphering): *the conversion of cipher text to plain text, i.e., reverse of encryption*

The Symmetric Cipher Model:

A symmetric cipher model is composed of five essential parts:



1. Plain Text (x): This is the original data/message that is to be communicated to the receiver by the sender. It is one of the inputs to the encryption algorithm.

2. Secret Key (k): It is a value/string/text file used by the encryption and decryption algorithm to encode and decode the plain text to cipher text and vice-versa respectively. It is independent of the encryption algorithm. It governs all the conversions in plain text. All the substitutions and transformations done depend on the secret key.

3. Encryption Algorithm (E): It takes the plain text and the secret key as inputs and produces Cipher Text as output. It implies several techniques such as substitutions and transformations on the plain text using the secret key.

$E(x, k) = y$

4. Cipher Text (y): It is the formatted form of the plain text (x) which is unreadable for humans, hence providing encryption during the transmission. It is completely dependent upon the secret key provided to the encryption algorithm. Each unique secret key produces a unique cipher text.

5. Decryption Algorithm (D): It performs reversal of the encryption algorithm at the recipient's side. It also takes the secret key as input and decodes the cipher text received from the sender based on the secret key. It produces plain text as output.

$D(y, k) = x$

Requirements for Encryption:

There are only two requirements that need to be met to perform encryption. They are,

1. Encryption Algorithm: There is a need for a very strong encryption algorithm that produces cipher texts in such a way that the attacker should be unable to crack the secret key even if they have access to one or more cipher texts.

2. Secure way to share Secret Key: There must be a secure and robust way to share the secret key between the sender and the receiver. It should be leakproof so that the attacker cannot access the secret key.

Substitution Cipher

Hiding some data is known as encryption. When plain text is encrypted it becomes unreadable and is known as ciphertext. In a Substitution cipher, any character of plain text from the given fixed set of characters is substituted by some other character from the same set depending on a key. For example with a shift of 1, A would be replaced by B, B would become C, and so on.

Note: A special case of Substitution cipher is known as Caesar cipher where the key is taken as 3.

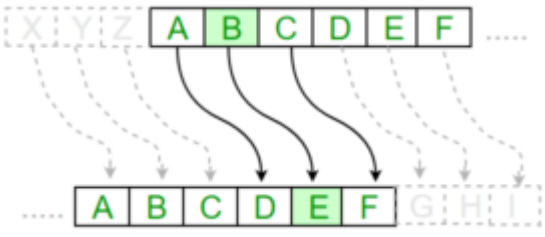
Mathematical representation

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

L

(Encryption Phase with shift n)

(Decryption Phase with shift n)



Examples:

Plain Text: I am studying Data Encryption
Key: 4
Output: M eq wxyhCmrk Hexe IrgvCtxmsr

Plain Text: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Key: 4
Output: EFGHIJKLMNOPQRSTUVWXYZabcd

Algorithm for Substitution Cipher:

Input:

- A String of both lower and upper case letters, called PlainText.
- An Integer denoting the required key.

Procedure:

- Create a list of all the characters.
- Create a dictionary to store the substitution for all characters.
- For each character, transform the given character as per the rule, depending on whether we're encrypting or decrypting the text.
- Print the new string generated.

Columnar Transposition Cipher

Given a plain-text message and a numeric key, cipher/de-cipher the given text using Columnar Transposition Cipher The Columnar Transposition Cipher is a form of transposition cipher just like Rail Fence Cipher. Columnar Transposition involves writing the plaintext out in rows, and then reading the ciphertext off in columns one by one.

Examples:

Encryption

Input : Geeks for Geeks

Key = HACK

Output : e kefGsGsrekoe_

Decryption

Input : e kefGsGsrekoe_

Key = HACK

Output : Geeks for Geeks

Encryption

Input : Geeks on work

Key = HACK

Output : e w_eoo_Gs kknr_

Decryption

Input : e w_eoo_Gs kknr_

Key = HACK

Output : Geeks on work

Encryption

In a transposition cipher, the order of the alphabets is re-arranged to obtain the cipher-text.

1. The message is written out in rows of a fixed length, and then read out again column by column, and the columns are chosen in some scrambled order.
2. Width of the rows and the permutation of the columns are usually defined by a keyword.
3. For example, the word HACK is of length 4 (so the rows are of length 4), and the permutation is defined by the alphabetical order of the letters in the keyword. In this case, the order would be “3 1 2 4”.
4. Any spare spaces are filled with nulls or left blank or placed by a character (Example: _).
5. Finally, the message is read off in columns, in the order specified by the keyword.

Encryption

Given text = Geeks for Geeks
Keyword = HACK **Length of Keyword** = 4 (no of rows) **Order of Alphabets in HACK** = 3124

H	A	C	K
3	1	2	4
G	e	e	k
s	_	f	o
r	_	G	e
e	k	s	_

Print Characters of column 1,2,3,4
Encrypted Text = e kefGsGsrekoe_

Decryption

1. To decipher it, the recipient has to work out the column lengths by dividing the message length by the key length.
2. Then, write the message out in columns again, then re-order the columns by reforming the key word.

Brute Force Attack

A **Brute force attack** is a well known breaking technique, by certain records, brute force attacks represented five percent of affirmed security ruptures. A brute force attack includes 'speculating' username and passwords to increase unapproved access to a framework. Brute force is a straightforward attack strategy and has a high achievement rate.

A few attackers use applications and contents as brute force devices. These instruments evaluate various secret word mixes to sidestep confirmation forms. In different cases, attackers attempt to get to web applications via scanning for the correct session ID. Attacker inspiration may incorporate taking data, contaminating destinations with malware, or disturbing help.

While a few attackers still perform brute force attacks physically, today practically all brute force attacks are performed by bots. Attackers have arrangements of usually utilized accreditations, or genuine client qualifications, got through security breaks or the dull web. Bots deliberately attack sites and attempt these arrangements of accreditations, and advise the attacker when they obtain entrance.

Block Cipher Design Principles

Block ciphers are built in the Feistel cipher structure. Block cipher has a specific number of rounds and keys for generating ciphertext.Block cipher is a type of encryption algorithm that processes fixed-size blocks of data, usually 64 or 128 bits, to produce ciphertext. The design of a block cipher involves several important principles to ensure the security and efficiency of the algorithm. Some of these principles are:

1. **Number of Rounds –** The number of Rounds is regularly considered in design criteria, it just reflects the number of rounds to be suitable for an algorithm to make it more complex, in DES we have 16 rounds ensuring it to be more secure while in AES we have 10 rounds which makes it more secure.
2. **Design of function F –** The core part of the Feistel Block cipher structure is the Round Function. The complexity of cryptanalysis can be derived from the Round function i.e. the increasing level of complexity for the round function would be greatly contributing to an increase in complexity. To increase the complexity of the round function, the avalanche effect is also included in the round function, as the change of a single bit in plain text would produce a mischievous output due to the presence of avalanche effect.
3. **Confusion and Diffusion:** The cipher should provide confusion and diffusion to make it difficult for an attacker to determine the relationship between the plaintext and ciphertext. Confusion means that the ciphertext should be a complex function of the key and plaintext, making it difficult to guess the key. Diffusion means that a small change in the plaintext should cause a significant change in the ciphertext, which makes it difficult to analyze the encryption pattern.
4. **Key Size:** The key size should be large enough to prevent brute-force attacks. A larger key size means that there are more possible keys, making it harder for an attacker to guess the correct one. A key size of 128 bits is considered to be secure for most applications.
5. **Key Schedule:** The key schedule should be designed carefully to ensure that the keys used for encryption are independent and unpredictable. The key schedule should also resist attacks that exploit weak keys or key-dependent properties of the cipher.
6. **Block Size:** The block size should be large enough to prevent attacks that exploit statistical patterns in the plaintext. A block size of 128 bits is generally considered to be secure for most applications.
7. **Non-linearity:** The S-box used in the cipher should be non-linear to provide confusion. A linear S-box is vulnerable to attacks that exploit the linear properties of the cipher.

8. **Avalanche Effect:** The cipher should exhibit the avalanche effect, which means that a small change in the plaintext or key should cause a significant change in the ciphertext. This ensures that any change in the input results in a complete change in the output.
9. **Security Analysis:** The cipher should be analyzed for its security against various attacks such as differential cryptanalysis, linear cryptanalysis, and brute-force attacks. The cipher should also be tested for its resistance to implementation attacks, such as side-channel attacks.

Overall, a good block cipher design should be resistant to various attacks, efficient, and easy to implement.

Data encryption standard (DES)

This article talks about the Data Encryption Standard (DES), a historic encryption algorithm known for its 56-bit key length. We explore its operation, key transformation, and encryption process, shedding light on its role in data security and its vulnerabilities in today's context.

What is DES?

Data Encryption Standard (DES) is a block cipher with a 56-bit key length that has played a significant role in data security**.** Data encryption standard (DES) has been found vulnerable to very powerful attacks 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 go 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 below:

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

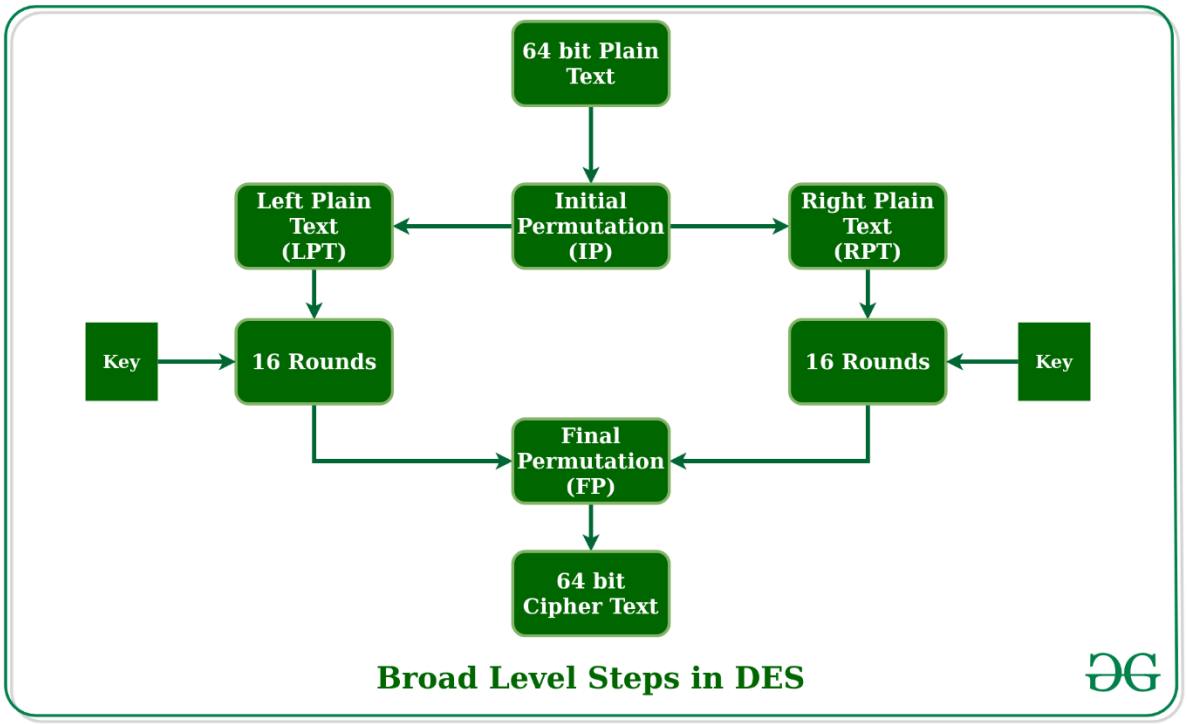
Figure - discording of every 8th bit of original key

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.

- In the first step, the 64-bit plain text block is handed over to an initial Permutation (IP) function.
- The initial permutation is performed on plain text.
- Next, the initial permutation (IP) produces two halves of the permuted block; saying Left Plain Text (LPT) and Right Plain Text (RPT).
- Now each LPT and RPT go through 16 rounds of the encryption process.
- In the end, LPT and RPT are rejoined and a Final Permutation (FP) is performed on the combined block
- The result of this process produces 64-bit ciphertext.



Initial Permutation (IP)

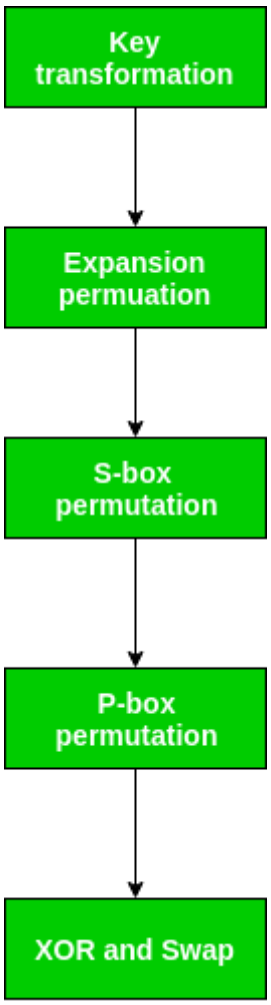
As we have noted, the initial permutation (IP) happens only once and it happens before the first round. It suggests how the transposition in IP should proceed, as shown in the figure. For example, it says that the IP replaces the first bit of the original plain text block with the 58th bit of the original 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 one 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. From the 48 we might obtain 64 or 56 bits based on requirement which helps us to recognize that this model is very versatile and can handle any range of requirements needed or provided. 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 to the first position, bit number 17 moves to the second position, and so on. If we observe the table , 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 the 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.

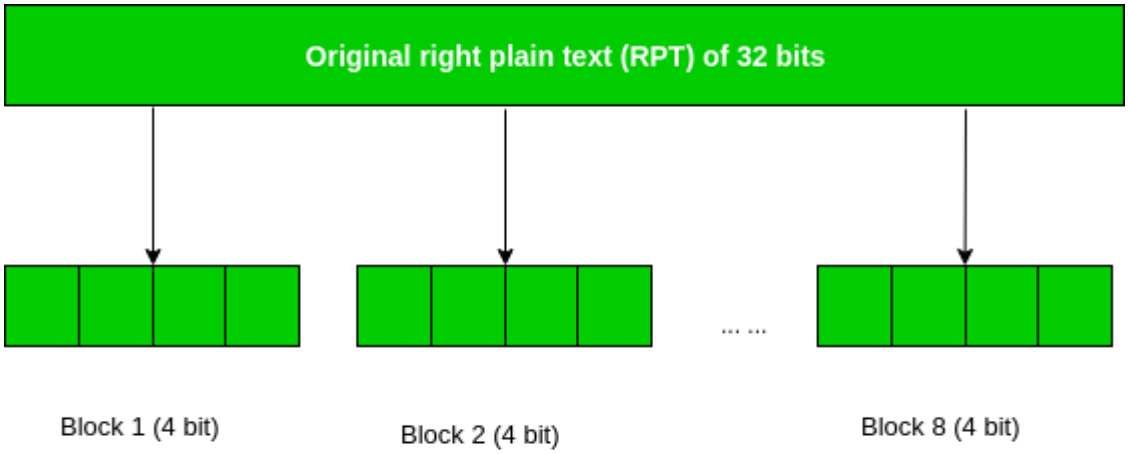


Figure - division of 32 bit RPT into 8 bit blocks

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**.

Image Steganography in Cryptography

The word **Steganography** is derived from two Greek words- 'stegos' meaning 'to cover' and 'grayfia', meaning 'writing', thus translating to 'covered writing', or 'hidden writing'. **Steganography** is a method of hiding secret data, by embedding it into an audio, video, image, or text file. It is one of the methods employed to protect secret or sensitive data from malicious attacks.

How is it different from cryptography?

Cryptography and steganography are both methods used to hide or protect secret data. However, they differ in the respect that cryptography makes the data unreadable, or hides the *meaning* of the data, while steganography hides the *existence* of the data.

In layman's terms, cryptography is similar to writing a letter in a secret language: people can read it, but won't understand what it means. However, the existence of a (probably secret) message would be obvious to anyone who sees the letter, and if someone either knows or figures out your secret language, then your message can easily be read.

If you were to use *steganography* in the same situation, you would hide the letter inside a pair of socks that you would be gifting the intended recipient of the letter. To those who don't know about the message, it would look like there was nothing more to your gift than the socks. But the intended recipient knows what to look for, and finds the message hidden in them.

Similarly, if two users exchanged media files over the internet, it would be more difficult to

determine whether these files contain hidden messages than if they were communicating using cryptography.
Cryptography is often used to supplement the security offered by steganography. Cryptography algorithms are used to encrypt secret data before embedding it into cover files.

Image Steganography –

As the name suggests, Image Steganography refers to the process of hiding data within an image file. The image selected for this purpose is called the **cover image** and the image obtained after steganography is called the **stego image**.

How is it done?

An image is represented as an N*M (in case of grayscale images) or N*M*3 (in case of color images) matrix in memory, with each entry representing the intensity value of a pixel. In image steganography, a message is embedded into an image by altering the values of some pixels, which are chosen by an encryption algorithm. The recipient of the image must be aware of the same algorithm in order to know which pixels he or she must select to extract the message.

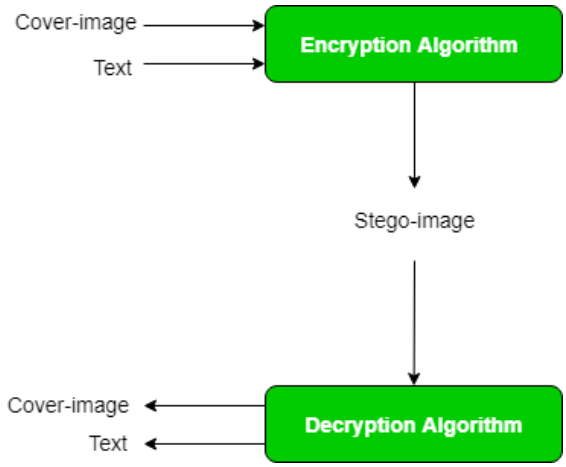


Figure – Process of Image Steganography

Is steganography a secure method of communication?

When steganography is employed alone, it is security by obscurity, which might result in the secret message being disclosed. Combining steganography and cryptography is the greatest way to disguise a message from adversaries while still protecting it in case it is detected.

In steganography, what algorithm is used?

His steganography approach entails concealing a huge amount of data (picture, audio, and text) within a colour bitmap (bmp) image. The image will be filtered and segmented in his study, with bits replacement applied to the appropriate pixels. These pixels are chosen at random rather than in order.
Detection of the message within the **cover image** is done by the process of **steganalysis**. This

can be done through comparison with the cover image, histogram plotting, or noise detection. While efforts are being invested in developing new algorithms with a greater degree of immunity against such attacks, efforts are also being devoted towards improving existing algorithms for steganalysis, to detect the exchange of secret information between terrorists or criminal elements.