Dr. Ammar Haider
Assistant Professor
School of Computing

CS3002 Information Security

Classical Cryptography

Source: Stallings CNS, chap 3

Cryptography: What

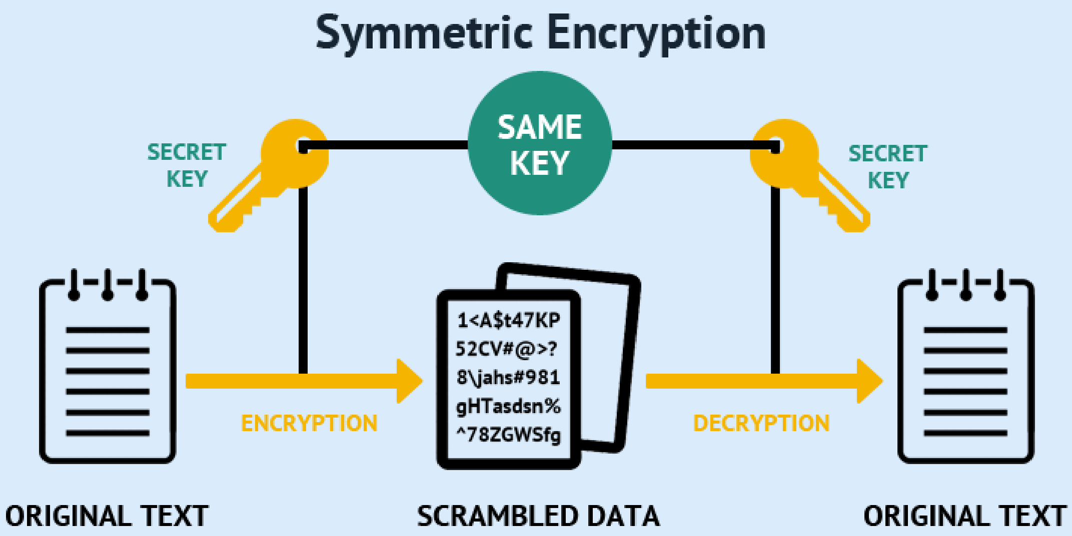
- Plaintext
  - Readable message or data that needs to be protected
- Encryption Algorithm (or Cipher)
  - Algorithm to perform various substitutions and transformations on the plaintext
- Secret key
  - Used as input to the algorithm, transformations depend on the key
- Ciphertext
  - Scrambled message produced as output

Cryptography: Why

- To achieve important properties of information security:
  - Confidentiality
  - Integrity
  - Authenticity
  - Authorization
  - Non repudiation

Symmetric Crypto

- AKA conventional cryptography
- Sender and receiver must both know the secret key
- Uses techniques like confusion and diffusion to encrypt/decrypt data



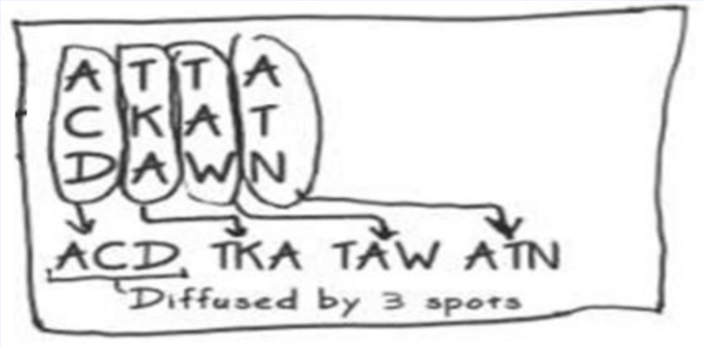
Symmetric Crypto: How?

- There are two primary operations. Note that these should be perfectly reversible!

(1) Substitution

Replace character(s) or bit-strings with other characters

(2) Permutation



Change the order of characters or bit strings.

Symmetric Crypto Limitations

Pros

- Very fast to compute
- Hardware acceleration available in many cases

Cons

- No mechanism of sharing the key secretly
- Managing separate keys for each **pair** of users, otherwise impersonation is possible
  - If Alice and Bob share a key. Imagine Trudy shares the same key with Alice for secure communication. Trudy may act as Alice and talk to Bob.

Classical Ciphers

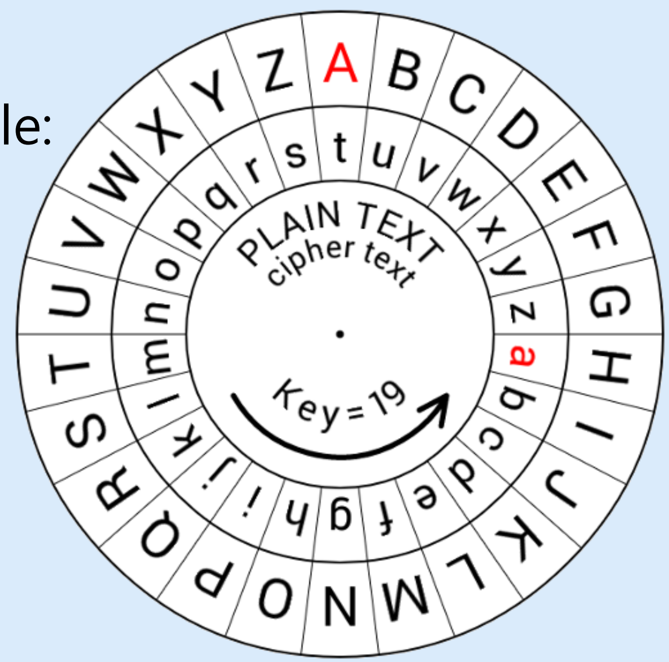
- We will start with studying classical ciphers – those used before the era of computers.
- Computers could break these ciphers easily, that’s why they became obsolete.

Substitution ciphers

Classical Substitution Ciphers

- Replace plaintext characters according to a translation table.
- Substitution rules become the secret key

Most well known example:
Caesar Cipher



Rotate the inner alphabet disk to change the key

When key=13, cipher is called **ROT-13**

Attacking the Ceaser cipher

- Ceaser cipher can be quite easily broken by a brute-force attack: The attacker tries every possible key on a piece of ciphertext until an intelligible translation into plaintext is obtained.
- On average, half of all possible keys must be tried to achieve success.
  - if there are X different keys, on average an attacker would discover the actual key after X/2 tries.

Attacking the Ceaser cipher

ciphertext

PHHW PH DIWHU WKH WRJD SDUWB

KEY

1 oggv og chvgt vjg vqic rctva
2 nffu nf bgufs uif uphb qbsuz
3 meet me after the toga party
4 ldds ld zesdq sgd snfz ozqsx
5 kccr kc ydrcp rfc rmey nyprw
6 jbbq jb xcqbo qeb qldx mxoqv
7 iaap ia wbpan pda pkcw lwnpu
8 hzzo hz vaozm ocz ojbv kvmot
9 gyyn gy uznyl nby niau julns
10 fxxm fx tymxk max mhzt itkmr
11 ewwl ew sxlwj lzw lgys hsjlq
12 dvvk dv rwkvi kyv kfxr grikp
13 cuuj cu qvjuh jxu jewq fqhjo
14 btti bt puitg iwt idvp epgin
15 assh as othsf hvs hcuo dofhm
16 zrrg zr nsgre gur gbtn cnegl
17 yqqf yq mrfqd ftq fasm bmdfk
18 xppe xp lqepc esp ezrl alcej
19 wood wo kpdob dro dyqk zkbdi
20 vnnc vn jocna cqn cxpj yjach
21 ummb um inbmz bpm bwoi xizbg
22 tlla tl hmaly aol avnh whyaf
23 skkz sk glzkx znk zumg vgxze
24 rjjy rj fkyjw ymj ytlf ufwyd
25 qiix qi ejxiv xli xske tevxc

Brute forcing in general

- Preceding example shows that there is more to brute-force than simply running through all possible keys.
- The analyst must be able to recognize plaintext as plaintext.
  - If the message is just plain text in English (or another language), then the result is easily understood by a human reader, but the task of recognizing the language would have to be automated.
  - If the text message has been compressed before encryption, then recognition is more difficult.
  - And if the message is some more general type of data, such as a word document or image or audio, the problem becomes even more difficult to automate.

Monoalphabetic substitution

- Caesar cipher is actually a special case of more general **monoalphabetic substitution** cipher

CIPHER ALPHABET

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

Figure 1

Classical Substitution Ciphers

Question

What is the size of 'key space' in the monoalphabetic substitution cipher assuming 26 letters?

a. 26
b. 26!
c. 2^{26}
d. 26^2

Key space = set of all possible keys

Cryptanalysis

- For such large key spaces brute forcing is no longer practical
  - Note: it is *unfeasible*, not impossible
- **Cryptanalysis** is the process of trying to work out the <u>encryption key</u> from a given ciphertext.
  - This type of attack exploits the characteristics of the algorithm (cipher) to attempt to deduce a specific plaintext or key
  - Attacker will analyze the ciphertext using different statistical tests – they must have prior knowledge of the nature of plaintext (plain English, photo, exe file etc.)

Cryptanalysis Techniques

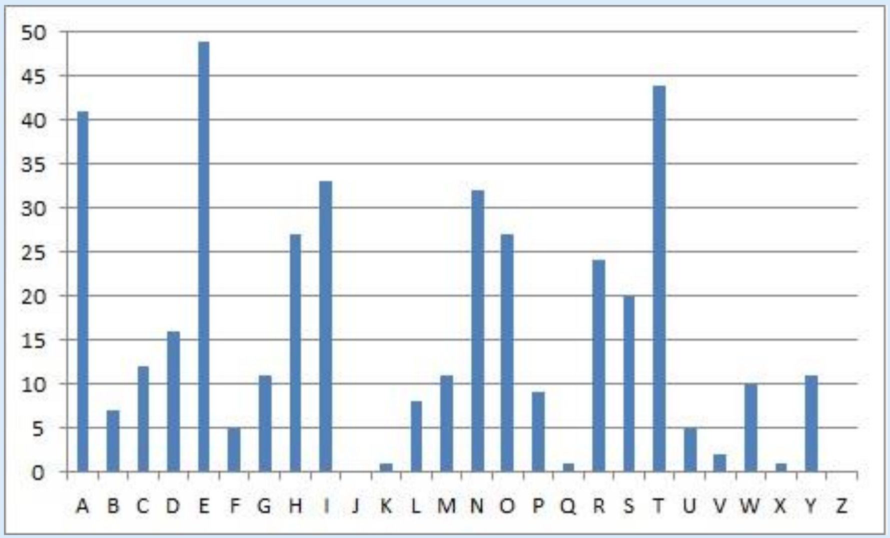
Frequency Analysis

- Compute frequencies of letters in ciphertext and compare with the typical frequencies in the target language.
  - e.g. in English texts, most frequent letters are E (13%), T (9.1%), A (8.2%), O (7.5%)

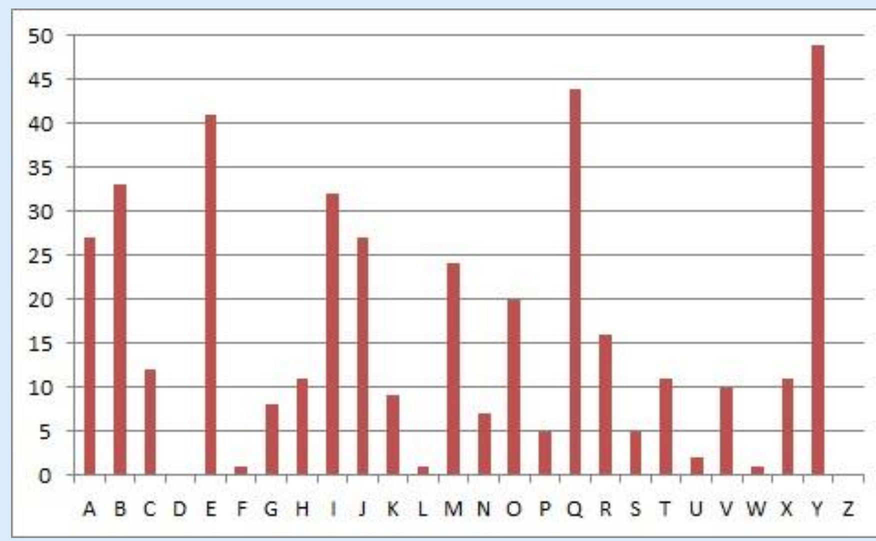
- Can also find frequency of groups of letters (di-grams and tri-grams): *an, in, the*

Cryptanalysis Techniques

Frequency Analysis Example 1



Plaintext (English)



Ciphertext

Cryptanalysis Techniques

Frequency Analysis Example 2

UKBYBIPOUZBCUFEEBORUKBYBHOBBRFESPVKBWFOFERVNBCVBZPRUBOFERVNBCVBPCYY
FVUFOFEIKNWFRFIKJNUPWRFIPOUNVNIPUBRNCUKBEFWWFDNCHXCYBOHOPYXPUBNCU
BOYNRVNIWNCPOJIOFHOPZRVFZIXUBORJRUBZRBCHNCBBONCHRJZSFWNVRJRUBZRPCYZ
PUKBZPUNVPWPCYVFZIXUPUNFCPWRVNBCVBRPYYNUNFCPWWJUKBYBIPOUZBCUIPOUN
VNIPUBRNCHOPYXPUBNCUBOYNRVNIWNCPOJIOFHOPZRNCRVNBCUNENVVFZIXUNCHPCY
VFZIXUPUNFCPWZPUKBZPUNVR

B	**36**
N	**34**
U	**33**
P	**32**
C	**26**

➔ **E**

➔ **T**

➔ **A**

NC	**11**
PU	**10**
UB	**10**
UN	**9**

Di-grams

➔ **IN**

➔ **AT**

UKB	**6**
RVN	**6**
FZI	**4**

Tri-grams

➔ **THE**

Hardening the Substitution Cipher

- Monoalphabetic ciphers are easy to break because they reflect the frequency data of the original alphabet.
- One possible countermeasure is to provide multiple substitutes, known as **homophones**, for a single letter.
  - e.g. letter e could be randomly assigned a number of different cipher symbols, such as 16, 74, 35, and 21
- If the number of symbols assigned to each letter is proportional to the relative frequency of that letter, then single-letter frequency information is completely obliterated.
- What about digram & trigram frequencies?

Hardening the Substitution Cipher

- To create a stronger cipher, we look into **polyalphabetic** substitution
- Most well-known example: **Vigenère Cipher**
- Uses Caesar's cipher with various different shifts, in order to hide the distribution of the letters.
- A ‘keyword’ defines the shifts used for each letter in the plaintext

Vigenère Cipher

- A keyword defines the shifts used in each letter in the text
- The key word is repeated many times to match the length of plaintext

Plaintext = “I attack you”

Keyword = “BOX”

(which translates to Ceaser shifts of “1, 14, 23”)

plaintext	I	a	t	t	a	c	k	y	o	u
key	1	14	23	1	14	23	1	14	23	1
ciphertext	J	o	q	u	o	z	l	m	l	v

Vigenère Cipher

<u>Vigenère square</u> is helpful when working by hand.

Header row is for plaintext.

First column represents the key letter.

	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

Cryptanalysis of Vigenère cipher

- Strength of Vigenère cipher is that there are multiple ciphertext letters for each plaintext letter, one for each unique letter of the keyword.
- e.g. In the preceding example, plaintext ‘t’ can be mapped to ‘u’, ‘h’ or ‘o’ (with the shifts of 1, 14 and 23 respectively).
- Thus, the letter frequency information is obscured.
- However, not all knowledge of the plaintext structure is lost.
- If attacker finds out the key length, say n, they can split the ciphertext in to n sections, and apply frequency analysis on each section separately.

Cryptanalysis of Vigenère cipher

- e.g. if key length = 3 is known, attacker can split the ciphertext in to 3 sections, and apply frequency analysis on each section separately.

ciphertext	J	o	q	u	o	z	l	m	l	v
Section 1	J			u			l			v
Section 2		o			o			m		
Section 3			q			z			l	

Cryptanalysis of Vigenère cipher

Guessing the key length

- Find repeated strings in the ciphertext. Their distance is expected to be a multiple of the key length. Compute the gcd of (most) distances.
- Example:

```
P. Text: TOBENOTORTOBE
Keyword: 1231231231231
C. Text: UQEFPRUQUUQEF
```

Diagraph	First Position	Second Position	Distance	Factors

Cryptanalysis of Vigenère cipher

Guessing the key length

- Find repeated strings in the ciphertext. Their distance is expected to be a multiple of the key length. Compute the gcd of (most) distances.
- Example:

```
P. Text: TOBENOTORTOBE
Keyword: 1231231231231
C. Text: UQEFPRUQUUQEF
```

Diagraph	First Position	Second Position	Distance	Factors
UQ	1	7	6	3, 2
UQ	7	10	3	3

Cryptanalysis of Vigenère cipher

Guessing the key length

- Find repeated strings in the ciphertext. Their distance is expected to be a multiple of the key length. Compute the gcd of (most) distances.
- Example:

```
P. Text: TOBENOTORTOBE
Keyword: 1231231231231
C. Text: UQEFPRUQUUQEF
```

Diagraph	First Position	Second Position	Distance	Factors
UQ	1	7	6	3, 2
UQ	7	10	3	3
EF	3	12	9	3, 3

Cryptanalysis of Vigenère cipher

Guessing the key length

- Find repeated strings in the ciphertext. Their distance is expected to be a multiple of the key length. Compute the gcd of (most) distances.
- Example:

```
P. Text: TOBENOTORTOBE
Keyword: 1231231231231
C. Text: UQEFPRUQUUQEF
```

Diagraph	First Position	Second Position	Distance	Factors
UQ	1	7	6	3, 2
UQ	7	10	3	3
EF	3	12	9	3, 3
QE	2	11	9	3, 3

More information and worked examples
https://crypto.interactive-maths.com/kasiski-analysis-breaking-the-code.html

Hardening Vigenère Cipher

- As seen earlier, Vigenère cipher can be attacked by breaking ciphertext into n sections, and frequency analyzing each of those sections separately.
- But this approach won't work if the keyword length is equal or more than that of plaintext.
  - So the key need not be repeated
  - This the idea applied in One-Time Pad cipher

One-Time Pad (OTP)

- To implement OTP, for each message use a **random key** that is at least as long as the message itself.
- Combine the key with the plaintext (e.g. similar to Vigenère cipher) to generate ciphertext.
- Each 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 message.
- Its name comes from the paper-pads on which random key streams were printed.
  - Two copies of each pad had to be created, one for sender other for receiver.
  - Each paper sheet was destroyed after use

One-Time Pad (OTP)

https://www.cryptomuseum.com/crypto/otp/index.htm

OTP Strength

When **applied correctly**, the OTP provides a truly unbreakable cipher.

How?

- It produces random output that bears no statistical relationship to the plaintext.
- Ciphertext contains no information whatsoever about the plaintext

OTP Practical or not?

OTP implementation rules are very strict. If any one of these rules is violated, we no longer have the guarantee of unbreakability.

1. The OTP should consist of truly random characters (noise).
2. The OTP (i.e. the key) should have the same length as the plaintext (or longer).
3. Only two copies of the OTP should exist.
4. The OTP should be used only once.
5. Both copies of the OTP are destroyed immediately after use.

In practice it is extremely challenging to

- generate large amounts of truly random keys (millions of characters); and…
- securely distribute those random keys

OTP Practical or not?

To illustrate the impracticality of the OTP, consider a simple example. Suppose Alice wants to send a 100 MB file to Bob using the OTP. She must first generate a 100 MB random key, which she then uses to encrypt the 100 MB file to produce the ciphertext. This 100 MB key must be securely transmitted to Bob before he can decrypt the ciphertext. If Alice and Bob wish to communicate regularly, they would need a new 100 MB key for each message, resulting in an enormous amount of key data that must be securely generated, stored, and transmitted.

https://eitca.org/cybersecurity/eitc-is-ccf-classical-cryptography-fundamentals/stream-ciphers/stream-ciphers-and-linear-feedback-shift-registers/examination-review-stream-ciphers-and-linear-feedback-shift-registers/what-are-the-limitations-of-the-one-time-pad-and-why-is-it-considered-impractical-for-most-real-world-applications/

Permutation (transposition) ciphers

changing order of characters in plaintext

Railfence cipher

- The plaintext is written down as a sequence of diagonals and then read off as a sequence of rows.
- e.g. to encipher the message “meet me after the party” with a rail fence of depth 3, we write the following

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

- The encrypted message is

mmthretetefeteatearpy

Railfence cipher

- The rail fence cipher is not very strong; the number of practical keys is small enough that a cryptanalyst can try them all by hand.

Columnar transposition ciphers

- In this cipher, message is written in a grid row by row, and read off column by column.
- In addition, order of columns is also permuted. This order becomes the key of the cipher

Message:
THIS IS TOP SECRET

Key:
3142

or Key="GAME" which translates to 3142, alphabetic ordering of these four letter is: A, E, G, M

Ciphertext:
HSSE SOC TIPR ITET
key defines the order in which we read columns

3	1	4	2
T	H	I	S
I	S	T	O
P	S	E	C
R	E	T	

Security of Permutation Ciphers

Question

- Are permutation ciphers susceptible to frequency analysis?

Security of Permutation Ciphers

- A transposition cipher is easily recognized because it has the same letter frequencies as the original plaintext.
- For the columnar transposition just shown, cryptanalysis is fairly straightforward and involves laying out the ciphertext in a matrix and playing around with column positions.
- The cipher can be made significantly more secure by performing more than one stage of transposition (i.e. by re-encrypting the ciphertext)