DETAILED LECTURE NOTES

(1)

INFORMATION SYSTEM SECURITY

UNIT-I

Introduction to Security Attacks

* **Information Security:** The protection of information and information system against unauthorized access or modification of information.

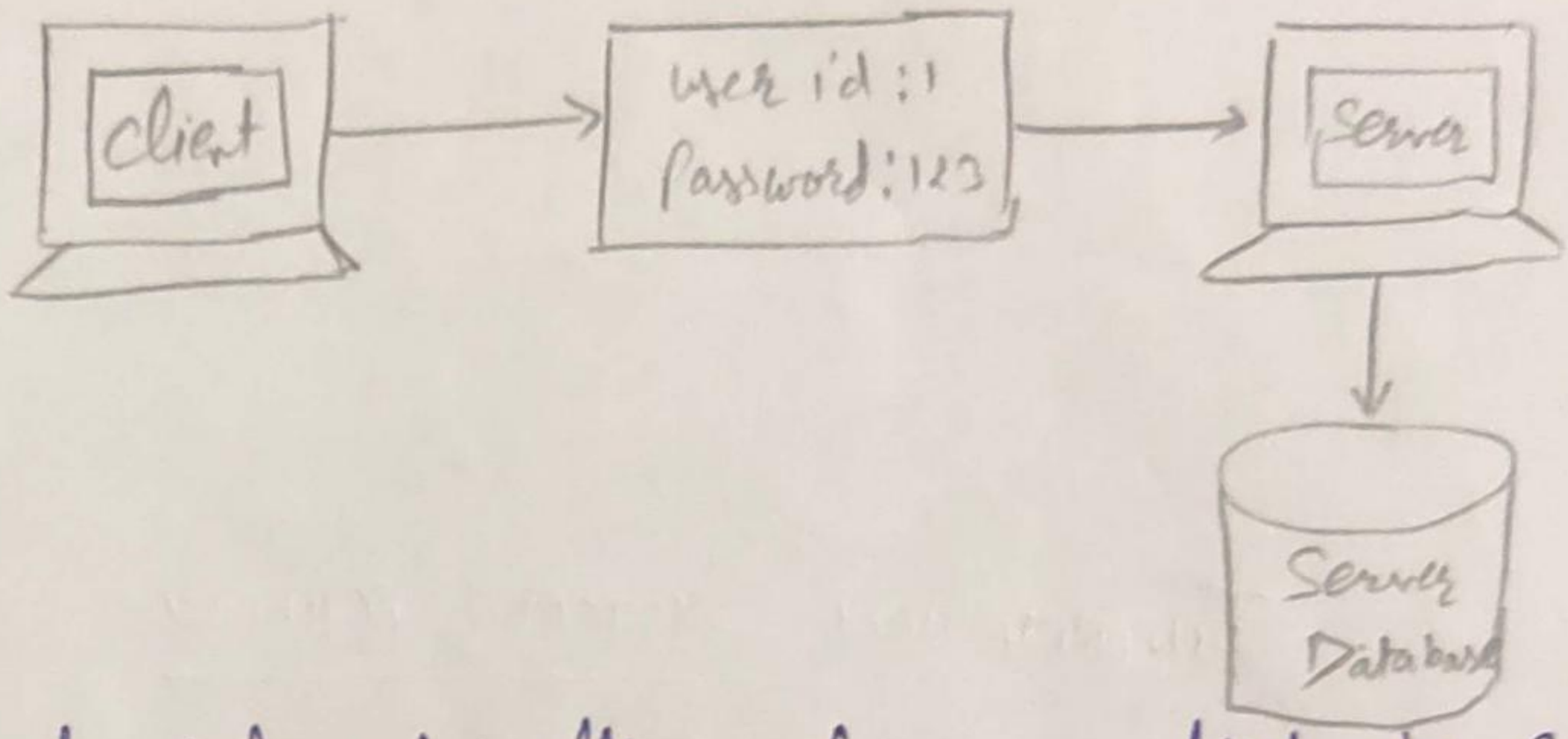
→ Info. can be anything like profile on social media, data in mobile phone, biometric etc.

* **Need for Security?** protect these assets, devices and services from being disrupted, stolen or exploited by ~~not~~ unathorized users, otherwise known as threat actors.

- These threats can be external or Internal and malicious or accidental in both origin and nature.

Eg. provide a user id and password to every user and use that Info to authenticate a user.

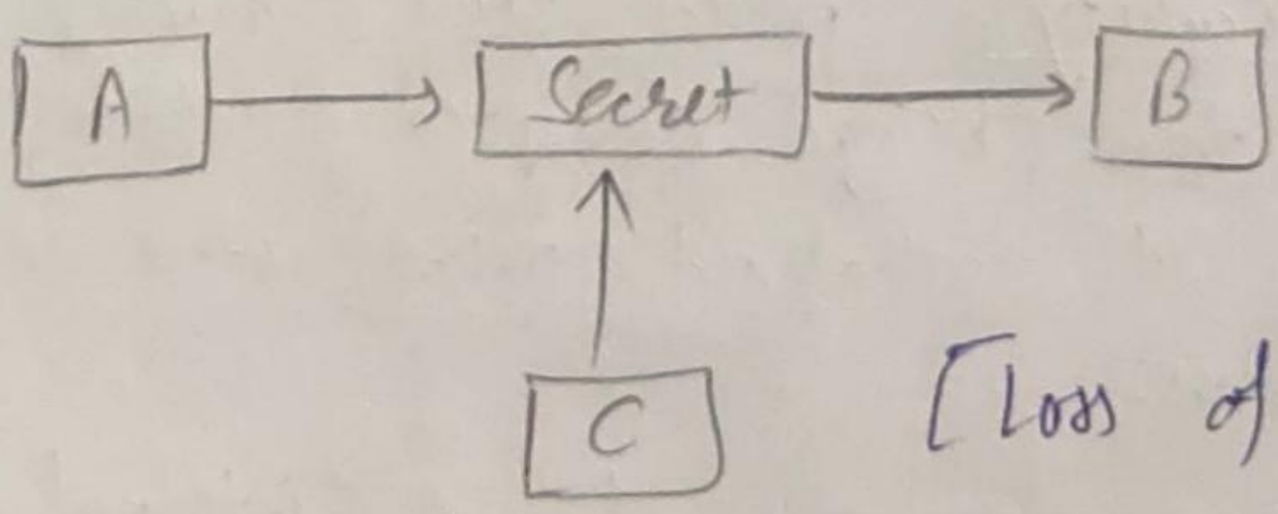
Scanned by TapScanner



(Ex of info. travelling from a client to server over the internet)

* <u>Principles of Security</u>: info. security prog. are build around 3 objectives, known as <u>CIA</u> (<u>Confidentiality</u>, <u>Integrity</u>, <u>Availability</u>).

1) <u>Confidentiality</u>: info. is not disclosed to ~~un~~ unauthorized individuals, entities and process.



[Loss of Confidentiality]

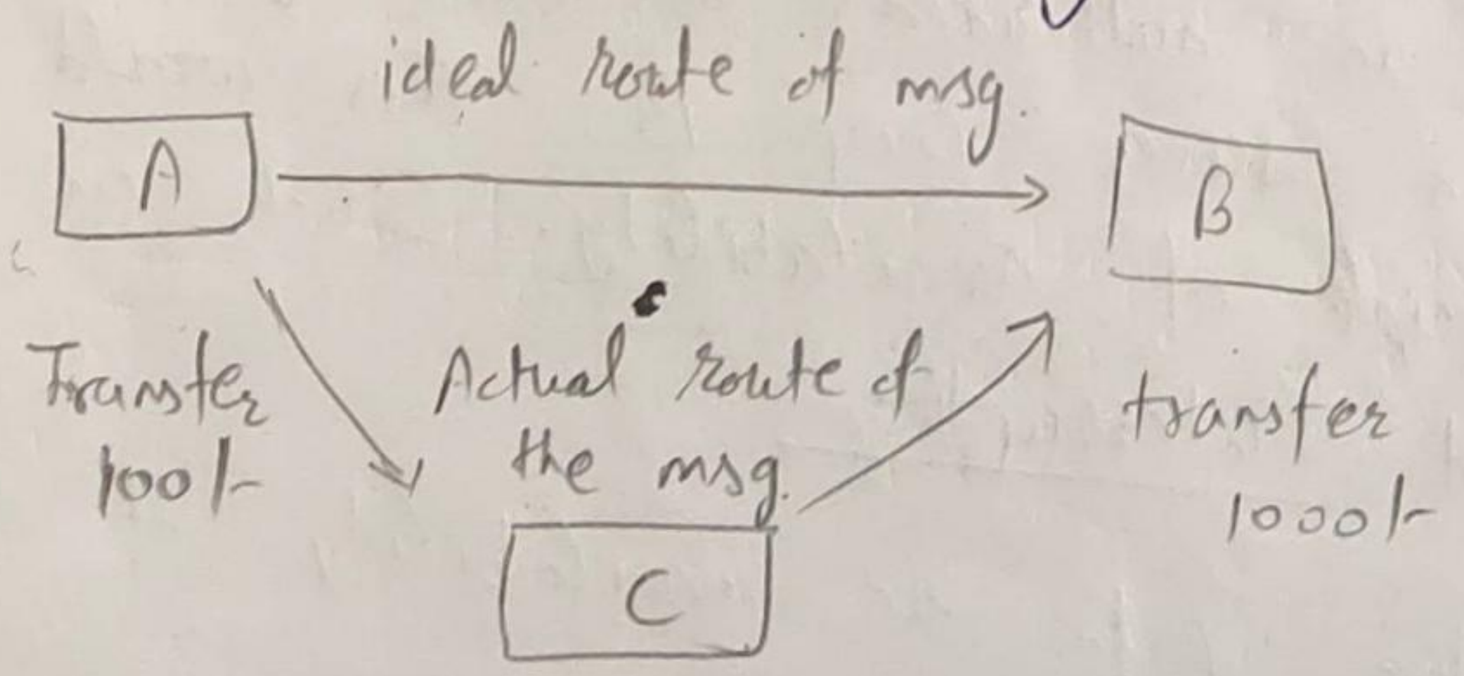
<u>eg.</u> user of Comp. A sends a msg to user of Comp. B, Another user C gets access to this msg., which is not desired and therefore, defeats the purpose of Confidentiality.

Scanned by TapScanner

Confidential email msg sent by A to B, which is accessed by C without the permission or knowledge of A and B. This type of attack is called as Interception.

"Interception causes loss of msg confidentiality."

2) Integrity: It means maintaining accuracy or completeness of data. This means data cannot be edited in an unauthorized way.



[Loss of integrity]

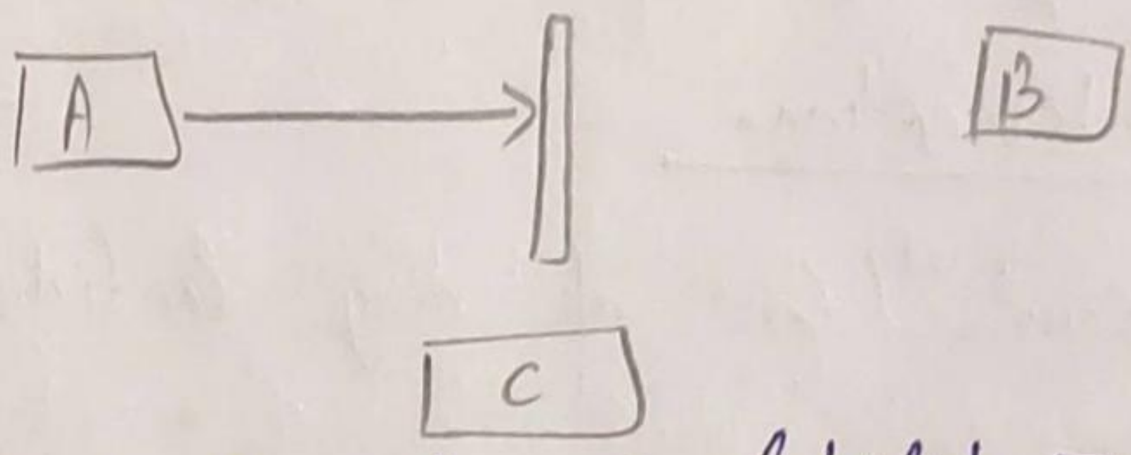
user A wants to sent a msg. to user B. But user C access the data & changes its contents and sends the changed msg to user B.

Scanned by TapScanner

User B does not knowing that contents of msg 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 msg. integrity."

3) Availability: means info. should be available to authorized parties at all times.



[Attack of Availability]

due to intentional actions of an unauthorized user C, an authorized user A may not be contact a server comp. B. This would defeat the principle of availability. Such an attack is called Interruption.

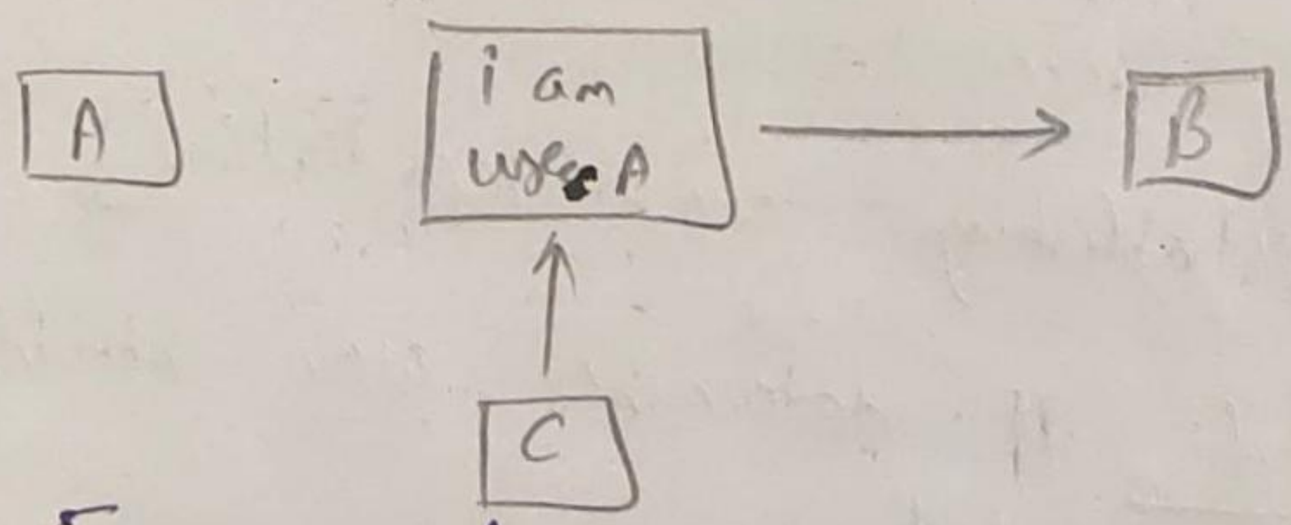
"Interruption puts the availability of resources in danger."

Apart from these principles we have some more principles:

Scanned by TapScanner

4) **Authentication:** authentication mechanisms help establish proof of identities. The authentication process ensure that the origin of a electronic msg. or document is correctly identified.

eg. Suppose user C sends an electronic document over the internet to user B. However, the trouble is that, user C had posed as user A when she sent document to user B. How would user B know that the msg. has come from user C, who is posing as user A ?



[Absence of authentication]

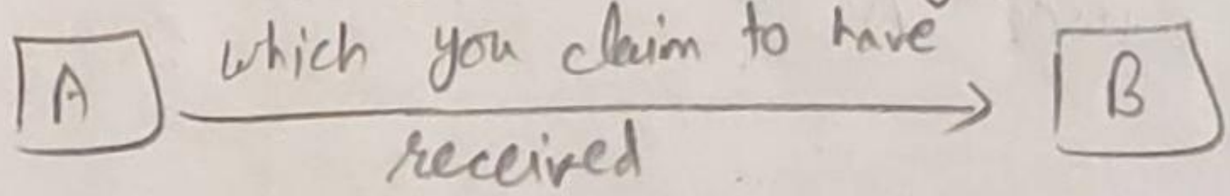
In case of user C, posing as user A, sending a fund transfer request to bank B. The bank might happily transfer the funds from A's account to C's account - after all, it would think that

user A has requested for fund transfer. This type of attack is called <u>febrication</u>.

" Febrication is possible in absence of proper authentication mechanism. "

5) <u>Non - Repudiation</u> : means one party cannot deny receiving a msg or a transaction nor can the other party deny sending a msg. or a transaction.

<u>eg.</u> user A could send a funds transfer request to bank B over the internet. After the bank performs the fund transfer as per A's instructions, A could claim that she never sent the funds transfer instruction to the bank.



[establishing non - repudiation]

6) <u>Access Control</u> : It determines <u>who</u> should be able to access <u>what</u>.

- Access control is broadly related to two areas :
  Role management : Concentrates on user side.
  Rule management : focuses on resources side.

"Access control specifies and controls who can access what."

Scanned by TapScanner

* Types of Attacks:

1) Attacks : A General View:

Types of attacks as understood by a common person

- Criminal attacks
- Publicity attacks
- Legal Attacks

Criminal Attacks: aim of the attackers is to max. financial gain by attacking comp. system.

Eg. fraud, Scams, Destruction, Identity theft, Intellectual property theft, Brand theft.

Publicity Attacks: occurs because the attackers want to see their names appear on T.V., new channels and news papers.

Legal Attacks: This form of attack is quite novel or unique. The attackers tries to make the judge or

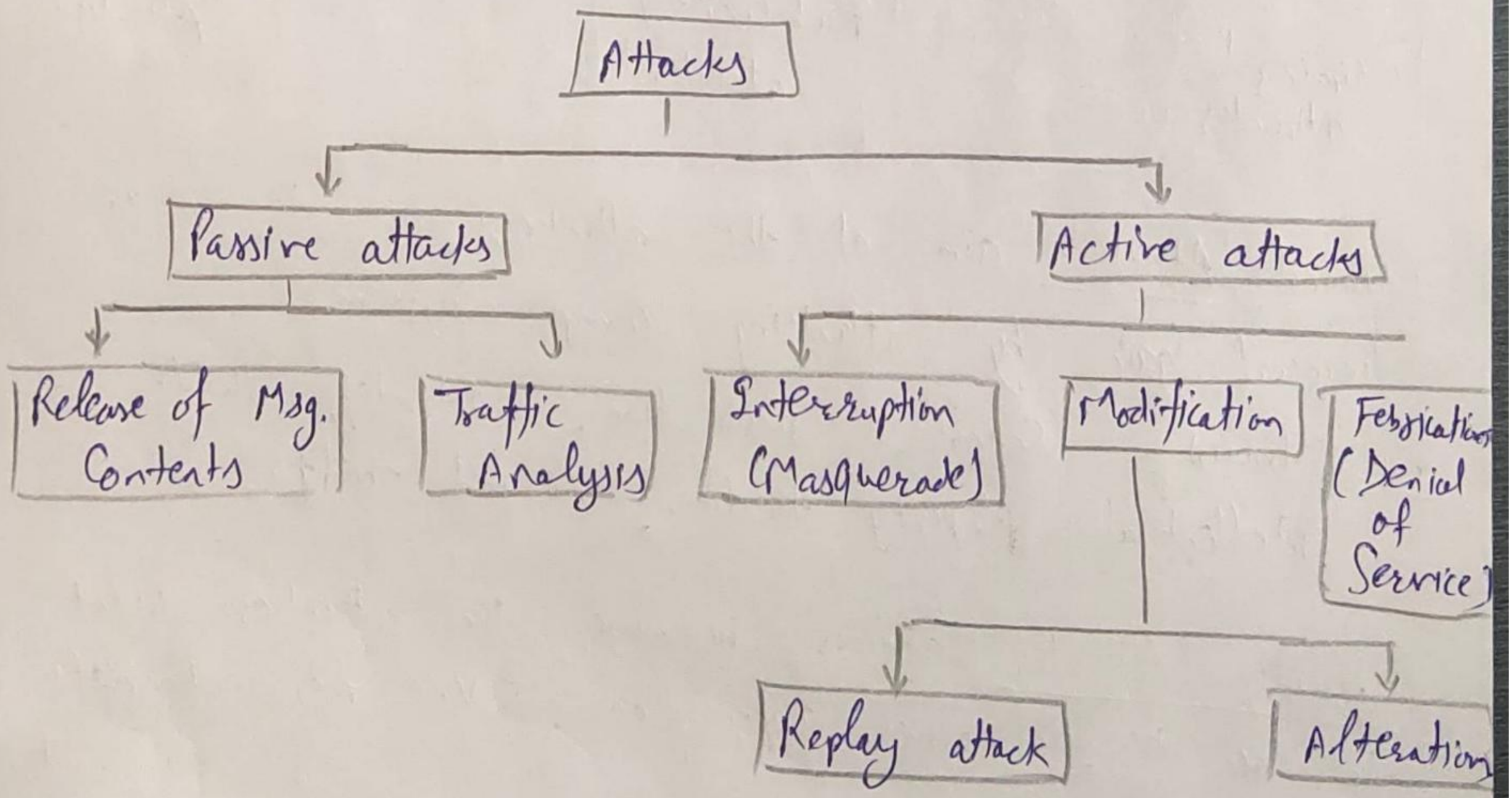
Scanned by TapScanner

jury doubtful about the security of a comp. sys.

eg. An attacker may sue a bank for a performing an online transaction, which she never wanted to perform. In court, she could innocently say sth like The bank's website asked me to enter a password and that is all that I provided; I do not know what happened thereafter. A judge is likely to sympathize with the attacker.

2) Attacks: A Technical View:

Interception, Febrication, Modification, Interruption



Scanned by TapScanner

POORNIMA
COLLEGE OF ENGINEERING

DETAILED LECTURE NOTES (5)

Passive Attacks: Attackers indulges in eavesdropping or monitoring of data transmission.

OR

attacker aims to obtain info. that is in transit.

"Passive attacks do not involve any modifications to the contents of an original msg."

General approach to deal with passive attacks is to think about prevention, rather than detection or corrective actions.

Release of Message Contents: When we send a confidential email msg to our friend, we desire that only she / he be able to access it. otherwise, the contents of msg. are released against our wishes to some one else.

Eg. Encode a msg. using a code lang. so that only desired parties understand the content of msg. bcoz only they know the code lang.

Scanned by TapScanner

Traffic Analysis Attacks: If such msg. are passing through, a passive attacker could try to figure out similarities b/w them to come up with some sort of pattern that provides her some clues regarding the comm. that is taking place. Such attempts of analysing (encoded) msg. to come up with likely patterns are the work of the traffic analysis attacks.

Active Attacks: Unlike passive attacks, the active attacks are based on modification of the original msg in some manner or the creation of a false msg. These attacks can not 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.

Masquerade: When an unauthorized entity pretends to be another entity.

Eg: user C might pose as user A & sends a msg to user B. User B might be led to believe that the msg. indeed came from user A.

Scanned by TapScanner

Replay Attacks: a user captures a sequence of events or some data units and re-sends them.

eg. Suppose user A wants to transfer some amount to user C's bank account. Both user A and C have accounts with bank B.

User A might send an electronic msg. to bank B, requesting for the funds transfer. User C could capture this msg. and send a second copy of the same to bank B. Bank B would have no idea that this is an unauthorized msg. 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: alteration of msg. involves some change to the original msg.

eg. Transfer 1000/- but got 10000/-

Scanned by TapScanner

3) A Practical Side of Attacks:

Security attacks in practice

- Application Level Attack
- Network Level Attacks

Application Level Attack: Attacker attempts to access, modify or prevent access to info. of a particular app. or to the app. itself.

eg. ~~obt~~ trying to obtain someone's Credit Card info. on the internet or changing the contents of a msg. to chang. the amount of transaction etc.

Network Level Attacks: These attacks generally aim at ~~redu~~ reducing the capabilities of a network by a no. of possible ~~meas~~ means.

4) Programs that Attacks:

Virus: a virus is a comp. prog. that attaches itself to another legitimate prog. and causes damage to the comp. sys. or to the n/w.

Worm: a worm does not perform any destructive actions and instead, only consumes system resources to bring it down.

Scanned by TapScanner

Trojan: A trojan horse allows an attacker to obtain some confidential info. about a comp. or a n/w.

5) **Specific Attacks:**

Sniffing and Spoofing: On the Internet, comp. exchange msg. with each other in the form of small blocks of data, called as packets.

→ A packet, like a postal envelope contains the actual data to be sent and the addressing info.

→ Attackers targets these packets as they travel from the source comp. to destination comp. over the internet.

→ These attacks take two main forms:

a) Packet sniffing (also called as snooping) / IP sniffing

b) Packet spoofing / IP spoofing.

→ The protocol used in this comm. is called Internet Protocol (IP).

Scanned by TapScanner

a) Packet Sniffing: It is a passive attack on an ongoing conversation. An attacket need not hijack a conversation, but instead, can simply observe (i.e. sniff) packets as they pass by.

→ To prevent an attacker from sniffing packets, the info. that is passing need to be protected in some ways. This can be done at two levels:

i) the data that is traveling can be encoded in some ways

ii) the transmission link itself can be encoded.

→ To read a packet, the attacker somehow needs to access it in the first place. So to do this is to control a comp. via which the traffic goes through.

→ Usually, this is a router. However, the routers are highly protected resources. Therefore, an attacker might not be able to attack it and instead, attack a less protected comp. on the same path.

b) Packet spoofing: An attacker sends packets with a false source address. than the receiver would inadvertently (without intention; accidently) send replies back to this forged address called as spoofed

Scanned by TapScanner

address. This can lead to three possible cases:

i) the attacker can intercept the reply: if the attacker is b/w the destination and the forged source, the attacker can see the reply and use the info. for hijacking attacks.

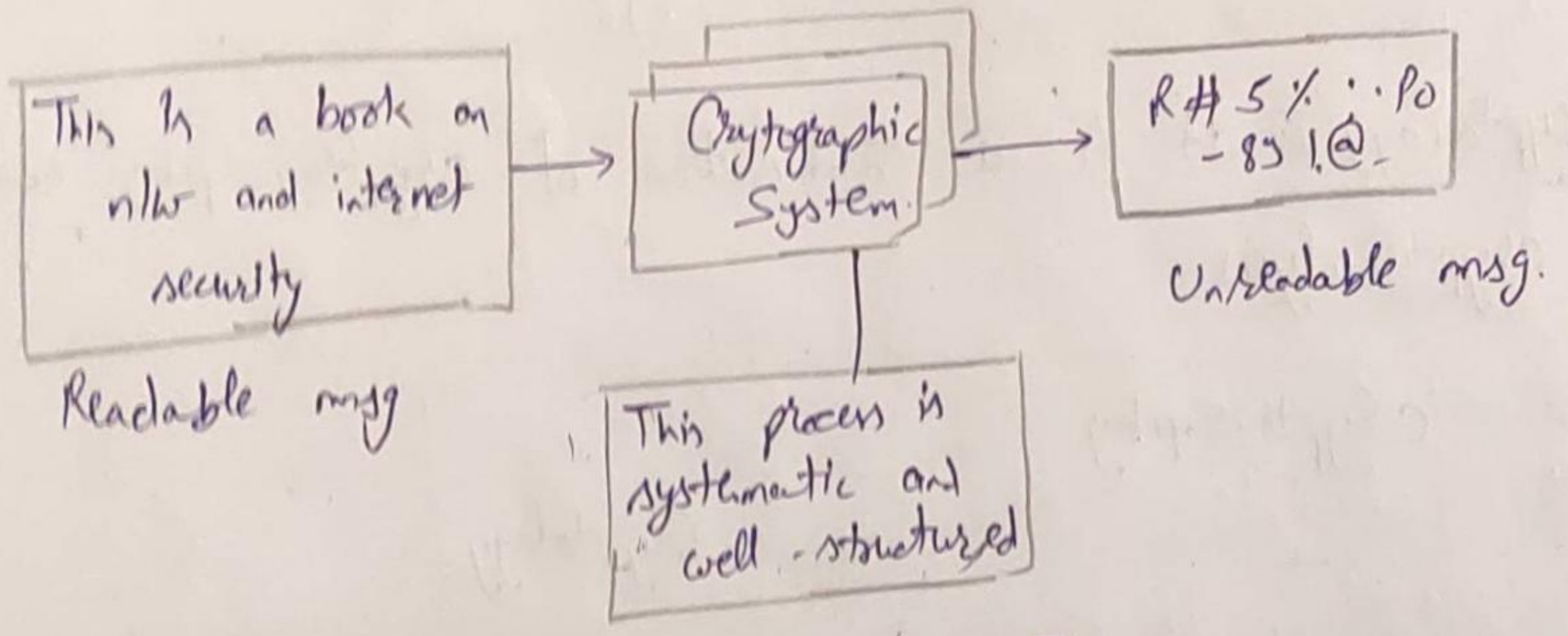
ii) The attacker need not see the reply: If the attacker's intention was a Denial of Service (DOS) attack, the attacker need not bother about the reply.

iii) The attacker does not want the reply: The attacker could simply be angry with the host, so it may put that host's address as the forged source add. and send the packet to the destination. The attacker does not want a reply from the destination, as it wants the host with the forged add. to receive it and get confused.

Scanned by TapScanner

DETAILED LECTURE NOTES

* **Cryptography:** it is the art and science of achieving security by encoding msg. to make them non-readable.

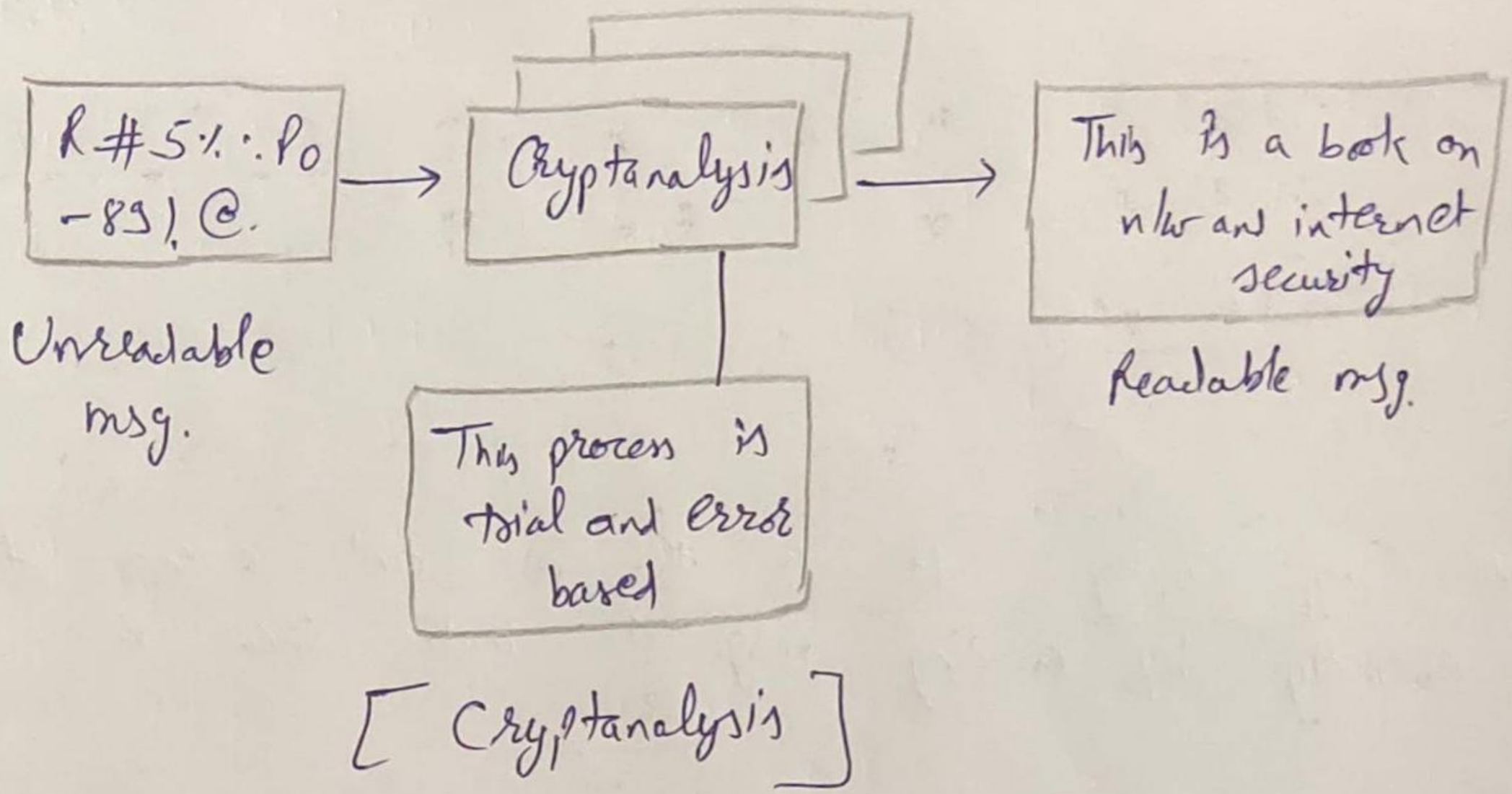


[Cryptographic System]

Cryptanalysis: it is a technique of decoding msg. from a non-readable format back to 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.

Scanned by TapScanner



[Cryptanalysis]

<u>Cryptology</u>: It is a Combination of cryptography and cryptanalysis.

Cryptography		
+	=	Cryptology
Cryptanalysis		

[cryptology]

* <u>Plain Text and Cipher Text</u>:

<u>Plain Text</u>: Any comm^n in the lang. that we speak - that is the human lang.

→ Msg. can be understood by anybody knowing the lang. as soon as the msg. is not codified in any manner.

→ <u>eg.</u> we speak with our family members, colleagues etc. we do not hide anything from them.

Scanned by TapScanner

1) "Clear text or plain text signifies a msg. that can be understood by the sender, the recipient and also by anyone else who gets an access to that msg."

2) Cipher Text: Each alphabet in the original msg. can be replaced by another to hide the original contents of the msg. The codified msg. is called as ~~ch~~ cipher text. Cipher means a code or a secret msg.

"When a plain text msg is codified using any suitable scheme, the resulting msg. is called as cipher text."

e.g. replace each alphabet with three place down the line.

I	MISS	YOU
L	PLVV	BRX

[Codification using the alphabet-replacement scheme]

Scanned by TapScanner

* Substitution Techniques:

1) **Caesar Cipher:** It was first proposed by Julius Caesar. It was the first eg. of substitution cipher.

→ In the substitution Cipher technique, the characters of a plain text msg. are replaced by other characters, no, or symbols.

→ Caesar cipher is a special case of substitution techniques wherein each alphabet in a msg. is replaced by an alphabet three places down the line.

eg. A R C H I $\xrightarrow[\text{by}]{\text{replaced}}$ D U F K L

"the characters of a plain text msg are replaced by other char., no or symbol."

→ Caesar cipher is a very weak scheme of hiding plain text msg.

→ for breaking the caesar cipher is to do the reverse of the caesar cipher process i.e. replace each alphabet in a cipher text msg. produced by Caesar Cipher with the alphabet i.e. three places up the line.

→ eg. replace A with X, B with Y, C with Z, D with A, E with B and so on.

Scanned by TapScanner

Algorithm to break Caesar cipher:

1) Read each alphabet in the cipher text msg. and search for it in the second row of the replacement table (i.e. second row of the table).

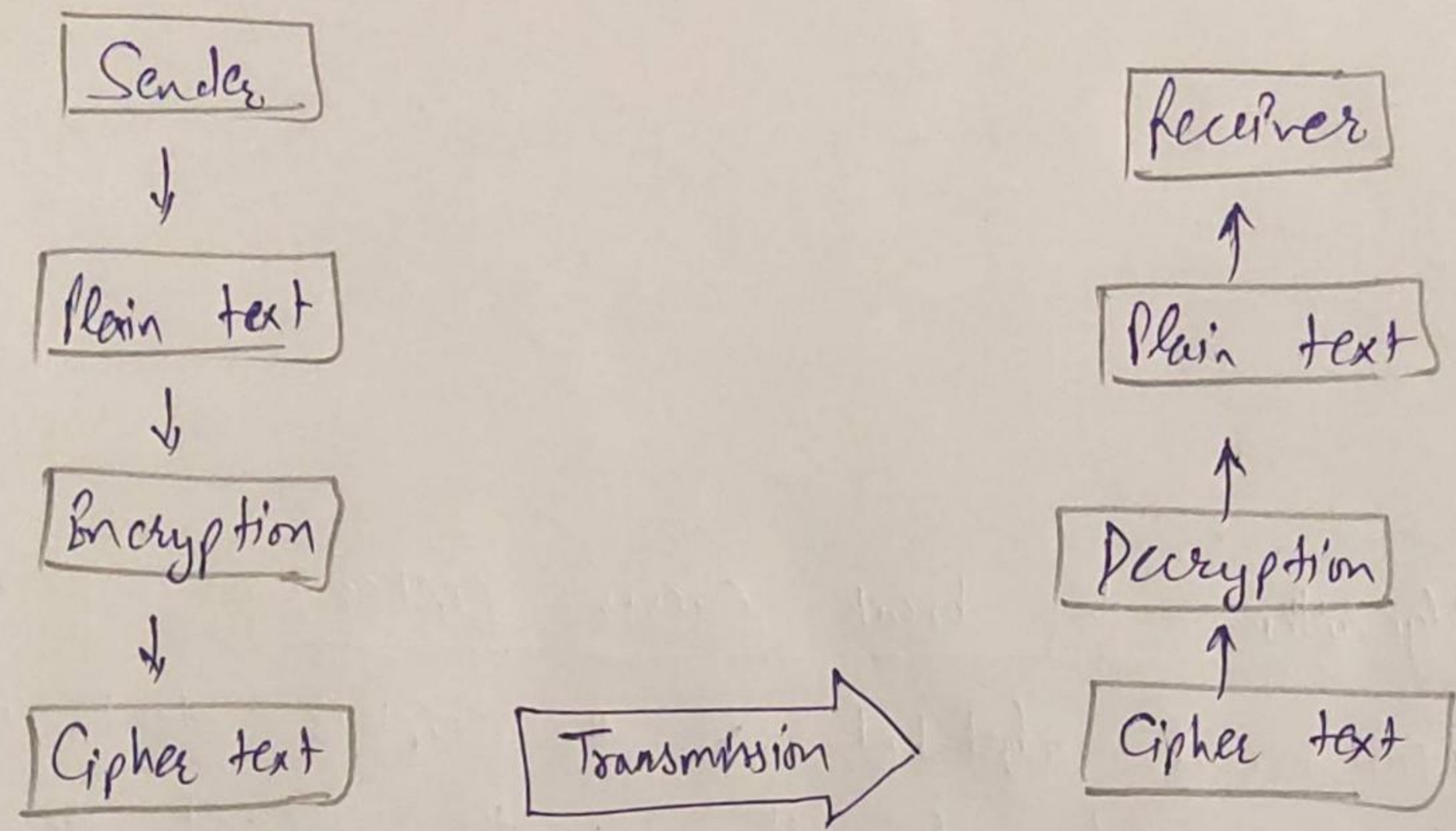
2) When a match is found, replace the alphabet in the cipher text msg. with the corresponding alphabet in the same column but the first row of the table (eg. if the alphabet in cipher text is J, replace it with G).

3) Repeat the process for all alphabets in the cipher text msg.

Eg. given a cipher text msg. L PLVV BRX, it is easy to work backwards and obtain the plain text I MISS YOU.

Cipher text	L	PLVV	BRX
Plain text	I	MISS	YOU

Scanned by TapScanner



[Elements of a crytographic operation]

Transforming a plain text msg. into cipher text

- Substitution Techniques
- Transposition Techniques

[Techniques for transforming plain text to cipher text]

2) <u>Modified version of Caesar Cipher:</u>

Caesar cipher is good in theory, but not so good in practice. So Let us assume that the cipher text alphabets corresponding to the original plain text alphabets may not necessarily be three places

Scanned by TapScanner

down the line, but instead, can be any places down the line. This can be complicate matter a bit. So we have 25 possibilities of replacement.

Algorithm to break the modified Caesar Cipher:

1) Let k be a no. equal to 1.
2) Read the complete cipher text msg.
3) Replace each alphabet in the cipher text msg. with an alphabet that is k positions down the order.
4) increment k by 1.
5) if k is less than 26, then go to step 2. otherwise, stop the process.
6) the original text msg. corresponding to the cipher text ~~msg~~ msg. is one of the 25 possibilities produced by the above steps.

Scanned by TapScanner

"A mechanism of encoding msg. so that they can be sent securely is called as Cryptography."

→ An attack on a cipher text msg., wherein the attacker attempts to use all possible permutations. (arranging the no. in order) & combinations, is called as a Brute force attack.

→ The process of trying to break any cipher text msg. to obtain the original plain text msg. itself is called as Cryptanalysis and the person attempting a cryptanalysis is called as a Cryptanalyst.

"Cryptanalyst is a person who attempts to break a cipher text msg. to obtain the original plain text msg. The process itself is called as Cryptanalysis."

→ Even the modified version of Caesar Cipher is not very secure. After all, the cryptanalyst needs to be aware of only the following points to break a cipher text msg. using the Brute-force attack, in this scheme:

1) Substitution technique was used to derive the cipher text from the original plain text.

Scanned by TapScanner

2) There are only 25 possibilities to try out.
3) The lang. of the plain text was English.

"A cryptanalyst attempting a Brute-Force attack tries all possibilities to derive the original plain text msg. from a given cipher text msg."

3) Mono-alphabetic Cipher:

→ Now imagine that rather than using a uniform scheme for all the alphabets in a given plain text msg., we decide to use random substitution.

→ Means, in a given plain text msg., each A can be replaced by any other alphabet (B through Z), each B can also be replaced by any other random alphabet (A or C through Z) and so on.

→ The crucial difference being, there is no relation b/w the replacement of B and replacement of A.

Scanned by TapScanner

→ That is, if we have decided to replace each A with D, we need not necessarily replace each B with E, we can replace each B with any other character.

"Mono alphabetic cipher pose a difficult problem for a cryptanalyst because it can be very difficult to crack thanks to the high no. of possible permutations and combination.".

4) <u>Homophonic Substitution Cipher</u>:

→ It is very similar to Mono-alphabetic Cipher.

→ Like a plain substitution cipher technique, we replace one alphabet with another.

→ However, the diff. b/w two techniques is that whereas the replacement alphabet set in case of the simple substitution technique is fixed (eg. Replace A with D, B with E etc.).

→ In case of Homophonic substitution cipher, one plain text alphabet can map to more than one cipher text alphabet.

→ <u>Eg.</u> A can be replaced by D, H, P, R;
B can be replaced by E, I, Q, S etc.

Scanned by TapScanner

"Homophonic Substitution Cipher also involves substitution of one plain text character with a cipher text character at a time, however the cipher text character can be any one of the chosen set."

5) Polygram Substitution Cipher:

→ In this rather than replacing one plain text alphabet with one cipher text alphabet at a time, a block of alphabet is replaced with another block.

Eg. HELLO could be replaced by YUQQW but, HELL could be replaced by a totally diff. cipher text block TEUI.

"Replaces one block of plain text with a block of cipher text - it does not work on a character-by-character basis."

Scanned by TapScanner

6) Polyalphabetic Substitution Cipher:

→ This cipher uses multiple one-character keys.

→ Each of the keys encrypts one plain text character.

→ The first key encrypts the first plain text character; the second key encrypts the second plain text character and so on.

→ After all the keys are used, they are recycled.

Main features are:

- it uses a set of related monoalphabetic substitution ~~cipher etc.~~ rules.
- it uses a key that determines which rule is used for which transformation.

eg. Vignere cipher.

→ In this algo. 26 Caesar cipher make up the mono-alphabetic substitution rules. There is a shifting mechanism, from a count of 0 to 25.

→ For each plain text letter, we have a corresponding substitution, which we call as the key letter.

eg. table 2.2 (page No. 48 Atul kahate) for key letter p and plain text letter q, the corresponding cipher text letter is at the intersection of row titled p and column titled q. For this case, cipher text would be F.

Scanned by TapScanner

7) Playfair Cipher:

→ It is also called as Playfair Square. cryptographic technique that is used for manual encryption of data.

→ This was invented by Charles Wheatstone in 1854.

→ Eventually the scheme came to be known by the name of Lord, playfair, who was Wheatstone's friend.

→ The playfair cipher was used by the British army in World War I and by the Australians in World War II.

Playfair encryption scheme uses two main processes:

1) Creation & Population of Matrix

2) Encryption process

Scanned by TapScanner

Step 1: Creation and Population of Matrix:

Use 5X5 matrix, which is used to store a keyword or phrase that becomes the key for encryption & decryption.

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 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.

eg. keyword : PLAYFAIR EXAMPLE.

make a matrix 5X5 for it.

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

Scanned by TapScanner

Row 1: The first row of the matrix

P	L	A	Y	F

None of the alphabets is duplicate. Therefore, write these one after other in same row. as per rule #1 defined.

Row 2:

I	R	E	X	M

first four letters (IREX) in continuation as per the rule #1 defined. However, after this, we have a repetition of alphabet A. Therefore, we discard the A as per rule #2 & choose the next alphabet which is M.

Row 3:

B	C	D	G	H

all of keyword alphabets are covered now. Therefore, there is nothing left to population in our matrix from third row onwards. Hence, rule #3 consider. & fill the remaining parts with the A - Z alphabets.

Scanned by TapScanner

Row 4:

K	N	O	Q	S

apply rule #3 & get the text i.e. KNOQS

Row 5:

T	U	V	W	Z

apply rule #3 & get the text i.e. TUVWZ

Step 2: Encryption Process:

1) Before executing these steps, plaintext msg. that we want to encrypt needs to be broken down into groups of two alphabets.

Eg. MY NAME IS ATUL

⇓

MY NA ME IS AT UL

The encryption process works on this broken-down msg.

2) if both the alphabets are the same (or only one is left), add an X after the first alphabet & encrypt the new pair and continue.

3) If both the alphabets in the pair appear in same row of our matrix, replace them with alphabets to their immediate right respectively. If the original pair is on the right side of the row, then wrapping around to the left side of the row happens.

4) if both the alphabets in the pair appear in the same column of our matrix, replace them with

Scanned by TapScanner

alphabets immediately below them respectively. If the original pair is on the right side of the row, then wrapping around to the top side of row happens.

5) if the alphabet are not in the same row or column, replace them with the alphabets in the same row respectively, but at the other pair of corner of the rectangle defined by the original pair. The order is quite significant here, The first encrypted alphabet of the pair is the one that is present on the same row as the first plaintext alphabet.

Explanation of encryption process :

1) First we break the original text into pairs of two alphabets each.

MY NA ME IS AT UL

2) first pair of alphabet is MY. Looking at the matrix, we see that the alphabet M and Y do not occur in the same row or column.

Scanned by TapScanner

therefore, we need to apply step #5. this means replace this text with the text diagonally opposite to it. XF is first cipher text block for MY pair.

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

3) Next text block to be encrypted is NA. again, step #5 will apply. & get OL cipher text.

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

4) Next plain text is ME. both the alphabet M and E are in same second row. Therefore, based on ~~ostep~~ step #3, cipher text 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

5) Next plain text is IS. apply Step #5 to get diagonal alphabets. Cipher text block ~~get~~ ~~AK.~~ would be MK.

Scanned by TapScanner

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

6) fifth block of plain text is AT. apply step #5 & get cipher text block 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

7) last plain text block is UL. both the alphabets U and L are in same column. Therefore, apply step #4 to get alphabet 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

So Owre plain text block MY NAME IS ATUL becomes XF OLIX MK PVLR.

Scanned by TapScanner

8) Hill Cipher :

→ It works on multiple letters at the same time.

→ It is a type of Polygraphic Substitution Cipher.

→ Lester Hill invented this in 1929.

→ this method makes use of mathematics.

To encrypt :

key, plain text

$$\boxed{C = KP \bmod 26}$$

Step 1: Choose a key (key matrix must be a square matrix)

eg. $VIEW = \begin{bmatrix} V & I \\ E & W \end{bmatrix} = \begin{bmatrix} 21 & 8 \\ 4 & 22 \end{bmatrix}_{2\times 2}$

eg. $QUICKNESS = \begin{bmatrix} Q & U & I \\ C & K & N \\ E & S & S \end{bmatrix} = \begin{bmatrix} 16 & 20 & 8 \\ 2 & 10 & 13 \\ 4 & 18 & 18 \end{bmatrix}_{3\times 3}$

Eg. plain text = ATTACK

Let key $= \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix}_{2\times 2}$. (It is given or can assume.)

Since, the key is 2×2 matrix, plain text should be converted into vectors of length 2.

It means if we have key $n \times n$ then convert it into $n \times 1$.

Scanned by TapScanner

So $\begin{bmatrix} A \\ T \end{bmatrix}_{2\times1}$ $\begin{bmatrix} T \\ A \end{bmatrix}_{2\times1}$ $\begin{bmatrix} C \\ K \end{bmatrix}_{2\times1}$

Now encryption begins:

1) 1st vector $\begin{bmatrix} A \\ T \end{bmatrix} = \begin{bmatrix} 0 \\ 19 \end{bmatrix}$, key $= \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix}$

$$C = K P \bmod 26$$

$$= \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} 0 \\ 19 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 2\times0 + 3\times19 \\ 3\times0 + 6\times19 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 57 \\ 114 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 5 \\ 10 \end{bmatrix}$$

Corresponding alphabets $= \begin{bmatrix} F \\ K \end{bmatrix}$

$\therefore$ plain text $\begin{bmatrix} A \\ T \end{bmatrix}$ becomes $\begin{bmatrix} F \\ K \end{bmatrix}$ i.e. $AT \rightarrow FK$

Scanned by TapScanner

2) 2nd vector is $\begin{bmatrix} T \\ A \end{bmatrix} = \begin{bmatrix} 19 \\ 0 \end{bmatrix}$

$$C = \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} 19 \\ 0 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 38 \\ 57 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 12 \\ 5 \end{bmatrix} = \begin{bmatrix} M \\ F \end{bmatrix}$$

TA becomes MF.

3) next vector is $\begin{bmatrix} C \\ K \end{bmatrix} = \begin{bmatrix} 2 \\ 10 \end{bmatrix}$

$$C = \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix} \begin{bmatrix} 2 \\ 10 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 34 \\ 66 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 8 \\ 14 \end{bmatrix} = \begin{bmatrix} I \\ O \end{bmatrix}$$

CK becomes IO.

Plain text = ATTACK
Cipher text = FKMFIO } Polyalphabetic Cipher.

To Decrypt: find Inverse of key matrix K^{-1}

$$\boxed{P = K^{-1} C \bmod 26}$$

Scanned by TapScanner

plain → ATTACK , key = $\begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix}$

Cipher → FKMFIO

$k^{-1} = \frac{1}{|k|} adj(k)$

determinant of matrix :

$d = \begin{vmatrix} a & b \\ c & d \end{vmatrix}$

$d = |ad - bc|$

eg: $d = \begin{vmatrix} 2 & 3 \\ 3 & 6 \end{vmatrix} = |12 - 9| = 3$

∴ determinant value $d = 3$

Now find multiplicative inverse of determinant

ie $dd^{-1} \equiv 1 \mod 26$ (identity matrix)

so $3 * d^{-1} \equiv 1 \mod 26$

⇕ this mean

$(3 * d^{-1}) \mod 26 = 1$

so $\boxed{d^{-1} = 9}$

Use Hit & Trial method

$1 \mod 26 = 1$

$3 * d^{-1} \mod 26 = 1$

$3 * 9 \mod 26 = 1$

$\boxed{27 \mod 26 = 1}$

(multiple of 26 or greater than 26)

Scanned by TapScanner

Now determinant $d = 3$

$d^{-1} = 9$

Now we will find adjacent of the matrix

Let $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ then $adj(A) = \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

$\therefore$ Here, $k = \begin{bmatrix} 2 & 3 \\ 3 & 6 \end{bmatrix}$ then $adj(k) = \begin{bmatrix} 6 & -3 \\ -3 & 2 \end{bmatrix}$

Before decryption, we have to remove -ve values (add 26 to -ve no.)

$\therefore adj(k) = \begin{bmatrix} 6 & -3+26 \\ -3+26 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 23 \\ 23 & 2 \end{bmatrix}$

Now $k^{-1} = \frac{1}{|k|} adj(k)$ or $\frac{1}{|d|} adj(k)$

OR

$|k^{-1}| \, adj(k) = d^{-1} \, adj(k)$

$\hookrightarrow$ determinant value.

$k^{-1} = 9 \begin{bmatrix} 6 & 23 \\ 23 & 2 \end{bmatrix} = \begin{bmatrix} 54 & 207 \\ 207 & 18 \end{bmatrix}$

Now find its modulo 26

$k^{-1} = \begin{bmatrix} 54 & 207 \\ 207 & 18 \end{bmatrix} \bmod 26 = \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix}$

Scanned by TapScanner

How we will decrypt

Cipher = FK MF 20

$$C = \begin{bmatrix} F \\ K \end{bmatrix} = \begin{bmatrix} 5 \\ 10 \end{bmatrix}$$

$\therefore$ plain text, $P = K^{-1} C \bmod 26$

$$= \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix} \begin{bmatrix} 5 \\ 10 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 260 \\ 305 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 0 \\ 19 \end{bmatrix} = \begin{bmatrix} A \\ T \end{bmatrix}$$

Similarly, $C = \begin{bmatrix} M \\ F \end{bmatrix} = \begin{bmatrix} 12 \\ 5 \end{bmatrix}$

Corresponding plain text $= \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix} \begin{bmatrix} 12 \\ 5 \end{bmatrix} \bmod 26$

$$= \begin{bmatrix} 149 \\ 390 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 19 \\ 0 \end{bmatrix} = \begin{bmatrix} T \\ A \end{bmatrix}$$

Scanned by TapScanner

Again, for the vector $\begin{bmatrix} I \\ O \end{bmatrix} = \begin{bmatrix} 8 \\ 14 \end{bmatrix}$

$$P = \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix} \begin{bmatrix} 8 \\ 14 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 366 \\ 452 \end{bmatrix} \bmod 26$$

$$= \begin{bmatrix} 2 \\ 10 \end{bmatrix} = \begin{bmatrix} C \\ K \end{bmatrix}$$

$\therefore$ $\boxed{\text{plain text : A T T A C K}}$

Scanned by TapScanner

* Transposition Techniques :

As we know, substitution techniques focus on substituting a plain text alphabet with a cipher text alphabet.

Transposition techniques differ from substitution technique that they do not simply replace one alphabet with another, they also perform some permutation over the plain text alphabets.

1) Rail Fence Technique :

step 1 : write down the plain text msg. as a sequence of diagonals.

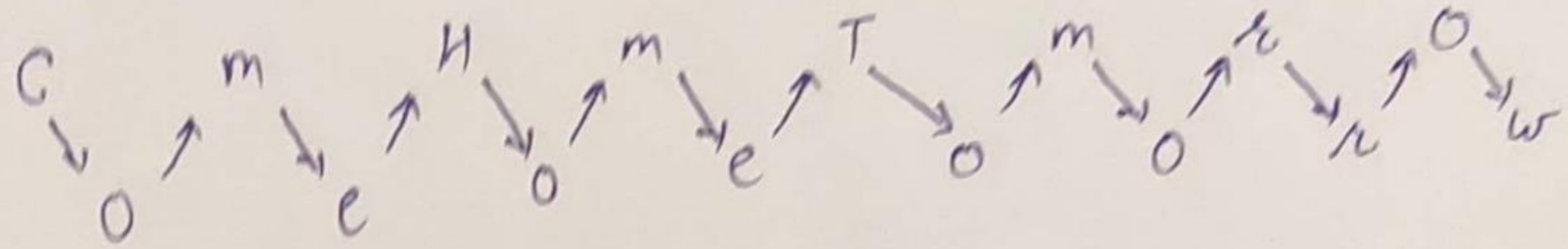
step 2 : Read the plain text written in step 1 as a sequence of rows.

"Rail fence techniques involves writing plain text as sequence of diagonals and then reading pt row-by-row to produce cipher text."

Scanned by TapScanner

Eg: Original plain text msg: Come Home Tomorrow

1) arrange the plain text msg. as a sequence of diagonals.



2) Now read the text row-by-row and write it sequentially. So

Cipher text msg: CmHmTmroeoeoorw

2) Simple Columnar Transposition Technique:

Step 1: Write the plain text msg row-by-row in a rectangle of a pre-defined size.

Step 2: Read the msg. 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.

Step 3: Thus msg. obtained, ~~which~~ is in cipher text msg.

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

Scanned by TapScanner

3) Verman Cipher (One - Time Pad):

POORNIMA
COLLEGE OF ENGINEERING

DETAILED LECTURE NOTES

Eg. Original plain text msg: Come home tomorrow

1) Let's consider rectangle six columns. & write the msg. in rectangle row-by-row.

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

2) Let's the order of column is random like 4, 6, 1, 2, 5 and 3, then read the text in the order of these columns.

3) Cipher text msg: eowoocmroerhmmto.

Simple Columnar Transposition Technique with Multiple Rounds:

Step 1 to 3 : same

Step 4: Repeat steps 1 to 3 as many times as desired.

"Cipher text produced by Simple columnar Transposition Technique with multiple rounds is much more complex to crack as compared to basic technique."

Scanned by TapScanner

3) Verman Cipher (One - Time Pad):

→ It is implemented using a random set of non-repeating characters as the i/p cipher text.

→ Once an i/p cipher text for transposition is used, it is never used again for any other msg. (hence the name one - time).

→ The length of i/p cipher text is equal to the original plain text.

Step 1: Treat each plain text alphabet as a no. in an increasing sequence, i.e. A = 0, B = 1, ... Z = 25.

Step 2: Do the same for each character of the input cipher text.

Step 3: Add each no. corresponding to the plain text alphabet to the corresponding input cipher text alphabet no.

Step 4: If the sum is greater than 26, subtract 26 from it.

Step 5: Translate each no. of the sum back to the corresponding alphabet. This gives the output cipher text.

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

Scanned by TapScanner

eg. plain text msg: How Are You
using a one-time pad NCBTZQARX to produce a cipher text msg. UQXTQUYFR

1) Plain text : H O W A R E Y O U
7 14 22 0 17 4 24 14 20

+

2) One time pad : 13 2 1 19 25 16 0 17 23
N C B T Z Q A R X

3) Initial total : 20 16 23 19 $\boxed{42}$ 20 24 $\boxed{31}$ $\boxed{43}$

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

4) Book Cipher / Running key Cipher:

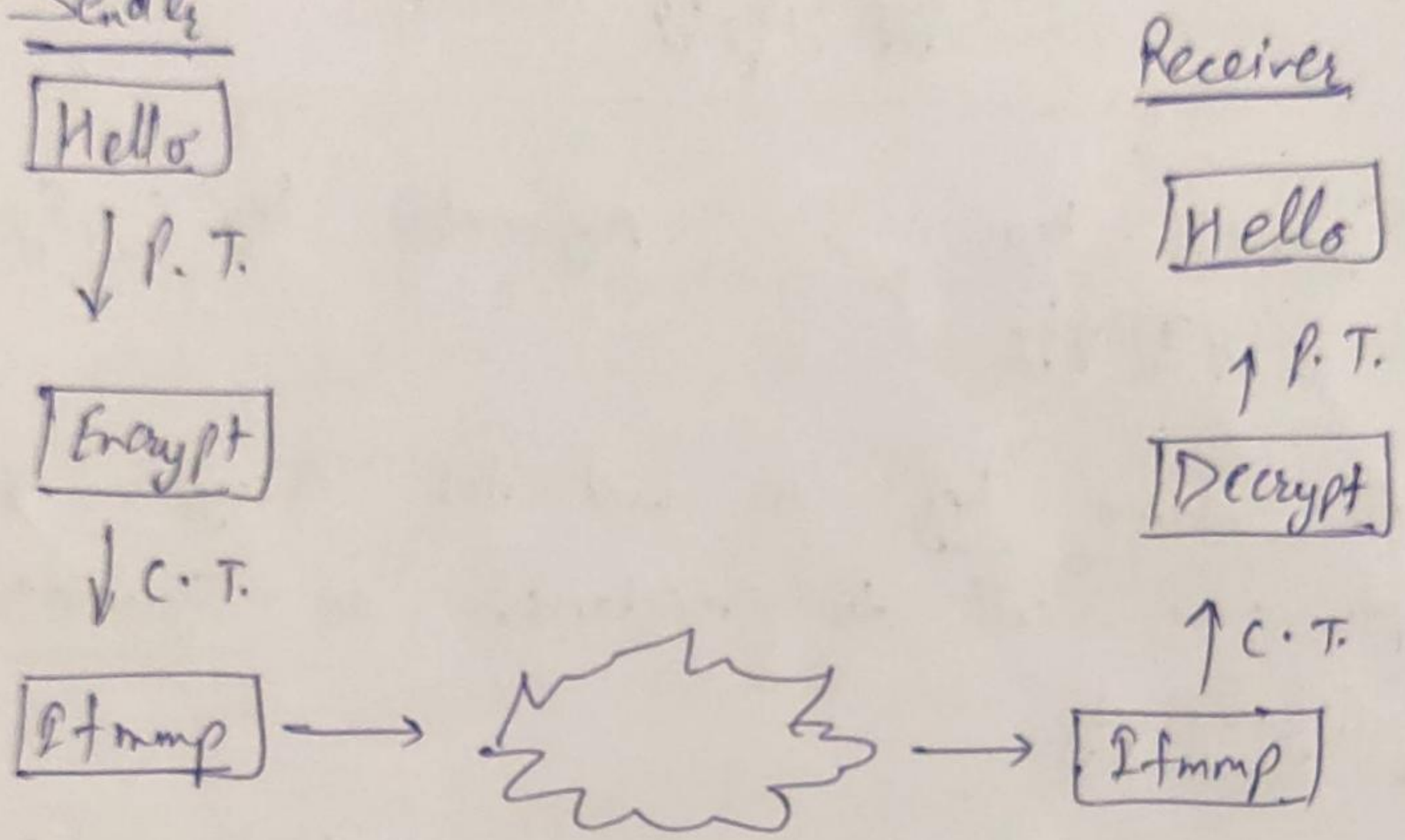
The Idea used in Book cipher, also called as Running key cipher is quite simple and is similar in principle to vernan cipher.

For producing cipher text, Some portion of text from a book is used, which serves the purpose of a one-time pad.

Thus, characters from a book are used as one-time pad and they are added to the input plain text msg. similar to the way a one-time pad works.

Scanned by TapScanner

* **Encryption & Decryption:** Encryption transforms a plain text msg. into cipher text, whereas decryption transforms a cipher text msg. back into plain text.



[Encryption & Decryption in the real world]

Every encryption and decryption process ~~two~~ has two aspects: the algorithm and the key used for encryption & decryption.

Inputs to encryption and decryption processes

- Algorithm
- Key.

Scanned by TapScanner

" the algo. used for encryption & decryption processes is usually known to everybody. However, it is key used for encryption & decryption that makes the process of cryptography secure."

Cryptography techniques

- Symmetric key Cryptography
- Asymmetric key cryptography.

→ If the same key is used for encryption & decryption ; we call the mechanism as Symmetric key Cryptography.

→ if two diff. 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 as Asymmetric key cryptography.

Scanned by TapScanner

* Symmetric and Asymmetric key :

1) Symmetric key Cryptography and the prob. of key distribution:

"A person wants to send a highly confidential letter to another person B, A and B both resides in the same city, but are seperated by a few miles and for some reason, cannot meet each other."

i) **Solution :** A puts the confidential letter in an envelope, seal it and sends it by post. A hope that no one opens it before it reaches B.

Problem : will not guarantee that the envelope does not get opened before it reaches B. After all, someone can open the envelope, read the confidential letter and resealed the envelope.

ii) **Solution :** hand - delivery mechanism.

problem : not fool proof.

As a result, we observe that no sol^n is completely acceptable. Either it is not fool proof or is not practically possible. This is the prob. of key distribution or key exchange. Since the sender and receiver will use the same key to lock and unlock, this is called symmetric key operation.

Scanned by TapScanner

2) Diffie-Hellman Key Exchange / Agreement Algo:

Let us assume that Alice and Bob want to agree upon a key to be used for encryption / decryption msg. that would be exchanged b/w them.

Algo:

i) Alice and Bob agree on two large prime no., n and g. These two integers need not to be kept secret. Alice and Bob can use an insecure channel to agree on them.

ii) Alice chose another large random no. x, & calculates A such that:

$$A = g^x \bmod n$$

iii) Alice sends the no. A to Bob.

iv) Bob independently chooses another large random integer y and calculates B such that:

$$B = g^y \bmod n$$

v) Bob sends the no. B to Alice.

vi) A now computes the secret key K1 as follows:

$$\boxed{K_1 = B^x \bmod n}$$

vii) B now computes the secret key K2 as follows:

$$\boxed{K_2 = A^y \bmod n}$$

Scanned by TapScanner

eg. i) Alice and Bob agree on two large prime no. n and g.

Let $n = 11$, $g = 7$

ii) $A = g^x \bmod n$

Let $x = 3$, then $A = 7^3 \bmod 11 = 343 \bmod 11 = 2$

iii) Alice sends 2 to Bob.

iv) $B = g^y \bmod n$

Let $y = 6$, then $B = 7^6 \bmod 11 = 117649 \bmod 11 = 4$

v) Bob sends 4 to alice.

vi) $K1 = B^x \bmod n$

$K_1 = 4^3 \bmod 11 = 64 \bmod 11 = 9$

vii) $K2 = A^y \bmod n$

$K_2 = 2^6 \bmod 11 = 64 \bmod 11 = 9$

Problem with this algo: man-in-the-middle attack.

i) Alice wants to comm. Bob securely. So they agree on two no. that is.

Let $n = 11$ and $g = 7$

(Symmetric key $K_1 = K_2 = K$)

Scanned by TapScanner

ii) Alice does not realize that the attacker Tom is listening quietly to the conversion b/w her & Bob. Tom simply picks up the value of n and g and also forwards them to Bob.

Alice	Tom	Bob
$n = 11, g = 7$	$n = 11, g = 7$	$n = 11, g = 7$

iii) assume that, Alice, Tom & Bob select random no. x and y

Alice	Tom	Bob
$x = 3$	$x = 8, y = 6$	$y = 9$

iv) Tom selects bot x & y? because Tom calculates both A and B

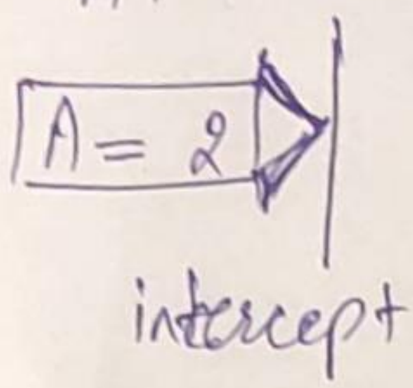
Alice	Tom	Bob
$A = g^x \bmod n$	$A = g^x \bmod n$	$B = g^y \bmod n$
$= 7^3 \bmod 11$	$= 7^8 \bmod 11$	$= 7^9 \bmod 11$
$= 343 \bmod 11$	$= 5764801 \bmod 11$	$= 40353607 \bmod 11$
$= 2$	$= 9$	$= 8$

$B = g^y \bmod n$
$= 7^6 \bmod 11$
$= 117649 \bmod 11$
$= 4$

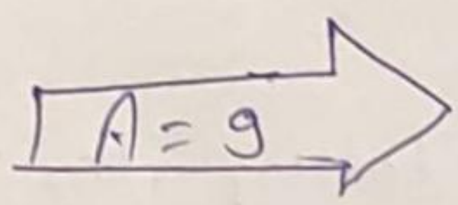
Scanned by TapScanner

v) Now,

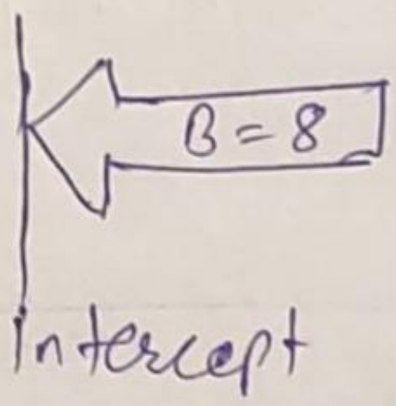
Alice | Tom | Bob

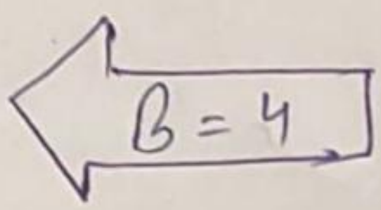


Tom intercepts the value of A sent by Alice to Bob and sends Bob his own A, instead.



Tom intercepts the value of B sent by Bob to Alice and sends Alice his own B, instead.





Alice	Tom	Bob
$A = 2,\ B = 4^*$	$A = 2,\ B = 8$	$A = 9^*,\ B = 8$

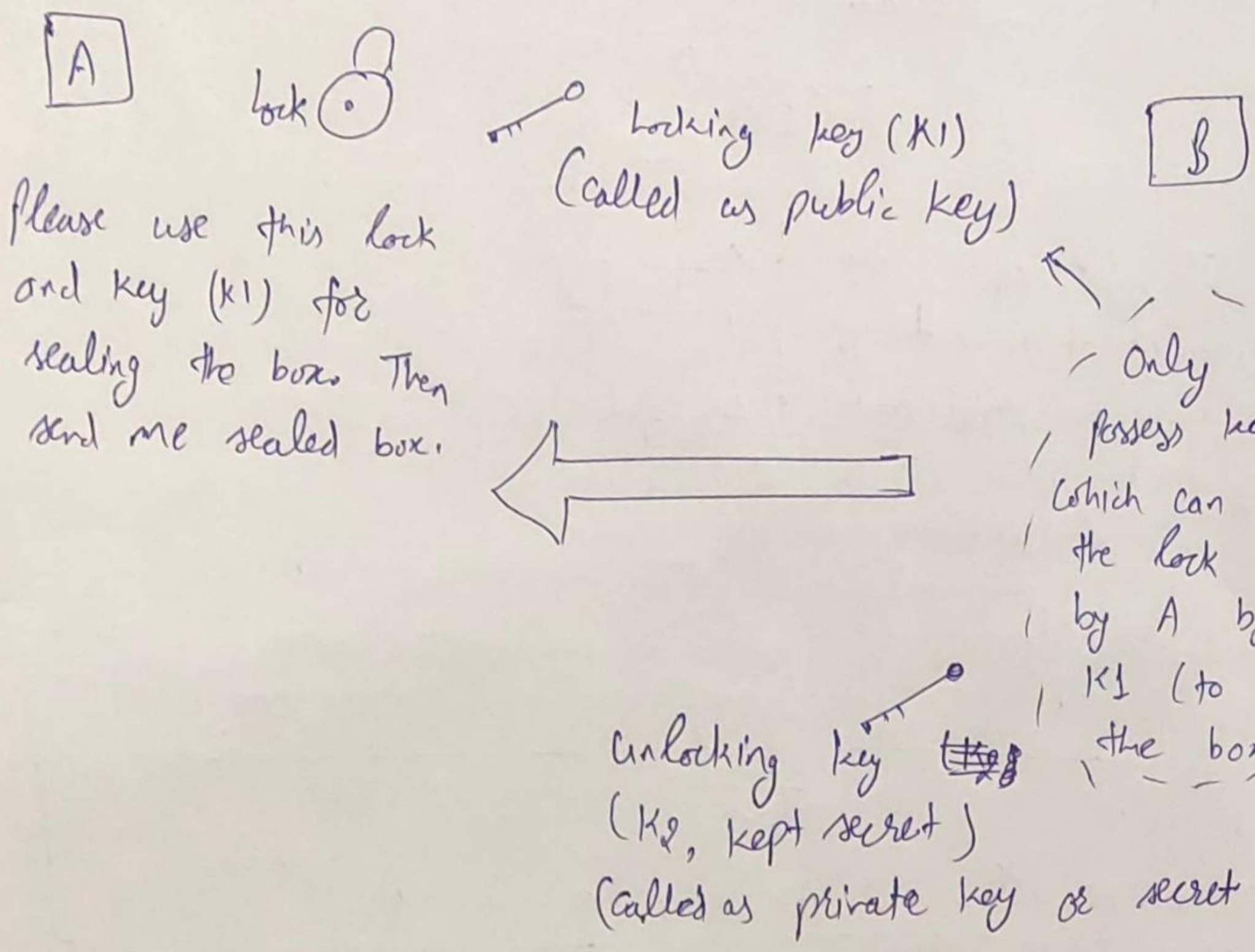
vi) Now Alice calculate only K_1, Bob only K_2 and Tom both K_1 & K_2

Alice	Tom	Bob
$K_1 = B^x \bmod n$	$K_1 = B^x \bmod n$	$K_2 = A^y \bmod n$
$= 4^3 \bmod 11$	$= 8^8 \bmod 11$	$= 9^9 \bmod 11$
$= 64 \bmod 11$	$= 5$	$= 5$
$= 9$	$K_2 = 2^6 \bmod 11$	
	$= 9$	
	$(= A^y \bmod n)$	

Scanned by TapScanner

As we see, the man-in-the-middle attack can work against the Diffie-Hellman key exchange algo. causing it to fail.

3) <u>Asymmetric key operation:</u>



[Use of key pair]

Scanned by TapScanner