01. COMPUTER SECURITY CONCEPTS

1.1 A Definition of Computer Security:

The ***NIST Computer Security Handbook*** defines computer security as ***protecting an automated information system*** to achieve the three main objectives: ***confidentiality, integrity, and availability.*** Think of it like a triple lock on your most valuable data:

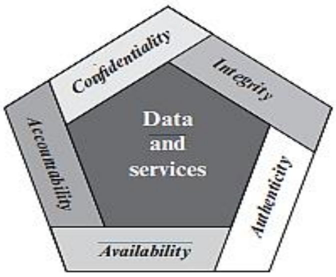


Figure 1.1 Essential Network and Computer Security Requirements

- **Confidentiality:** This is like a padlock ***ensuring only authorized individuals can access private information.*** Imagine ***student grades;*** they should only be accessible to ***students, their parents,*** and ***authorized school personnel.***
- **Integrity:** This is like a ***tamper-proof seal*** guaranteeing ***information remains accurate and unchanged.*** Consider a hospital's patient allergy data; any alteration could have abnormal consequences.
- **Availability:** This is like a well-oiled door making sure ***authorized users can access information and systems promptly,*** like students accessing online course materials without delays.

1.2 Deeper Dive into the CIA Triad:

- **Confidentiality** goes beyond data secrecy. It also ***includes privacy,*** which ***empowers individuals to control their personal information,*** deciding who can ***collect, store, and access*** it.
- **Integrity includes two aspects:**
  - **Data integrity:** ***Information (stored or transmitted) and programs remain unaltered unless authorized.*** Imagine financial transactions; any unauthorized change could lead to chaos.
  - **System integrity:** The ***system functions as intended, free from malicious manipulation.*** Think of an air traffic control system; unauthorized tampering could endanger countless lives.
- **Availability** ensures ***systems function promptly*** and ***authorized users have uninterrupted access.*** Example include ***Google Colaboratory.***

1.3 Beyond the CIA Triad:

While the CIA triad forms the core of computer security, additional concepts are crucial:

- **Authenticity:** ***Verifying the genuineness of users and data sources.*** Imagine verifying a login to ensure you're not talking to a hacker pretending to be your customer.

- **Accountability:** ***Tracing actions back to responsible individuals.*** This supports non-repudiation ***(preventing denial of actions)*** and helps identify responsible parties in case of breaches.

1.4 Examples of Different Impact Levels:

The importance of these concepts depends on the potential impact of a security breach. Different assets warrant different levels of protection:

- **High:** ***Student grade information (confidentiality) or hospital patient allergy data (integrity)*** have high impact breaches causing severe or even life-threatening consequences.

- **Moderate:** A ***university website (availability) or online forum (integrity)*** might have moderate impact breaches causing inconvenience or reputational damage.

- **Low:** A ***publicly available directory information*** (confidentiality) have low impact breaches causing minimal harm.

1.5 The Challenges of Computer Security:

1. **Complex Mechanisms**: Security systems use ***complicated methods and rules.*** They need to be smart to understand and stop different types of threats.

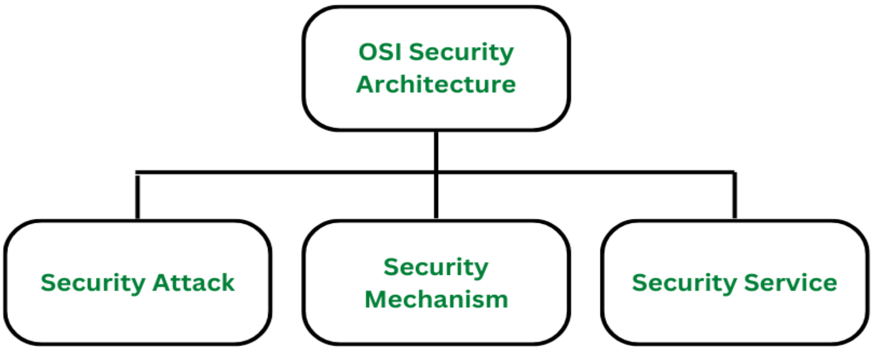
2. **Attackers Advantage**: People who want to harm the system only need to find one weak point, but the people protecting the system have to make sure every point is strong.

3. **Placement of Security Measures**: Deciding where to put security protections is important and needs careful thought. These protections could be physical (like a lock) or logical (like a password).

4. **Protection of Secret Information**: ***Keeping secret information (like passwords or keys) safe is another challenge.*** This information needs to be managed carefully.

02. OSI SECURITY ARCHITECTURE

The OSI (Open Systems Interconnection) Security Architecture is a ***systematic approach to providing security at each layer of the OSI model.*** It defines ***security services and security mechanisms*** that can be used at each of the ***seven layers*** of the OSI model to ***provide security for data transmitted over a network.***



The architecture focuses on three main concepts:

1. **Security Attack:** Any ***action that compromises the security of information*** owned by an organization.
2. **Security Mechanism:** A ***process (or a device)*** that is designed to ***detect, prevent, or recover from a security attack.***
3. **Security Service:** This is a service that ***improves the safety of computer systems and the transfer of information*** in a company. It's like a guard that keeps the company's digital information safe.

Security attacks can be further classified into two sub-categories:

- **Passive Attack:** Attacks in which a ***third-party intruder tries to access the message/content/data being shared*** by the sender and receiver by keeping a close watch on the transmission.
- **Active Attack:** These attacks involve some ***modification of the data stream*** or the ***creation of a false stream.***

The OSI security architecture is ***internationally acceptable*** as it lays the flow of providing safety in an organization. It is a very important aspect of ***computer networking.*** The OSI Security model ***provides a standard architecture of security*** of the data for an organization. It is important to provide a well-defined and well-organized structure for the security so that the organizations can decide upon the security measure in a planned manner and the chances of data breaching or any such security threats are minimized.

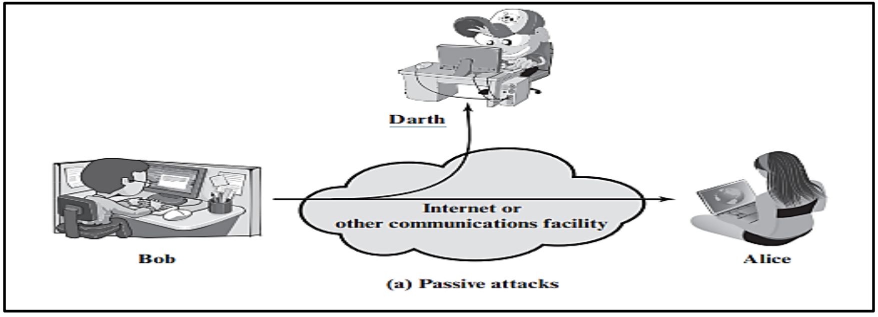
03. SECURITY ATTACKS

A **security attack** is an act made upon a system with the ***goal to obtain unauthorized access to information or resources.*** It is usually carried out by the people those who ***bypass the security rules set by companies*** or on personal devices to carry out these attacks.

Classification of Attacks:

1. Passive Attacks:

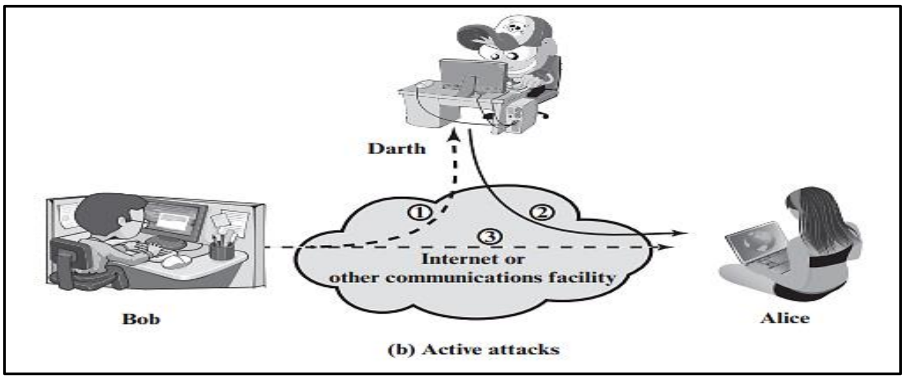
- ***Eavesdropping:*** Monitoring or observing transmissions without altering data.
- ***Traffic Analysis:*** ***Observing patterns of messages*** to extract information about communication. Not actual content is revealed, just metadata is visible.
- ***Prevention:*** Encryption can mask message contents; however, traffic analysis remains a challenge.
- ***Detection:*** ***Passive attacks are difficult to detect*** as they do not alter data; ***prevention, especially through encryption, is emphasized.***



(a) Passive attacks

2. Active Attacks:

- ***Masquerade:*** One entity pretends to be another, often involving ***capturing and replaying authentication sequences.***
- ***Replay:*** Someone ***quietly capture the data and retransmit*** to produce an ***unauthorized effect.***
- ***Modification of Messages: Alteration of legitimate messages*** or ***delaying/reordering*** to produce an unauthorized effect.
- ***Denial of Service (DoS):*** Prevents normal use or management of communication facilities. **Recent example:** Geopolitical events in Ukraine. The attack significantly impacted ***Ukrainian government networks.***
- ***Detection and Prevention: Active attacks are challenging to prevent due to diverse vulnerabilities,*** but detection and recovery measures are implemented.



(b) Active attacks

Threats and Attacks:

- ***Threat:*** This is when there's a chance that something bad could happen to a system by taking advantage of its weak points.
- ***Attack:*** An ***intelligent and deliberate attempt to disrupt security services,*** violating the security policy.

In summary, security attacks are classified into passive and active categories. ***Passive attacks focus on information observation,*** while ***active attacks involve modifications or creation of false streams.*** Prevention measures, such as encryption, are emphasized for passive attacks, while ***active attacks require a combination of detection, recovery, and deterrence efforts.*** Threats and attacks highlight potential dangers exploiting vulnerabilities, emphasizing the need for a comprehensive security approach.

04. SECURITY SERVICES

STATE AND EXPLAIN DIFFERENT SECURITY SERVICES WITH RESPECT TO OSI X.800 MODEL. (EXAM 2019)

The ***OSI X.800 model*** defines ***five main categories*** of security services that can be implemented at different layers of the OSI model to protect data transmitted over a network. Here's a breakdown of each service with respect to the model:

1. Authentication:

- **Objective:**

  Making sure that the person or system you're communicating with is really who they say they are.

- **Types:**
  - **Peer Entity Authentication:** Used at connection establishment to ***verify the identities of both parties*** involved in the communication.
  - **Data-Origin Authentication:** ***Verifies the source of a single data unit*** without establishing a connection.

2. Access Control:

- **Objective:** ***Prevent unauthorized access to resources.***
- **Function:** ***Controls*** who can ***access a resource, under what conditions,*** and ***what they can do with it.***

3. Data Confidentiality:

- **Objective:** Protect data from ***unauthorized disclosure.***
- **Types:**
  - **Connection Confidentiality:** ***Secures all user data on a connection.***
  - **Connectionless Confidentiality:** Protects user data in a ***single data block.***
  - **Selective-Field Confidentiality:** ***Encrypts specific fields*** within a data block.
  - **Traffic-Flow Confidentiality:** Hides information about ***communication patterns*** such as ***source, destination, tunnel, etc.***

4. Data Integrity:

- **Objective:** Ensure data received is exactly as sent by an authorized entity.
- **Types:**
  - **Connection Integrity with Recovery:** ***Detects and attempts to recover*** from any ***modification, insertion, deletion, or replay*** of data within a connection.

  - **Connection Integrity without Recovery:** ***Only detects data integrity violations*** without recovery.
  - **Connectionless Integrity:** ***Detects data modification in a single data block.***
  - **Selective-Field Connection Integrity:** ***Verifies the integrity of specific fields within a data block*** on a connection.
  - **Selective-Field Connectionless Integrity:** Verifies the integrity of specific fields within a ***single connectionless data block.***

5. Non-repudiation:

- **Objective:** Prevent either party from ***denying participation in a communication.***
- **Types:**
  - **Non-repudiation, Origin:** ***Provides proof that a specific party sent a message.***
  - **Non-repudiation, Destination:** ***Provides proof that a specific party received a message.***

Availability Service:

- Not explicitly listed in X.800, but crucial for overall security.
- **Objective:** Ensure ***timely access to systems and data*** for authorized users.
- **Function:** Protects against ***denial-of-service attacks*** by ***managing and controlling system resources effectively.***

Remember, these services can be implemented at different layers of the OSI model ***depending on the specific needs and desired level of security.*** Choosing the right service and layer ensures efficient and effective protection for your data and communications.

05. SECURITY MECHANISM

The security mechanisms defined in OSI X.800 are ***divided into those that are implemented in a specific protocol layer,*** such as TCP or an application-layer protocol, and those that are not specific to any particular protocol layer. Here are some of the ***specific and pervasive security mechanisms:***

Specific Security Mechanisms:

1. **Encipherment:** The use of ***mathematical algorithms*** to transform data into a form that is not readily intelligible.
2. **Digital Signature:** Digital signature is like a special mark added to data. It lets the person who gets the data ***check where it came from and that it hasn't been tampered,*** helping to prevent fakes.
3. **Access Control:** A variety of mechanisms that ***enforce access rights to resources.***
4. **Data Integrity:** A variety of mechanisms used to assure the integrity of a data unit or stream of data units.

Pervasive Security Mechanisms:

1. **Trusted Functionality:** The data which is perceived to be correct ***with respect to some criteria*** (e.g., as established by a security policy).
2. **Security Label:** Security label is a tag attached to a resource (***like a piece of data***) that ***identifies its security properties.***
3. **Event Detection:** ***Detection of security-relevant events.*** For example, if someone tries to log into a system with the wrong password too many times, that's a security-relevant event that should be detected.
4. **Security Audit Trail:** Security audit trail refers to the gathering of data that might be used to help with a ***security check.***

These mechanisms are ***designed to recover from specific attacks*** at various protocol layers. They are ***used to implement security services*** and might operate by themselves or with others to provide a particular service. For example, encipherment can be achieved by two famous techniques named ***Cryptography and Encipherment.*** The level of data encryption is dependent on the algorithm used for encipherment.

06. NETWORK SECURITY MODEL

Network Security is like Securing a Building Just like you secure a building with valuables inside, ***network security aims to ensure safe communication*** and ***protect against unauthorized access.***

Network security model is responsible for:

1. Establishing a Secure Connection:

- **Secure Route**: Parties communicate over the Internet using ***specific routes and protocols (like TCP/IP)*** to ensure the ***clear and safe exchange of messages.***

2. Protecting the Message:

- **Transformation**: ***Messages are encrypted to create a secret code,*** ensuring that even if intercepted, they can't be understood without the key.
- **Key Distribution**: A ***secret key*** is shared between parties but kept ***hidden from others.***

3. Trusted Third Party Involvement:

In some cases, a ***trusted third party is involved to manage the secure sharing of the secret key*** or ***settle disputes*** about message authenticity, similar to having a mutual friend who holds a spare house key.

Essential Tasks in Network Security Model:

1. **Algorithm Design**: Creating a complex secret code (algorithm) that can't be cracked by opponents, ***ensuring intercepted messages can't be read.***
2. **Secret key Generation**: Generating a secret key that works with the algorithm, ***known only to authorized individuals/parties.***

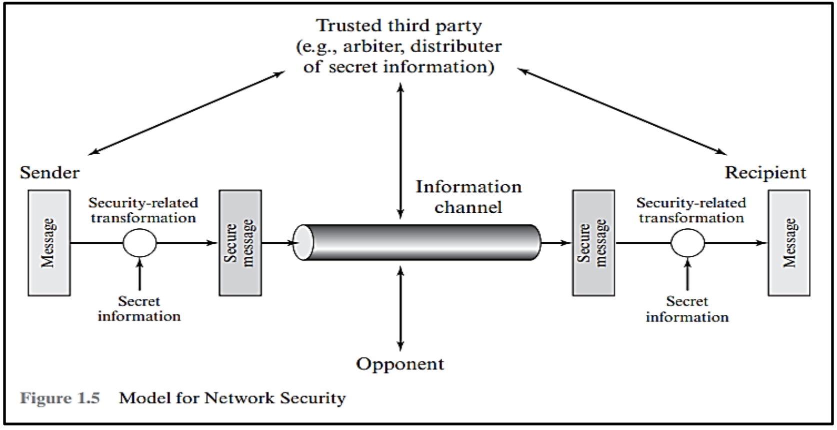


Figure 1.5 Model for Network Security

3. **Secret Distribution**: Securely sharing the secret key without exposing it to potential threats.

4. **Protocol Specification**: Agreeing on ***specific rules and processes (protocol)*** for using the ***secret key and algorithm*** to ensure ***secure message transmission.***

4. Protection Against Unauthorized Access:

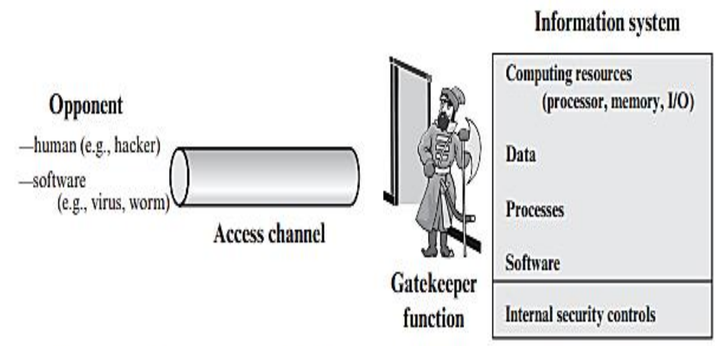


Figure 1.6 Network Access Security Model

- **Gatekeeper Functions**: Implementing security measures to prevent unauthorized access, including ***password-based logins*** and ***screening for threats like viruses or worms.***
- **Internal Security Controls**: Implementing measures to ***monitor and control access*** even after initial security checkpoints.

07. CLASSICAL ENCRYPTION TECHNIQUES

Classical encryption techniques are traditional methods of securing information before modern encryption methods were developed in the 1970s.

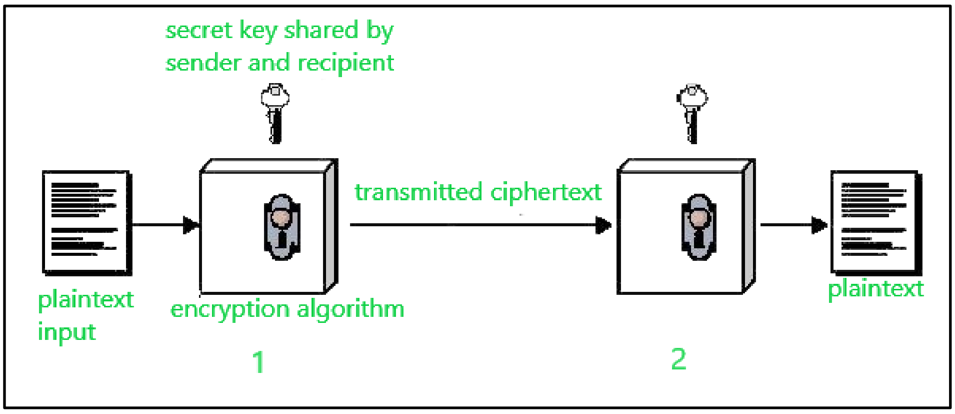
- **Symmetric Encryption:**
  - ***Classical encryption primarily uses symmetric encryption,*** where the same key is used to both lock (encrypt) and unlock (decrypt) the information.
- **Plaintext and Ciphertext:**
  - In classical encryption, the ***original message is called the "plaintext,"*** and it is transformed into a ***coded message called the "ciphertext"*** through encryption. The ***reverse process, turning ciphertext back into plaintext, is called decryption.***

- **Shared Secret Key:**
  - ***Classical encryption relies on a shared secret key between the parties communicating.*** This key is used to both encrypt and decrypt the information.
- **Common Algorithms:**
  - Common classical encryption algorithms include ***Substitution & Transposition ciphers.***
- **Cryptography and Cryptanalysis:**
  - The study of ***securing communication through encryption is called cryptography,*** while ***methods for deciphering messages without knowing the encryption details called as cryptanalysis.***
- **Usage:**
  - Classical encryption techniques, especially symmetric encryption, are still widely used in various applications, especially when a shared secret key can be securely managed.

07.A] SYMMETRIC CIPHER MODEL

EXPLAIN A MODEL OF CONVENTIONAL CRYPTOSYSTEM. (EXAM 2017)

Conventional cryptosystems, also known as ***symmetric encryption systems,*** rely on a ***single shared secret key*** for both encryption and decryption. This means ***both the sender and receiver need to have the same key*** to secure communication.



Here's a breakdown of the model:

Components:

- **Plaintext:** The original, unencrypted message you want to keep secret.
- **Secret Key:** A shared piece of information, ***like a password or string of characters,*** used for encryption and decryption.
- **Encryption Algorithm:** A ***mathematical process*** that transforms the plaintext into an unreadable format called ***ciphertext*** using the secret key.
- **Ciphertext:** The ***scrambled, unreadable version*** of the plaintext generated by the encryption algorithm.

- **Decryption Algorithm:** The process that reverses the encryption, transforming the ciphertext back into the original plaintext using the same secret key.

Steps:

1. **Encryption:** The sender uses the encryption algorithm and the secret key to convert the plaintext into ciphertext.
2. **Transmission:** The ciphertext is transmitted through an insecure channel (e.g., internet).
3. **Decryption:** The receiver uses the same secret key and the decryption algorithm to convert the ciphertext back into the original plaintext.

Security:

- The ***security of a conventional cryptosystem relies heavily on the secrecy of the shared key.*** If the key falls into the wrong hands, anyone can decrypt the communication.
- ***Strong encryption algorithms and long and complex keys are crucial for protecting against brute-force attacks*** where attackers try to guess the key.

Advantages:

- **Fast and efficient:** Since only one key is involved, encryption and decryption are typically faster compared to ***public-key cryptography.***
- **Suitable for large-scale data:** Efficient for encrypting large amounts of data due to its ***speed and minimal computational cost.***

Disadvantages:

- **Key management:** ***Securely sharing and managing the secret key can be challenging,*** especially when dealing with multiple users.
- **Key exposure risk:** If the key is compromised, ***all past and future communications using that key are vulnerable.***

Remember, no encryption system is fool proof. Understanding the model and its limitations is crucial for choosing the right security measures for your specific needs.

07.A.1] CRYPTOGRAPHY

Cryptography is the ***science and practice*** of ***securing communication and information*** through the use of ***encryption.*** It plays a crucial role in ensuring the ***confidentiality, integrity, and authenticity of data*** in various communication systems. Cryptographic systems are characterized by ***three*** independent dimensions:

1. **Type of Operations**: Cryptography transforms plaintext (the original message) into ciphertext (the encrypted message) using two main principles:
   - **Substitution**: Each element in the plaintext (like a bit, letter) is replaced by another element.

- **Transposition**: The ***elements in the plaintext are rearranged.*** The key point here is that ***no information is lost during these operations***.

2. **Number of Keys Used**: The type of encryption depends on the number of keys used:

   - **Symmetric Encryption**: If the ***sender and receiver use the same key,*** the system is called symmetric. It is also known as ***secret key cryptography.***

   - **Asymmetric Encryption**: If the sender and receiver use different keys, the system is known as asymmetric, ***public-key*** cryptography.

3. **Processing of Plaintext**: This refers to ***how the plaintext is handled during encryption:***

   - **Block Cipher**: This method processes the input one block of elements at a time, producing an output block for each input block.

   - **Stream Cipher**: This method processes the input elements continuously, producing output one element at a time.

In summary, cryptography is a method of ***protecting information by transforming it into an unreadable format.*** It can be ***characterized by the type of operations it uses, the number of keys used, and the way the plaintext is processed.*** The goal is to ensure that the information can only be read by the intended recipient.

07.A.2] CRYPTANALYSIS AND BRUTE-FORCE ATTACK

Cryptanalysis

Cryptanalysis is an approach to breaking encryption that ***relies on understanding the nature of the encryption algorithm,*** often combined with knowledge of the general characteristics of the plaintext or access to sample plaintext–ciphertext pairs. There are several ***types of cryptanalytic attacks*** based on the information known to the attacker:

1. **Ciphertext-Only Attack**
   - The ***attacker only has access to the ciphertext*** and may not know the encryption algorithm and secret key.
   - It is the most ***challenging scenario for the attacker.***
2. **Known Plaintext Attack**
   - The attacker ***knows some plaintext and its corresponding ciphertext,*** exploiting this knowledge to deduce the key.
   - The knowledge of ***plaintext-ciphertext pairs aids in determining the secret key with ease***.
3. **Chosen Plaintext Attack**
   - The attacker can ***choose plaintext messages and obtain their corresponding ciphertexts*** generated with the secret key.
4. **Chosen Ciphertext Attack**
   - The attacker ***cannot choose ciphertexts*** and obtain their corresponding plaintexts with the secret key. This attack focuses on ***analyzing existing ciphertexts*** and attempting to decrypt them using various techniques ***without direct control over their generation.***

Brute-Force Attack

A brute-force attack is a ***straightforward method where the attacker systematically tries every possible secret key until an intelligible translation of the ciphertext into plaintext is obtained.***

Key points about brute-force attacks:

- **Objective**: Recover/Find out the key used in encryption.
- **Process**: Try every possible key until successful.
- **Success Probability**: On an average, half of all possible keys need to be tried to achieve success.
- **Defence**: The defense of an encryption algorithm is its ***ability to make the process of finding the correct key (decrypting the data) extremely time-consuming*** and ***resource-intensive***.

Cryptanalysis and brute-force attacks emphasize the importance of designing encryption algorithms that are resistant to known methods of attack and have a sufficiently ***large key space*** to ***make exhaustive searches impractical.*** These considerations are vital in ensuring the security and confidentiality of encrypted data.

07.B] SUBSTITUTION TECHNIQUES

In classical encryption techniques, two fundamental building blocks are substitution and transposition. ***Substitution involves replacing elements in the plaintext*** (letters, bits, or symbols) with other elements. When applied to letters, it is commonly known as a substitution cipher.

07.B.1] CAESAR CIPHER:

The Caesar cipher is one of the earliest and simplest ciphers, attributed to ***Julius Caesar.*** The Caesar cipher is one of the simplest encryption techniques ever created. It's a type of ***substitution cipher,*** where ***each letter of the plaintext is replaced with another letter*** (fixed number of positions down the alphabet).

Example:

Let's say you want to encrypt the message "HELLO WORLD" with a ***shift value of 3.*** This means you will shift each letter by 3 positions down the alphabet:

Plaintext: H E L L O W O R L D

Ciphertext: K H O O R Z R Q R G

As you can see, each letter has been shifted 3 positions down. 'H' becomes 'K', 'E' becomes 'H', and so on.

Formulas:

The Caesar cipher uses simple mathematical formulas for encryption and decryption:

Encryption:

- ***C = (P + n) mod 26***

where:

- C is the ciphertext letter
- P is the plaintext letter
- n is the shift value (between 0 to 25)
- mod 26 ensures the result stays within the alphabet (A-Z)

Decryption:

- ***P = (C - n) mod 26***

The decryption formula is simply the encryption formula reversed.

Additional Notes:

- The Caesar cipher is ***very weak and easily broken,*** even by hand.
- It's only ***suitable for educational purposes*** or ***secret messages of no real importance.***
- Stronger encryption techniques exist that use more complex algorithms and longer keys, making them much harder to crack.

07.B.2] MONOALPHABETIC CIPHERS:

In monoalphabetic ciphers, each letter in the plaintext is replaced by a single, fixed cipher alphabet. This substitution technique ***significantly increases the key space compared to the Caesar cipher.*** In this cipher, each letter is mapped to exactly one other letter. The key in a monoalphabetic cipher is the mapping of each letter in the alphabet to another letter. Instead of having only 25 possible keys (as in Caesar cipher), monoalphabetic ciphers have ***4.03 * 10^26 possible keys.***

Let's take a simple example:

Step 1: Define the Secret Key

First, we need to define a key for our monoalphabetic cipher. Let's use the following key:

Mapping English alphabets with cipher key:

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

In this key, A maps to Q, B maps to W, C maps to E, and so on.

Step 2: Encryption

Now, let's encrypt a plaintext message using this key. Suppose our plaintext is HELLO.

Plain: H E L L O

Cipher: I T S S G

So, the ciphertext for HELLO is ITSSG.

Step 3: Decryption

To decrypt the message, we simply reverse the process using the same key:

Cipher: I T S S G

Plain: H E L L O

So, the plaintext for ITSSG is HELLO.

A ***monoalphabetic cipher is like a secret code*** where you swap out each letter in your message with another letter. For example, if 'A' becomes 'Q', 'B' becomes 'W', and so on, then the word 'HELLO' would be written as 'ITSSG.

But here's the thing: This kind of cipher ***isn't very secure.*** Why? Because in any language, some letters are used more often than others. In English, for example, 'E' is the most common letter. So, if you see a lot of 'W's in the coded message, you might guess that 'W' stands for 'E'. This is called ***frequency analysis.*** You can also look at common pairs of letters (like 'th' in English), or even whole words. If you see a three-letter word in the code that you think is 'the', that can give you three letters at once.

Once you start guessing some of the letters, you can use those guesses to help make more guesses, and so on, until you've cracked the whole code.

To make the ciphertext harder to crack, you could use ***multiple code letters for each real letter.*** For example, sometimes 'A' could be 'Q', and sometimes it could be 'Z'. This makes it harder to use frequency analysis. But even then, patterns might still show up that could give the code away.

So, while monoalphabetic ciphers are a fun introduction to codes, they're not secure enough for serious use. Modern codes use much more complex methods to keep messages secret.

0.7.B.3] PLAYFAIR CIPHER:

The Playfair cipher is based on ***symmetric encryption technique*** and was the first literal ***digraph substitution cipher.*** It was ***invented in 1854 by Charles Wheatstone*** but was named after ***Lord Playfair*** who promoted the use of the cipher.

The Playfair Cipher is a ***multiple-letter encryption cipher*** that ***treats digraph in the plaintext as single units and encrypts them into ciphertext digraph.*** It utilizes a ***5x5 matrix*** of letters constructed using a keyword. The matrix is filled in by placing the letters of the keyword ***(excluding duplicates)*** from ***left to right and top to bottom.*** The remaining spaces are filled with the remaining letters in alphabetic order, ***treating 'I' and 'J' as one letter.***

Rules for Encryption:

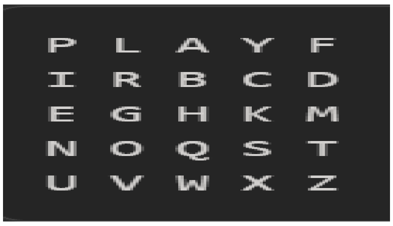
1. ***Repeating plaintext letters*** in the same pair are separated with a ***filler letter (commonly 'X').***
2. If two plaintext letters fall in the ***same row***, each is ***replaced by the letter to its right.***
3. If two plaintext letters fall in the same column, each is ***replaced by the letter beneath it.***
4. Otherwise, each plaintext letter in a pair is ***replaced by the letter in its own row and the column occupied by the other plaintext letter.***

Despite being considered unbreakable for a long time due to the large number of possible digraphs, the ***Playfair Cipher is relatively easy to break through cryptanalysis, as it still preserves much of the structure of the plaintext language.*** The cipher was used as the standard field system by the ***British Army in World War I*** and had significant use by the ***U.S. Army and Allied forces*** during ***World War II.***

Steps Involved:

Step 1: Key Matrix

The first step is to create a key table. This is a 5x5 matrix of letters, based on a keyword. Let's use the keyword ***PLAYFAIR***. We remove any duplicate letters from the keyword, then fill in the rest of the table with the remaining letters of the alphabet in order (excluding 'J' to fit the 5x5 grid).



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

Step 2: Encryption

Now, let's encrypt a plaintext message using this key. Suppose our plaintext is ***HELLO WORLD.***

We break the plaintext into ***digraphs (groups of two letters).*** If a digraph has two of the same letters, or if there's a single letter left over at the end, we insert an 'X' to separate them: HE LX LO WO RL DX.

Then, for each digraph, we find the letters in the key table by applying the rules for encryption:

Applying these rules, **HE LX LO WO RL DX → KG YV RV VQ GR CZ.**

Step 3: Decryption

So, **KG YV RV VQ GR CZ** decrypts back to HE LX LO WO RL DX, and removing the 'X's gives us back HELLO WORLD.

That's a basic example of a Playfair cipher! Remember, while this cipher is more ***secure than a simple monoalphabetic cipher,*** it's still ***not very secure by modern standards.*** Modern encryption algorithms use much more complex methods to secure information.

0.7.B.4] HILL CIPHER:

Hill cipher is a type of encryption method that uses **linear algebra and matrices to transform plaintext into ciphertext.** It works by ***dividing the plaintext into fixed size column vectors,*** and ***multiplying each block by a key matrix to get the ciphertext block.*** The key matrix must be invertible, which means that it has a unique inverse matrix that can be used to decrypt the ciphertext.

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
0	1	2	3	4	5	6	7	8	9
K	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**
10	11	12	13	14	15	16	17	18	19
U	**V**	**W**	**X**	**Y**	**Z**				
20	21	22	23	24	25				

To illustrate how hill cipher works, let us take an example with a ***2x2 key matrix*** and a plaintext of "ACT". The key matrix is:

$$\begin{bmatrix}6 & 24\\13 & 16\end{bmatrix}$$

The plaintext "ACT" is converted into numbers using the scheme A=0, B=1, ..., Z=25. Then, it is divided into blocks of size 2. Since the plaintext has an odd number of letters, we append a dummy letter Z (25) at the end to make it even. The plaintext blocks are:

$$\begin{bmatrix}0\\2\end{bmatrix}, \begin{bmatrix}19\\25\end{bmatrix}$$

To encrypt each vector, we multiply it by the key matrix modulo 26. For example, the first block is encrypted as:

Block 01:

$$\begin{bmatrix}6 & 24\\13 & 16\end{bmatrix}\begin{bmatrix}0\\2\end{bmatrix} \bmod 26 = \begin{bmatrix}48\\32\end{bmatrix} \bmod 26 = \begin{bmatrix}22\\6\end{bmatrix}$$

Block 02:

$$\begin{bmatrix}6 & 24\\13 & 16\end{bmatrix}\begin{bmatrix}19\\25\end{bmatrix} \bmod 26 = \begin{bmatrix}714\\569\end{bmatrix} \bmod 26 = \begin{bmatrix}12\\23\end{bmatrix}$$

The encrypted block is then converted back into letters using the same scheme. The first block becomes "WG". Similarly, the second block is encrypted as "MX". The final ciphertext is "WGMX".

To decrypt the ciphertext, we need to find the ***inverse of the key matrix modulo 26.*** This can be done by using the formula:

$$K^{-1} = \frac{1}{det(K)} \cdot adj(K)(mod26)$$

The decrypted block is then converted back into letters using the same scheme. The first block becomes "AC". Similarly, the second block is decrypted as "TX". The final plaintext is "ACT", where X is the dummy letter that can be ignored.

0.7.B.5] POLYALPHABETIC CIPHER

A ***polyalphabetic cipher*** is a type of encryption technique where ***each letter in a plaintext is substituted with a different letter,*** but ***not in a fixed way like a simple Caesar cipher.*** Instead, it uses ***multiple substitution alphabets*** to create a more complex and secure encryption. This means the ***same letter in the plaintext can be encrypted to different letters*** in the ciphertext, ***depending on its position in the message and the key used.***

Here's what makes ***polyalphabetic ciphers stronger than monoalphabetic*** ones:

- **Frequency analysis:** In monoalphabetic ciphers, the ***most frequent letter in the ciphertext usually corresponds to the most frequent letter in the plaintext*** (e.g., "E" in English). This pattern can be easily exploited to break the code. Polyalphabetic ciphers avoid this by using different alphabets, making it ***harder to identify letter frequencies.***
- **Statistical analysis:** Similar to frequency analysis, looking at ***letter pairs or longer sequences*** can help break monoalphabetic ciphers. ***Polyalphabetic ciphers mix things up,*** making statistical analysis less effective.

Vigenère Cipher:

The **Vigenère cipher** is one of the most famous polyalphabetic ciphers. It uses a ***secret key*** (a secret phrase) to determine which alphabet to use for each letter of the message. The Vigenère cipher was ***considered secure for centuries*** due to its complexity compared to monoalphabetic ciphers. However, with advancements in cryptanalysis, it was eventually broken. The ***key weakness lies in the repeating keyword,*** which creates predictable patterns in the ciphertext.

Example:

Let's encrypt the message **"we are discovered save yourself"** using the keyword **"deceptive":**

- **Keyword**: deceptive(deceptive...)
- **Plaintext**: wearediscoveredsaveyourself
- **Ciphertext**: ZICVTWQNGRZGVTWAVZHCQYGLMGJ

Let's break down the encryption process step by step to understand how the ciphertext ***"ZICVTWQNGRZGVTWAVZHCQYGLMGJ"*** is derived from the plaintext ***"wearediscoveredsaveyourself"*** using the keyword ***"deceptive"*** in the Vigenère cipher.

Step 1: Repeat the Keyword

First, ***we repeat the keyword "deceptive" to match the length of the plaintext:***

Keyword: deceptive(deceptive...)

Plaintext: wearediscoveredsaveyourself

Step 2: Assign Numerical Values

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
0	1	2	3	4	5	6	7	8	9
K	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**
10	11	12	13	14	15	16	17	18	19
U	**V**	**W**	**X**	**Y**	**Z**				
20	21	22	23	24	25				

Plaintext	**w**	**e**	**a**	**r**	**e**	**d**	**i**	**s**	**c**	**o**	**v**	**e**	**r**	**e**	**d**	**s**	**a**	**v**	**e**	**y**	**o**	**u**	**r**	**s**	**e**	**l**	**f**
Numeric	22	4	0	17	4	3	8	18	2	14	21	4	17	4	3	18	0	21	4	24	14	20	17	18	4	11	5
Keyword	**d**	**e**	**c**	**e**	**p**	**t**	**i**	**v**	**e**	**d**	**e**	**c**	**e**	**p**	**t**	**i**	**v**	**e**	**d**	**e**	**c**	**e**	**p**	**t**	**i**	**v**	**e**
Numeric	3	4	2	4	15	19	8	21	4	3	4	2	4	15	19	8	21	4	3	4	2	4	15	19	8	21	4

Step 3: Encrypt Each Letter

Now, we encrypt each letter of the plaintext using the corresponding letter of the repeated keyword. ***Encryption involves adding the numeric values of the plaintext and the keyword*** modulo 26.

Plaintext	22	4	0	17	4	3	8	18	2	14	21	4	17	4	3	18	0	21	4	24	14	20	17	18	4	11	5
Keyword	3	4	2	4	15	19	8	21	4	3	4	2	4	15	19	8	21	4	3	4	2	4	15	19	8	21	4
Ciphertext	25	8	2	21	19	22	16	13	6	17	25	6	21	19	22	0	21	25	7	2	16	24	6	11	12	6	9

Step 4: Convert Back to Letters

Finally, we convert the numeric values of the ciphertext back to letters based on their positions in the alphabet:

Ciphertext: Z I C V T W Q N G R Z G V T W N C Z I C Q Y G R L G J

0.7.B.6] ONE TIME PAD

One time pad cipher is a type of substitution cipher that ***uses a random key of the same length as the message*** to transform each letter of the message into a different letter. The ***key is used only once and then discarded,*** making the cipher unbreakable.

For example, suppose you want to encrypt the message ***"HELLO"*** with the key ***"XMCKL".*** You can use the following table to convert each letter into a number from 0 to 25:

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**
0	1	2	3	4	5	6	7	8	9
K	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**
10	11	12	13	14	15	16	17	18	19
U	**V**	**W**	**X**	**Y**	**Z**				
20	21	22	23	24	25				

The message “HELLO” becomes:

7 4 11 11 14

The key “XMCKL” becomes:

23 12 2 10 11

To encrypt the message, you add the message and the key modulo 26, which means you divide the sum by 26 and take the remainder. For example, the first letter is encrypted as:

$$7 + 23 \bmod 26 = 30 \bmod 26 = 4$$

The encrypted numeric value is then converted back into a letter using the same table. The first letter becomes E. Similarly, the rest of the letters are encrypted as:

$$4 + 12 \bmod 26 = 16 \bmod 26 = 16$$

$$11 + 2 \bmod 26 = 13 \bmod 26 = 13$$

$$11 + 10 \bmod 26 = 21 \bmod 26 = 21$$

$$14 + 11 \bmod 26 = 25 \bmod 26 = 25$$

The encrypted letters are Q, N, V, and Z. The ***final ciphertext is “EQNVZ”.***

To decrypt the ciphertext, you subtract the key from remainder values, which means you add 26 if the difference is negative. For example, the first letter is decrypted as:

$$4 - 23 \bmod 26 = -19 \bmod 26 = 7$$

The decrypted letter is then converted back into a letter using the same table. The first letter becomes H. Similarly, the rest of the letters are decrypted as:

$$16 - 12 \bmod 26 = 4\mathrm{mod}\ 26 = 4$$

$$13 - 2 \bmod 26 = 11\mathrm{mod}\ 26 = 11$$

$$21 - 10 \bmod 26 = 11\mathrm{mod}\ 26 = 11$$

$$25 - 11 \bmod 26 = 14\mathrm{mod}\ 26 = 14$$

The decrypted letters are H, E, L, L, and O. The ***final plaintext is “HELLO”.***

07.B] TRANSPOSITION TECHNIQUES (Refer Tutorial No: 01)