

# MAS 433: Cryptography

## Lecture 2

### Classical Ciphers

Wu Hongjun

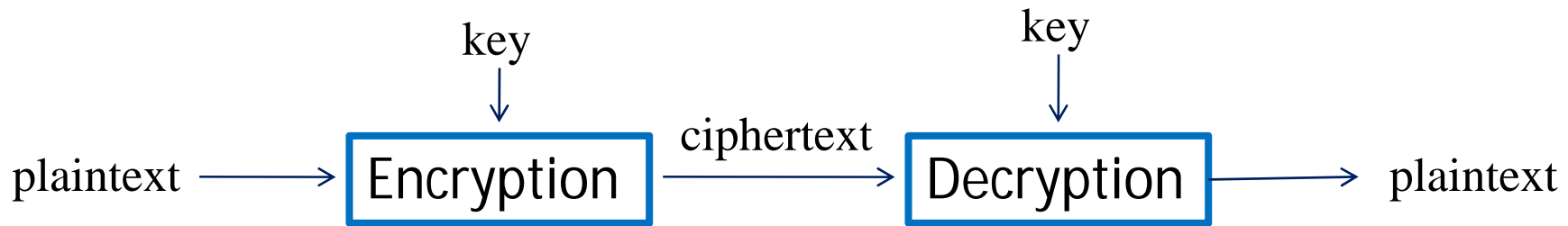
# Lecture Outline

- **Classical ciphers**
  - **Shift cipher (Caesar cipher)**
  - **Substitution cipher, frequency cryptanalysis**
  - **Vigenere cipher**
  - **Playfair cipher**
  - **Transposition (permutation) cipher**
- Symmetric key encryption
- Hash function and Message Authentication Code
- Public key encryption
- Digital signature
- Key establishment and management
- Introduction to other cryptographic topics

# Recommended reading for classical ciphers

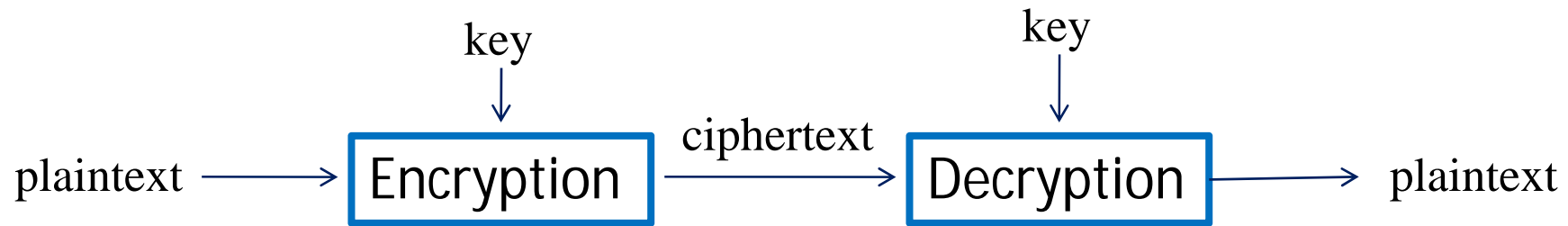
- Cryptography Theory and Practice
  - Section 1.1.1, 1.1.2, 1.1.4 and 1.1.6
  - Section 1.2.2, 1.2.3
- Wikipedia
  - History of cryptography  
[http://en.wikipedia.org/wiki/History\\_of\\_cryptography](http://en.wikipedia.org/wiki/History_of_cryptography)
  - Classical ciphers  
[http://en.wikipedia.org/wiki/Classical\\_cipher](http://en.wikipedia.org/wiki/Classical_cipher)

# Encryption/Decryption



- Encryption
  - the process of transforming information to make it unreadable, except those possessing special knowledge (key)
- Decryption
  - the inverse of encryption
- Cipher
  - the algorithm or device used for encryption/decryption
- Key: the secret information being used in a cipher
- Plaintext: the message to be encrypted
- Ciphertext: the encrypted message

# Symmetric Key Cipher




- Symmetric key cipher
  - the key used for encryption is the same as that used for decryption
- Classical ciphers
  - developed before computer era
  - all the classical ciphers are symmetric key ciphers

# Shift cipher (Caesar cipher)

- Key
  - an integer;  $1 \leq K \leq 25$  (for English with 26 letters)
- Encryption
  - Each letter in the plaintext  $P$  is replaced with the  $K$ 'th letter following that letter (alphabetical order)
- Decryption
  - Each letter in the ciphertext  $C$  is replaced with the  $K$ 'th letter before that letter

# Shift cipher (Caesar cipher)

Plaintext = cryptographyisfun  
 $K = 2$   
Ciphertext =



---

Formally, let

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

then encryption:  $c_i = (p_i + K) \bmod 26$

decryption:  $p_i = (c_i - K) \bmod 26$

For example, to encrypt 'y':

$$'y' = 24 \Rightarrow 24 + 2 \bmod 26 = 0 \Rightarrow 0 = 'A'$$

# Shift cipher (Caesar cipher)

- Caesar cipher
  - Shift cipher with  $K = 3$
  - Used by Rome troops
- How about the security of shift cipher?
  - It is difficult for a person who has never heard about shift cipher to break it
  - But for a person who knows how this cipher works, shift cipher is too weak
    - Only 25 possible keys
      - $\Rightarrow$  try every possible key to break it (brute force)



# Substitution Cipher

- An invertible secret substitution table  $S$  (the key)
  - Encryption:  $c_i = S(p_i)$
  - Decryption:  $p_i = S^{-1}(c_i)$
- Example
  - Let the secret table  $S$  be given as

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	A	D	C	Z	H	W	Y	G	O	Q	X	S	V	T	R	N	M	S	K	J	I	P	F	E	U
  - Then
    - Plaintext:        b e c a u s e
    - Ciphertext:      A Z

# Substitution Cipher

- Security of substitution cipher
  - Brute force search for the key is infeasible
    - key space size (the number of possible substitution tables) is huge:  $26! \approx 4 \times 10^{26} \approx 2^{88.4}$
    - If we try one billion keys per second, it takes about 13 billion years to try all the keys
  - Thought to be unbreakable
    - until the invention of frequency analysis

# Frequency Analysis

- Invented by Arabian scientist al-Kindi in the 9th century
- Main idea
  - the encryption of substitution cipher does not randomize the frequency of occurrence of letters properly
  - calculating the frequency of occurrence of letters in ciphertext; comparing those frequency with the frequency of occurrence of letters in the language to determine the substitution table

(whenever there is non-randomness in an encryption system, there is a potential attack!)

# Frequency Analysis

- Probabilities of occurrences of the 26 letters (English)

letter	probability	letter	probability
<i>A</i>	.082	<i>N</i>	.067
<i>B</i>	.015	<i>O</i>	.075
<i>C</i>	.028	<i>P</i>	.019
<i>D</i>	.043	<i>Q</i>	.001
<i>E</i>	.127	<i>R</i>	.060
<i>F</i>	.022	<i>S</i>	.063
<i>G</i>	.020	<i>T</i>	.091
<i>H</i>	.061	<i>U</i>	.028
<i>I</i>	.070	<i>V</i>	.010
<i>J</i>	.002	<i>W</i>	.023
<i>K</i>	.008	<i>X</i>	.001
<i>L</i>	.040	<i>Y</i>	.020
<i>M</i>	.024	<i>Z</i>	.001

# Frequency Analysis

- Probabilities of occurrences of two consecutive letters, called digrams, are given as follows:

th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

# Frequency Analysis

- The 16 most common trigrams in English are:

the and tha ent ing ion tio for  
nde has nce edt tis oft sth men


# Frequency Analysis

- Example: Given the following ciphertext encrypted with substitution cipher, how to recover plaintext?  
(in this example, we use capital letter to indicate ciphertext, use small letter to indicate plaintext)

YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ  
NDIFEFMZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ  
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ  
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

# Frequency Analysis

- Step 1. Compare the frequency of letters in ciphertext with that of English:



letter	frequency	letter	frequency
<i>A</i>	0	<i>N</i>	9
<i>B</i>	1	<i>O</i>	0
<i>C</i>	15	<i>P</i>	1
<i>D</i>	13	<i>Q</i>	4
<i>E</i>	7	<i>R</i>	10
<i>F</i>	11	<i>S</i>	3
<i>G</i>	1	<i>T</i>	2
<i>H</i>	4	<i>U</i>	5
<i>I</i>	5	<i>V</i>	5
<i>J</i>	11	<i>W</i>	8
<i>K</i>	1	<i>X</i>	6
<i>L</i>	0	<i>Y</i>	10
<i>M</i>	16	<i>Z</i>	20

letter	probability	letter	probability
<i>A</i>	.082	<i>N</i>	.067
<i>B</i>	.015	<i>O</i>	.075
<i>C</i>	.028	<i>P</i>	.019
<i>D</i>	.043	<i>Q</i>	.001
<i>E</i>	.127	<i>R</i>	.060
<i>F</i>	.022	<i>S</i>	.063
<i>G</i>	.020	<i>T</i>	.091
<i>H</i>	.061	<i>U</i>	.028
<i>I</i>	.070	<i>V</i>	.010
<i>J</i>	.002	<i>W</i>	.023
<i>K</i>	.008	<i>X</i>	.001
<i>L</i>	.040	<i>Y</i>	.020
<i>M</i>	.024	<i>Z</i>	.001

'Z' appears most often, very likely  $S('e') = 'Z'$



# Frequency Analysis

- Step 2. digram in ciphertext

assume  $S('e') = 'Z'$

- 1) ZW appears four times  
=> 'W' may be 'r,s,n,d,a'
- 2) WZ does not appear  
=> 'W' is not 'r'
- 3) W appears 8 times (0.047)  
=> 'W' is more likely to be 'd'

=> likely,  $S('d') = 'W'$

th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

letter	probability	letter	probability
A	.082	N	.067
B	.015	O	.075
C	.028	P	.019
D	.043	Q	.001
E	.127	R	.060
F	.022	S	.063
G	.020	T	.091
H	.061	U	.028
I	.070	V	.010
J	.002	W	.023
K	.008	X	.001
L	.040	Y	.020
M	.024	Z	.001

# Frequency Analysis

- Step 3. digram in ciphertext

assume  $S('d') = 'W'$

'RW' appears twice

$\Rightarrow$  'R' may be 'e, n'

'e' is assumed to be 'Z'

$\Rightarrow$  'R' may be 'n'

th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

# Frequency Analysis

- After the first three steps, we obtain:

```
-----end-----e----ned---e-----  
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ  
  
-----e-----e-----n--d---en-----e-----e  
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ  
  
-e---n-----n-----ed---e---e--ne-nd-e-e--  
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ  
  
-ed-----n-----e-----ed-----d---e--n  
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR
```

# Frequency Analysis

- Step 4. digram in ciphertext

assume  $S('e') = 'Z'$

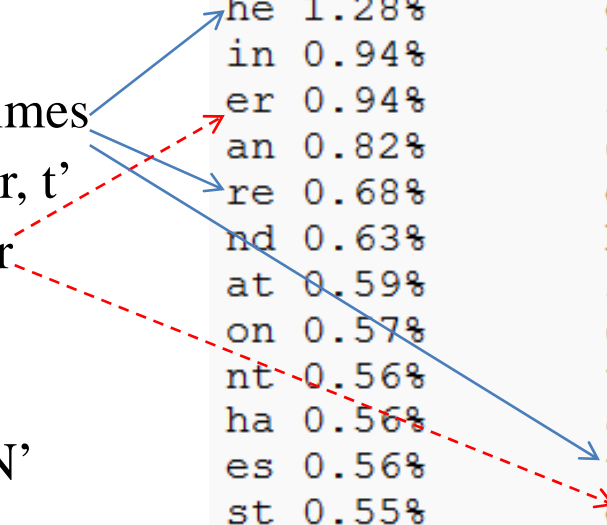
1) NZ appears three times

=> 'N' may be 'h, r, t'

2) ZN does not appear

=> 'N' is not 'r, t'

=> likely,  $S('h') = 'N'$



th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

# Frequency Analysis

- Step 5. Trigram in ciphertext
  - Now there is ne-ndhe in plaintext, ‘-’ denotes ‘C’ in ciphertext
  - From the distribution of trigram in English, ‘and’ appears with relatively high probability
    - Likely ‘C’ is ‘a’

=> Likely  $S('a') = 'C'$

# Frequency Analysis

- Now we obtain:

-----end-----a---e-a--nedh--e-----a-----  
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ

h-----ea---e-a---a---nhad-a-en--a-e-h--e  
NDIFEFMZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

he-a-n-----n-----ed---e---e--neandhe-e--  
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-a---nh---ha---a-e-----ed-----a-d--he--n  
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

# Frequency Analysis

- Step 6. Determine ‘M’
  - ‘M’ is the second most common ciphertext letter
    - ‘M’ may be ‘t,a,o,i,n,s,h,r’
  - There is ‘RNM’ in ciphertext, so it is ‘nh-’ in plaintext, suggests ‘h-’ begins a word, so ‘M’ represents a vowel
    - ‘M’ may be ‘o, i’
  - ‘CM’ appears once
    - ‘C’ is ‘a’
    - Likely ‘M’ is ‘i’ since the distribution of ‘ai’ is more than ‘ao’ in English

letter	probability	letter	probability
<i>A</i>	.082	<i>N</i>	.067
<i>B</i>	.015	<i>O</i>	.075
<i>C</i>	.028	<i>P</i>	.019
<i>D</i>	.043	<i>Q</i>	.001
<i>E</i>	.127	<i>R</i>	.060
<i>F</i>	.022	<i>S</i>	.063
<i>G</i>	.020	<i>T</i>	.091
<i>H</i>	.061	<i>U</i>	.028
<i>I</i>	.070	<i>V</i>	.010
<i>J</i>	.002	<i>W</i>	.023
<i>K</i>	.008	<i>X</i>	.001
<i>L</i>	.040	<i>Y</i>	.020
<i>M</i>	.024	<i>Z</i>	.001

# Frequency Analysis

- Now we obtain

-----iend-----a-i-e-a-inedhi-e-----a---i-  
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBCTXCDDUMJ

h-----i-ea-i-e-a---a-i-nhad-a-en--a-e-hi-e  
NDIFEFMZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

he-a-n-----in-i-----ed---e---e-ineandhe-e--  
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-a--inhi--hai--a-e-i--ed-----a-d--he--n  
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR



# Frequency Analysis

- Determine 'J' by considering digram
  - 'JN' appears twice in ciphertext, and 'N' is 'h'
  - 'th' is the most frequent digram

⇒ Likely 'J' is 't'

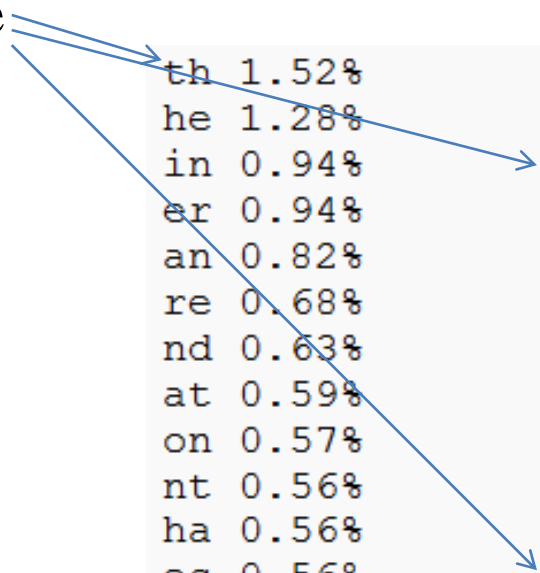
# Frequency Analysis

- Determine 'Y' by considering digram

- 'JY' appears once

- 'Y' is not 'h,e'

- ⇒ Likely 'Y' is 'o'



th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

# Frequency Analysis

- Determine ‘D’
  - Four occurrence of ‘MD’
  - ‘M’ is ‘i’
  - ‘D’ may be ‘n,t,s’
  - Likely ‘D’ is ‘s’

th 1.52%	en 0.55%	ng 0.18%
he 1.28%	ed 0.53%	of 0.16%
in 0.94%	to 0.52%	al 0.09%
er 0.94%	it 0.50%	de 0.09%
an 0.82%	ou 0.50%	se 0.08%
re 0.68%	ea 0.47%	le 0.08%
nd 0.63%	hi 0.46%	sa 0.06%
at 0.59%	is 0.46%	si 0.05%
on 0.57%	or 0.43%	ar 0.04%
nt 0.56%	ti 0.34%	ve 0.04%
ha 0.56%	as 0.33%	ra 0.04%
es 0.56%	te 0.27%	ld 0.02%
st 0.55%	et 0.19%	ur 0.02%

# Frequency Analysis

- Determine ‘F’
  - ‘HNCMF’ in ciphertext
  - ‘chaiF’
  - Likely ‘F’ is ‘r’

# Frequency Analysis

- We now obtain

o-r-riend-ro--arise-a-inedhise--t---ass-it  
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ

hs-r-riseasi-e-a-orationhadta-en--ace-hi-e  
NDIFEFMZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

he-asnt-oo-in-i-o-redso-e-ore-ineandhesett  
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-ac-inhischair-aceti-ted--to-ardsthes-n  
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

# Frequency Analysis

- With further guessing, we obtain:

Our friend from Paris examined his empty glass with surprise, as if evaporation had taken place while he wasn't looking. I poured some more wine and he settled back in his chair, face tilted up towards the sun. –

# How to improve substitution cipher?

- Problem with substitution cipher
  - The frequency of occurrence of letters in plaintext is not randomized by the substitution cipher
- How to hide the statistics of a language after encryption?
  - Three main approaches
    - Homophonic substitution
    - Polyalphabetic substitution
      - Example: Vigenere
    - Polygraphic substitution
      - Example: Playfair cipher

# How to improve substitution cipher?

- Homophonic substitution
  - the substitution is no longer bijective
  - may consider it as one-to-many mapping
  - Example:
    - suppose that there are 1000 symbols in ciphertext
    - ‘E’ is encrypted to one of 127 ciphertext symbols (randomly)
    - ‘Z’ is encrypted to one ciphertext symbol
    - .....
    - so the symbols in ciphertext appear uniformly distributed
  - How to break it (for a long message) ?
    - Hint: digram pattern



# How to improve substitution cipher?

- Polyalphabetic substitution
  - Use multiple substitution alphabets
    - to hide the statistical feature of a language
  - Vigenere is the best-known polyalphabetic cipher
    - Inventor: Giovan Battista Bellaso, 1553
    - Believed unbreakable until 19th century
- Polygraphic substitution
  - Use multiple-letter substitution
  - Playfair cipher

# Vignere Cipher

- Use a number of shift ciphers
- Definition

Plaintext:  $P = (\mathbb{Z}_{26})^n$

Ciphertext:  $C = (\mathbb{Z}_{26})^n$

Key:  $K = (\mathbb{Z}_{26})^m$  (key consists of  $m$  letters)

Encryption:  $C_i = (P_i + K_i \bmod m) \bmod 26$

Decryption:  $P_i = (C_i - K_i \bmod m) \bmod 26$

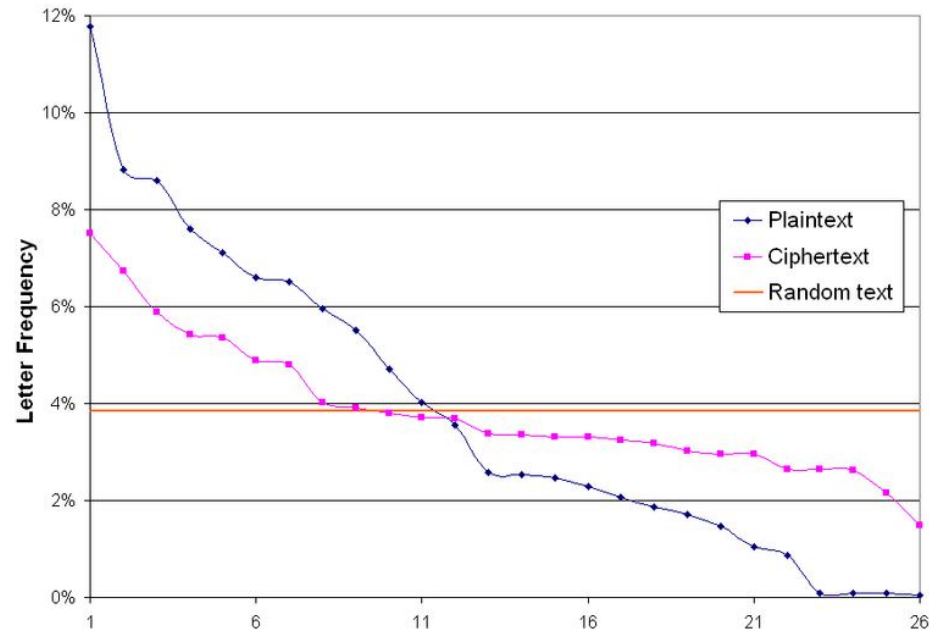
- Example

– Plaintext:      c   r   y   p   t   o   g   r   a   p   h   y  
– Key:              L   U   C   K  
– Ciphertext:

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
0	1	<u>2</u>	3	4	5	6	7	8	9	10	<u>11</u>	12	<u>13</u>	14	15	16	17	18	19	20	21	22	23	24	25

# Vigenere Cipher

- The Vigenère cipher masks the characteristic letter frequencies of English plaintexts, but some patterns remain (wiki diagram)



# Cryptanalysis of Vigenere Cipher

- Cryptanalysis approach
  - Find the length of the key
  - Then for each letter in the key, the problem becomes a simple shift cipher
    - use frequency cryptanalysis to break it (determining one letter is enough)
- How to find the key length  $m$  ?
  - Two methods
    - Kasiski test
    - Index of coincidence

# Cryptanalysis of Vigenere Cipher

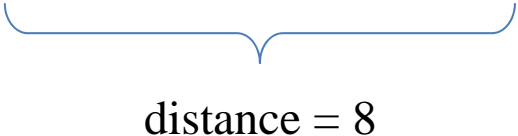
- Kasiski test
  - Based on the observation:
    - two identical segments of plaintext will be encrypted to the same ciphertext if their distance is the multiple of  $m$
  - Algorithm:
    1. Search for pairs of identical segments of length at least 3
    2. Record distances between the two segments:  $\Delta_1, \Delta_2, \dots$
    3.  $m$  is a divisor of  $\gcd(\Delta_1, \Delta_2, \dots)$
- length at least 3, why?

# Cryptanalysis of Vigenere Cipher

- Kasiski test (contd.)

– Example:

P: t h e s u n a n d t h e m a n i n t h e m o o n  
Key: K I N G K I N G K I N G K I N G K I N G K I N G  
C: D P R Y E V N T N B U K W I A O X B U K W W B T



distance = 8

# Cryptanalysis of Vigenere Cipher

- Index of coincidence
  - simple idea
    - If the value of  $m$  is guessed correctly, then the distribution of the ciphertext letters  $\{C_{\beta+m \times i}\}$  would be close to the distribution of the English letters for any constant  $\beta$ 
      - since  $\{C_{\beta+m \times i}\}$  are generated from the same substitution table
    - If the value of  $m$  is guessed wrongly, then the distribution of the ciphertext letters  $\{C_{\beta+m \times i}\}$  would be random

# Cryptanalysis of Vigenere Cipher

- Index of coincidence (contd.)

Definition: Suppose  $X = x_1x_2 \cdots x_n$  is a string of  $n$  alphabetic characters. The index of coincidence of  $X$ , denoted  $I_c(x)$ , is defined to be the probability that two random elements of  $X$  are identical.

Denote the numbers of A,B,C, ..., Z in  $X$  by  $f_0, f_1, f_2, \dots, f_{25}$  (respectively). We can choose two elements of  $X$  in  $\binom{n}{2}$  ways. For each  $i$ ,  $0 \leq i \leq 25$ , there are  $\binom{f_i}{2}$  ways of choosing both elements to be  $i$ . Thus

$$I_c = \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i-1)}{n(n-1)} \approx \frac{\sum_{i=0}^{25} f_i^2}{n^2} = \sum_{i=0}^{25} p_i^2$$



# Cryptanalysis of Vigenere Cipher

- Index of coincidence (contd.)
  - If  $X$  is a string of English language text,  $I_c = 0.065$

- If  $X$  is a random string, then

$$I_c = 1/26 = 0.038$$

letter	probability	letter	probability
<i>A</i>	.082	<i>N</i>	.067
<i>B</i>	.015	<i>O</i>	.075
<i>C</i>	.028	<i>P</i>	.019
<i>D</i>	.043	<i>Q</i>	.001
<i>E</i>	.127	<i>R</i>	.060
<i>F</i>	.022	<i>S</i>	.063
<i>G</i>	.020	<i>T</i>	.091
<i>H</i>	.061	<i>U</i>	.028
<i>I</i>	.070	<i>V</i>	.010
<i>J</i>	.002	<i>W</i>	.023
<i>K</i>	.008	<i>X</i>	.001
<i>L</i>	.040	<i>Y</i>	.020
<i>M</i>	.024	<i>Z</i>	.001

# Cryptanalysis of Vigenere Cipher

- Index of coincidence (contd.)

Algorithm: Guess the value of  $m$ . For  $\beta = 0, 1, 2, 3, \dots, m-1$ , compute the value of  $I_c$  for each set  $\{C_{\beta+m \cdot i}\}$ . If the values of  $I_c$  are all close to 0.065, then the value of  $m$  is guessed correctly.

# Cryptanalysis of Vigenere Cipher

- Example: ciphertext from a Vigenere cipher

CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQM~~Q~~EQERBW  
RVXUOAKXAOSXXWEAHBWGJMMQM~~N~~KGRFVGXWTRZXWIAK  
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX  
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLLCHR  
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT  
AMRVLCRREMNDGLXRRIMGNSNRWCHRQHAEYEVTAQE~~B~~BI  
PEEWEVKAKOEWADREMXMTBHHCHRTKDNVRZCHRCLQOHP  
WQAI IWXNRMGWOI I FKEE

# Cryptanalysis of Vigenere Cipher

- Example (contd.)
  - Kasiski test
    - Ciphertext string CHR occurs in five places, beginning at positions 1, 166, 236, 276 and 286. Thus the distances are 165, 235, 275, 285.  $\gcd(165, 235, 275, 285) = 5$
  - Indices of coincidences
    - $m = 1, I_c = 0.045$
    - $m = 2, I_c = 0.046, 0.041$
    - $m = 3, I_c = 0.043, 0.050, 0.047$
    - $m = 4, I_c = 0.042, 0.039, 0.045, 0.040$
    - $m = 5, I_c = 0.063, 0.068, 0.069, 0.061, 0.072$

# Playfair cipher

- Invented by Charles Wheatstone in 1854
- Its use was promoted by Lyon Playfair
- Digram substitution
- Step1: Generate the key table
- Step2: Encrypt the message

# Playfair cipher

- Step1: Generate the key table
  - 5 by 5 table containing 25 characters
    - I and J are treated in the same way
  - Fill the table with the letters of the key, dropping any duplicate letters
  - Fill the remaining spaces with the rest of the letters of the alphabet in order
  - Example: for the key “playfair example” (wiki)

P L A Y F<sub>A</sub>  
 I R E X<sub>A</sub> M<sub>PLE A</sub>  
 B C D<sub>EF</sub> G H<sub>I=J</sub>  
 K<sub>LM</sub> N<sub>P</sub> Q<sub>R</sub> S  
 T U V W<sub>XY</sub> Z



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

# Playfair cipher

- Step2: Encrypt the message
  - Step 2.1: Break a message into digrams. If both letters are the same in the digram (or if there is only one letter left), add an “X” after the first letter.
    - Example: the message  
“Hide the gold in the tree stump” becomes  
HI DE TH EG OL DI NT HE TR EX ES TU MP

# Playfair cipher

- Step2: Encrypt the message
  - Step 2.2: Encrypt each digram by applying the rules:
    - Rule 1: If the letters appear on the same row of your table, replace them with the letters to their immediate right respectively (wrapping around to the left side of the row if a letter in the original pair was on the right side of the row).
      - Example: Encrypt the digram OR → YZ



*	*	*	*	*
*	O	Y	R	Z
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*



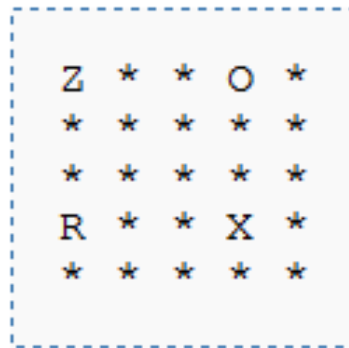
# Playfair cipher

- Step2: Encrypt the message
  - Step 2.2: Encrypt each digram by applying the rules:
    - Rule 2: If the letters appear on the same column of your table, replace them with the letters immediately below respectively (wrapping around to the top side of the column if a letter in the original pair was on the bottom side of the column).
      - Example: Encrypt the digram OR → BY

*	*	O	*	*
*	*	B	*	*
*	*	*	*	*
*	*	R	*	*
*	*	Y	*	*

# Playfair cipher

- Step2: Encrypt the message
  - Step 2.2: Encrypt each digram by applying the rules:
    - Rule 3: If the letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair. The order is important – the first letter of the encrypted pair is the one that lies on the same **row** as the first letter of the plaintext pair.
      - Example: Encrypt the digram OR → ZX



# Playfair cipher examples

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

HI

Shape: Rectangle  
Rule: Pick Same Rows,  
Opposite Corners

BM

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

DE

Shape: Column  
Rule: Pick Items Below Each  
Letter, Wrap to Top if Needed

OD

P	L	A	Y	F
I	R	E	X	M

TH

Shape: Rectangle  
Rule: Pick Same Rows,  
Opposite Corners

ZB

B	C	D	G	H
K	N	O	Q	S
T	U	V	W	Z

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

EX

Shape: Row  
Rule: Pick Items to Right of Each  
Letter, Wrap to Left if Needed

XM

# Transposition (permutation) cipher

- Definition:

Let  $m$  be a positive integer. Let  $\mathcal{P} = \mathcal{C} = (\mathbb{Z}_{26})^m$  and let  $\mathcal{K}$  consist of all permutations of  $\{1, \dots, m\}$ . For a key (i.e., a permutation)  $\pi$ , we define

$$e_{\pi}(x_1, \dots, x_m) = (x_{\pi(1)}, \dots, x_{\pi(m)})$$

and

$$d_{\pi}(y_1, \dots, y_m) = (y_{\pi^{-1}(1)}, \dots, y_{\pi^{-1}(m)}),$$

where  $\pi^{-1}$  is the inverse permutation to  $\pi$ .

# Transposition (permutation) cipher

- Example:

Suppose  $m = 6$  and the key is the following permutation  $\pi$ :

$x$	1	2	3	4	5	6
$\pi(x)$	3	5	1	6	4	2

plaintext: shesellsseashellsbytheseashore

1) Partition the plaintext into groups of 6 letters

shesel | lsseas | hellsb | ythese | ashore

2) Rearrange the 6 letters in each group

EESLSH | SALSES | LSHBLE | HSYEET | HRAEOS

ciphertext: EESLSHSALSESLSHBLEHSYEETHRAEOS

# Summary of classical ciphers

- Shift ciphers
- Substitution ciphers
  - frequency cryptanalysis
- Vigenere cipher
  - Kasiski test
  - Index of coincidence
- Playfair cipher
- Transposition (permutation) cipher