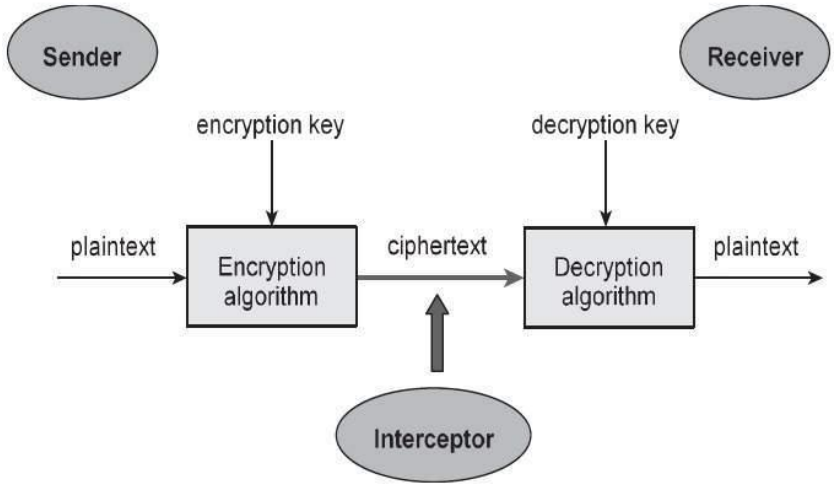
What is cryptography?

Cryptography is a method of protecting information and communications through the use of codes, so that only those for whom the information is intended can read and process it.

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**.



The illustration 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.

The objective of this simple cryptosystem is that at the end of the process, only the sender and the receiver will know the plaintext.

Components of a Cryptosystem

The various components of a basic cryptosystem are as follows −

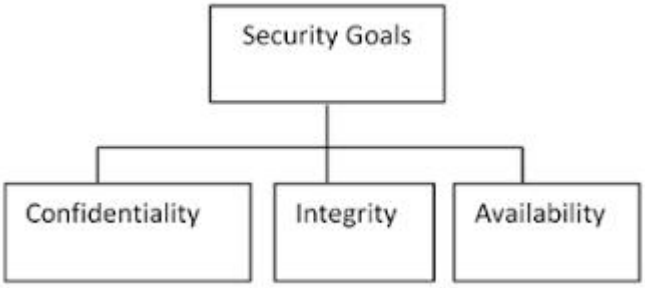
- **Plaintext.** It is the data to be protected during transmission.
- **Encryption Algorithm.** It is a mathematical process that produces a ciphertext for any given plaintext and encryption key. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.
- **Ciphertext.** It is the scrambled version of the plaintext produced by the encryption algorithm using a specific the encryption key. The ciphertext is not guarded. It flows on 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 in order to compute the plaintext.

For a given cryptosystem, a collection of all possible decryption keys is called a **key space**.

An **interceptor** (an attacker) is an unauthorized entity who attempts to determine the plaintext. He can see the ciphertext and may know the decryption algorithm. He, however, must never know the decryption key.

The Three Security Goals are **Confidentiality, Integrity and Availability**.



1. **Confidentiality:** Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information. A loss of confidentiality is the unauthorized disclosure of information. It includes the following two concepts:
    - *Data confidentiality*: Assures that private or confidential information is not made available or disclosed to unauthorized individuals.
    - *Privacy*: Assures that individuals control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed.
2. **Integrity:** Guarding against improper information modification or destruction, including ensuring information nonrepudiation and authenticity. A loss of integrity is the unauthorized modification or destruction of information. It includes the following two concepts:
    - *Data*: Assures information and programs are changed only in a specified and authorized manner.
    - *System*: Assures that a system performs its intended function in an unimpaired manner, free from unauthorized manipulation of the system.

    Integrity models have three goals:

- Prevent unauthorized users from making modifications to data or programs
- Prevent authorized users from making improper or unauthorized modifications
- Maintain internal and external consistency of data and programs.

    An example of integrity checks is balancing a batch of transactions to make sure that all the information is present and accurately accounted for.

3. **Availability:** Ensuring timely and reliable access to and use of information. A loss of availability is the disruption of access to or use of information or an information system. Assures that system works promptly and services are not denied to authorized

users. Information security professionals usually address three common challenges to availability:

- **Denial of service (DoS)** due to intentional attacks or because of undiscovered flaws in implementation (for example, a program written by a programmer who is unaware of a flaw that could crash the program if a certain unexpected input is encountered)
- Loss of information system capabilities because of natural disasters (fires, floods, storms, or earthquakes) or human actions (bombs or strikes)
- Equipment failures during normal use

Cryptographic Attacks

The basic intention of an attacker is to break a cryptosystem and to find the plaintext from the ciphertext. To obtain the plaintext, the attacker only needs to find out the secret decryption key, as the algorithm is already in public domain.

Hence, he applies maximum effort towards finding out the secret key used in the cryptosystem. Once the attacker is able to determine the key, the attacked system is considered as *broken* or *compromised*.

Based on the methodology used, attacks on cryptosystems are categorized as follows −

- **Ciphertext Only Attacks (COA)** − In this method, the attacker has access to a set of ciphertext(s). He does not have access to corresponding plaintext. COA is said to be successful when the corresponding plaintext can be determined from a given set of ciphertext. Occasionally, the encryption key can be determined from this attack. Modern cryptosystems are guarded against ciphertext-only attacks.
- **Known Plaintext Attack (KPA)** − In this method, the attacker knows the plaintext for some parts of the ciphertext. The task is to decrypt the rest of the ciphertext using this information. This may be done by determining the key or via some other method. The best example of this attack is *linear cryptanalysis* against block ciphers.
- **Chosen Plaintext Attack (CPA)** − In this method, the attacker has the text of his choice encrypted. So he has the ciphertext-plaintext pair of his choice. This simplifies his task of determining the encryption key. An example of this attack is *differential cryptanalysis* applied against block ciphers as well as hash functions. A popular public key cryptosystem, RSA is also vulnerable to chosen-plaintext attacks.
- **Dictionary Attack** − This attack has many variants, all of which involve compiling a 'dictionary'. In simplest method of this attack, attacker builds a dictionary of ciphertexts and corresponding plaintexts that he has learnt over a period of time. In future, when an attacker gets the ciphertext, he refers the dictionary to find the corresponding plaintext.

- **Man in Middle Attack (MIM)** − The targets of this attack are mostly public key cryptosystems where key exchange is involved before communication takes place.

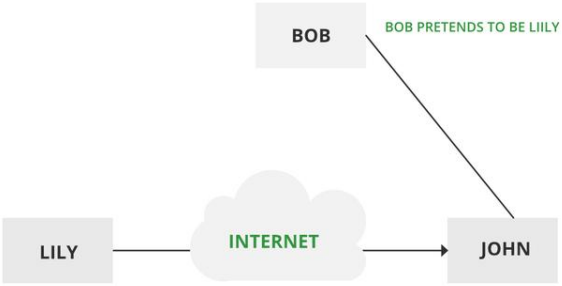
- Host *A* wants to communicate to host *B*, hence requests public key of *B*.
- An attacker intercepts this request and sends his public key instead.
- Thus, whatever host *A* sends to host *B*, the attacker is able to read.
- In order to maintain communication, the attacker re-encrypts the data after reading with his public key and sends to *B*.

Active attacks: An Active attack attempts to alter system resources or affect their operations. Active attacks involve some modification of the data stream or the creation of false statements. Types of active attacks are as follows:

- Masquerade
- Modification of messages
- Repudiation
- Replay
- Denial of Service

Masquerade –

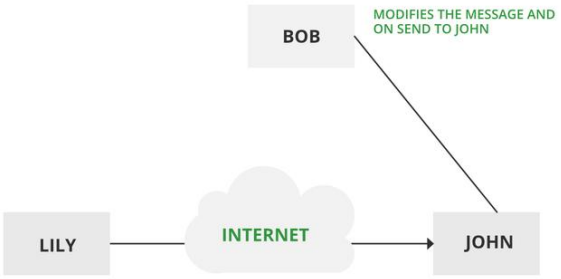
A masquerade attack takes place when one entity pretends to be a different entity. A Masquerade attack involves one of the other forms of active attacks. If an authorization procedure isn't always absolutely protected, it is able to grow to be extraordinarily liable to a masquerade assault. Masquerade assaults may be performed using the stolen passwords and logins, with the aid of using finding gaps in programs, or with the aid of using locating a manner across the authentication process.



Masquerade Attack

Modification of messages –

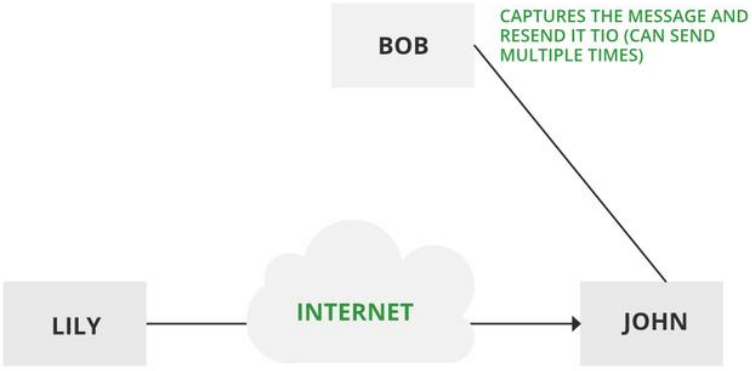
It means that some portion of a message is altered or that message is delayed or reordered to produce an unauthorized effect. Modification is an attack on the integrity of the original data. It basically means that unauthorized parties not only gain access to data but also spoof the data by triggering denial-of-service attacks, such as altering transmitted data packets or flooding the network with fake data. Manufacturing is an attack on authentication. For example, a message meaning "Allow JOHN to read confidential file X" is modified as "Allow Smith to read confidential file X".



Modification of messages

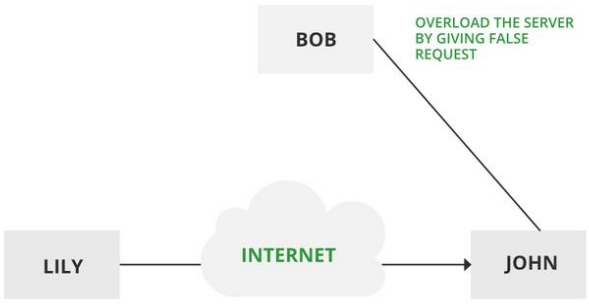
Repudiation –
This attack occurs when the network is not completely secured or the login control has been tampered with. With this attack, the author's information can be changed by actions of a malicious user in order to save false data in log files, up to the general manipulation of data on behalf of others, similar to the spoofing of e-mail messages.

Replay –
It involves the passive capture of a message and its subsequent transmission to produce an authorized effect. In this attack, the basic aim of the attacker is to save a copy of the data originally present on that particular network and later on use this data for personal uses. Once the data is corrupted or leaked it is insecure and unsafe for the users.



Replay

Denial of Service –
It prevents the normal use of communication facilities. This attack may have a specific target. For example, an entity may suppress all messages directed to a particular destination. Another form of service denial is the disruption of an entire network either by disabling the network or by overloading it with messages so as to degrade performance.

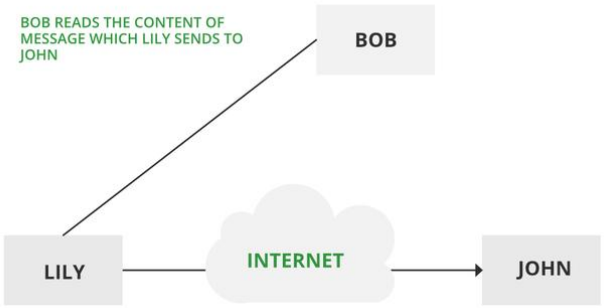


Denial of Service

Passive attacks: A Passive attack attempts to learn or make use of information from the system but does not affect system resources. Passive Attacks are in the nature of eavesdropping on or monitoring transmission. The goal of the opponent is to obtain information that is being transmitted. Types of Passive attacks are as follows:

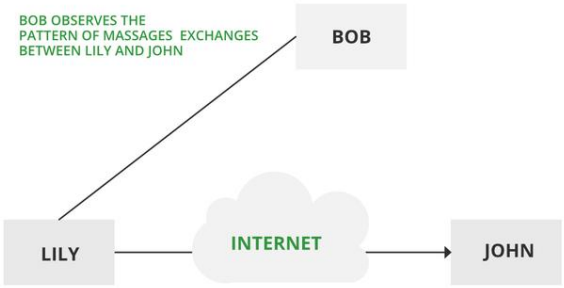
- The release of message content
- Traffic analysis

The release of message content –
Telephonic conversation, an electronic mail message, or a transferred file may contain sensitive or confidential information. We would like to prevent an opponent from learning the contents of these transmissions.



Passive attack

Traffic analysis –
Suppose that we had a way of masking (encryption) information, so that the attacker even if captured the message could not extract any information from themessage. The opponent could determine the location and identity of communicating host and could observe the frequency and length of messages being exchanged. This information might be useful in guessing the nature of the communication that was taking place. The most useful protection against traffic analysis is encryption of SIP traffic. To do this, an attacker would have to access the SIP proxy (or its call log) to determine who made the call.



Traffic analysis

Symmetric encryption is also referred to as **conventional encryption** or **single-key encryption.** It was the only type of encryption in use prior to the development of public-key encryption. It remains by far the most widely used of the two types of encryption.

A symmetric encryption scheme has five ingredients:

1. **Plain text:** This is the Original intelligible message or data that is fed in to the algorithm as input.
2. **Encryption Algorithm**: The encryption algorithm performs various substitutions and transformation on the plain text to convert it into ciphertext.
3. **Secret Key:** The secret key is also input to the encryption algorithm. The key is a value independent of the plain text. The algorithm will produce a different output depending on the specific key being used at the time. The exact substitutions and transformations performed by the algorithm depend on the key.
4. **Ciphertext:** This is the scrambled message produced as output. It depends on the plain text and the secret key. For a given message, two different keys will produce different ciphertexts. The ciphertext is an apparently random stream of data and, as it stands, is unintelligible.
5. **Decryption Algorithm:** This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key as the input and produces the original plain text.

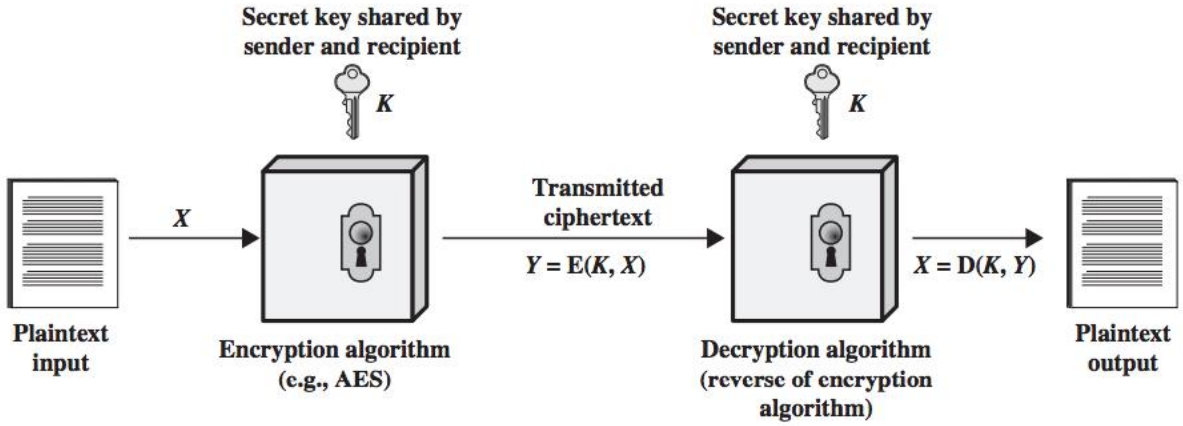


Fig 1 Simplified Model of Conventional Encryption

There are two requirements for secure use of conventional encryption-

- We need a strong encryption algorithm. At a minimum, we would like the algorithm to be such that an opponent who known the algorithm and has access to one or more ciphertext would be unable to decipher the ciphertext or figure out the key. Usually, this requirement is stated in a stronger form. The opponent should be unable to decrypt ciphertext or discover the key even if he or she is in possession of a number of ciphertext together with the plain text that produce each ciphertext
- Sender and Receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure. If someone can discover the key and knows the algorithm, all information using this key is readable.

Symmetric key cryptography is a type of encryption scheme in which the similar key is used both to encrypt and decrypt messages. Such an approach of encoding data has been largely used in the previous decades to facilitate secret communication between governments and militaries.

Symmetric-key cryptography is called a shared-key, secret-key, single-key, one-key and eventually private-key cryptography. With this form of cryptography, it is clear that the key should be known to both the sender and the receiver that the shared. The complexity with this approach is the distribution of the key.

Symmetric key cryptography schemes are usually categorized such as stream ciphers or block ciphers. Stream ciphers work on a single bit (byte or computer word) at a time and execute some form of feedback structure so that the key is repeatedly changing.

A block cipher is so-called because the scheme encrypts one block of information at a time utilizing the same key on each block. In general, the same plaintext block will continually encrypt to the same ciphertext when using the similar key in a block cipher whereas the same plaintext will encrypt to different ciphertext in a stream cipher.

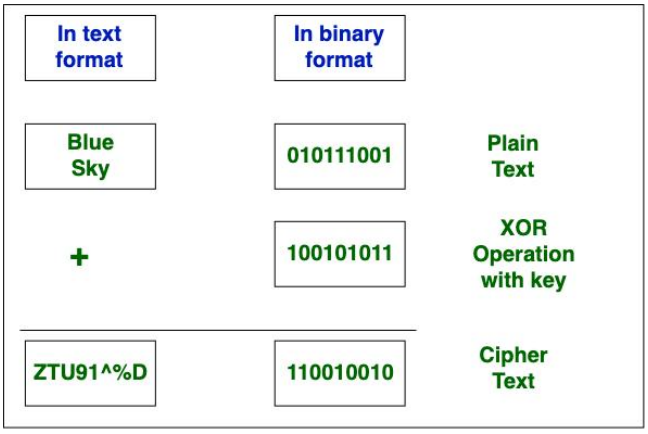
Block ciphers can operate in one of several modes which are as follows −

- Electronic Codebook (ECB) mode is the simplest application and the shared key can be used to encrypt the plaintext block to form a ciphertext block. There

are two identical plaintext blocks will always create the same ciphertext block. Although this is the most common mode of block ciphers, it is affected to multiple brute-force attacks.

- Cipher Block Chaining (CBC) mode insert a feedback structure to the encryption scheme. In CBC, the plaintext is exclusively-ORed (XORed) with the prior ciphertext block prior to encryption. In this mode, there are two identical blocks of plaintext not encrypt to the similar ciphertext.
- Cipher Feedback (CFB) mode is a block cipher implementation as a selfsynchronizing stream cipher. CFB mode enable data to be encrypted in units lower than the block size, which can be beneficial in some applications including encrypting interactive terminal input. If it is using 1-byte CFB mode. Each incoming character is located into a shift register the similar size as the block, encrypted, and the block transmitted. At the receiving side, the ciphertext is decrypted and the more bits in the block are discarded.
- Output Feedback (OFB) mode is a block cipher implementation conceptually same to a synchronous stream cipher. OFB avoids the similar plaintext block from making the same ciphertext block by using an internal feedback structure that is independent of both the plaintext and ciphertext bitstreams.

Block Cipher and **Stream Cipher** belongs to the symmetric key cipher. These two block ciphers and stream cipher are the methods used for converting the plain text into ciphertext.
The main difference between a **Block cipher** and a **Stream cipher** is that a block cipher converts the plain text into cipher text by taking plain text's block at a time. While stream cipher Converts the plain text into cipher text by taking 1 byte of plain text at a time.



Stream Cipher

Let's see the difference between them:

S.NO	Block Cipher	Stream Cipher
1.	Block Cipher Converts the plain text into cipher text by taking plain text's block at a time.	Stream Cipher Converts the plain text into cipher text by taking 1 byte of plain text at a time.
2.	Block cipher uses either 64 bits or more than 64 bits.	While stream cipher uses 8 bits.
3.	The complexity of block cipher is simple.	While stream cipher is more complex.

4.	Block cipher Uses confusion as well as diffusion.	While stream cipher uses only confusion.
5.	In block cipher, reverse encrypted text is hard.	While in-stream cipher, reverse encrypted text is easy.
6.	The algorithm modes which are used in block cipher are ECB (Electronic Code Book) and CBC (Cipher Block Chaining).	The algorithm modes which are used in stream cipher are CFB (Cipher Feedback) and OFB (Output Feedback).
7.	Block cipher works on transposition techniques like rail-fence technique, columnar transposition technique, etc.	While stream cipher works on substitution techniques like Caesar cipher, polygram substitution cipher, etc.
8.	Block cipher is slow as compared to a stream cipher.	While stream cipher is fast in comparison to block cipher.

Symmetric encryption techniques

A **symmetric encryption** is any technique where the same key is used to both encrypt and decrypt the data. The Caesar Cipher is one of the simplest symmetric encryption techniques, and of course, one of the easiest to crack.

Since then, cryptologists have invented many more symmetric encryption techniques, including the ones used today to encrypt data like passwords.

Vigenère Cipher

French cryptologists invented the Vigenère Cipher in the mid 1500s. The cipher was considered especially strong, and author Lewis Caroll even called it “unbreakable” in 1868. It was indeed much stronger than the Caesar Cipher, but as we’ll see, it can definitely be cracked.

Encryption

The Vigenère cipher uses an entire word as the shift key, as opposed to the Caesar Cipher’s single shift amount.

Imagine that we want to encrypt the phrase VERSAILLES and use a shift key of CHEESE.

First, we need to repeat the shift key to line up with each of the letters in the phrase:

Original	**V**	**E**	**R**	**S**	**A**	**I**	**L**	**L**	**E**	**S**

Original	**V**	**E**	**R**	**S**	**A**	**I**	**L**	**L**	**E**	**S**
Shift key	C	H	E	E	S	E	C	H	E	E

Now we replace each letter of the original text according to the Vigenère table:

For the first letter "V", we select the row that starts with "V". Then since the corresponding shift key letter is "C", we move to the column that has a header of "C". The letter at the intersection of the "V" row and "C" column is "X". Thus, we encrypt "V" as "X".

Original	**V**	**E**	**R**	**S**	**A**	**I**	**L**	**L**	**E**	**S**
Shift key	C	H	E	E	S	E	C	H	E	E
Encrypted	X	?	?	?	?	?	?	?	?	?

The letter at the intersection of the "E" row and "H" column is "L", so we encrypt "E" as "L".

Original	**V**	**E**	**R**	**S**	**A**	**I**	**L**	**L**	**E**	**S**
Shift key	C	H	E	E	S	E	C	H	E	E
Encrypted	X	L	?	?	?	?	?	?	?	?

If we keep going, we'll end up with the encrypted text "XLVWSMNSIW".

Original	**V**	**E**	**R**	**S**	**A**	**I**	**L**	**L**	**E**	**S**
Shift key	C	H	E	E	S	E	C	H	E	E
Encrypted	X	L	V	W	S	M	N	S	I	W

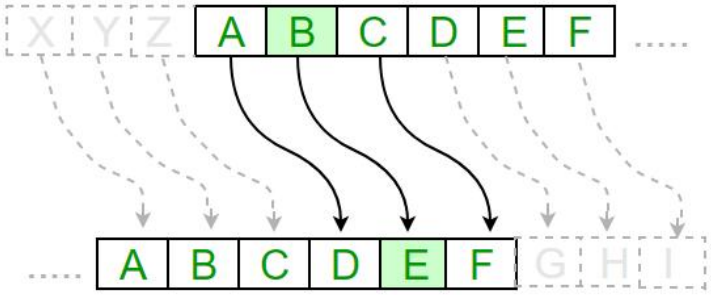
Caesar Cipher in Cryptography

The Caesar Cipher technique is one of the earliest and simplest methods of encryption technique. 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.

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.

(Encryption Phase with shift n)

(Decryption Phase with shift n)



Examples :

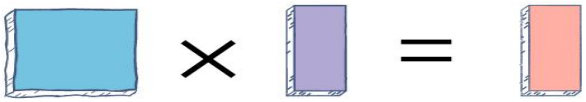
Text : ABCDEFGHIJKLMNOPQRSTUVWXYZ
Shift: 23
Cipher: XYZABCDEFGHIJKLMNOPQRSTUVW

Text : ATTACKATONCE
Shift: 4
Cipher: EXXEGOEXSRGI

Hill cipher

The Hill cipher is a polygraphic substitution cipher built on concepts from Linear Algebra. The Hill cipher makes use of modulo arithmetic, matrix multiplication, and matrix inverses; hence, it is a more mathematical cipher than others. The Hill cipher is also a block cipher, so, theoretically, it can work on arbitrary sized blocks.

Polygraphic substitution is a uniform substitution where a block of letters is substituted by a word, character, number, etc.



The Hill cipher is built on matrix multiplication

Since the Hill cipher is fairly complex, let's encrypt the text "CODE" and, later, decrypt the resulting ciphertext to understand how the Hill cipher works. To keep the example simple, we will use a straightforward substitution scheme where the letter A is mapped to 0, B is mapped to 1, etc. to stick to a 2x2 key matrix. The complexity of the Hill cipher increases with the size of the key matrix.

Encryption

Encrypting with the Hill cipher is built on the following operation:

$$E(K, P) = (K*P) \bmod 26$$

Where K is our key matrix and P is the plaintext in vector form. Matrix multiplying these two terms produces the encrypted ciphertext. Let's do so step by step:

1. Pick a keyword to encrypt your plaintext message. Let's work with the random keyword "DCDF". Convert this keyword to matrix form using your substitution scheme to convert it to a numerical 2x2 key matrix.

2. Next, we will convert our plaintext message to vector form. Since our key matrix is 2x2, the vector needs to be 2x1 for matrix multiplication to be possible. In our case, our message is four letters long so we can split it into blocks of two and then substitute to get our plaintext vectors.

3. Now, we can matrix multiply the key matrix with each 2x1 plaintext vector, take the moduli of the resulting 2x1 vectors by 26, and concatenate the results to get "WWVA", the final ciphertext.