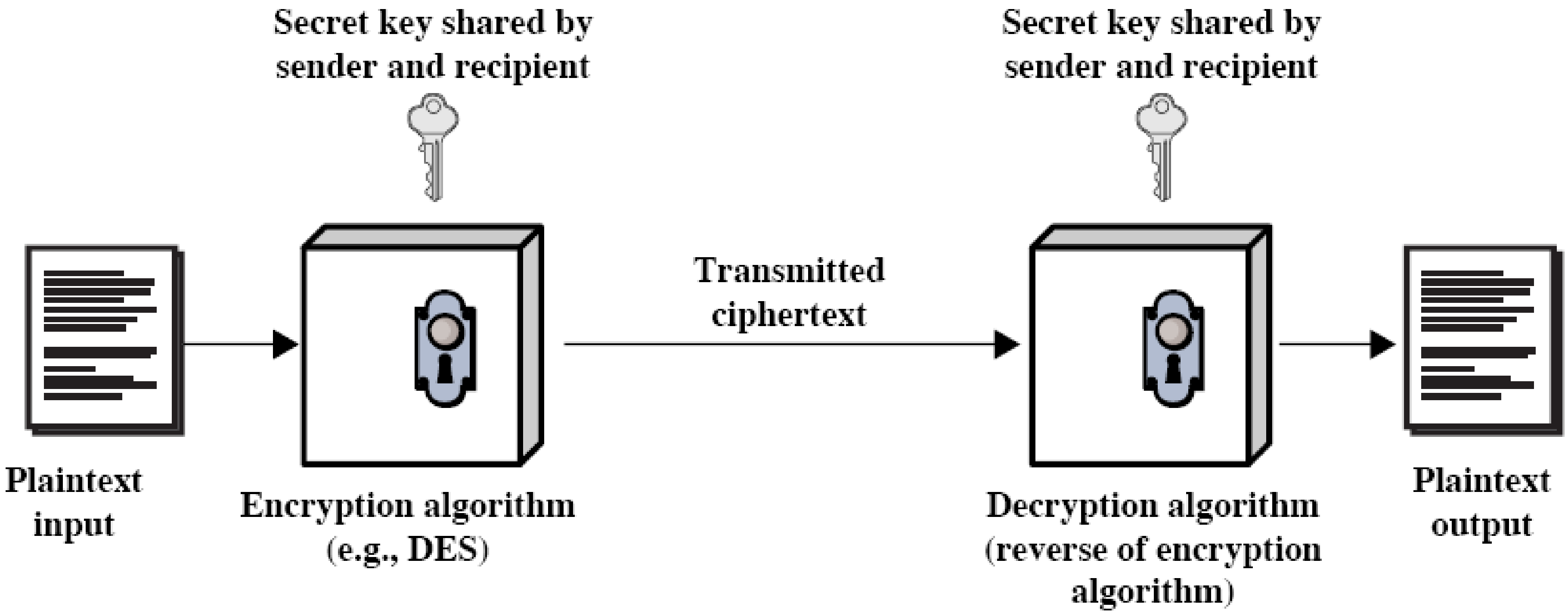
Classical Encryption Techniques

CSS 322 – Security and Cryptography

Contents

- Terminology and Models
- Requirements, Services and Attacks
- Substitution Ciphers
  - Caesar, Monoalphabetic, Polyalphabetic, One-time Pad, …
- Transposition Ciphers
- Steganography

Basic Encryption Model


Secret key shared by sender and recipient
Secret key shared by sender and recipient
Transmitted ciphertext
Plaintext input
Encryption algorithm (e.g., DES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Terminology

- Plaintext
  - The original message we want to keep secret
- Encryption algorithm
  - Takes the plaintext (and key) and produces modified (preferably unintelligible) output, i.e. the ciphertext
- Secret key
  - Used as input to encryption algorithm to change the output produced by the algorithm
- Ciphertext
  - The encrypted message
- Decryption algorithm
  - Takes the ciphertext and key to produce the original plaintext message

Requirements for Security

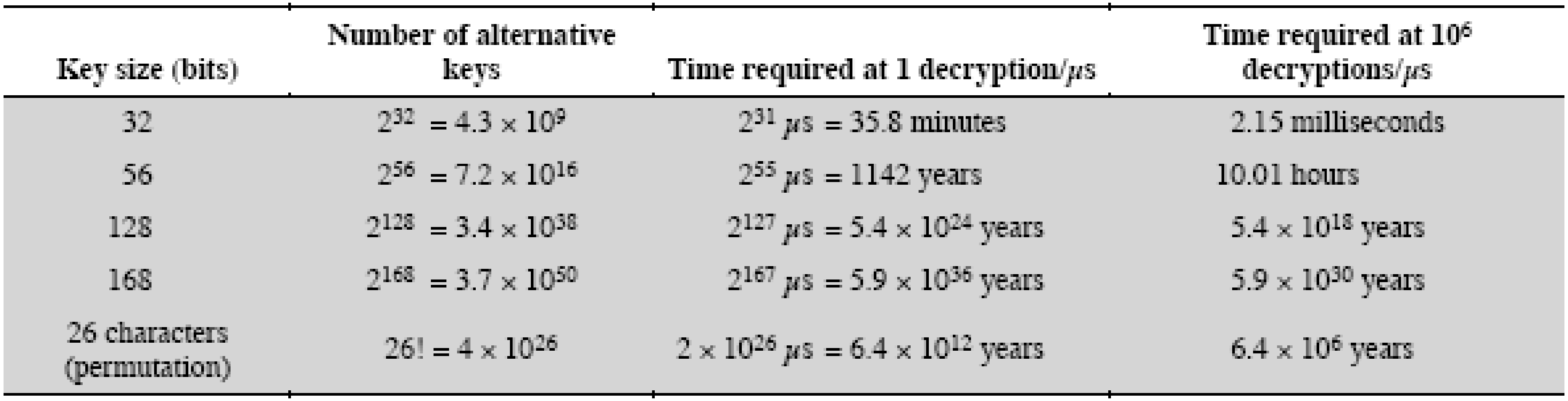
- For a symmetric encryption system to be secure, it must:
  1. Have a strong encryption algorithm. Given the algorithm and ciphertext, an attacker cannot obtain the key or plaintext.
  2. Sender and receiver have knowledge of the secret key (and keep it secret).
- Do not have to keep algorithm secret – only the key
  - Allows for mass and cheap manufacturing of devices that perform symmetric key encryption

Cryptographic Systems

- Can be characterised by:
  - Operations used to transform plaintext to ciphertext
    - Substitution: replace one element in plaintext with another
    - Transposition: rearrange elements
  - Number of keys used
    - Sender/receiver use same key: symmetric, secret-key, shared-key or conventional cryptosystem
    - Sender/receiver use different keys: asymmetric or public-key cryptosystem
  - Processing of plaintext
    - Block cipher: encrypt (and decrypt) fixed size block of plaintext at a time
    - Stream cipher: continuous input of plaintext and output of ciphertext

Forms of Attacks

- Brute Force Attack
  - Try every key possible until readable text is obtained from the ciphertext
  - On average, number of guesses is half the key space



Key size (bits)	Number of alternative keys	Time required at 1 decryption/μs	Time required at 10^6 decryptions/μs
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	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	$2^{167}\ \mu s = 5.9 \times 10^{36}$ years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	$2 \times 10^{26}\ \mu s = 6.4 \times 10^{12}$ years	6.4×10^6 years

- Cryptanalysis
  - Use knowledge of algorithm and/or plaintext patterns to “intelligently” decipher the ciphertext
  - Attacks differ based on amount of information known to attacker
- Cryptanalyst or attacker tries to break the system

Cryptanalytic Attacks

- Assume encryption algorithm is known by attacker
- Types of attacks (and known information)
  - Ciphertext only
    - Ciphertext
  - Known plaintext
    - Ciphertext; One or more plaintext/ciphertext pairs
  - Chosen plaintext
    - Ciphertext; Plaintext chosen by attacker, and corresponding ciphertext
  - Chosen ciphertext
    - Ciphertext; Ciphertext chosen by attacker, and corresponding decrypted plaintext
  - Chosen text
    - Ciphertext; Plaintext chosen by attacker, and corresponding ciphertext; Ciphertext chosen by attacker, and corresponding decrypted plaintext

Measure of Security

- Unconditionally Secure
  - Ciphertext does not contain enough information to derive plaintext or key (even if attacker had all the time in the world!)
  - One-time pad is only known scheme
  - Not practically feasible
- Computationally Secure
  - If either:
    - Cost of breaking the cipher exceeds value of encrypted information
    - Time required to break cipher exceeds useful lifetime of encrypted information
  - Hard to estimate value/lifetime of some information
  - Harder to estimate how much effort needed to break a cipher

Caesar Cipher

- Julius Caesar shifts each letter by three positions in the alphabet:

```
Plain  (p): 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 (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 C
```

- Example:

```
Cipher:  VHFXULWBDQGFUBSWRJUDSKD
Plain:   ?
```

- Can be generalised to shift by *k* positions
- Assume each letter is assigned a number (a=0, b=1, …)

$$C = E(p) = (p + k)\,\mathrm{mod}(26)$$

$$p = D(C) = (C - k)\,\mathrm{mod}(26)$$

Breaking the Caesar Cipher

- Try all 25 keys (brute force attack)
  - E.g. k=1, k=2, k=3, …
  - If the language of the plaintext is known, often can recognise correct plaintext
    - What if you don’t know the language?
    - What if it is compressed?
    - …

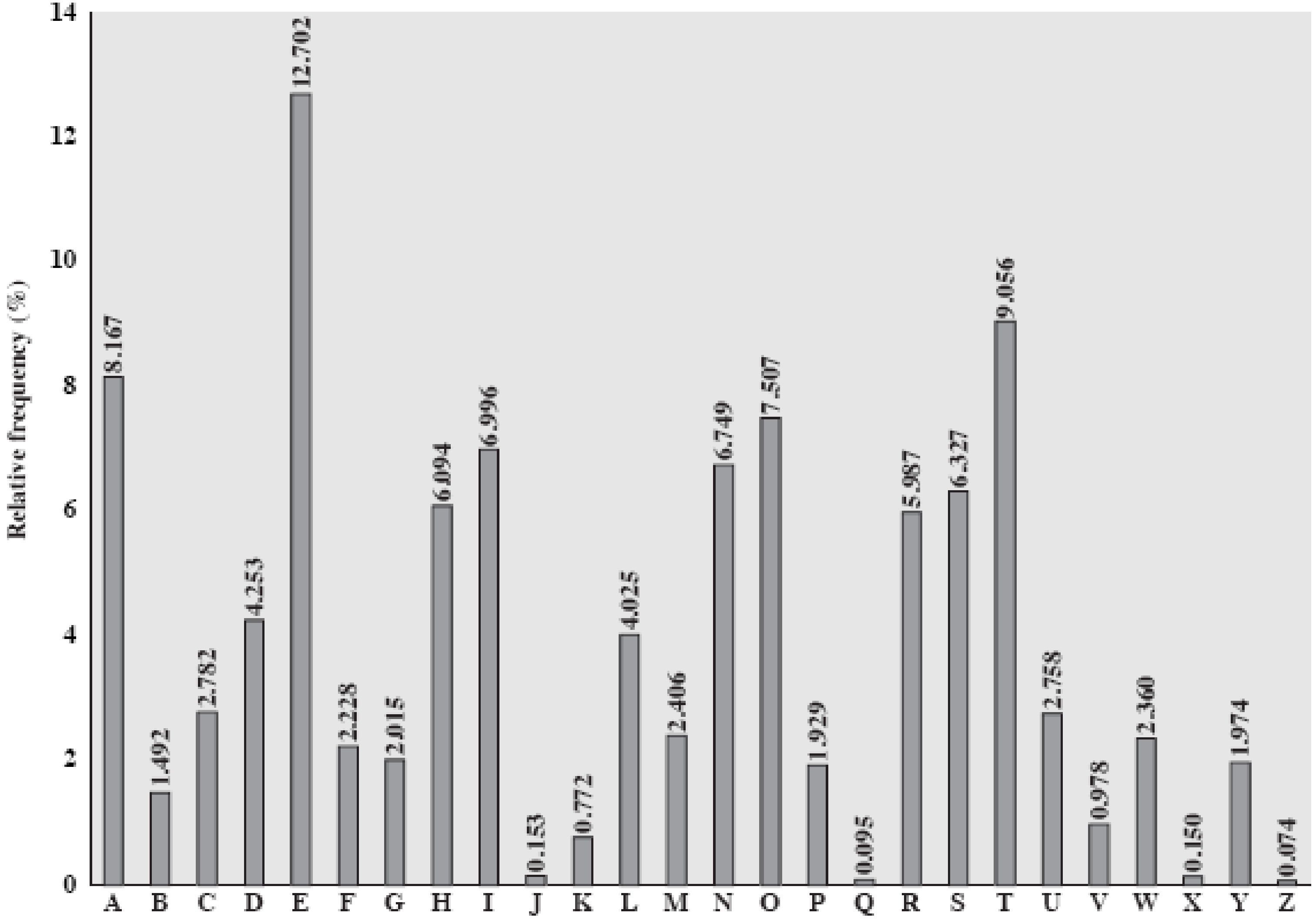
Monoalphabetic Ciphers

- Allow any permutation of characters to be key, e.g.

```
Plain  (p):  a b c d e … w x y z
Cipher (C):  D Z G L S … B T F Q
```

- Number of keys: $26! > 4 \times 10^{26}$
  - 6.4×10^{6} years to try every key
- But knowledge of language statistics makes it easy to break
  - E.g. if attacker knows the message is in plain English can use known patterns in English language
    - Frequency of letters
    - Frequency of pairs of letters (digrams) and triples of letters (trigrams)
    - Known or expected words in plaintext

Frequency of Letters in English



Breaking Monoalphabetic Ciphers

```
Cipher: UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
        VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
        EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ
```

- Relative frequency of letters

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
M	6.67								

- What can we guess?
  - P and Z are e and t (what order?)
  - Use also statistics of digrams (th, he, an, in) and trigrams (the, and, tha, ent). E.g. ZW = th
  - Continue analysis and trial-and-error

Playfair Cipher

- Construct 5x5 matrix based on a keyword

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

  - Keyword starts matrix (omit duplicate letters)
  - I/J is treated as one letter
  - Fill in remainder of matrix with other letters in alphabetical order
- To encrypt, consider pairs of letters

Playfair Cipher Rules

1. If plaintext has pair of identical letters, then use a filler such as x before encrypting. E.g.
   - balloon = ba lx lo xo nx
2. If pair is on same row, then cipher text becomes the letters to the right
   - E(be) = CN
3. If pair is on same column, the cipher text becomes the letters below
   - E(tv) = NT
4. Else, plaintext is replaced by letter in same row as itself and same column as other letter
   - E(ir) = AS, E(ud) = QE

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

Playfair Cipher Analysis

- Improves on monoalphabetic ciphers because much harder to use statistics about relative frequency of letters and digrams
- But still relatively easy to break using language/frequency analysis

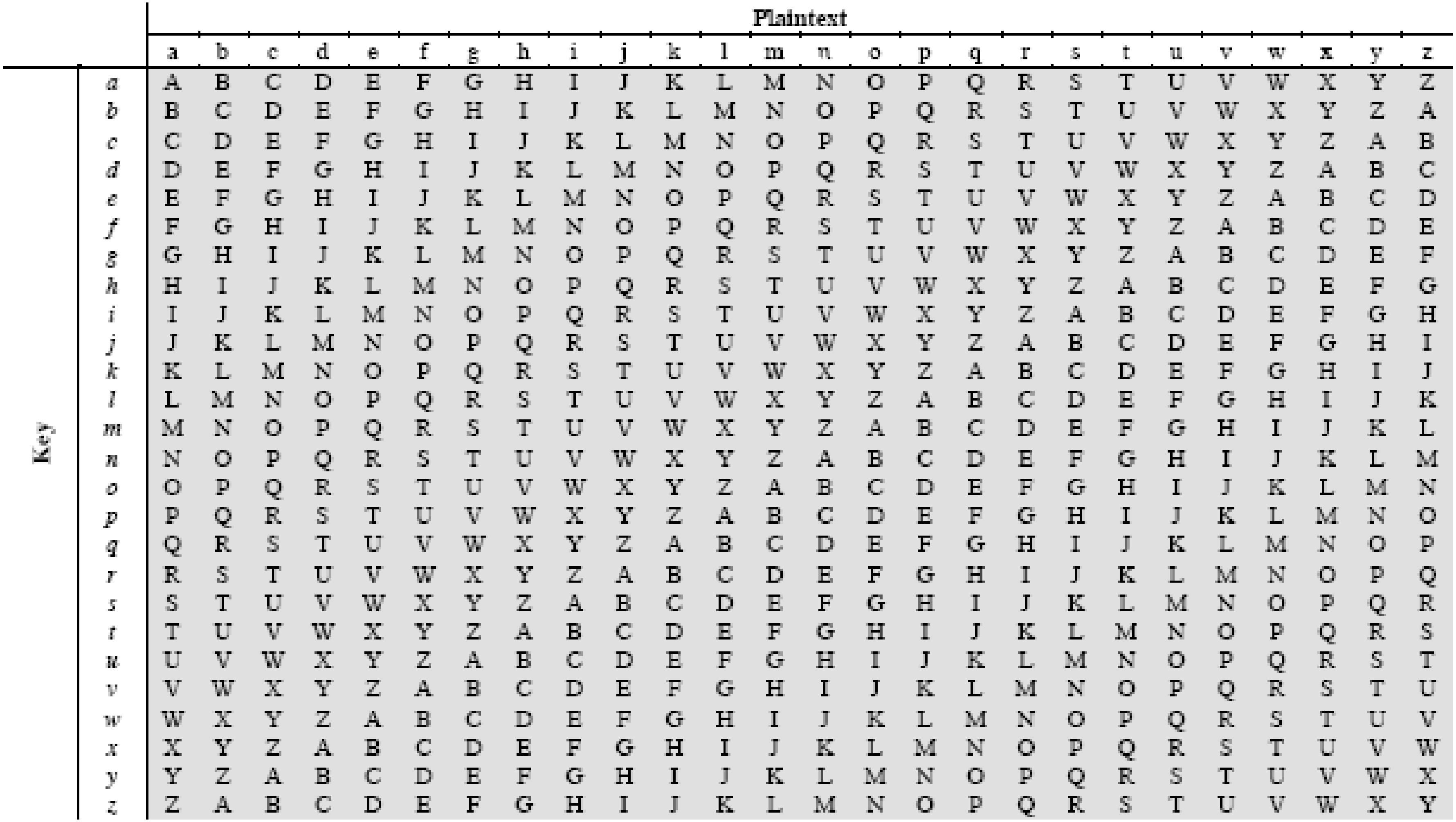
Vigenère Cipher

- Example of polyalphabetic cipher
  - A key determines which one of a set of monoalphabetic ciphers to use
- Uses the 26 Caesar ciphers
  - The keyword letter determines which Caesar cipher to use (a → k=0, b → k=1, …)
  - Conceptually, a look-up table can be used (next slide)
- Example:

```
Plain  (p):  internettechnologies
Keyword   :  sirindhornsirindhorn
Cipher (C):  AVKMEQLHKRUPEWYRNWVF
```

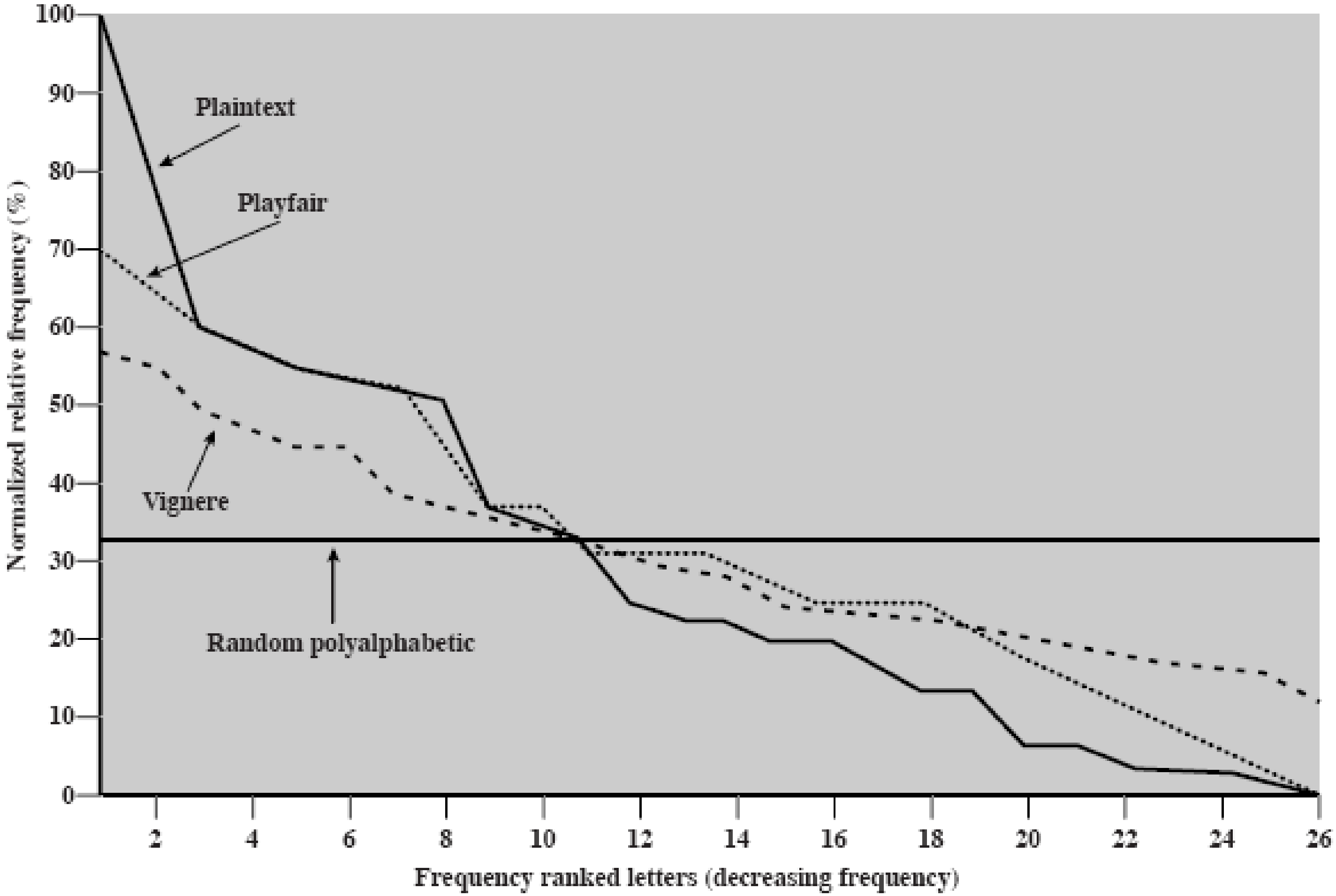
- Information about letter frequency is hidden (multiple ciphertext letters for each plaintext letter)
- But still some information can be used by attacker

Vigenère Table



		Plaintext																									
		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
Key	*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

Information on Letter Frequencies



Breaking the Vigenère Cipher

- Attacker makes following guesses/assumptions:
  - If monoalphabetic code used, then relative frequency of cipher letters should match those of English language
  - If frequency does not match, assume Vigenère
  - Repeated sequences of ciphertext are generated if 2 sequences of plaintext are integer multiple of keyword length apart
    - With long ciphertext, analyst can detect length of keyword
    - Can then analyse individual monoalphabetic ciphers
- Improving on Vigenère
  - Depends upon the construction of the key
    - Use very long keyword
    - Concatenate keyword with plaintext to create new keyword
  - Ultimate security
    - Use keyword the same length as plaintext and no statistical relationship with plaintext
    - Called *One-time pad*

One-time Pad

- Only known scheme that is unbreakable (unconditionally secure)
  - Knowing the entire ciphertext, an attacker cannot obtain plaintext
- Use key which is same length as plaintext and no statistical relationship with plaintext

```
Ciphertext: ANKYODKYUREPFJBYOJDSPLREYIUNOFDOIUERFPLUYTS
Key1      : pxlmvmsydoftyrvzwc_tnlebnecvgdupahfzzlmnyih
Plaintext1: mr_mustard_with_the_candlestick_in_the_hall

Key2      : mfugpmiydgaxgoufhklllmhsqdqogtewbqgfyovuhwt
Plaintext2: miss_scarlet_with_the_knife_in_the_library_
```

- If attacker discovered keys, no way to pick correct plaintext
- Many different keys may produce readable plaintext

One-time Pad

- Truly random key produces truly random ciphertext
  - Attacker cannot take advantage of frequency of letters
- In theory, one-time pad is ultimate security
- In practice,
  - Messages are often quite long
  - Requires many large random keys – this is not easy (look at random number generators later)
  - Each message needs a different key of same length to be distributed between sender and receiver – very resource consuming and hard to use
- One-time pad has very little practical use

Transposition Techniques: Rail Fence

- Re-arrange the letters in the plaintext
- Rail-fence technique of depth N (example: N=3)
  - Write letters in a diagonal and then obtain ciphertext by reading rows

```
Plain:  internettechnologiesandapplications

        i  e  e  e  n  o  e  n  p  i  t  n
         n  r  t  c  o  g  s  d  p  c  i  s
          t  n  t  h  l  i  a  a  l  a  o

Cipher:  IEEENOENPITNNRTCOGSDPCISTNTHLIAALAO
```

  - Easy to break
    - Use frequency of letters etc to determine the depth

Transposition Techniques: Rows/Columns

- Write plaintext in rows, and then read (to obtain ciphertext) in columns AND re-order the columns

```
Plain:  securityandcryptography
Key:    3 1 5 6 2 4

        s e c u r i
        t y a n d c
        r y p t o g
        r a p h y a

Cipher: EYYARDOYSTRRICGACAPPUNTH
```

- Harder than rail fence to analyse, but still relatively easy using letter frequency, digrams and trigrams and trying different columns

Transposition Techniques

- Significant increase in security by applying the transposition a second (and third …) time

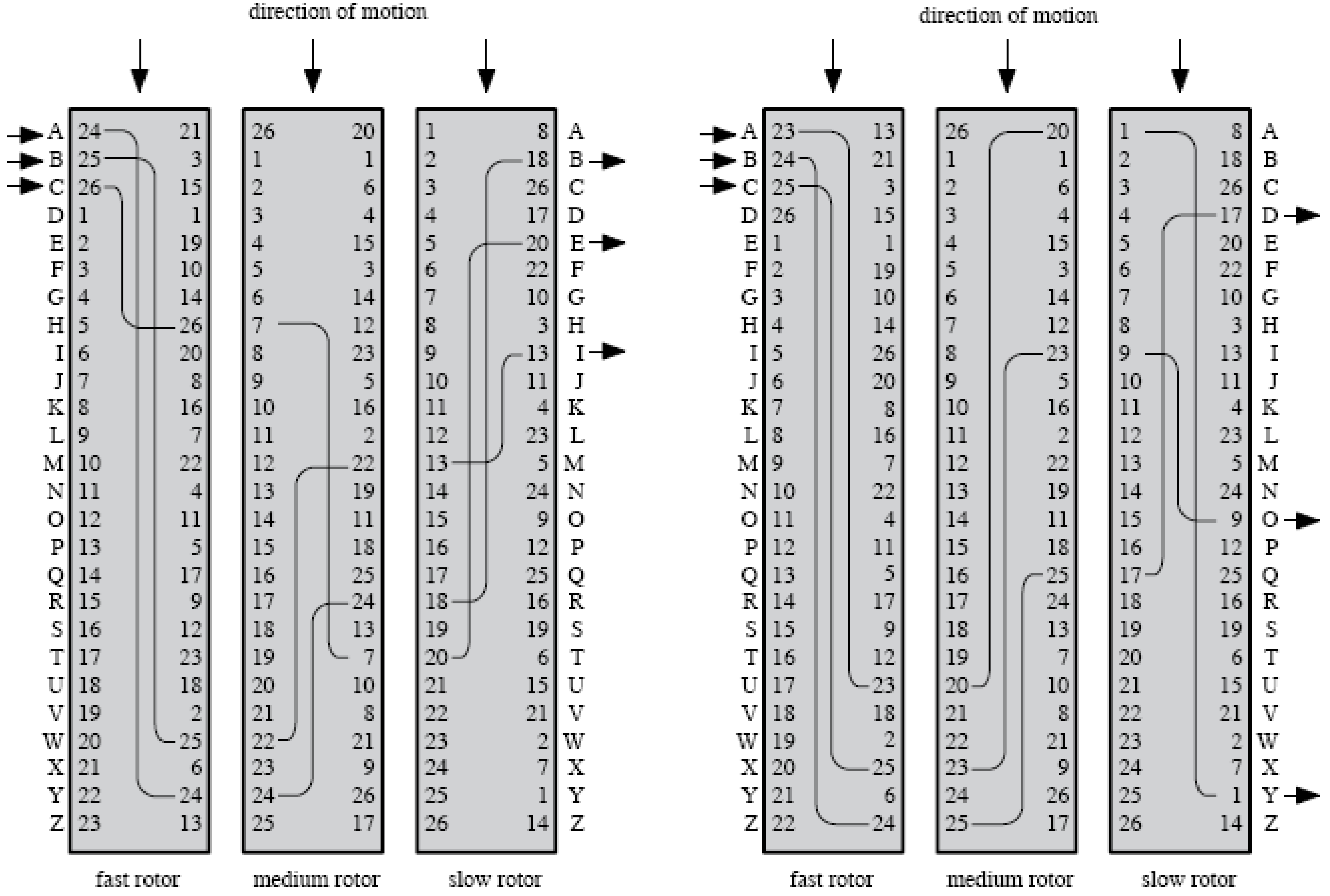
```
Input:  EYYARDOYSTRRICGACAPPUNTH
Key:    3 1 5 6 2 4

        e y y a r d
        o y s t r r
        i c g a c a
        p p u n t h

Output: YYCPRRCTEOIPDRAHYSGUATAN
```

- There is much less structure in output after two transpositions
  - Much harder to analyse

Rotor Machines

- Multiple stages of encryption can be used for substitution (e.g. mono-/poly-alphabetic) and transposition ciphers
- Rotor machines were early application of this
  - Principle was basis for Enigma cipher used by Germany in WW2
- Machine has multiple cylinders
  - Monoalphabetic substitution cipher for each cylinder
  - Output of one cylinder is input to next cylinder
  - Plaintext is input to first cylinder; ciphertext is output of last cylinder
  - Entering a plaintext letter causes last cylinder to rotate its cipher
  - Complete rotation of one cylinder causes previous cylinder to rotate its cipher
- Principle is used in Data Encryption Standard (DES)

Rotor Machines



(a) Initial setting

	fast rotor		medium rotor		slow rotor		
A	24	21	26	20	1	8	A
B	25	3	1	1	2	18	B
C	26	15	2	6	3	26	C
D	1	1	3	4	4	17	D
E	2	19	4	15	5	20	E
F	3	10	5	3	6	22	F
G	4	14	6	14	7	10	G
H	5	26	7	12	8	3	H
I	6	20	8	23	9	13	I
J	7	8	9	5	10	11	J
K	8	16	10	16	11	4	K
L	9	7	11	2	12	23	L
M	10	22	12	22	13	5	M
N	11	4	13	19	14	24	N
O	12	11	14	11	15	9	O
P	13	5	15	18	16	12	P
Q	14	17	16	25	17	25	Q
R	15	9	17	24	18	16	R
S	16	12	18	13	19	19	S
T	17	23	19	7	20	6	T
U	18	18	20	10	21	15	U
V	19	2	21	8	22	21	V
W	20	25	22	21	23	2	W
X	21	6	23	9	24	7	X
Y	22	24	24	26	25	1	Y
Z	23	13	25	17	26	14	Z

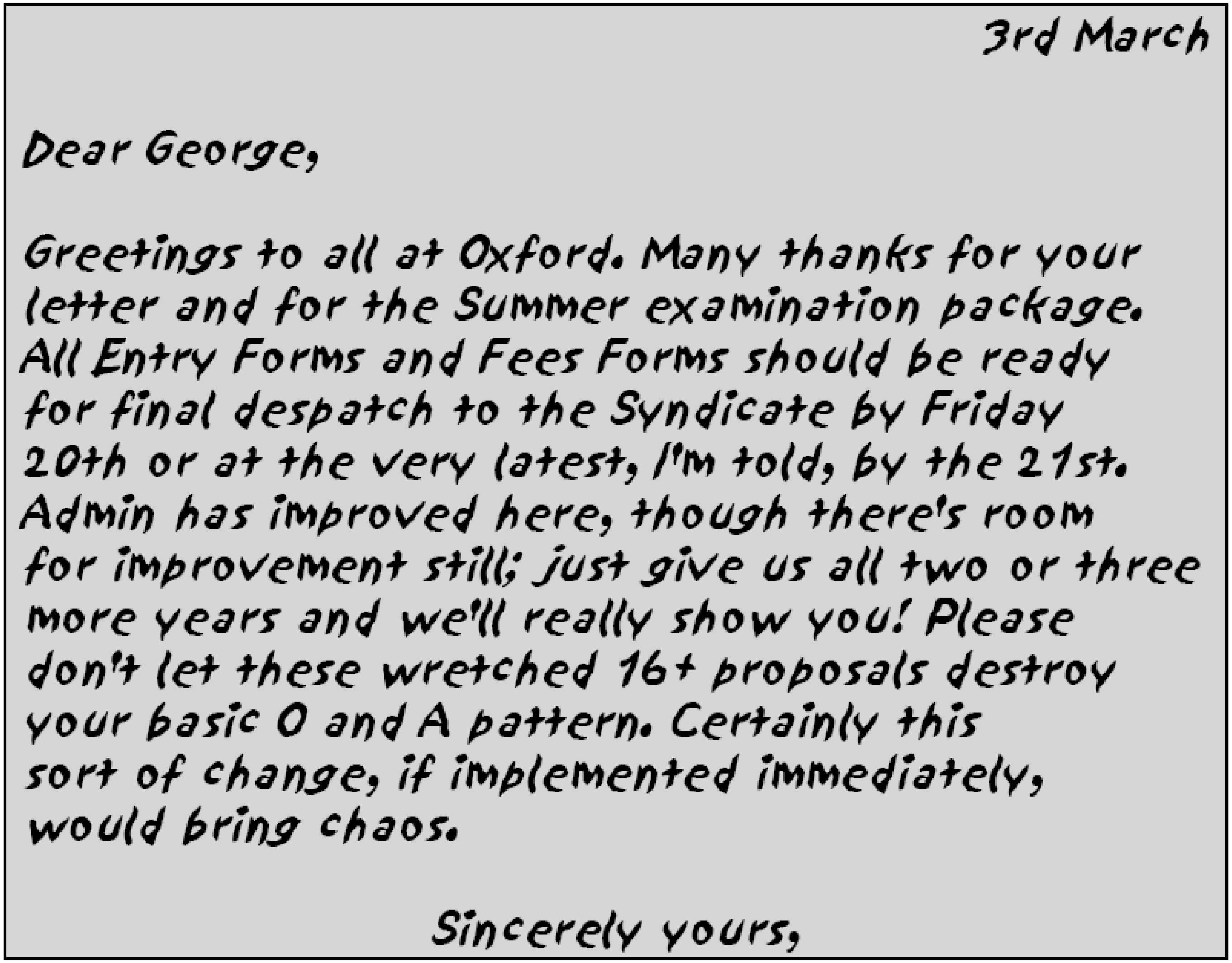
(b) Setting after one keystroke

	fast rotor		medium rotor		slow rotor		
A	23	13	26	20	1	8	A
B	24	21	1	1	2	18	B
C	25	3	2	6	3	26	C
D	26	15	3	4	4	17	D
E	1	1	4	15	5	20	E
F	2	19	5	3	6	22	F
G	3	10	6	14	7	10	G
H	4	14	7	12	8	3	H
I	5	26	8	23	9	13	I
J	6	20	9	5	10	11	J
K	7	8	10	16	11	4	K
L	8	16	11	2	12	23	L
M	9	7	12	22	13	5	M
N	10	22	13	19	14	24	N
O	11	4	14	11	15	9	O
P	12	11	15	18	16	12	P
Q	13	5	16	25	17	25	Q
R	14	17	17	24	18	16	R
S	15	9	18	13	19	19	S
T	16	12	19	7	20	6	T
U	17	23	20	10	21	15	U
V	18	18	21	8	22	21	V
W	19	2	22	21	23	2	W
X	20	25	23	9	24	7	X
Y	21	6	24	26	25	1	Y
Z	22	24	25	17	26	14	Z

Steganography

- Hide a real message in a fake, but meaningful, message
- Assumes recipient knows the method of ‘hiding’
- Examples:
  - Selected letters in a document are marked to form the hidden message
  - Invisible ink (letters only become visible when exposed to a chemical or heat)
  - Using selected bits in images or videos to carry the message
    - E.g. Change the last bits in each frame of a video to carry the message – difficult to notice difference in quality of video
- Advantages
  - Does not ‘look like’ you are hiding anything
- Disadvantages
  - Once attacker knows your method, everything is lost
  - Can be inefficient (need to send lot of information to carry small message)

Steganography Example



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,

Summary

- Cryptographic systems assume:
  - Attacker knows algorithm and ciphertext
  - Attack tries to discover plaintext and/or keys
- Attacks can be by cryptanalysis (intelligent analysis of information) or brute force (try every possible key)
- Cryptographic algorithms use transposition and substitution
- Many old (and insecure) algorithms:
  - Caesar, Monoalphabetic, Vignere, Playfair, Rotor machine, …
- Next we look at algorithms suitable for practical use in today’s computers and networks