Unit 1

Introduction to the Concepts of Security

MKK

INTRODUCTION

- Computer data often travels from one computer to another, leaving the safety of its protected physical surroundings.
- Once the data is out of hand, people with bad intention could modify or forge your data, either for amusement or for their own benefit.
- Cryptography can reformat and transform our data, making it safer on its trip between computers.
- The technology is based on the essentials of secret codes, augmented by modern mathematics that protects our data in powerful ways.

MKK

THE NEED FOR SECURITY

- Most previous computer applications had no, or at best, very little security. This continued for a number of years until the importance of data was truly realized.
- Until then, computer data was considered to be useful, but not something to be protected. When computer applications were developed to handle financial and personal data, the real need for security was felt like never before.
- People realized that data on computers is an extremely important aspect of modern life.

Therefore, various areas in security began to gain prominence.

Two typical examples of such security mechanisms were as follows:

- Provide a user identification and password to every user, and use that information to authenticate a user.
- Encode information stored in the databases in some fashion, so that it is not visible to users who do not have the right permission.

Soon, people realized the basic security measures were not quite enough as Internet took the world by storm.

Modern Nature of Attacks

Changes in computer-based systems are mainly due to the speed at which things happen and the accuracy that we get, as compared to the traditional world.

- salient features of the modern nature of attacks, as follows:

1. Automating Attacks: An automated threat is a type of computer security threat to a computer network or web application, characterised by the malicious use of automated tools such as Internet bots. Automated threats are popular on the internet as they can complete large amounts of repetitive tasks with almost no cost to execute.

2. Privacy Concerns: Collecting information about people and later (mis)using it is turning out to be a huge problem these days. The so-called data mining applications gather, process, and tabulate all sorts of details about individuals.

3. Distance Does not Matter: A modern thief would perhaps not like to wear a mask and attempt a robbery! Instead, it is far easier and cheaper to attempt an attack on the computer systems of the bank while sitting at home! It may be far more prudent for the attacker to break into the bank's servers, or steal credit card/ATM information from the comforts of his/her home or place of work.

SECURITY APPROACHES

Trusted Systems

- A trusted system is a computer system that can be trusted to a specified extent to enforce a specified security policy.
- Trusted systems often use the term reference monitor.
- Naturally, following are the expectations from the reference monitor:

(a) It should be tamper-proof.

(b) It should always be invoked.

(c) It should be small enough so that it can be tested independently.

Security Models

- An organization can take several approaches to implement its security model.

1. No Security:

In this simplest case, the approach could be a decision to implement no security at all.

2. Security through Obscurity:

In this model, a system is secure simply because nobody knows about its existence and contents.

3. Host Security:

In this scheme, the security for each host is enforced individually. This is a very safe approach, but the trouble is that it cannot scale well.

4. Network Security:

Host security is tough to achieve as organizations grow and become more diverse. In this technique, the focus is to control network access to various hosts and their services, rather than individual host security. This is a very efficient and scalable model.

MKK

Security-Management Practices

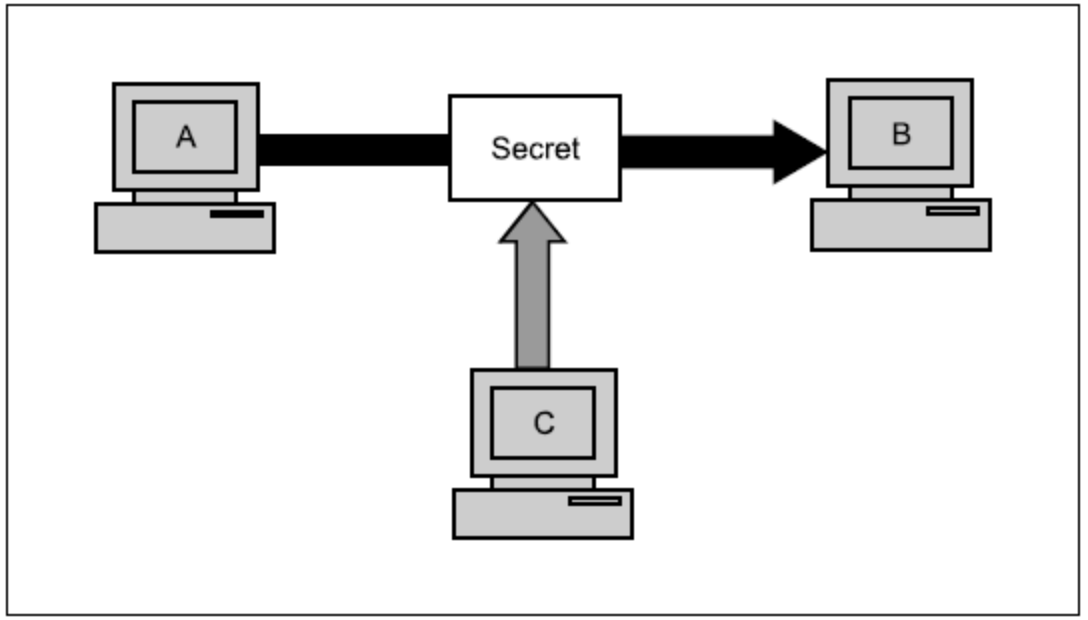
- Good security-management practices always talk of a security policy being in place.
- A good security policy generally takes care of four key aspects, as follows.
  - Affordability How much money and effort does this security implementation cost?
  - Functionality What is the mechanism of providing security?
  - Cultural Issues Does the policy complement the people's expectations, working style and beliefs?
  - Legality Does the policy meet the legal requirements?

PRINCIPLES OF SECURITY

The Principles of Security can be classified as follows:

Confidentiality:

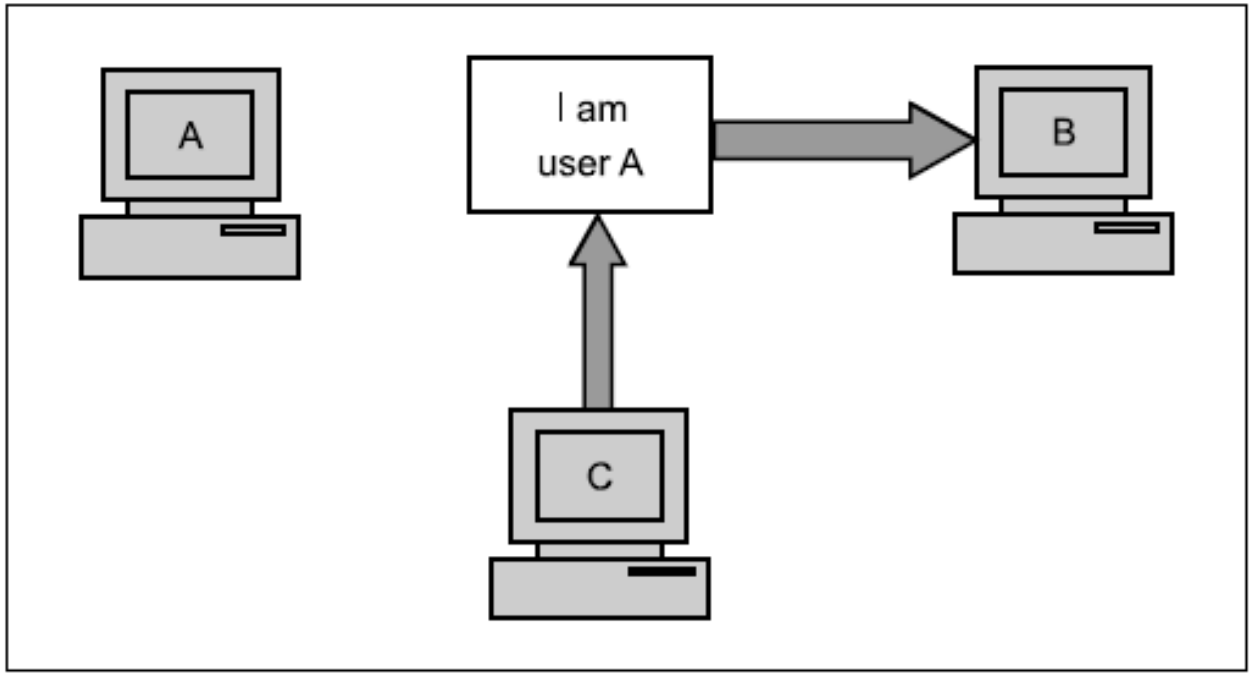
- The degree of confidentiality determines the secrecy of the information. The principle specifies that only the sender and receiver will be able to access the information shared between them. Confidentiality compromises if an unauthorized person is able to access a message.
- For example, let us consider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



MKK

Authentication:

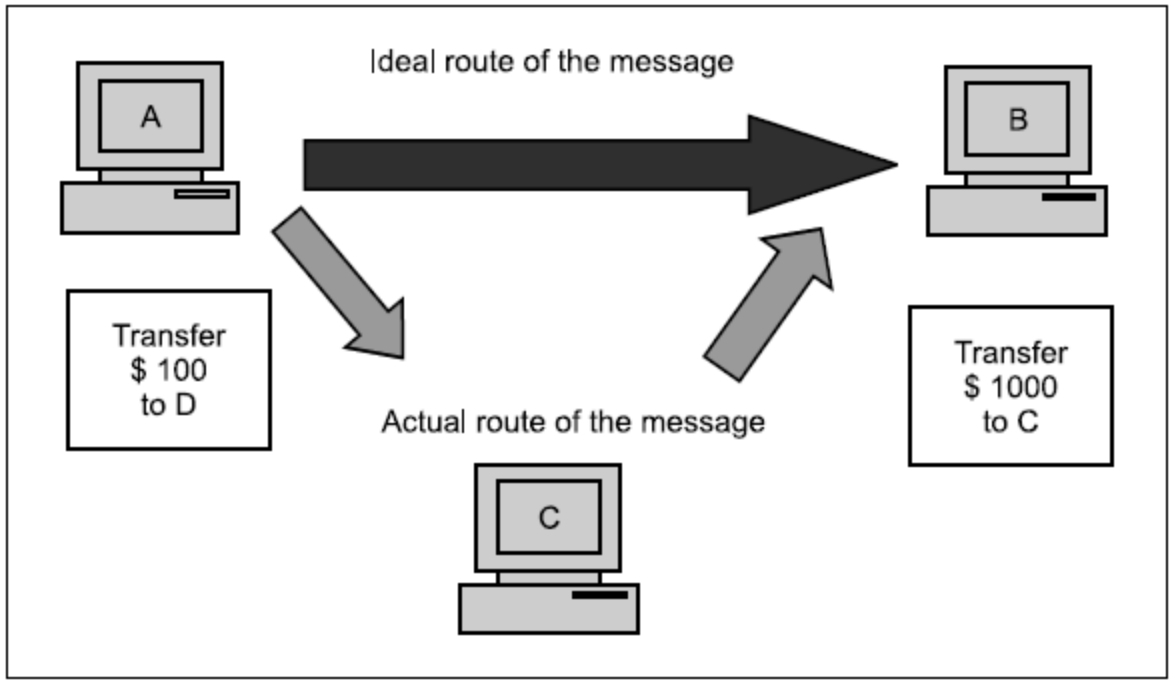
- Authentication is the mechanism to identify the user or system or the entity. It ensures the identity of the person trying to access the information. The authentication is mostly secured by using username and password. The authorized person whose identity is preregistered can prove his/her identity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

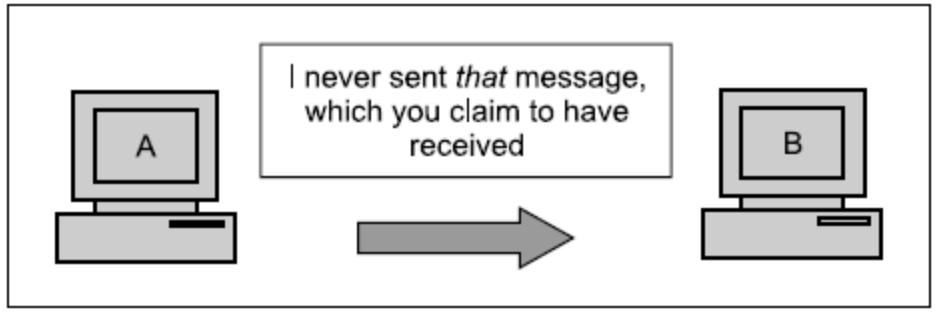
Example:
suppose A write a check for $100 to pay for goods bought from the US.
However, when A see in his next account statement, he will startled to see that the check resulted in a payment of $1000!
This is the case for loss of message integrity.
Here, user C tampers with a message originally sent by user A, which is actually destined for user B. User C somehow manages to access it, change its contents, and send the changed message to user B. User B has no way of knowing that the contents of the message were changed after user A had sent it. User A also does not know about this change. This type of attack is called modification.
Modification causes loss of message integrity.



MKK

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.



- These are the four chief principles of security. There are two more: ***access control and availability***, which are not related to a particular message, but are linked to the overall system as a whole.

MKK

Access control:

- The principle of access control determines who should be able to access what.
  For instance, we should be able to specify that user A can view the records in a database, but cannot update them. However, user B might be allowed to make updates as well. An access-control mechanism can be set up to ensure this.
- Access control is broadly related to two areas: ***role management and rule management.***
- Role management concentrates on the user side (which user can do what), whereas rule management focuses on the resources side (which resource is accessible, and under what circumstances).

Availability:

- The principle of availability states that resources (i.e. information) should be available to authorized parties at all times.
- For example, due to the intentional actions of another unauthorized user C, an authorized user A may not be able to contact a server computer B, as shown in Fig. This would defeat the principle of availability. Such an attack is called *interruption*.

MKK

The OSI standard for Security Model (titled OSI Security Model 7498-2). This also defines seven layers of security in the form of

- Authentication
- Access control
- Non-repudiation
- Data integrity
- Confidentiality
- Assurance or availability
- Notarization or signature

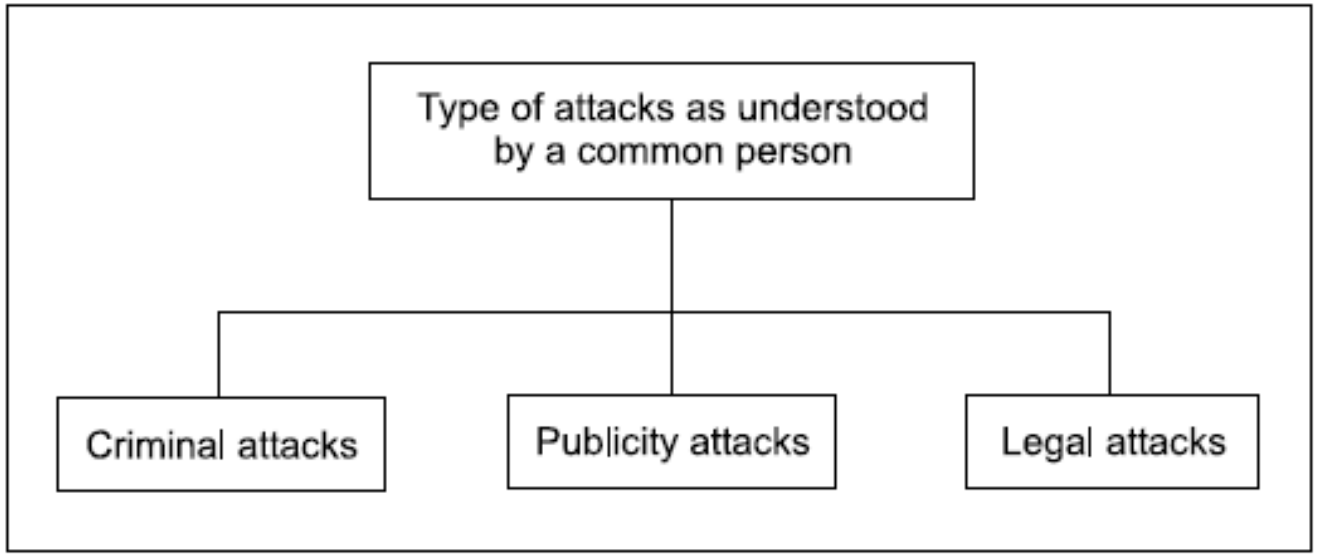
MKK

Types of Attacks

- We shall classify attacks with respect to two views: the common person's view and a technologist's view.

Attacks: A General View

From a common person's point of view, we can classify attacks into three categories



MKK

1. Criminal Attacks

Criminal attacks are the simplest to understand. Here, the sole aim of the attackers is to maximize financial gain by attacking computer systems, like:

- *Fraud*: concentrate on manipulating some aspects of electronic currency
- *Scams*: the most common ones being sale of services, auctions, multilevel marketing schemes, general merchandise, and business opportunities,etc., A very common example is the Nigeria scam
- *Destruction*: Some sort of grudge is the motive behind such attacks. where authorized users of these sites failed to log in or access these sites.
- *Identity theft*: best understood quote-Why steal from someone when you can just become that person?
- *Intellectual property theft*: ranges from stealing companies' trade secrets, databases, digital music and videos, electronic documents and books, software, and so on.
- *Brand theft*: It is quite easy to set up fake Web sites that look like real Web sites. The attackers use these details to then access the real site, causing an identity theft.

2. Publicity Attacks

- Publicity attacks occur because the attackers want to see their names appear on television news channels and newspapers. History suggests that these types of attackers are usually not hardcore criminals.
- One form of publicity attacks is to damage (or deface) the Web pages of a site by attacking it.

3. Legal Attacks

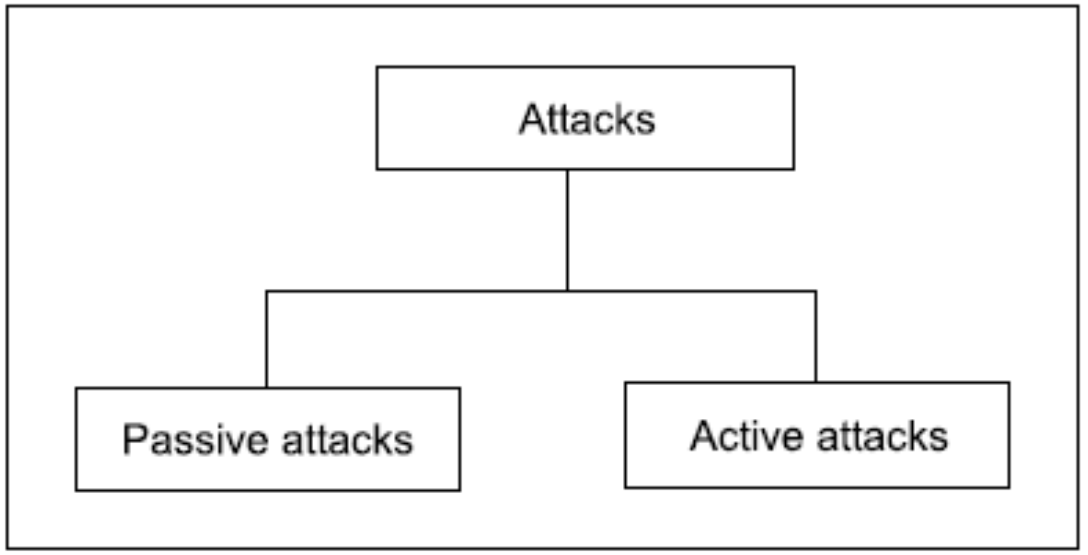
- This form of attack is quite novel and unique. Here, the attacker tries to make the judge or the jury doubtful about the security of a computer system.
- This works as follows.
  - The attacker attacks the computer system, and the attacked party (say a bank or an organization) manages to take the attacker to the court. While the case is being fought, the attacker tries to convince the judge and the jury that there is inherent weakness in the computer system and that he/she has done nothing wrongful.
- The aim of the attacker is to exploit the weakness of the judge and the jury in technological matters.

MKK

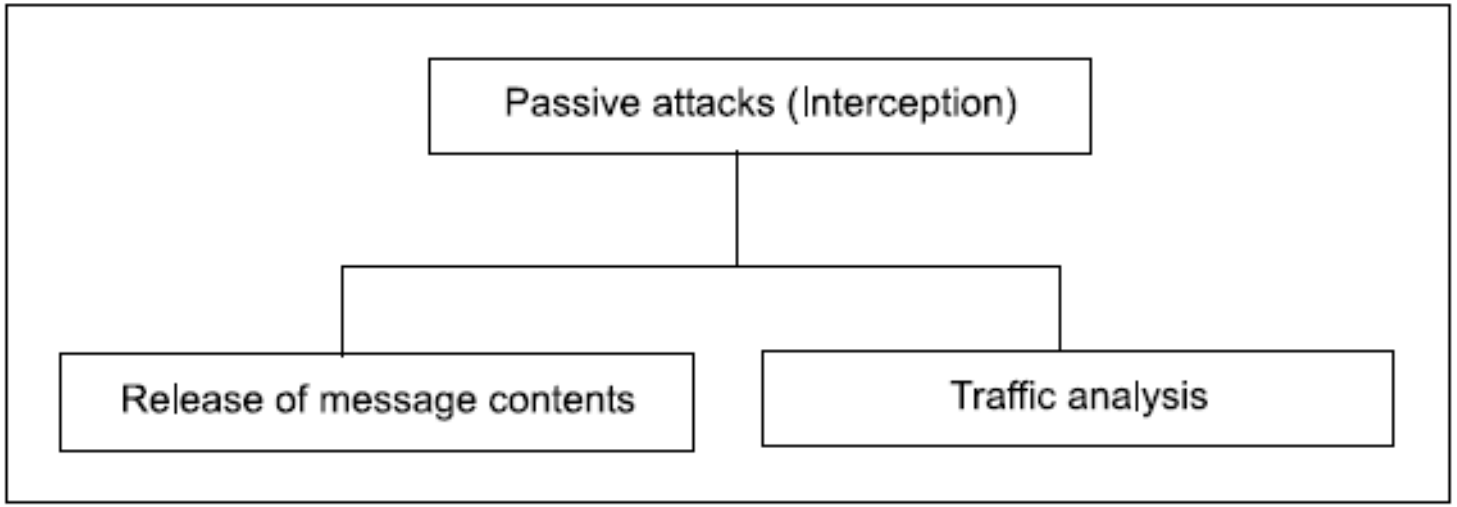
Attacks: A Technical View

- Attacks are further grouped into two types:

  passive attacks and active attacks



(a) ***Passive Attacks*** are those wherein the attacker indulges in eavesdropping or monitoring of data transmission. In other words, the attacker aims to obtain information that is in transit. The term passive indicates that the attacker does not attempt to perform any modifications to the data.



MKK

- Passive attacks into two sub-categories:
- ***Release of message contents*** is quite simple to understand. When you send a confidential email message to your friend, you desire that only he/she be able to access it. Otherwise, the contents of the message are released against our wishes to someone else. Using certain security mechanisms, we can prevent the release of message contents. For example, we can encode messages using a code language, so that only the desired parties understand the contents of a message, because only they know the code language. However, if many such messages are passing through, a passive attacker could try to figure out similarities between them to come up with some sort of pattern that provides her some clues regarding the communication that is taking place. Such attempts of analyzing (encoded) messages to come up with likely patterns are the work of the ***traffic-analysis attack.***

(b) ***Active Attacks*** Unlike passive attacks, the active attacks are based on the modification of the original message in some manner, or in the creation of a false message. These attacks cannot be prevented easily. However, they can be detected with some effort, and attempts can be made to recover from them.

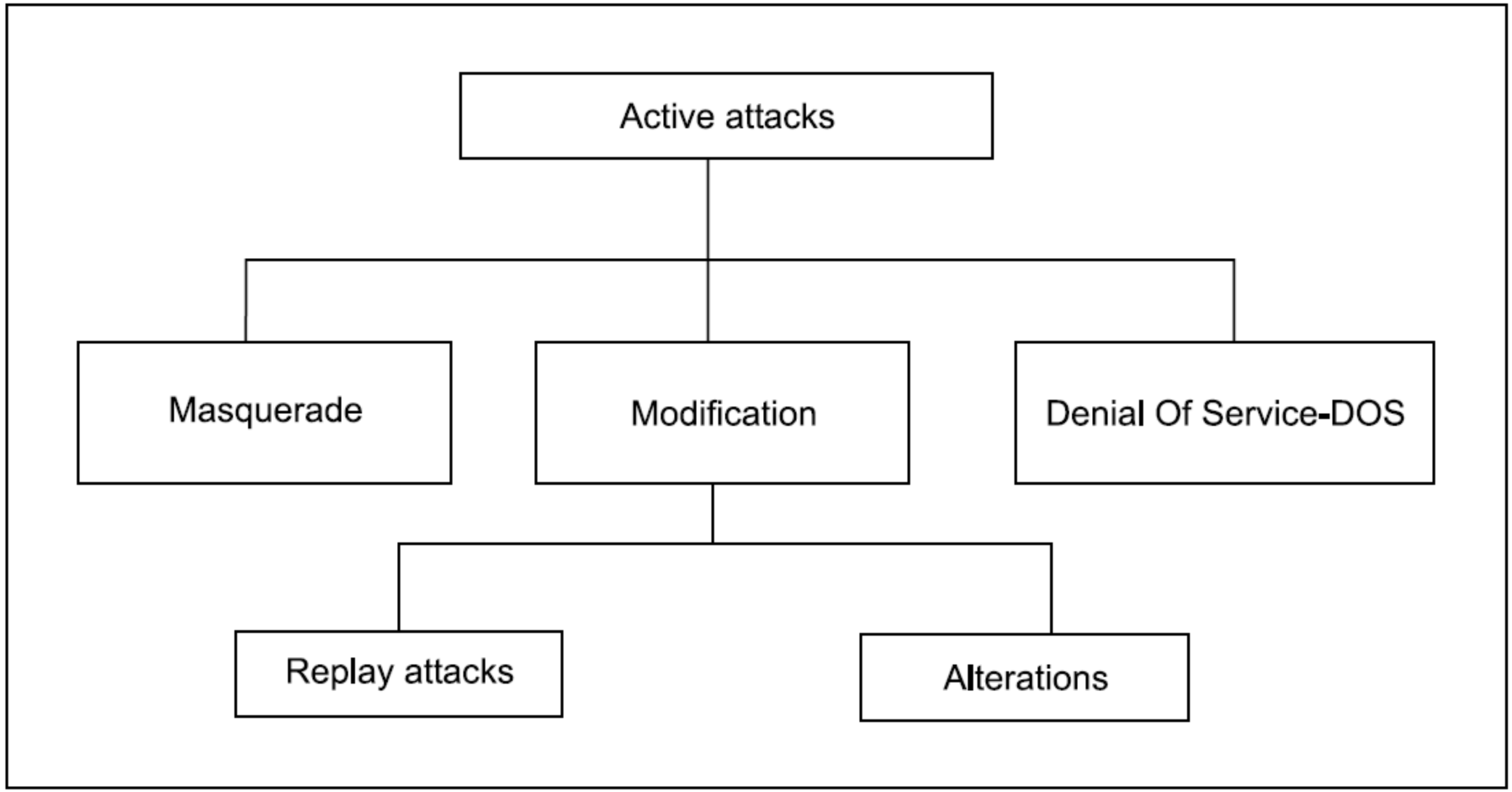
- These attacks can be in the form of interruption, modification and fabrication.

In active attacks, the contents of the original message are modified in some way.

● Trying to pose as another entity involves masquerade attacks.

● Modification attacks can be classified further into replay attacks and alteration of messages.

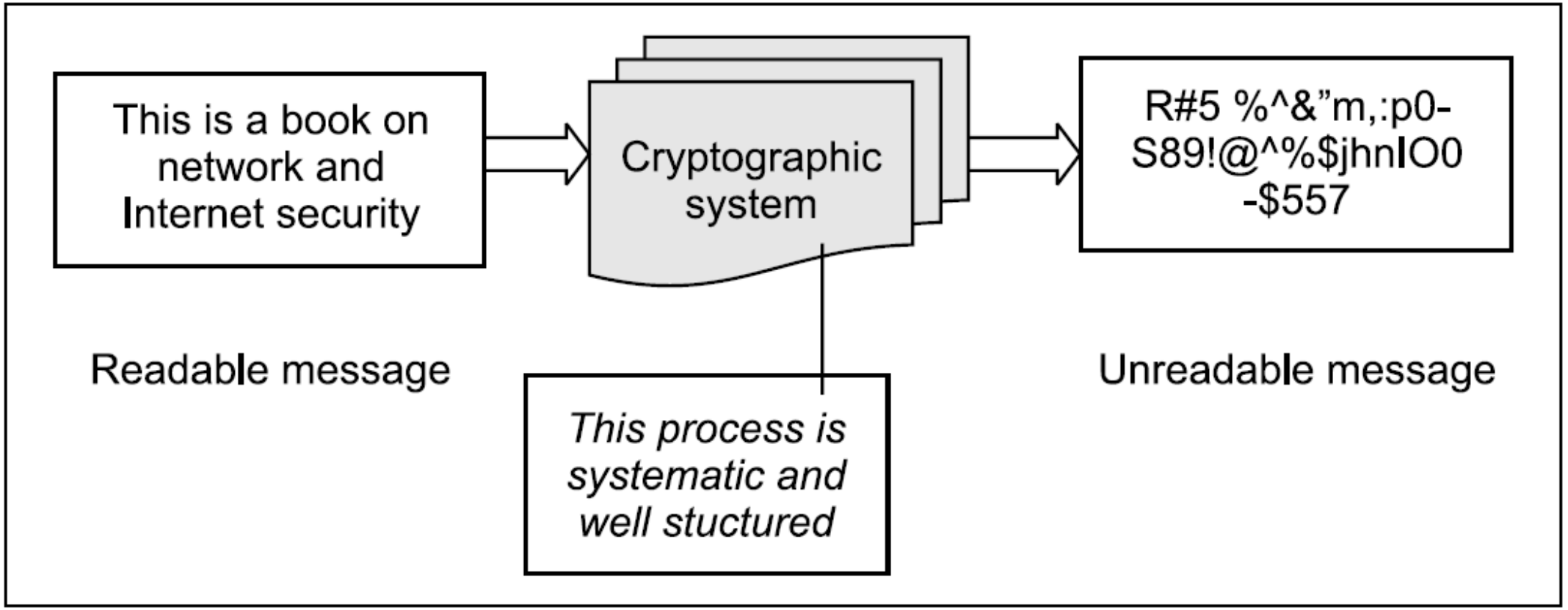
● Fabrication causes Denial Of Service (DOS) attacks.


Active attacks
Masquerade
Modification
Denial Of Service-DOS
Replay attacks
Alterations

- ***Masquerade*** is caused when an unauthorized entity pretends to be another entity. As we have seen, user C might pose as user A and send a message to user B. User B might be led to believe that the message indeed came from user A.
- In a ***replay attack***, a user captures a sequence of events, or some data units, and re-sends them. For instance, suppose user A wants to transfer some amount to user C's bank account. Both users A and C have accounts with bank B. User A might send an electronic message to bank B, requesting for the funds transfer. User C could capture this message, and send a second copy of the same to bank B. Bank B would have no idea that this is an unauthorized message, and would treat this as a second, and different, funds transfer request from user A. Therefore, user C would get the benefit of the funds transfer twice: once authorized, once through a replay attack.
- ***Alteration of messages*** involves some change to the original message. For instance, suppose user A sends an electronic message Transfer $1000 to D's account to bank B. User C might capture this, and change it to Transfer $10000 to C's account
- ***Denial Of Service (DOS)*** attacks make an attempt to prevent legitimate users from accessing some services, which they are eligible for. For instance, an unauthorized user might send too many login requests to a server using random user ids in quick succession, so as to flood the network and deny other legitimate users to use the network facilities.

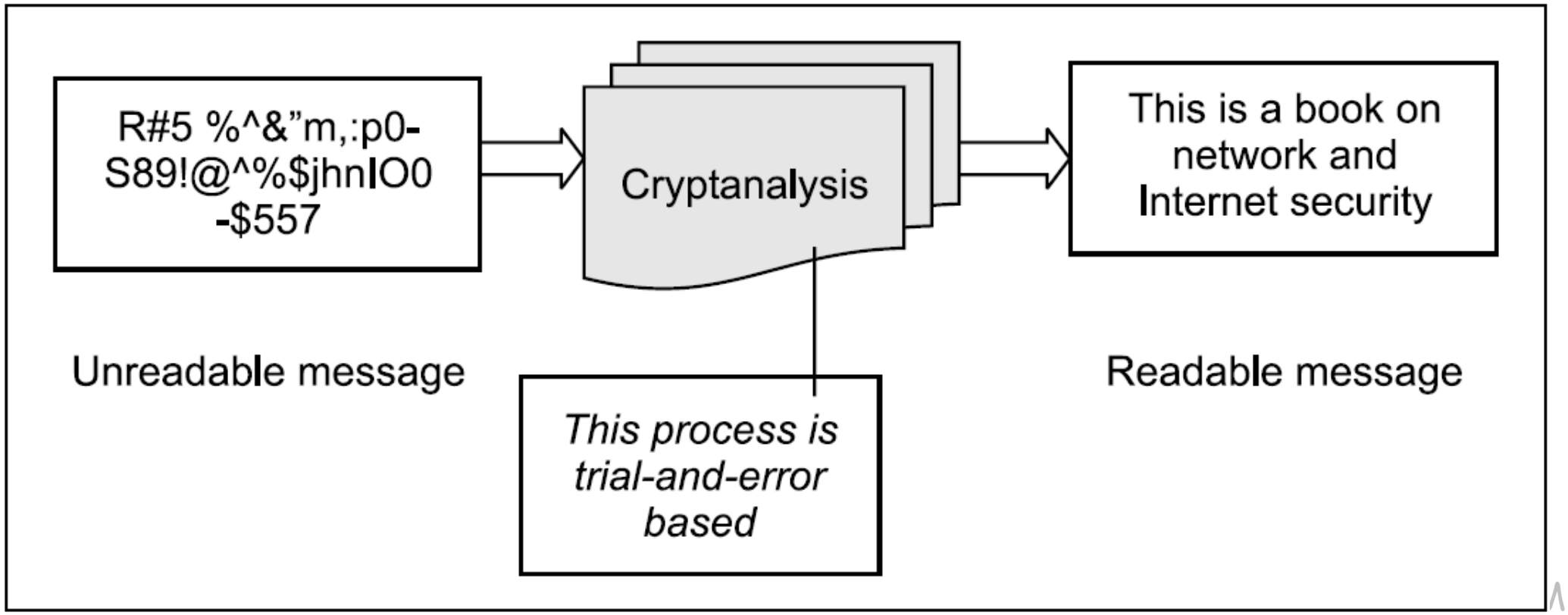
CRYPTOGRAPHY TECHNIQUES

- ***Cryptography*** is the art of achieving security by encoding messages to make them non-readable.



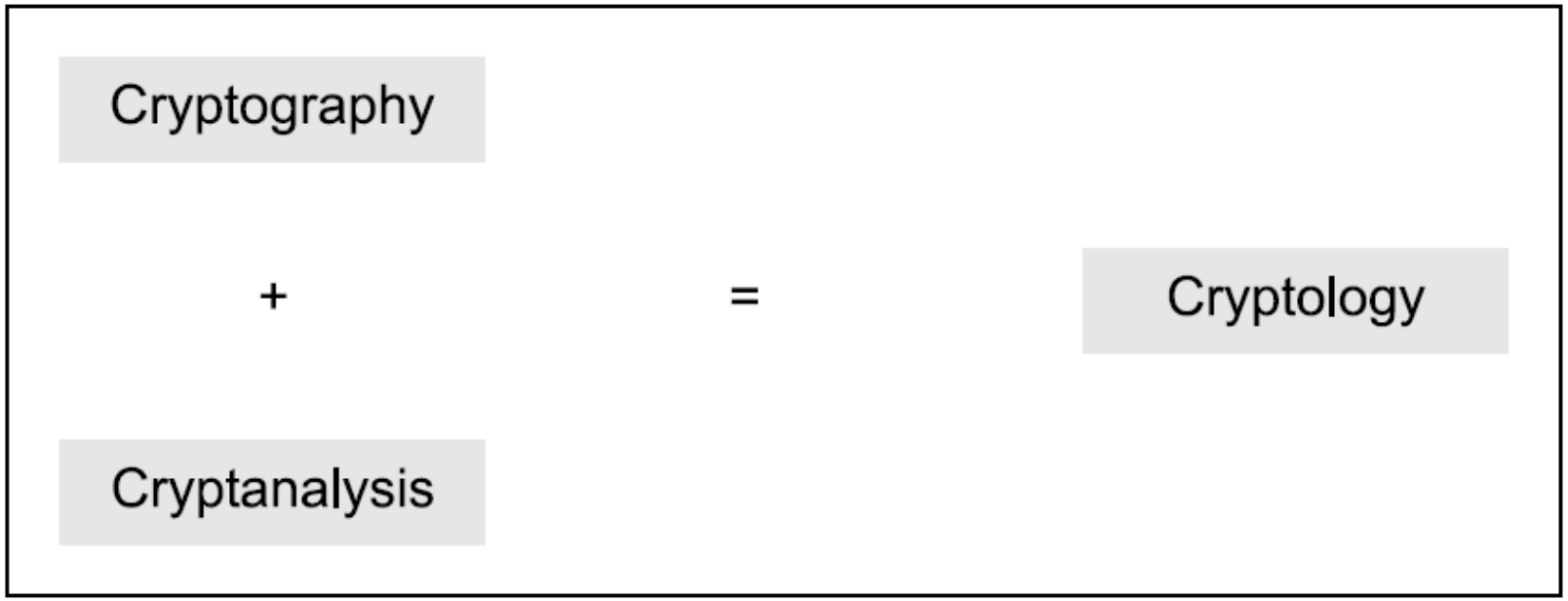
Cryptographic system

- ***Cryptanalysis*** is the technique of decoding messages from a non-readable format back to a readable format without knowing how they were initially converted from readable format to non readable format.
- In other words, it is like breaking a code.



Cryptanalysis

- ***Cryptology*** is a combination of cryptography and cryptanalysis.



Cryptography + Cryptanalysis = Cryptology

Plain text and Cipher text

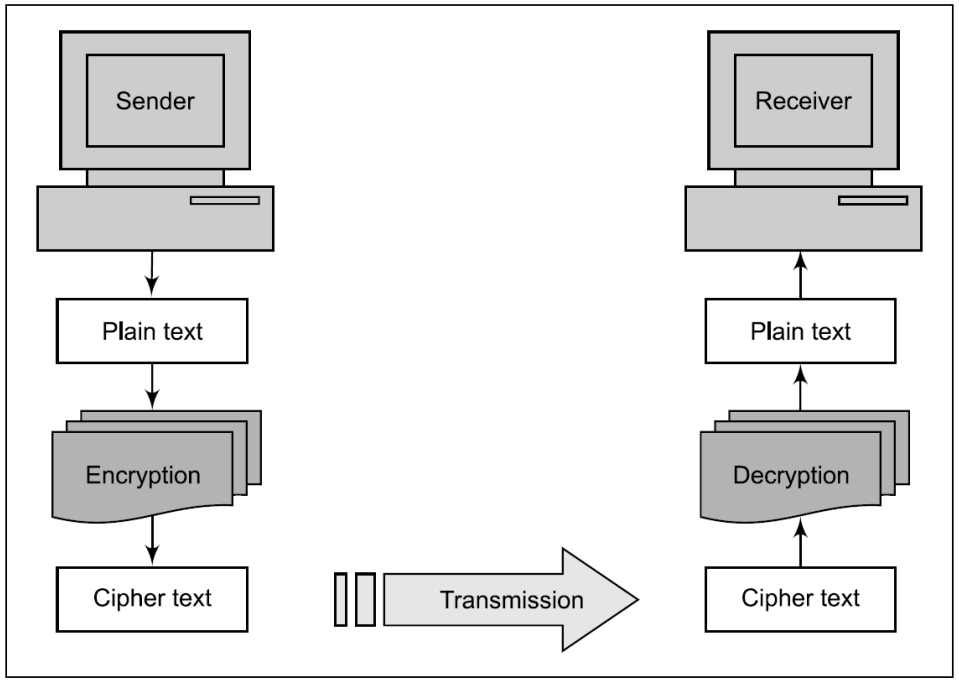
- Any communication in the language —that is the human language—takes the form of plain text or clear text.
- *Clear text, or plain text, signifies a message that can be understood by the sender, the recipient, and also by anyone else who gets access to that message.*
- For instance, if we replace each alphabet in the conversation with another character.
  - As an example, replace each alphabet with the alphabet that is actually three alphabets down the order. So, each A will be replaced by D, B will be replaced by E, C will be replaced by F, and so on. To complete the cycle, each W will be replaced by Z, each X will be replaced by A, each Y will be replaced by B and each Z will be replaced by C.
- We can summarize this scheme:

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

A scheme for codifying messages (replacing each alphabet with an alphabet three places down the line)

MKK

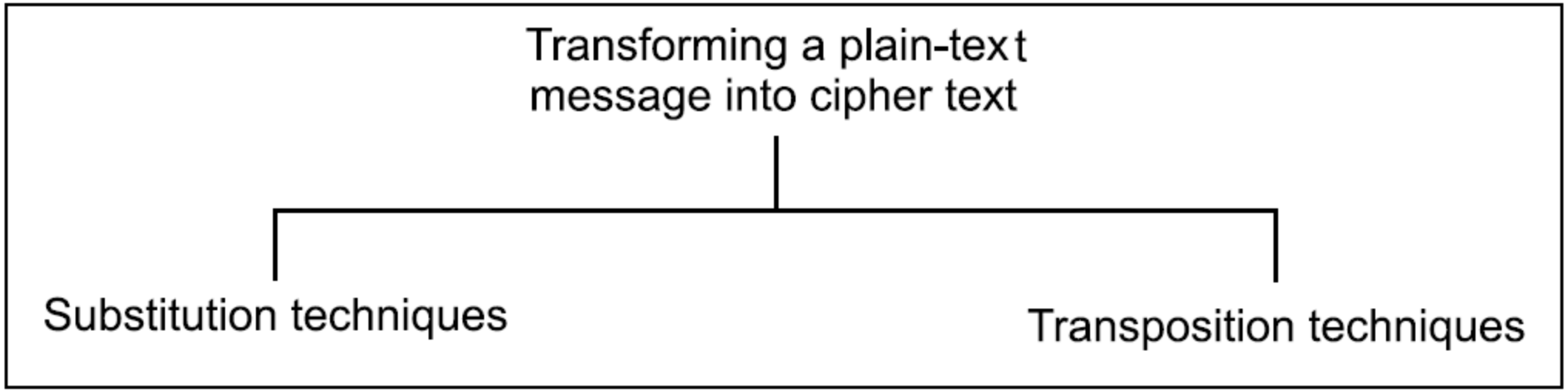
- Of course, there can be many variants of such a scheme. It is not necessary to replace each alphabet with the one that is three places down the order. It can be the one that is four, five or more places down the order.
- The codified message is called **cipher text**.
- Cipher means a code or a secret message.
- *When a plain-text message is codified using any suitable scheme, the resulting message is called cipher text.*



Elements of a cryptographic operation

MKK

- There are two primary ways in which a plain-text message can be codified to obtain the corresponding cipher text: **substitution** and **transposition**.



*Note that when the two approaches are used together, we call the technique **product cipher**.*

MKK

Substitution Techniques

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

Caesar cipher (or) shift cipher

- The earliest known use of a substitution cipher and the simplest was by Julius Caesar.
- The Caesar cipher involves replacing each letter of the alphabet with the letter standing 3 places further down the alphabet.
  - e.g., plain text : pay more money
  - Cipher text: sdb pruh prqhb

Note that the alphabet is wrapped around, so that letter following „z‟ is „a‟.

- For each plaintext letter p, substitute the cipher text letter c such that

$$C = E(p) = (p+3) \bmod 26$$

- A shift may be any amount, so that general Caesar algorithm is

$$C = E(p) = (p+k) \bmod 26$$

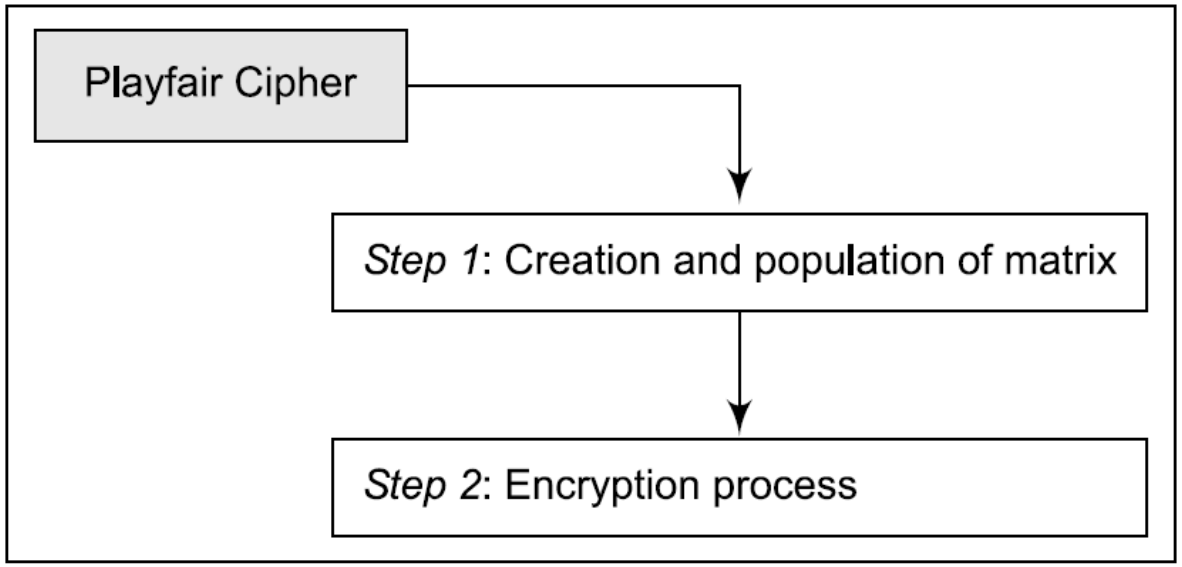
- Where k takes on a value in the range 1 to 25. The decryption algorithm is simply

$$P = D(C) = (C-k) \bmod 26$$

MKK

Playfair Cipher

- The Playfair cipher, also called Playfair square, is a cryptographic technique used for manual encryption of data. This scheme was invented by Charles Wheatstone in 1854. The Playfair cipher was used by the British army in World War I and by the Australians in World War II.
- This was possible because the Playfair cipher is quite fast to use and does not demand any special equipment to be used. It was used to protect important but not very critical information, so that by the time the cryptanalysts could break it, the value of the information was nullified anyway!
- The Playfair encryption scheme uses two main processes



MKK

Step 1: Creation and Population of Matrix

- The Playfair cipher makes use of a 5 x 5 matrix (table), which is used to store a keyword or phrase that becomes the key for encryption and decryption. The way this is entered into the 5 x 5 matrix is based on some simple rules

1. Enter the keyword in the matrix row-wise: left-to-right, and then top-to-bottom.
2. Drop duplicate letters.
3. Fill the remaining spaces in the matrix with the rest of the English alphabets (A-Z) that were not a part of our keyword. While doing so, combine I and J in the same cell of the table. In other words, if I or J is a part of the keyword, disregard both I and J while filling the remaining slots.

MKK

For example, suppose that our keyword is **PLAYFAIR EXAMPLE**. Then, the 5 x 5 matrix containing our keyword will look as shown
in Fig.

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

Keyword matrix for our example

- Our keyword was PLAYFAIR EXAMPLE
- Our first two rows were P L A Y F and I R E X M.

- Let us now change the font of the alphabets in our keyword which are covered anywhere in these two rows, in italics. Let us also indicate duplicates with an underscore. This would make our keyword look like this:

PLAYFAIR EXAMPLE

As we can see, all of our keyword alphabets are covered now (because every alphabet is (a) either in italics, indicating that it is a part of our matrix, or is (b) underlined, indicating that it is a duplicate).

MKK

Step 2: Encryption Process

- To encrypt a message using the Playfair cipher, the plaintext message is first divided into pairs of letters. If there is an odd number of letters, a "dummy" letter such as "X" is added at the end to make the message even. Each pair of letters is then encrypted using the following steps:

1. *If the two letters are the same, add a "dummy" letter such as "X" between them.*
2. *Locate the two letters in the grid and find their positions (row and column).*
3. *If the two letters are in the same row, replace each letter with the letter to its right (wrapping around to the beginning of the row if necessary).*
4. *If the two letters are in the same column, replace each letter with the letter below it (wrapping around to the top of the column if necessary).*
5. *If the two letters are not in the same row or column, replace each letter with the letter in the same row but in the column of the other letter.*

- To decrypt a message encrypted with the Playfair cipher, the reverse process is used.

MKK

- Let us now take a concrete example to illustrate the process of encrypting some text using a keyword.
- Our keyword is **PLAYFAIR EXAMPLE** and the original text is **MY NAME IS ATUL**. We know that the matrix for our keyword is as shown in Fig.

PLAYFAIR EXAMPLE

Keyword

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

Matrix

MKK

1. First we break the original text into pairs of two alphabets each. This means that our original text would now look like this:

MY NA ME IS AT UL

2. Now, we apply our Playfair cipher algorithm to this text. The first pair of alphabets is **MY** In this case, this text is **XF**, which is our first cipher text block l step # 5

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

Alphabet Pair 1

3. Our next text block to be encrypted is **NA**. Again, Step #5 will apply.

As we can see, our second block of cipher text is OL.

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

Alphabet Pair 2

4. Now third block of plain text, which is **ME.** Step #3 will apply.

cipher-text block would be **IX**.

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

Alphabet Pair 3

MKK

5. Now fourth block of plain text, which is **IS.** Step #5 will apply. Cipher-text block would be **MK**.

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

Alphabet Pair 4

6. Fifth block of plain text, which is **AT.** Step #5 will apply. Cipher-text block would be **PV**.

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

Alphabet Pair 5

7. We will now take a look at the sixth and last block of plain text, which is **UL**. We can see that the two alphabets **U** and **L** are in the same column. Therefore, we need to apply the logic of Step #4 to get the alphabets **LR**.

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

Alphabet Pair 6

MKK

- Thus, our plain-text blocks **MY NA ME IS AT UL** becomes **XF OL IX MK PV LR**.
- To decrypt a message encrypted with the Playfair cipher, the reverse process is used.

Hill Cipher

- The Hill cipher works on multiple letters at the same time. Hence, it is a type of polygraphic substitution cipher. Lester Hill invented this in 1929.
- The Hill cipher has its roots in the matrix theory of mathematics.
- More specifically, we need to know how to compute the inverse of a matrix.

Hill cipher example

1. Treat every letter in the plain-text message as a number, so that A = 0, B = 1,...,Z=25.

2. The plain-text message is organized as a matrix of numbers based on the above conversion. For example, if our plain text is CAT. Based on the above step, we know that C = 2, A = 0, and T = 19. Therefore, our plain-text matrix would look as follows:

$$\begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix}$$

3. Now, our plain-text matrix is multiplied by a matrix of randomly chosen keys. The key matrix consists of size nxn, where n is the number of rows in our plain-text matrix. For example, we take the following key matrix:

$$\begin{pmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{pmatrix}$$

MKK

4. Now multiply the two matrices, as shown below:

$$\begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix} \quad X \quad \begin{pmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{pmatrix} \quad = \quad \begin{pmatrix} 31 \\ 216 \\ 325 \end{pmatrix}$$

5. Now compute a mod 26 value of the above matrix. That is take the remainder after dividing the above matrix values by 26. That is

$$\begin{pmatrix} 31 \\ 216 \\ 325 \end{pmatrix} \quad \text{mod } 26 = \quad \begin{pmatrix} 5 \\ 8 \\ 13 \end{pmatrix}$$

6. (This is because: 31/26=1 with a remainder of 5: which goes in the above matrix, and so on).

7. Now, translating the numbers to alphabets, 5=F, 8=1, and 13 = N. Therefore, our cipher text is ***FIN***.

MKK

8. For decryption, take the cipher-text matrix and multiply it by the inverse of our original key matrix The inverse of our original key matrix is

1. For decryption, take the cipher-text matrix and multiply it by the inverse of our original key matrix. The inverse of our original key matrix is

$$\begin{pmatrix} 8 & 5 & 10 \\ 21 & 8 & 21 \\ 21 & 12 & 8 \end{pmatrix} \quad X \quad \begin{pmatrix} 5 \\ 8 \\ 13 \end{pmatrix} \quad = \quad \begin{pmatrix} 210 \\ 442 \\ 305 \end{pmatrix}$$

2. Now we need to take modulo 26 of this matrix, as follows.

$$\begin{pmatrix} 210 \\ 442 \\ 305 \end{pmatrix} \quad \text{mod } 26 = \quad \begin{pmatrix} 2 \\ 0 \\ 19 \end{pmatrix}$$

3. Thus, our plain-text matrix contains 2, 0, 19; which corresponds to 2 = C, 0 = A, and 19 = T. This gives is the original plain text back successfully.

MKK

TRANSPOSITION TECHNIQUES

- Transposition techniques differ from substitution techniques in the way that they do not simply replace one alphabet with another, but they also perform some permutation over the plain text

Rail-Fence Technique

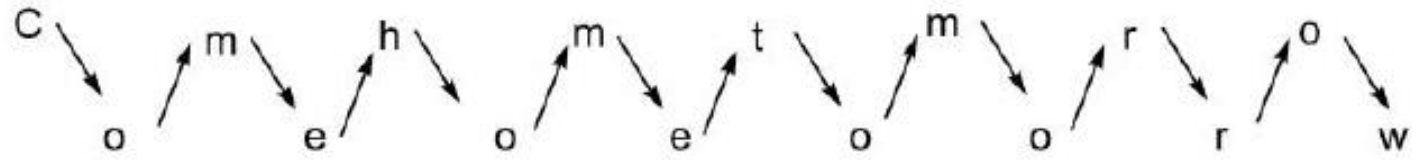
- The rail-fence technique is an example of transposition. It uses a simple algorithm

1. Write down the plain-text message as a sequence of diagonals.
2. Read the plain text written in *Step* 1 as a sequence of rows.

- Let us illustrate the rail-fence technique with a simple example. Suppose that we have a plain-text message :*Come home tomorrow*. How would we transform that into a cipher-text message using the rail-fence technique?

Original plain-text message: ***Come home tomorrow***

1. After we arrange the plain-text message as a sequence of diagonals, it would look as follows (write the first character on the first line i.e. *C*, then second character on the second line, i.e. *o*, then the third character on the first line, i.e. *m*, then the fourth character on the second line, i.e. *e*, and so on). This creates a zigzag sequence, as shown below.



2. Now read the text row by row, and write it sequentially. Thus, we have: ***Cmhmtmrooeoeoorw*** as the cipher text.

- As the figure shows, the plain-text message ‘Come home tomorrow’ transforms into ‘Cmhmtmrooeoeoorw’ with the help of rail-fence technique.
- Rail-fence technique involves writing plain text as a sequence of diagonals and then reading it row by row to produce cipher text.
- It should be quite clear that the rail-fence technique is quite simple for a cryptanalyst to break into.
- It has very little sophistication built in.

MKK

Simple Columnar Transposition Technique

1. Basic Technique

- Variations of the basic transposition technique such as rail-fence technique exist. Which we shall call *simple columnar transposition technique*

1. Write the plain-text message row by row in a rectangle of a pre-defined size.
2. Read the message column by column. However, it need not be in the order of columns 1, 2, 3, etc. It can be any random order such as 2, 3, 1, etc.
3. The message thus obtained is the cipher-text message.

- Let us examine the simple columnar transposition technique with an example. Consider the same plaintext message 'Come home tomorrow'. Let us understand how it can be transformed into cipher text using this technique.
- Like the rail-fence technique, the simple columnar transposition Technique is also quite simple to break into.
- To make matters complex for a cryptanalyst, we can modify the simple columnar transposition technique to add another twist: *perform more than one round of transposition using the same technique*.

Original plain-text message: ***Come home tomorrow***

1. Let us consider a rectangle with six columns. Therefore, when we write the message in the rectangle row by row (suppressing spaces), it would look as follows:

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6
C	o	m	e	h	o
m	e	t	o	m	o
r	r	o	w		

2. Now, let us decide the order of columns as some random order, say 4, 6, 1, 2, 5 and 3. Then read the text in the order of these columns.
3. The cipher text thus obtained would be ***eowoocmroerhmmto***.

- The simple columnar transposition technique simply arranges the plain text as a sequence of rows of a rectangle that are read in columns randomly.

MKK

2. Simple Columnar Transposition Technique with Multiple Rounds

- To improve the basic simple columnar transposition technique, we can introduce more complexity. The idea is to use the same basic procedure as used by the simple columnar transposition technique, but to do it more than once. That adds considerably more complexity for the cryptanalyst.

1. Write the plain text message row by row in a rectangle of a pre-defined size.
2. Read the message column by column. However, it need not be in the order of columns 1, 2, 3 etc. It can be any random order such as 2, 3, 1, etc.
3. The message thus obtained is the cipher text message of round 1.
4. Repeat steps 1 to 3 as many times as desired.

Cipher text produced by the simple columnar transposition technique with multiple rounds is much more complex to crack as compared to the basic technique.

MKK

Original plain-text message: ***Come home tomorrow***

1. Let us consider a rectangle with six columns. Therefore, when we write the message in the rectangle row by row, it would look as follows:

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6
C	o	m	e	h	o
m	e	t	o	m	o
r	r	o	w		

2. Now, let us decide the order of columns as some random order, say 4, 6, 1, 2, 5, 3. Then read the text in the order of these columns.
3. The cipher text thus obtained would be ***eowoocmroerhmmto*** in round 1.
4. Let us perform Steps 1 through 3 once more. So, the tabular representation of the cipher text after round 1 is as follows:

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6
e	o	w	o	o	c
m	r	o	e	r	h
m	m	t	o		

5. Now, let us use the same order of columns, as before, that is 4, 6, 1, 2, 5, and 3. Then read the text in the order of these columns.
6. The cipher text thus obtained would be ***oeochemmormorwot*** in round 2.
7. Continue like this if more number of iterations is desired, otherwise stop.

MKK

Vernam Cipher (One-Time Pad)

- The Vernam cipher, whose specific subset is called one-time pad, is implemented using a random set of non-repeating characters as the input cipher text. The most significant point here is that once an input cipher text for transposition is used, it is never used again for any other message (hence the name onetime).
- The length of the input cipher text is equal to the length of the original plain text.

1. Treat each plain-text alphabet as a number in an increasing sequence, i.e. A = 0, B = 1, … Z = 25.
2. Do the same for each character of the input cipher text.
3. *Add* each number corresponding to the plain-text alphabet to the corresponding input cipher-text alphabet number.
4. If the sum thus produced is greater than 26, subtract 26 from it.
5. Translate each number of the sum back to the corresponding alphabet. This gives the output cipher text.

MKK

- Let us apply the Vernam cipher algorithm to a plain-text message HOW ARE YOU using a one-time pad NCBTZQARX to produce a cipher-text message UQXTRUYFR

1. Plain text	H 7	O 14	W 22	A 0	R 17	E 4	Y 24	O 14	U 20
	+								
2. One-time pad	13 N	2 C	1 B	19 T	25 Z	16 Q	0 A	17 R	23 X
3. Initial Total	20	16	23	19	42	20	24	31	43
4. Subtract 26, if > 25	20	16	23	19	16	20	24	5	17
5. Cipher text	U	Q	X	T	Q	U	Y	F	R

- The Vernam Cipher was first implemented at AT&T with the help of a device called the Vernam machine.

MKK

Vernam Cipher uses a one-time pad, which is discarded after a single use, and therefore, is suitable only for short messages.

ENCRYPTION AND DECRYPTION

- In technical terms, the process of encoding plaintext messages into cipher text messages is called encryption.
- The reverse process of transforming cipher-text messages back to plain text messages is called decryption.
- Every encryption and decryption process has two aspects: the algorithm and the key used for encryption and decryption.
- In general, the algorithm used for encryption and decryption processes is usually known to everybody. However, it is the key used for encryption and decryption that makes the process of cryptography secure.

Broadly, there are two cryptographic mechanisms, depending on what keys are used. If the same key is used for encryption and decryption, we call the mechanism symmetric key cryptography. However, if two different keys are used in a cryptographic mechanism, wherein one key is used for encryption, and another, different key is used for decryption; we call the mechanism asymmetric key cryptography.

Symmetric and Asymmetric Key Cryptography

- Symmetric key cryptography involves using the same secret key to encrypt and decrypt the data. The encryption key is shared between the sender and the receiver of the message. This means that both the sender and receiver have the same key, and they use it to encrypt and decrypt messages. The main advantage of symmetric key cryptography is its speed, as it can encrypt and decrypt large amounts of data quickly. However, the main disadvantage is the challenge of securely sharing the key between the sender and receiver without it being intercepted by a third party.
- *Advanced Encryption Standard (AES):* This is a widely used symmetric key encryption algorithm that uses a block cipher to encrypt data.
- *Data Encryption Standard (DES):* This is another popular symmetric key encryption algorithm that uses a block cipher. However, DES is considered less secure than AES and is no longer recommended for use.

- Asymmetric key cryptography, also known as public key cryptography, uses two different keys, a public key and a private key. The sender encrypts the message using the receiver's public key, and the receiver decrypts it using their private key. The public key can be freely distributed, as it does not reveal any information about the private key. Asymmetric key cryptography is more secure than symmetric key cryptography because it does not require the sender and receiver to share a secret key. However, it is slower and requires more computational power than symmetric key cryptography.
- *RSA:* This is a popular asymmetric key encryption algorithm that is widely used for secure data transmission, digital signatures, and encryption of email messages.
- *Elliptic Curve Cryptography (ECC):* This is a relatively new asymmetric key encryption algorithm that uses elliptic curves instead of prime numbers to generate keys. ECC is considered more efficient and secure than other asymmetric key algorithms, such as RSA.

MKK

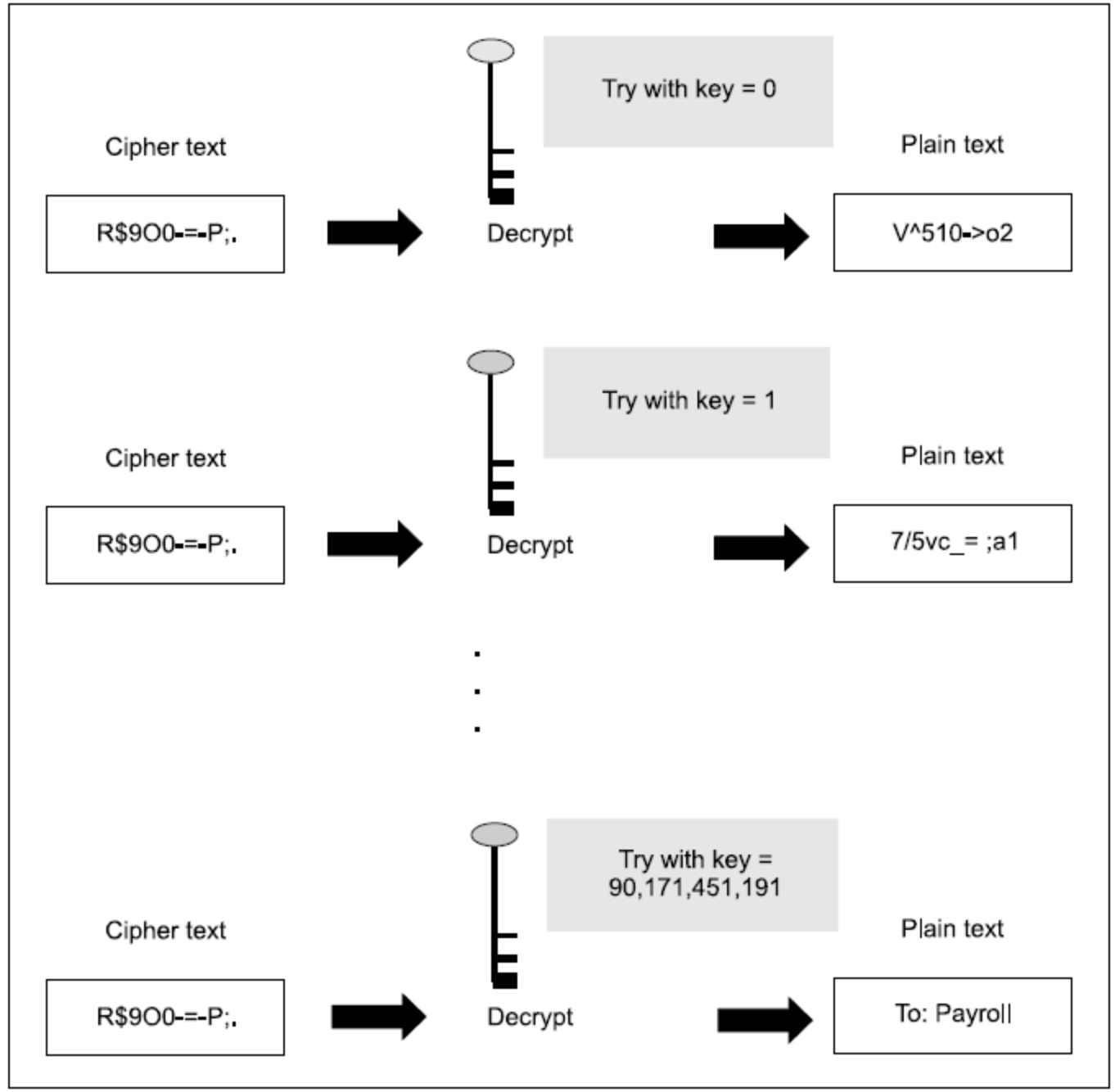
Steganography

- Steganography is a technique that facilitates hiding of a message that is to be kept secret inside other messages. This results in the concealment of the secret message itself! Historically, the sender used methods such as invisible ink, tiny pin punctures on specific characters, minute variations between handwritten characters, pencil marks on handwritten characters, etc.
- Of late, people hide secret messages within graphic images. For instance, suppose that we have a secret message to send. We can take another image file and we can replace the last two rightmost bits of each byte of that image with (the next) two bits of our secret message. The resulting image would not look too different, and yet carry a secret message inside! The receiver would perform the opposite trick: it would read the last two bits of each byte of the image file, and reconstruct the secret message.
- Steps to be followed:
  - Choose an image
  - Encode the message
  - Save the image
  - Send the image

MKK

KEY RANGE AND KEY SIZE

- key range refers to the number of possible keys that can be used in a cryptographic algorithm. The larger the key range, the more secure the algorithm is against brute force attacks.
- Key size, on the other hand, refers to the length of the key used in a cryptographic algorithm. It is usually measured in bits. The longer the key, the more secure the algorithm is against various attacks. For example, a key size of 128 bits is considered secure for most applications today, while a key size of 256 bits is considered even more secure.
- In general, the key range and key size are important factors in determining the strength and security of a cryptographic algorithm. A larger key range and key size generally provide greater security, but also require more computational resources to process. It is important to strike a balance between security and performance when selecting key ranges and key sizes for cryptographic algorithms.


Try with key = 0
Cipher text
Plain text
R$9O0-=-P;.
Decrypt
V^510->o2
Try with key = 1
Cipher text
Plain text
R$9O0-=-P;.
Decrypt
7/5vc_= ;a1
Try with key =
90,171,451,191
Cipher text
Plain text
R$9O0-=-P;.
Decrypt
To: Payroll

MKK

- The figure shows, the attacker has access to the cipher-text block and the encryption/decryption algorithm. He/she also knows the key range (a number between 0 and 100 billion). The attacker now starts trying every possible key, starting from 0. After every decryption, he/she looks at the generated plain text.
- If the attacker notices that the decryption has yielded unintelligent plain text, she continues the process with the next key in the sequence. Finally, he/she is able to find the right key with a value 90,171,451,191, which yields the plain text To: Payroll.
- At the simplest level, the key size can be just 1 bit. This means that the key can be either 0 or 1. If the key size is 2, the possible key values are 00, 01, 10, 11.
- From a practical viewpoint, a 40-bit key takes about 3 hours to crack. However, a 41-bit key would take 6 hours, a 42-bit key takes 12 hours, and so on. This means that every additional bit doubles the amount of time required to crack the key.
- This is shown in following Fig.

Thus, with every incremental bit, the attacker has to perform double the number of operations as compared to the previous key size. It is found that for a 56-bit key, it takes 1 second to search 1 percent of the key range.

A 2-bit binary number has four possible states:

00
01
10
11

If we have one more bit to make it a 3-bit binary number, the number of possible states also doubles to eight, as follows:

000
001
010
011
100
101
110
111

In general, if an *n-bit* binary number has *k* possible states, an *n* + 1 *bit* binary number will have 2*k* possible states.

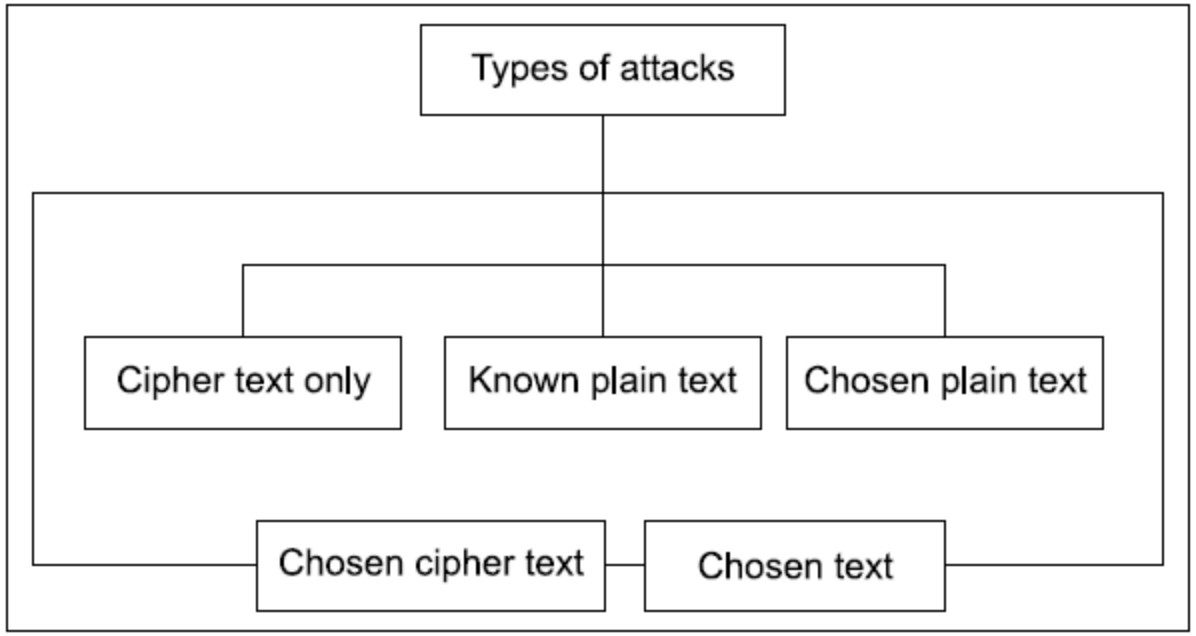
Taking this argument further, it takes about 1 minute to search about half of the key range

Key size on bits	*Time required to search 1 percent of the key space*	*Time required to search 50 percent of the key space*
56	1 second	1 minute
57	2 seconds	2 minutes
58	4 seconds	4 minutes
64	4.2 minutes	4.2 hours
72	17.9 hours	44.8 days
80	190.9 days	31.4 years
90	535 years	321 centuries
128	146 billion millennia	8 trillion millennia

MKK

POSSIBLE TYPES OF ATTACKS

- There are five possibilities for an attack on this message



MKK

1. Cipher-Text Only Attack

In this type of attack, the attacker does not have any clue about the plain text. He/She has some or all of the cipher text.

2. Known Plain-Text Attack

In this case, the attacker knows about some pairs of plain text and corresponding cipher text for those pairs. Using this information, the attacker tries to find other pairs, and therefore, know more and more of the plain text. Examples of such known plain texts are company banners, file headers, etc., which are found commonly in all the documents of a particular company.

3. Chosen Plain-Text Attack

Here, the attacker selects a plain-text block, and tries to look for the encryption of the same in the cipher text. Here, the attacker is able to choose the messages to encrypt. Based on this, the attacker intentionally picks patterns of cipher text that result in obtaining more information about the key.

4. Chosen Cipher-Text Attack

In the chosen cipher-text attack, the attacker knows the cipher text to be decrypted, the encryption

algorithm that was used to produce this cipher text, and the corresponding plain-text block. The attacker's job is to discover the key used for encryption. However, this type of attack is not very commonly used.

5. Chosen-Text Attack

The chosen-text attack is essentially a combination of chosen plain-text attack and chosen cipher-text

attack.

Summary of types of attacks

Attack	*Things known to the attacker*	*Things the attacker wants to find out*
Cipher text only	• Cipher text of several messages, all of which are encrypted with the same encryption key • Algorithm used	• Plain-text messages corresponding to these cipher text messages • Key used for encryption
Known cipher text	• Cipher text of several messages, all of which are encrypted with the same encryption key • Plain-text messages corresponding to the above cipher text messages • Algorithm used	• Key used for encryption • Algorithm to decrypt cipher text with the same key
Chosen plain text	• Cipher text and associated plain-text messages • Chooses the plain-text to be encrypted	• Key used for encryption • Algorithm to decrypt cipher text with the same key
Chosen cipher text	• Cipher text of several messages to be decrypted • Corresponding plain-text messages	• Key used for encryption
Chosen text	• Some of the above	• Some of the above

MKK

That's all about

UNIT 1

MKK