## Chapter -2

## CLASSICAL ENCRYPTIONTECHNIQUES

### SYMMETRIC CIPHER MODEL

A symmetric encryption scheme has five ingredients (Figure 2.1):

• **Plaintext:** This is the original intelligible message or data that is fed into the algorithm as input.

• **Encryption algorithm:** The encryption algorithm performs various substitu-tions and transformations on the plaintext.

• **Secret key:** The secret key is also input to the encryption algorithm. The key is a value independent of the plaintext and of the algorithm. The algorithm will produce a different output depending on the specific key being used at the time. The exact substitutions and transformations performed by the algorithm depend on the key.

• **Ciphertext:** This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different ciphertexts. The ciphertext is an apparently random stream of data and, as it stands, is unintelligible.

• **Decryption algorithm:** This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key and produces the original plaintext.

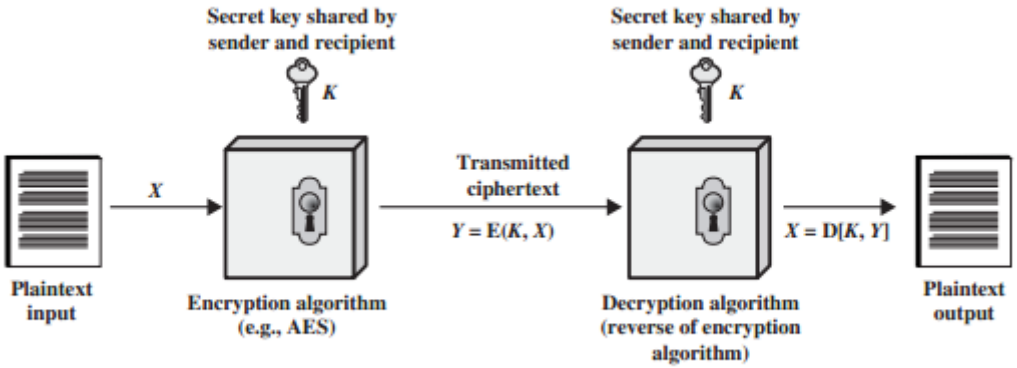



Figure 2.1 Simplified Model of Symmetric Encryption

In other words, we do not need to keep the algorithm secret; we need to keep only the key secret. This feature of symmetric encryption is what makes it feasible for widespread use.

Let us take a closer look at the essential elements of a symmetric encryp-tion scheme, using Figure 2.2. A source produces a message in plaintext,

$X = [X\,1\,, X\,2\,\ldots\ldots X_M]$

The elements of are letters in some finite alphabet. Traditionally, the alphabet usually consisted of the 26 capital letters. Nowadays, the binary alphabet {0, 1} is typically used. For encryption, a key of the form

$K = [K1, K2\ldots\ldots K_J]$

is generated. If the key is generated at the message source, then it must also be provided to the destination by means of some secure channel. Alternatively, a third party could generate the key and securely deliver it to both source and destination.

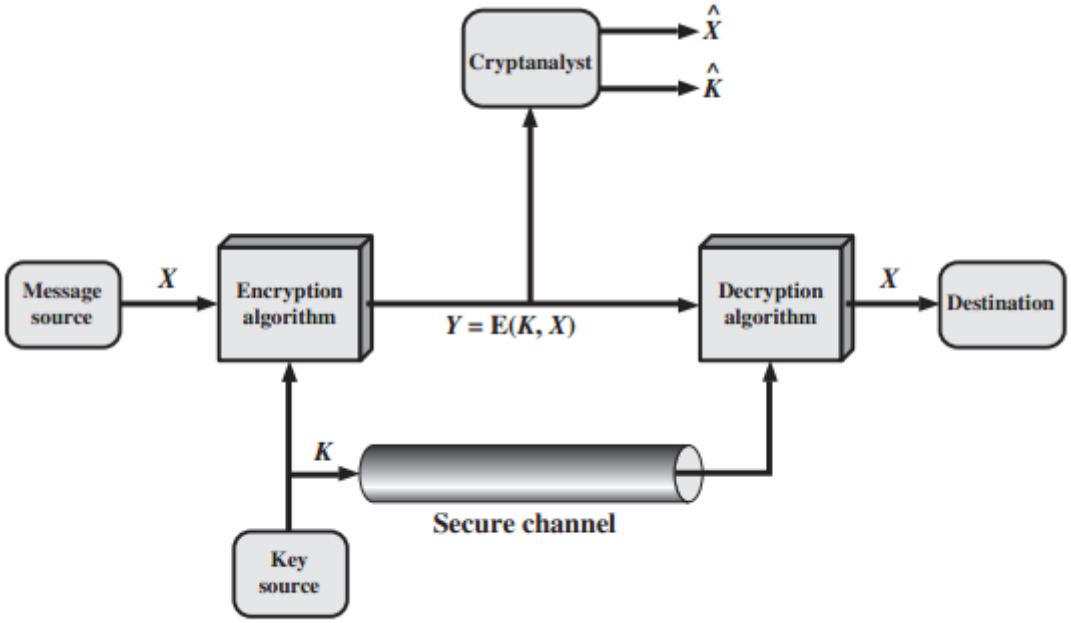



Figure 2.2 Model of Symmetric Cryptosystem

With the message and the encryption key as input, the encryption algorithm forms the ciphertext

$Y=[Y1,Y2\ldots\ldots Y_N]$

We can write this as Y = E(K, X)

This notation indicates that is produced by using encryption algorithm E as a function of the plaintext , with the specific function determined by the value of the key .

The intended receiver, in possession of the key, is able to invert the transformation: X = D(K, Y)

An opponent, observing but not having access to or , may attempt to recover or or both and . It is assumed that the opponent knows the encryption (E) and decryption (D) algorithms. If the opponent is interested in only this particular message, then the focus of the effort is to recover X by generating a plaintext estimate X.

Often, however, the opponent is interested in being able to read future messages as well, in which case an attempt is made to recover K by generating an estimate K.

## Cryptanalysis and Brute-Force Attack

Typically, the objective of attacking an encryption system is to recover the key in use rather than simply to recover the plaintext of a single ciphertext. There are two general approaches to attacking a conventional encryption scheme:

• **Cryptanalysis:** Cryptanalytic attacks rely on the nature of the algorithm plus perhaps some knowledge of the general characteristics of the plaintext or even some sample plaintext–ciphertext pairs. This type of attack exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or to deduce the key being used.

• **Brute-force attack:** The attacker tries every possible key on a piece of cipher-text until an intelligible translation into plaintext is obtained. On average, half of all possible keys must be tried to achieve success.

Table 2.1 summarizes the various types of cryptanalytic attacks based on the amount of information known to the cryptanalyst.

Table 2.1 Types of Attacks on Encrypted Messages

| Type of Attack | Known to Cryptanalyst |
|---|---|
| Ciphertext Only | • Encryption algorithm<br>• Ciphertext |
| Known Plaintext | • Encryption algorithm<br>• Ciphertext<br>• One or more plaintext–ciphertext pairs formed with the secret key |
| Chosen Plaintext | • Encryption algorithm<br>• Ciphertext<br>• Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key |
| Chosen Ciphertext | • Encryption algorithm<br>• Ciphertext<br>• Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |
| Chosen Text | • Encryption algorithm<br>• Ciphertext<br>• Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key<br>• Ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key |

The **ciphertext-only attack** is the easiest to defend against because the opponent has the least amount of information to work with. In many cases, however, the analyst has more information. The analyst may be able to capture one or more plaintext messages as well as their encryptions.

For example, a file that is encoded in the Postscript format always begins with the same pattern, or there may be a standardized header or banner to an electronic funds transfer message, and so on

Closely related to the **known-plaintext attack** is what might be referred to as a probable-word attack. If the opponent is working with the encryption of some general prose message, he or she may have little knowledge of what is in the message. However, if the opponent is after some very specific information, then parts of the message may be known.

As another example, the source code for a program developed by Corporation X might include a copyright statement in some standardized position.

If the analyst is able somehow to get the source system to insert into the system a message chosen by the analyst, then a **chosen-plaintext attack** is possible.

Table 2.1 lists two other types of attack: **chosen ciphertext** and **chosen text.** These are less commonly employed as cryptanalytic techniques but are nevertheless possible avenues of attack.

A **brute-force attack** involves trying every possible key until an intelligible translation of the ciphertext into plaintext is obtained. On average, half of all possible keys must be tried to achieve success. Table 2.2 shows how much time is involved for various key spaces. Results are shown for four binary key sizes. The 56-bit key size is used with the Data Encryption Standard (DES) algorithm, and the 168-bit key size is used for triple DES. The minimum key size specified for Advanced Encryption Standard (AES) is 128 bits.

Table 2.2 Average Time Required for Exhaustive Key Search

| Key Size (bits) | Number of Alternative Keys | Time Required at 1 Decryption/$\mu$s | Time Required at $10^6$ Decryptions/$\mu$s |
|---|---|---|---|
| 32 | $2^{32} = 4.3 \times 10^9$ | $2^{31}\mu s = 35.8$ minutes | 2.15 milliseconds |
| 56 | $2^{56} = 7.2 \times 10^{16}$ | $2^{55}\mu s = 1142$ years | 10.01 hours |
| 128 | $2^{128} = 3.4 \times 10^{38}$ | $2^{127}\mu s = 5.4 \times 10^{24}$ years | $5.4 \times 10^{18}$ years |
| 168 | $2^{168} = 3.7 \times 10^{50}$ | $2^{167}\mu s = 5.9 \times 10^{36}$ years | $5.9 \times 10^{30}$ years |
| 26 characters (permutation) | $26! = 4 \times 10^{26}$ | $2 \times 10^{26}\mu s = 6.4 \times 10^{12}$ years | $6.4 \times 10^6$ years |

For each key size, the results are shown assuming that it takes 1 μs to perform a single decryption, which is a reason-able order of magnitude for today's machines. With the use of massively parallel organizations of microprocessors, it may be possible to achieve processing rates many orders of magnitude greater.

# How to Defend Against Brute Force Attacks

- **Increase password length**: More characters equal more time to brute force crack
- **Increase password complexity**: More options for each character also increase the time to brute force crack
- **Limit login attempts**: Brute force attacks increment a counter of failed login attempts on most directory services – a good defense against brute force attacks is to lock out users after a few failed attempts, thus nullifying a brute force attack in progress
- **Implement Captcha**: Captcha is a common system to verify a human is a human on websites and can stop brute force attacks in progress
- **Use multi-factor authentication**: Multi-factor authentication adds a second layer of security to each login attempt that requires human intervention which can stop a brute force attack from success

# SUBSTITUTION TECHNIQUES

The two basic building blocks of all encryption techniques are substitution and transposition.

A substitution technique is one in which the letters of plaintext are replaced by other letters or by numbers or symbols. 1 If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns.

1. **Caesar Cipher:-** The earliest known, and the simplest, use of a substitution cipher was by Julius Caesar. The Caesar cipher involves replacing each letter of the alphabet with the let-ter standing three places further down the alphabet. For example,

```
plain:  meet me after the toga party
cipher: PHHW PH DIWHU WKH WRJD SDUWB
```

Note that the alphabet is wrapped around, so that the letter following Z is A. We can define the transformation by listing all possibilities, as follows:

```
plain:   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:  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
```

Let us assign a numerical equivalent to each letter:

| a | b | c | d | e | f | g | h | i | j | k | l | m |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

| n | o | p | q | r | s | t | u | v | w | x | y | z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

Then the algorithm can be expressed as follows. For each plaintext letter $p$, substitute the ciphertext letter $C$:[2]

$$C = \mathrm{E}(3, p) = (p + 3) \bmod 26$$

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

$$C = \mathrm{E}(k, p) = (p + k) \bmod 26 \qquad \textbf{(2.1)}$$

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

$$p = \mathrm{D}(k, C) = (C - k) \bmod 26 \qquad \textbf{(2.2)}$$

If it is known that a given ciphertext is a Caesar cipher, then a brute-force cryptanalysis is easily performed: simply try all the 25 possible keys.

**Process of Simple Substitution Cipher**

- Write the alphabets A, B, C,...,Z in the natural order.
- The sender and the receiver decide on a randomly selected permutation of the letters of the alphabet.
- Underneath the natural order alphabets, write out the chosen permutation of the letters of the alphabet. For encryption, sender replaces each plaintext letters by substituting the permutation letter that is directly beneath it in the table. This process is shown in the following illustration. In this example, the chosen permutation is K,D, G, ..., O. The plaintext 'point' is encrypted to 'MJBXZ'.

Here is a jumbled Ciphertext alphabet, where the order of the ciphertext letters is a key.

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

**Monoalphabetic Ciphers**

With only 25 possible keys, the Caesar cipher is far from secure. A dramatic increase in the key space can be achieved by allowing an arbitrary substitution. Before proceeding, we define the term permutation. A permutation of a finite set of elements  is an ordered sequence of all the elements of  , with each element appearing exactly once. For example, if, there are six permutations of  :

abc, acb, bac, bca, cab, cba

In general, there are $n!$ permutations of a set of $n$ elements, because the first element can be chosen in one of $n$ ways, the second in $n - 1$ ways, the third in $n - 2$ ways, and so on.

Recall the assignment for the Caesar cipher:

```
plain:  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: 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
```

If, instead, the "cipher" line can be any permutation of the 26 alphabetic characters, then there are 26! or greater than $4*10^{26}$ possible keys. This is 10 orders of magnitude greater than the key space for DES and would seem to eliminate brute-force techniques for cryptanalysis. Such an approach is referred to as a monoalphabetic

substitution cipher, because a single cipher alphabet (mapping from plain alphabet to cipher alphabet) is used per message.

There is, however, another line of attack. If the cryptanalyst knows the nature of the plaintext (e.g., non-compressed English text), then the analyst can exploit the regularities of the language.

Example:

The ciphertext to be solved is

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
```

As a first step, the relative frequency of the letters can be determined and compared to a standard frequency distribution for English, such as is shown in Figure 2.5 (based on [LEWA00])

| | | | | |
|---|---|---|---|---|
| P 13.33 | H 5.83 | F 3.33 | B 1.67 | C 0.00 |
| Z 11.67 | D 5.00 | W 3.33 | G 1.67 | K 0.00 |
| S 8.33 | E 5.00 | Q 2.50 | Y 1.67 | L 0.00 |
| U 8.33 | V 4.17 | T 2.50 | I 0.83 | N 0.00 |
| O 7.50 | X 4.17 | A 1.67 | J 0.83 | R 0.00 |
| M 6.67 | | | | |

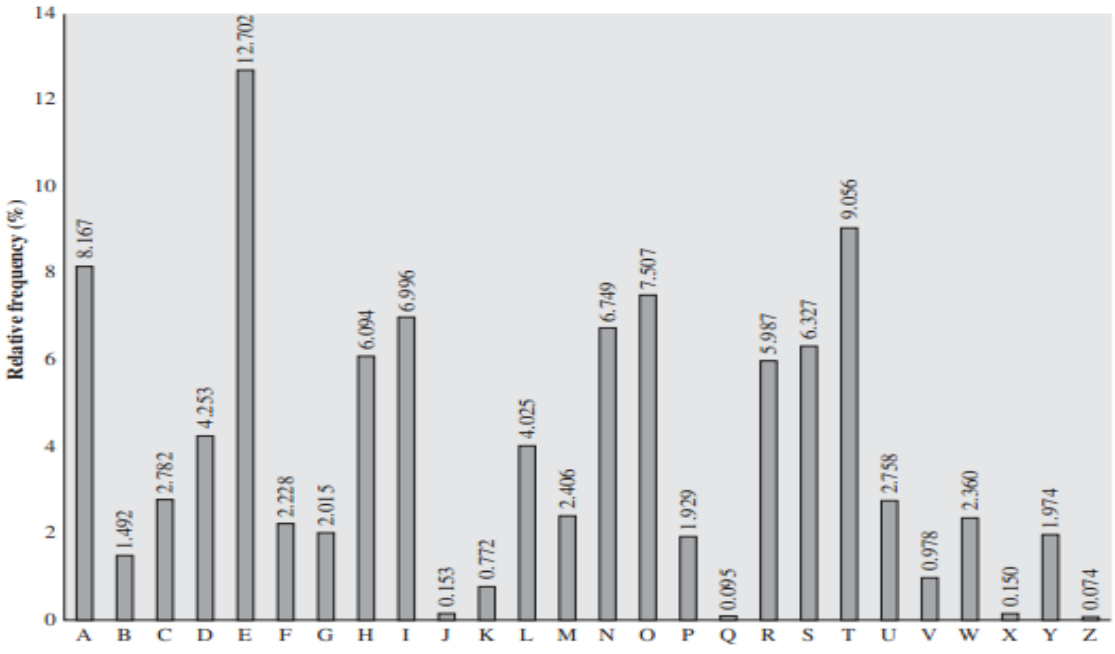



Figure 2.5 Relative Frequency of Letters in English Text

Comparing this breakdown with Figure 2.5, it seems likely that cipher letters P and Z are the equivalents of plain letters e and t, but it is not certain which is which. The letters S, U, O, M, and H are all of relatively high

frequency and probably correspond to plain letters from the set {a, h, i, n, o, r, s}. The letters with the lowest frequencies (namely, A, B, G, Y, I, J) are likely included in the set {b, j, k, q, v, x, z}.

```
UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
 t a         e  e te  a that e e a        a
VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
   e t   ta t ha e ee  a e  th    t  a
EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
 e  e e tat  e   the   t
```

Only four letters have been identified, but already we have quite a bit of the message. Continued analysis of frequencies plus trial and error should easily yield a solution from this point. The complete plaintext, with spaces added between words, follows:

```
it was disclosed yesterday that several informal but
direct contacts have been made with political
representatives of the viet cong in moscow
```

Monoalphabetic ciphers are easy to break because they reflect the frequency data of the original alphabet. A countermeasure is to provide multiple substitutes, known as homophones, for a single letter. For example, the letter e could be assigned a number of different cipher symbols, such as 16, 74, 35, and 21, with each homophone assigned to a letter in rotation or randomly.

**Playfair Cipher**

The best-known multiple-letter encryption cipher is the Playfair, which treats diagrams in the plaintext as single units and translates these units into ciphertext diagrams.

The Playfair algorithm is based on the use of a 5 × 5 matrix of letters constructed using a keyword. Here is an example, solved by Lord Peter Wimsey in Dorothy Sayers's Have His Carcase:

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

In this case, the keyword is monarchy. The matrix is constructed by filling in the letters of the keyword (minus duplicates) from left to right and from top to bot-tom, and then filling in the remainder of the matrix with the remaining letters in alphabetic order. The letters I and J count as one letter. Plaintext is encrypted two letters at a time, according to the following rules:

1. Repeating plaintext letters that are in the same pair are separated with a filler letter, such as x, so that balloon would be treated as ba lx lo on.

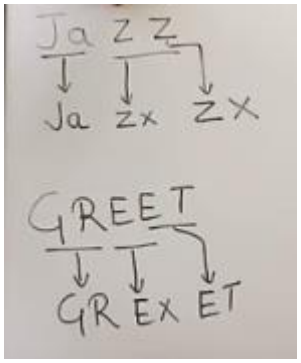
2. Two plaintext letters that fall in the same row of the matrix are each replaced by the letter to the right, with the first element of the row circularly following the last. For example, ar is encrypted as RM.

3. Two plaintext letters that fall in the same column are each replaced by the letter beneath, with the top element of the column circularly following the last. For example, mu is encrypted as CM.

4. Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row and the column occupied by the other plaintext letter. Thus, hs becomes BP and ea becomes IM (or JM, as the encipherer wishes).

Make pair of Two alphabets.

Example: If same letters, then add 'X'.




## Example:

**Key Given:** "PlayFair Example"

**Plain Text:** "Hide the gold in the tree stump"

Matrix:

| | | | | |
|---|---|---|---|---|
| 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 |

**Plain text Pair**: HI DE TH EG OL DI NT HE TR EX ES TU MP

**CIPHER TEXT: BM OD ZB XD NA BE KU DM UI XM MO UV IF**

Despite this level of confidence in its security, the Playfair cipher is relatively easy to break, because it still leaves much of the structure of the plaintext language intact. A few hundred letters of ciphertext are generally sufficient.

## Hill Cipher

Another interesting multiletter cipher is the Hill cipher, developed by the mathe-matician Lester Hill in 1929.

**THE HILL ALGORITHM:** This encryption algorithm takes successive plaintext letters and substitutes for them ciphertext letters. The substitution is determined by linear equations in which each character is assigned a numerical value (a=0, b=1, c=2…….z=25). For m=3 system can be defined as

$$c_1 = (k_{11}p_1 + k_{12}p_2 + k_{13}p_3) \bmod 26$$
$$c_2 = (k_{21}p_1 + k_{22}p_2 + k_{23}p_3) \bmod 26$$
$$c_3 = (k_{31}p_1 + k_{32}p_2 + k_{33}p_3) \bmod 26$$

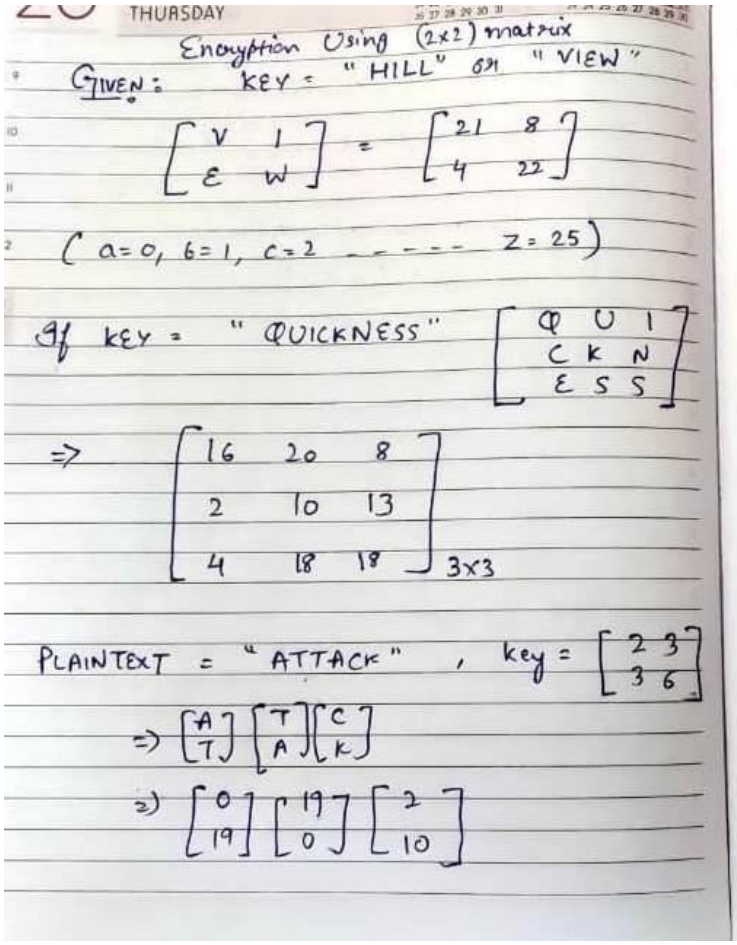
This can be expressed in terms of row vectors and matrices:[7]

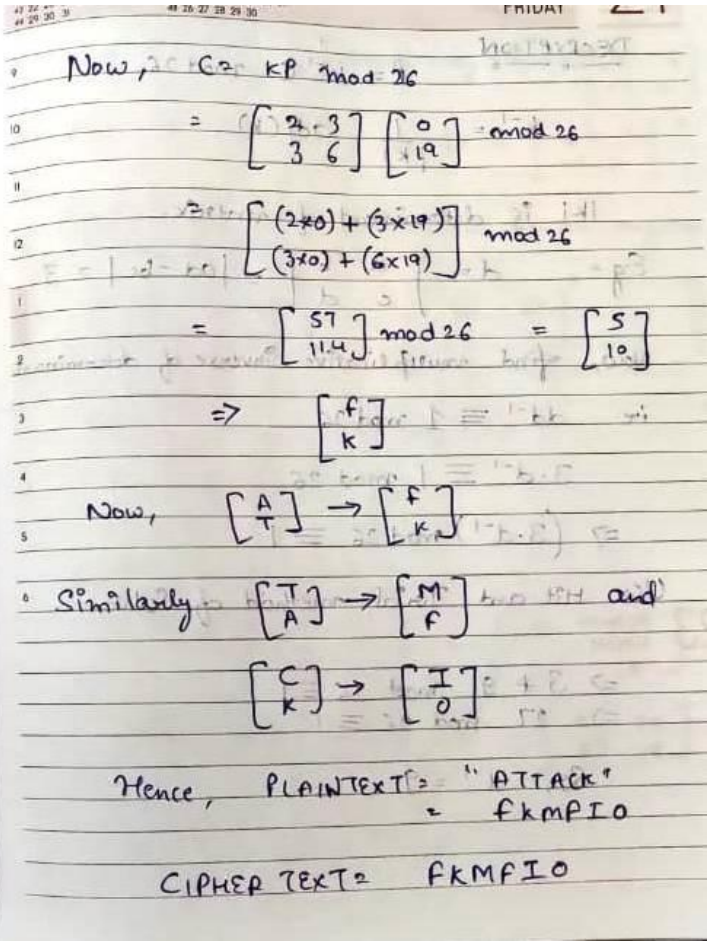
$$(c_1\ c_2\ c_3) = (p\ p_2\ p_3)\begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix} \bmod 26$$

or

$$\mathbf{C} = \mathbf{PK} \bmod 26$$

Example: Given Key- HILL


THURSDAY
Encryption Using (2x2) matrix
GIVEN: KEY = "HILL" OR "VIEW"
[V I; E W] = [21 8; 4 22]
(a=0, b=1, c=2 ----- z=25)
If KEY = "QUICKNESS" [Q U I; C K N; E S S]
=> [16 20 8; 2 10 13; 4 18 18] 3x3
PLAINTEXT = "ATTACK", key = [2 3; 3 6]
=> [A T][T A][C K]
=> [0 19][19 0][2 10]


Now, C = KP mod 26
= [2 3; 3 6][0 19] mod 26
= [(2x0)+(3x19); (3x0)+(6x19)] mod 26
= [57 114] mod 26 = [5 10]
=> [F K]
Now, [A T] -> [F K]
Similarly [T A] -> [M F] and
[C K] -> [I O]
Hence, PLAINTEXT = "ATTACK"
= fkmfio
CIPHER TEXT = FKMFIO

2018 SEPTEMBER

22 DAY 265-100 SATURDAY

DECRYPTION

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

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

$|K|$ is determinant of matrix.

Eg = $d = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = |ad - bc| = 3$

Now, find multiplicative inverse of determinant

ie $dd^{-1} \equiv 1 \mod 26$

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

$\Rightarrow (3 \cdot d^{-1}) \mod 26 \equiv 1$

Use Hit and Trial method of Solve.

23 DAY 266-099 SUNDAY

$\Rightarrow 3 * 9 \mod 26 \equiv 1$

$\Rightarrow 27 \mod 26 \equiv 1$

So, $d^{-1} = 9$

SEPTEMBER 2018

24 DAY 267-098 MONDAY

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

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

Before, decryption, we have to remove negative values, so add 26.

$\begin{bmatrix} 6 & -3+26 \\ -3+26 & 2 \end{bmatrix} = \begin{bmatrix} 6 & 23 \\ 23 & 2 \end{bmatrix}$

Now, $K^{-1} = \frac{1}{|K|} adj(K)$

$= \frac{1}{|d|} adj(K)$

$= d^{-1} adj(K)$

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

Now, find modulo 26 $= \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix}$

$K^{-1} = \begin{bmatrix} 2 & 25 \\ 25 & 18 \end{bmatrix}$

TUESDAY

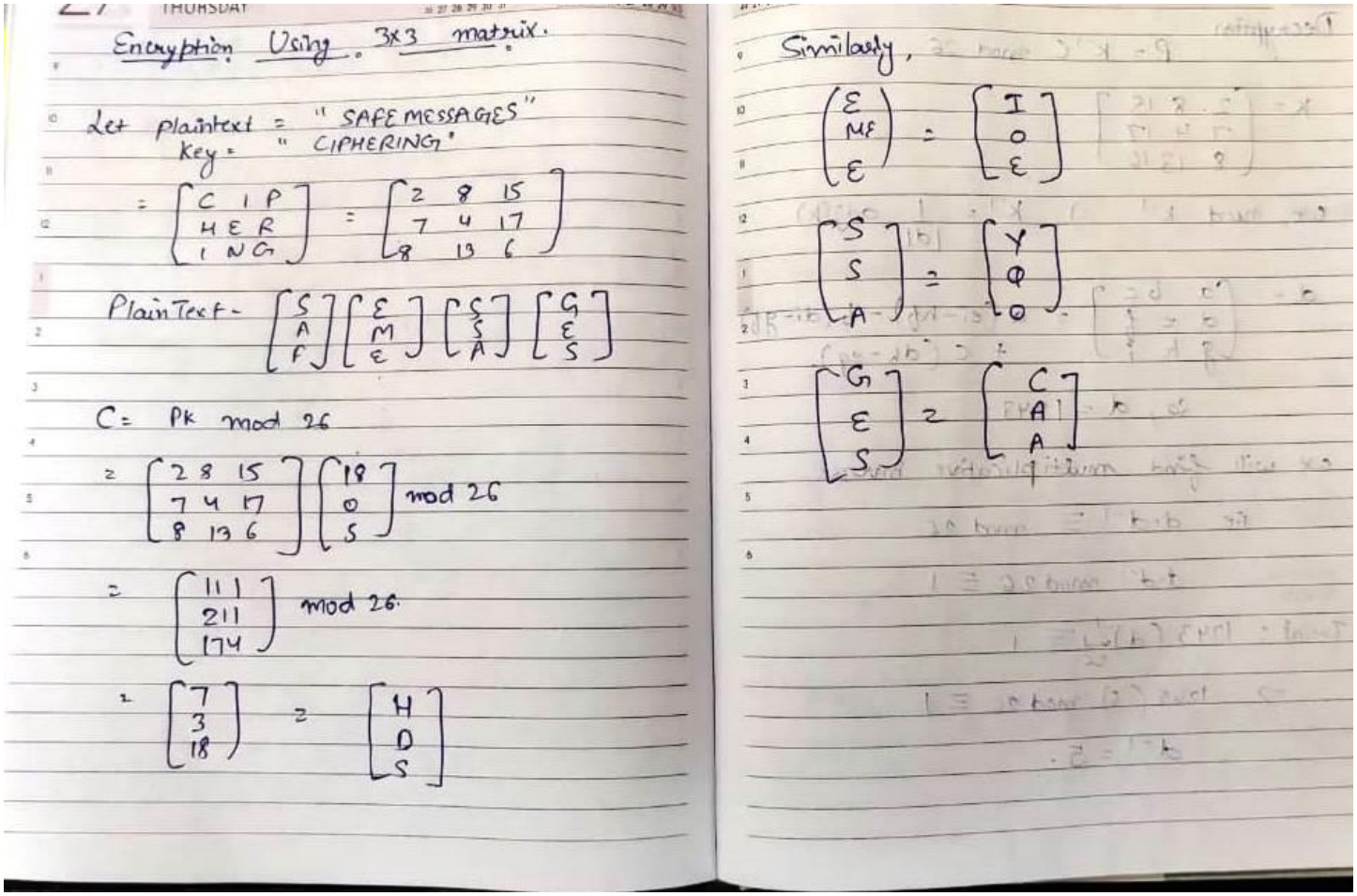
$C = \begin{bmatrix} F \\ K \end{bmatrix} \begin{bmatrix} M \\ F \end{bmatrix} \begin{bmatrix} J \\ O \end{bmatrix}$

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

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

$= \begin{bmatrix} 260 \\ 305 \end{bmatrix} \mod 26 = \begin{bmatrix} 0 \\ 19 \end{bmatrix} = \begin{bmatrix} A \\ T \end{bmatrix}$

Similarly $\begin{bmatrix} M \\ F \end{bmatrix} = \begin{bmatrix} I \\ A \end{bmatrix}$ and $\begin{bmatrix} 9 \\ 0 \end{bmatrix} = \begin{bmatrix} C \\ K \end{bmatrix}$



Encryption Using 3×3 matrix.

Let plaintext = "SAFEMESSAGES"
Key = "CIPHERING"

$$= \begin{bmatrix} C & I & P \\ H & E & R \\ I & N & G \end{bmatrix} = \begin{bmatrix} 2 & 8 & 15 \\ 7 & 4 & 17 \\ 8 & 13 & 6 \end{bmatrix}$$

Plain Text - $\begin{bmatrix} S \\ A \\ F \end{bmatrix} \begin{bmatrix} E \\ M \\ E \end{bmatrix} \begin{bmatrix} S \\ S \\ A \end{bmatrix} \begin{bmatrix} G \\ E \\ S \end{bmatrix}$

$C = Pk \mod 26$

$$= \begin{bmatrix} 2 & 8 & 15 \\ 7 & 4 & 17 \\ 8 & 13 & 6 \end{bmatrix} \begin{bmatrix} 18 \\ 0 \\ S \end{bmatrix} \mod 26$$

$$= \begin{bmatrix} 111 \\ 211 \\ 174 \end{bmatrix} \mod 26.$$

$$= \begin{bmatrix} 7 \\ 3 \\ 18 \end{bmatrix} = \begin{bmatrix} H \\ D \\ S \end{bmatrix}$$

Similarly,

$$\begin{pmatrix} E \\ ME \\ E \end{pmatrix} = \begin{bmatrix} I \\ O \\ E \end{bmatrix}$$

$$\begin{bmatrix} S \\ S \\ A \end{bmatrix} = \begin{bmatrix} Y \\ Q \\ O \end{bmatrix}$$

$$\begin{bmatrix} G \\ E \\ S \end{bmatrix} = \begin{bmatrix} C \\ A \\ A \end{bmatrix}$$

Decryption $P = K^{-1}C \mod 26$

$$K = \begin{bmatrix} 2 & 8 & 15 \\ 7 & 4 & 17 \\ 8 & 13 & 16 \end{bmatrix}$$

We need $K^{-1} \Rightarrow K^{-1} = \frac{1}{|d|} adj(K)$

$$d = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = a(ei-hf) - b(di-gf) + c(dh-eg).$$

So, $d = 1243$

We will find multiplicative inverse.

i.e $d \cdot d^{-1} \equiv \mod 26$

30 DAY 273-092 SUNDAY

$d \cdot d^{-1} \mod 26 \equiv 1$

Hit & Trial : $1243\,(d)^{-1}_{\mod 26} \equiv 1$

$\Rightarrow 1243\,(5) \mod 26 \equiv 1$

$d^{-1} = 5.$

Now, adj (K)

$$\begin{bmatrix} 2 & 8 & 15 \\ 7 & 4 & 17 \\ 8 & 13 & 16 \end{bmatrix} = \begin{bmatrix} -197 & 94 & 59 \\ 147 & -108 & 38 \\ 76 & 71 & -48 \end{bmatrix}$$

Now, remove negative signs & Transpose.

Transpose =

$$\begin{bmatrix} -197 & 147 & 76 \\ 94 & -108 & 71 \\ 59 & 38 & -48 \end{bmatrix}$$

Removing negatives with multiples of 26.

$$\begin{bmatrix} -197+26(8) & 147 & 76 \\ 94 & -108+26(5) & 71 \\ 59 & 38 & -48+26(2) \end{bmatrix}$$

$$= \begin{bmatrix} 11 & 147 & 76 \\ 94 & 22 & 71 \\ 59 & 38 & 4 \end{bmatrix}$$

Now, $K^{-1} = d^{-1}\, adj\,[K]$

$= 5 * \begin{bmatrix} 11 & 147 & 76 \\ 94 & 22 & 71 \\ 59 & 38 & 4 \end{bmatrix} \mod 26$

$$= \begin{bmatrix} 3 & 7 & 16 \\ 2 & 5 & 17 \\ 9 & 8 & 20 \end{bmatrix}$$

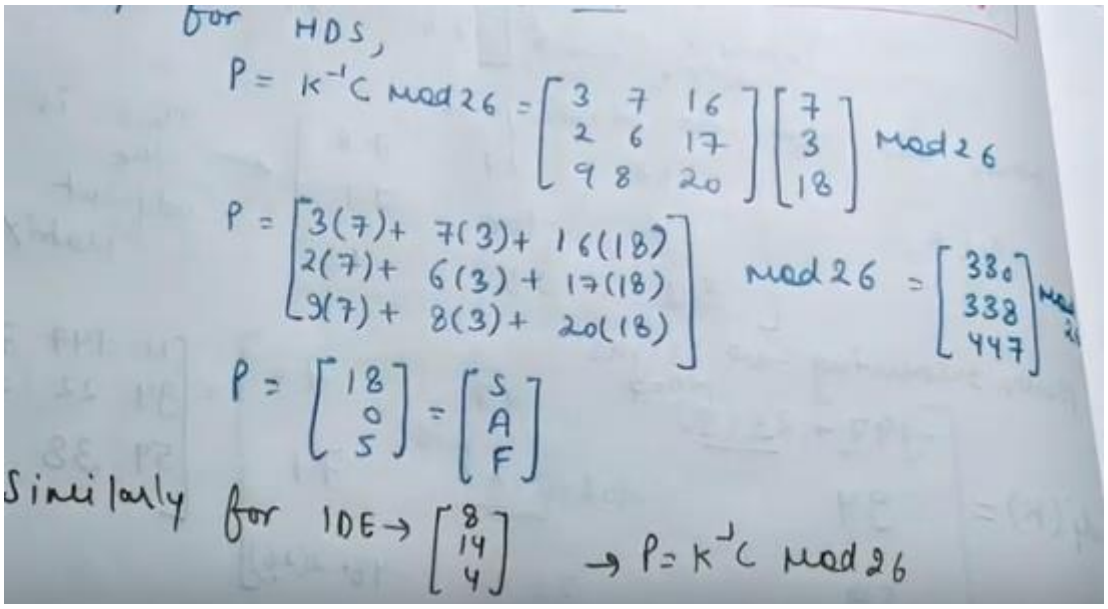
$$\begin{bmatrix} 9 & 8 & 20 \end{bmatrix}$$

Now, we have decryption formula,

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

Cipher = HDS IOE YQO CAA

Now, for HDS,




## Polyalphabetic Ciphers

Another way to improve on the simple monoalphabetic technique is to use different monoalphabetic substitutions as one proceeds through the plaintext message. The general name for this approach is polyalphabetic substitution cipher . All these techniques have the following features in common:

1. A set of related monoalphabetic substitution rules is used.

2. A key determines which particular rule is chosen for a given transformation.

**VIGENERE CIPHER** The best known, and one of the simplest, polyalphabetic ciphers is the Vigenère cipher. In this scheme, the set of related monoalphabetic substitution rules consists of the 26 Caesar ciphers with shifts of 0 through 25. Each cipher is denoted by a key letter, which is the ciphertext letter that substitutes for the plaintext letter a. Thus, a Caesar cipher with a shift of 3 is denoted by the key value d.

We can express the Vigenère cipher in the following manner. Assume a sequence of plaintext letters $P = p_0, p_1, p_2, \ldots, p_{n-1}$ and a key consisting of the sequence of letters $K = k_0, k_1, k_2, \ldots, k_{m-1}$, where typically $m < n$. The sequence of ciphertext letters $C = C_0, C_1, C_2, \ldots, C_{n-1}$ is calculated as follows:

$$\begin{aligned} C = C_0, C_1, C_2, \ldots, C_{n-1} &= \mathrm{E}(K, P) = \mathrm{E}[(k_0, k_1, k_2, \ldots, k_{m-1}), (p_0, p_1, p_2, \ldots, p_{n-1})] \\ &= (p_0 + k_0) \bmod 26, (p_1 + k_1) \bmod 26, \ldots, (p_{m-1} + k_{m-1}) \bmod 26, \\ &\quad (p_m + k_0) \bmod 26, (p_{m+1} + k_1) \bmod 26, \ldots, (p_{2m-1} + k_{m-1}) \bmod 26, \ldots \end{aligned}$$

A general equation of the encryption process is

$$C_i = (p_i + k_{i \bmod m}) \bmod 26$$

Similarly, decryption is a generalization of Equation

$$p_i = (C_i - k_{i \bmod m}) \bmod 26$$

Example:

```
key:          deceptivedeceptivedeceptive
plaintext:    wearediscoveredsaveyourself
ciphertext:   ZICVTWQNGRZGVTWAVZHCQYGLMGJ
```

Expressed numerically, we have the following result.

| key | 3 | 4 | 2 | 4 | 15 | 19 | 8 | 21 | 4 | 3 | 4 | 2 | 4 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| plaintext | 22 | 4 | 0 | 17 | 4 | 3 | 8 | 18 | 2 | 14 | 21 | 4 | 17 | 4 |
| ciphertext | 25 | 8 | 2 | 21 | 19 | 22 | 16 | 13 | 6 | 17 | 25 | 6 | 21 | 19 |

| key | 19 | 8 | 21 | 4 | 3 | 4 | 2 | 4 | 15 | 19 | 8 | 21 | 4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| plaintext | 3 | 18 | 0 | 21 | 4 | 24 | 14 | 20 | 17 | 18 | 4 | 11 | 5 |
| ciphertext | 22 | 0 | 21 | 25 | 7 | 2 | 16 | 24 | 6 | 11 | 12 | 6 | 9 |

Consequently, in both cases, r is encrypted using key letter e , e is encrypted using key letter p , and d is encrypted using key letter t . Thus, in both cases, the ciphertext sequence is VTW.

We indicate this above by underlining the relevant ciphertext letters and shading the relevant ciphertext numbers.

The periodic nature of the keyword can be eliminated by using a nonrepeating keyword that is as long as the message itself. Vigenère proposed what is referred to as an autokey system , in which a keyword is concatenated with the plaintext itself to provide a running key. For our example,

```
key:          deceptivewearediscoveredsav
plaintext:    wearediscoveredsaveyourself
ciphertext:   ZICVTWQNGKZEIIGASXSTSLVVWLA
```

Even this scheme is vulnerable to cryptanalysis. Because the key and the plain-text share the same frequency distribution of letters, a statistical technique can be applied.

Another method of using Vignere cipher is using Vignere table as shown

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

**Example:**

Plaintext: C R Y P T O G R A P H Y
Key: L U C K L U C K L U C K
Ciphertext: N L A Z E I I B L J J I

**VERNAM CIPHER:** The ultimate defense against such a cryptanalysis is to choose a keyword that is as long as the plaintext and has no statistical relationship to it. Such a system was introduced by an AT&T engineer

named Gilbert Vernam in 1918. His system works on binary data (bits) rather than letters. The system can be expressed succinctly as follows (Figure 2.7):

$$c_i = p_i \oplus k_i$$

where

$p_i$ = $i$th binary digit of plaintext
$k_i$ = $i$th binary digit of key
$c_i$ = $i$th binary digit of ciphertext
$\oplus$ = exclusive-or (XOR) operation

Compare this with Equation (2.3) for the Vigenère cipher.

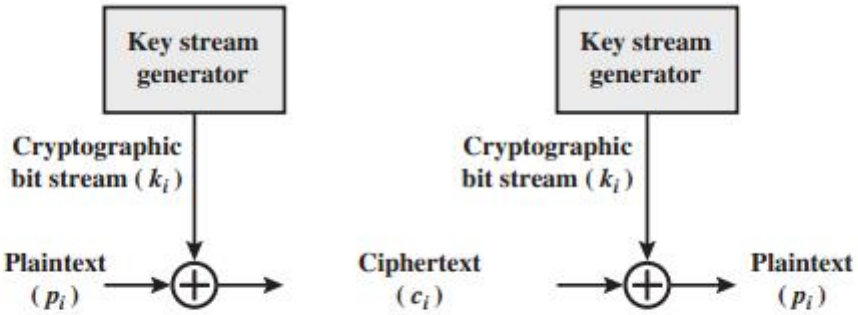



Figure 2.7 Vernam Cipher

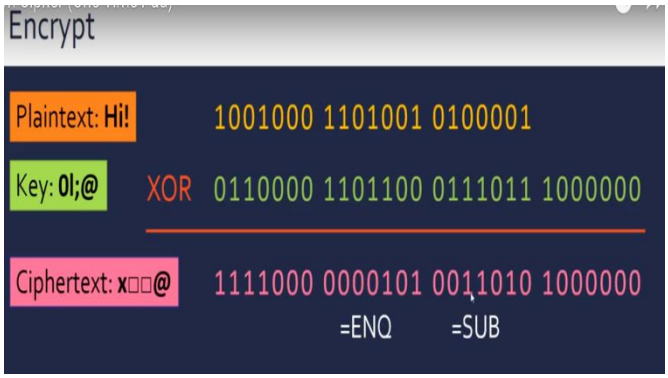
Thus, the ciphertext is generated by performing the bitwise XOR of the plain-text and the key. Because of the properties of the XOR, decryption simply involves the same bitwise operation:

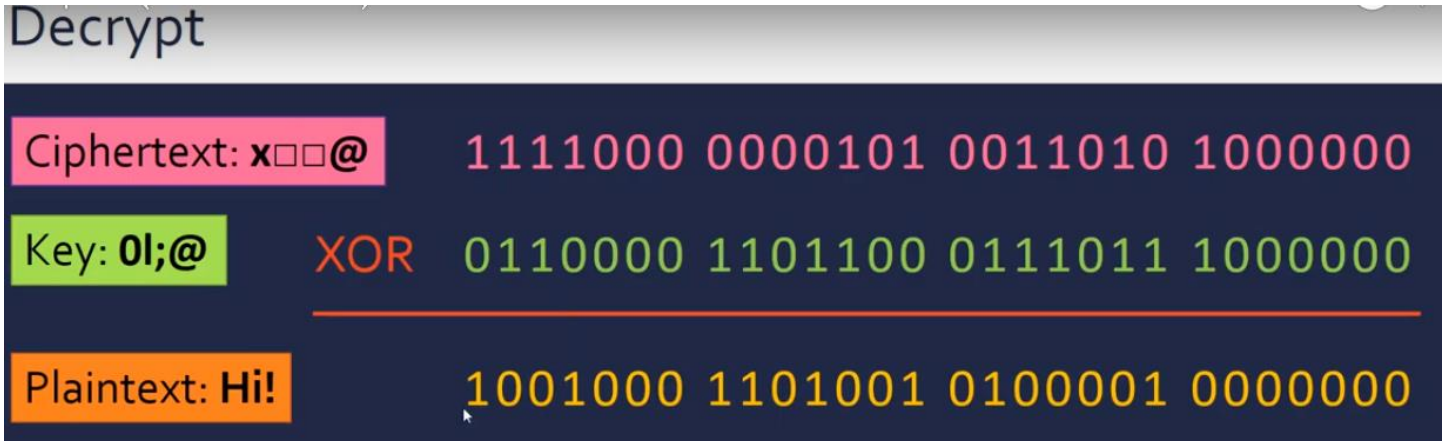
$$p_i = c_i \oplus k_i$$

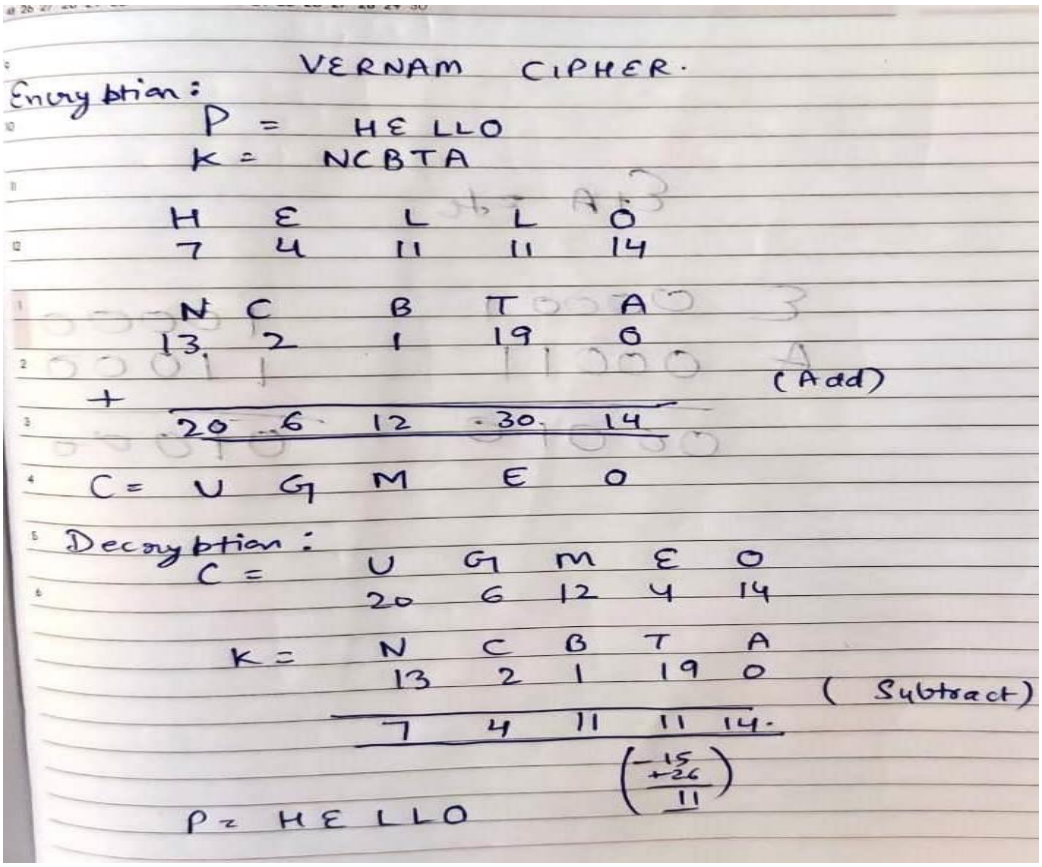
# ALPHABET IN BINARY, CAPITAL LETTERS

| | |
|---|---|
| A | 01000001 |
| B | 01000010 |
| C | 01000011 |
| D | 01000100 |
| E | 01000101 |
| F | 01000110 |
| G | 01000111 |
| H | 01001000 |

| | |
|---|---|
| I | 01001001 |
| J | 01001010 |
| K | 01001011 |
| L | 01001100 |
| M | 01001101 |
| N | 01001110 |
| O | 01001111 |
| P | 01010000 |
| Q | 01010001 |
| R | 01010010 |
| S | 01010011 |
| T | 01010100 |
| U | 01010101 |
| V | 01010110 |
| W | 01010111 |
| X | 01011000 |
| Y | 01011001 |
| Z | 01011010 |






**One-Time Pad**

An Army Signal Corp officer, Joseph Mauborgne, proposed an improvement to the Vernam cipher that yields the ultimate in security. Mauborgne suggested using a random key that is as long as the message, so that the key need not be repeated. In addition, the key is to be used to encrypt and decrypt a single message, and then is discarded. Each new message requires a new key of the same length as the new message. Such a scheme, known as a one-time pad , is unbreakable. It produces random output that bears no statistical relationship to the plaintext. Because the ciphertext contains no information whatsoever about the plaintext, there is simply no way to break the code.

An example should illustrate our point. Suppose that we are using a Vigenère scheme with 27 characters in which the twenty-seventh character is the space character, but with a one-time key that is as long as the message. Consider the ciphertext.

ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS

We now show two different decryptions using two different keys:

```
ciphertext: ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS
key:        pxlmvmsydofuyrvzwc tnlebnecvgdupahfzzlmnyih
plaintext:  mr mustard with the candlestick in the hall
```

The one-time pad offers complete security but, in practice, has two fundamental difficulties:

1. There is the practical problem of making large quantities of random keys. Any heavily used system might require millions of random characters on a regular basis. Supplying truly random characters in this volume is a significant task.

2. Even more daunting is the problem of key distribution and protection. For every message to be sent, a key of equal length is needed by both sender and receiver. Thus, a mammoth key distribution problem exists.

Because of these difficulties, the one-time pad is of limited utility and is useful primarily for low-bandwidth channels requiring very high security.

The one-time pad is the only cryptosystem that exhibits what is referred to as perfect secrecy.

**TRANSPOSITION TECHNIQUES**

All the techniques examined so far involve the substitution of a ciphertext symbol for a plaintext symbol. A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters. This technique is referred to as a transposition cipher.

The simplest such cipher is the rail fence technique, in which the plaintext is written down as a sequence of diagonals and then read off as a sequence of rows. For example, to encipher the message “meet me after the toga party” with a rail fence of depth 2, we write the following:

```
m e m a t r h t g p r y
  e t e f e t e o a a t
```

The encrypted message is **MEMATRHTGPRYETEFETEOAAT.**

A more complex scheme is to write the message in a rectangle, row by row, and read the message off, column by column, but permute the order of the columns. The order of the columns then becomes the key to the algorithm. For example,

```
Key:          4 3 1 2 5 6 7
Plaintext:    a t t a c k p
              o s t p o n e
              d u n t i l t
              w o a m x y z
Ciphertext:   TTNAAPTMTSUOAODWCOIXKNLYPETZ
```

Thus, in this example, the key is 4312567. To encrypt, start with the column that is labeled 1, in this case column 3. Write down all the letters in that column. Proceed to column 4, which is labeled 2, then column 2, then column 1, then columns 5, 6, and 7.

**ROTOR MACHINES**

Multiple stages of encryption can produce an algorithm that is significantly more difficult to cryptanalyze. This is as true of substitution ciphers as it is of transposition ciphers. Before the introduction of DES, the most important application of the principle of multiple stages of encryption was a class of systems known as rotor machines.

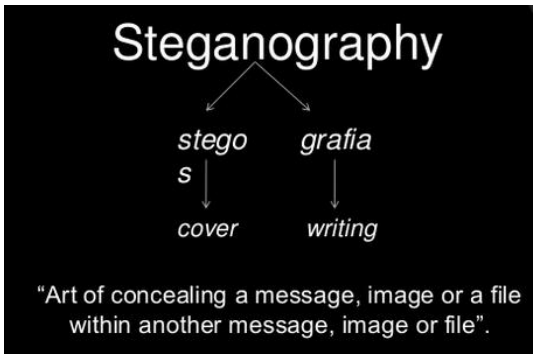
The basic principle of the rotor machine is illustrated in Figure 2.8. The machine consists of a set of independently rotating cylinders through which electrical pulses can flow. Each cylinder has 26 input pins and 26 output pins, with internal wiring that connects each input pin to a unique output pin. For simplicity, only three of the internal connections in each cylinder are shown.

Example: Enigma II

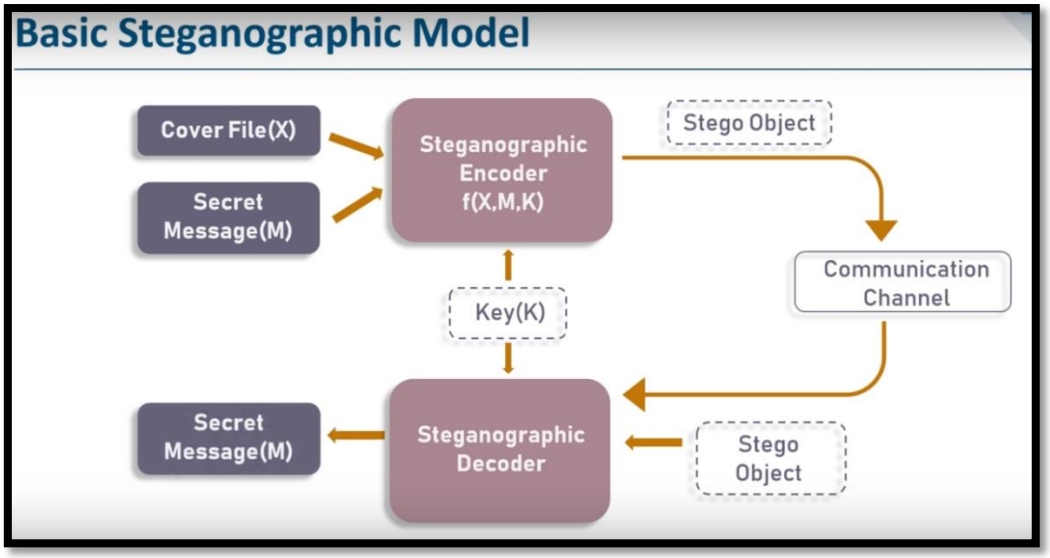

Enigma II
Encryption
Machine

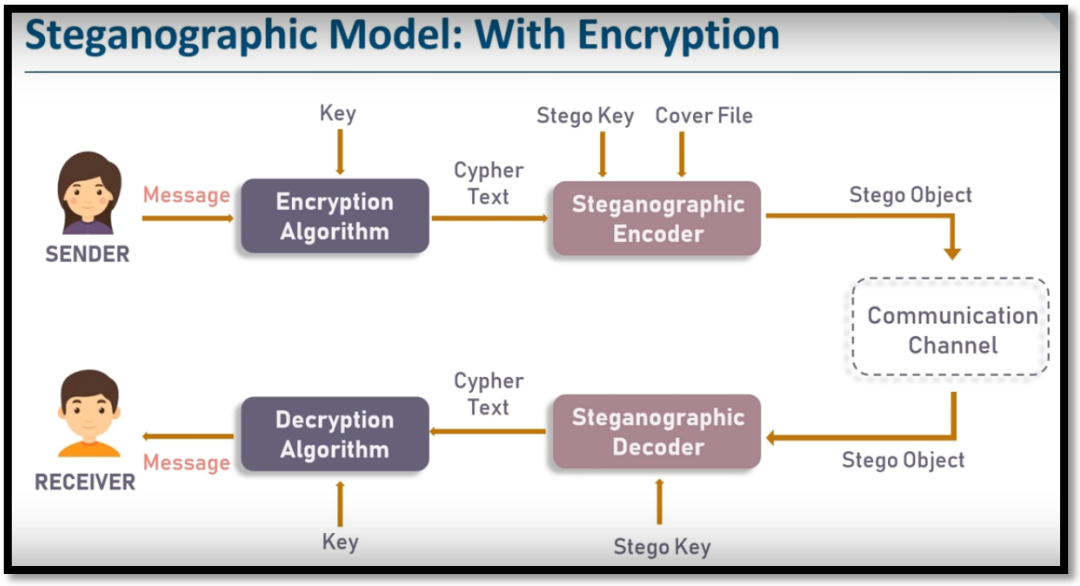

7-6-5 DOG
ATTACK AT DAWN
Enigma II
Encryption
Machine

**STEGANOGRAPHY**

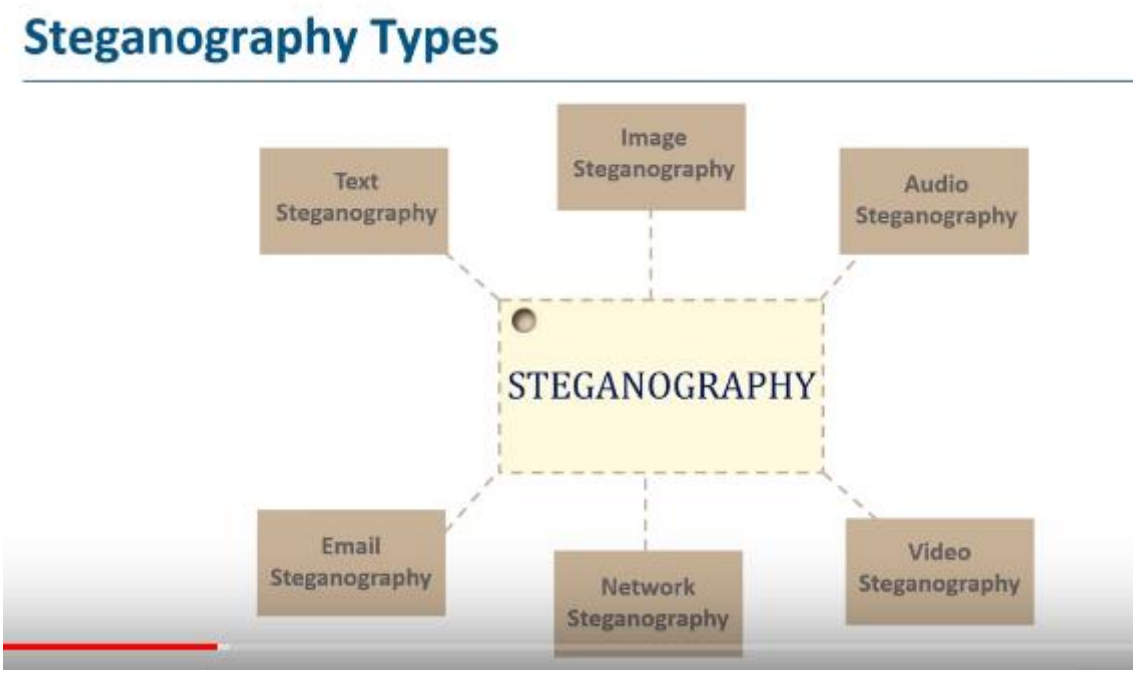



A plaintext message may be hidden in one of two ways. The methods of steganography conceal the existence of the message, whereas the methods of cryp-tography render the message unintelligible to outsiders by various transformations of the text.

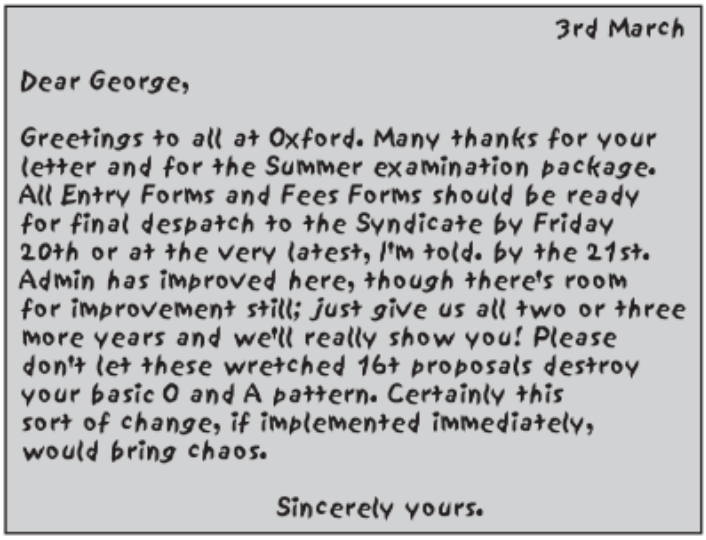





A simple form of steganography, but one that is time-consuming to construct, is one in which an arrangement of words or letters within an apparently innocuous text spells out the real message. For example, the sequence of first letters of each word of the overall message spells out the hidden message. Figure 2.9 shows an example in which a subset of the words of the overall message is used to convey the hidden message. See if you can decipher this; it's not too hard.




**Text Based Steganography:**



3rd March

Dear George,

Greetings to all at Oxford. Many thanks for your letter and for the Summer examination package. All Entry Forms and Fees Forms should be ready for final despatch to the Syndicate by Friday 20th or at the very latest, I'm told, by the 21st. Admin has improved here, though there's room for improvement still; just give us all two or three more years and we'll really show you! Please don't let these wretched 16+ proposals destroy your basic O and A pattern. Certainly this sort of change, if implemented immediately, would bring chaos.

Sincerely yours.

Figure 2.9 A Puzzle for Inspector Morse
*(From* The Silent World of Nicholas Quinn, *by Colin Dexter)*

Some examples are the following

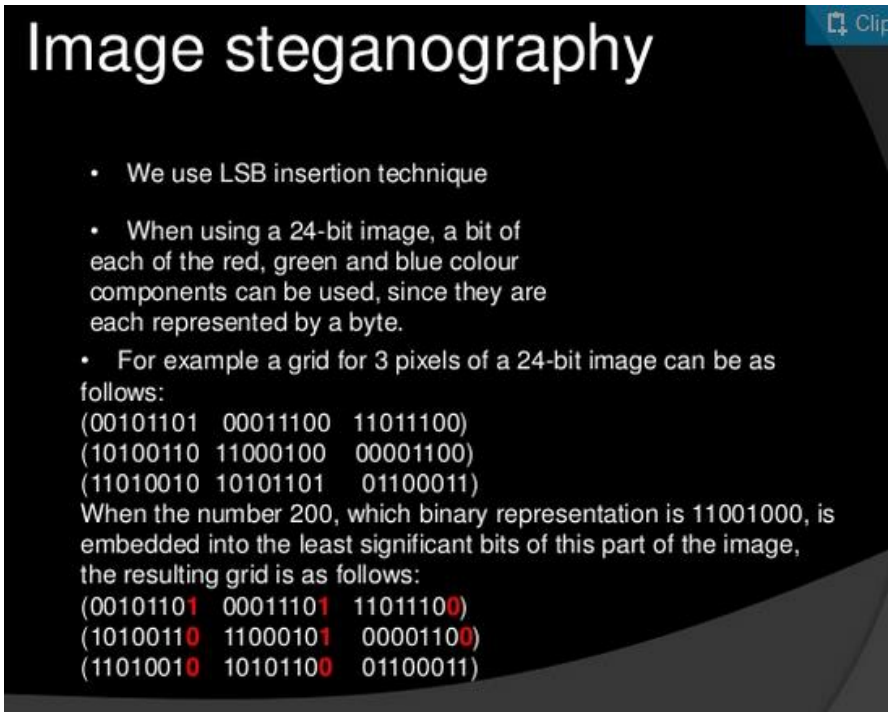
**Character marking:** Selected letters of printed or typewritten text are over- written in pencil. The marks are ordinarily not visible unless the paper is held at an angle to bright light.


**Invisible ink:** A number of substances can be used for writing but leave no visible trace until heat or some chemical is applied to the paper.

**Pin punctures:** Small pin punctures on selected letters are ordinarily not visible unless the paper is held up in front of a light.

**Typewriter correction ribbon:** Used between lines typed with a black ribbon, the results of typing with the correction tape are visible only under a strong light.

**Image Steganography**




Steganography has a number of drawbacks when compared to encryption. It requires a lot of overhead to hide a relatively few bits of information, although using a scheme like that proposed in the preceding paragraph may make it more effective.